Christie
Pandoras Box

User Manual

Pandoras Box 8.3.0 &
Widget Designer 6.5.1

Table of Contents

1 Hazard Warning 11
2 Conformity and Electrical Information 15
3 What's New? 17
4 What's New in Widget Designer 46
5 Product Overview 61
6 Pandoras Box 66
6.1 First Steps and Troubleshooting 67
6.1.1 Master / Client Remote Setup 67
6.1.2 How to Connect via VNC Remote 69
6.1.3 How to See the Taskbar 73
6.1.4 How to Ping and Check Your IP Adress 74
6.1.5 How to Check and Change the Resolution 76
6.1.6 How to Access Log Files 77
6.1.7 How to Install a New Pandoras Box Version 79
6.1.8 How to Copy Files to a Remote PC 80
6.1.9 How to Copy Content to Your Hard Drive 81
6.1.10 How to Copy Content to the Project 84
6.1.11 Basic Steps in the User Interface 86
6.2 Content 90
6.2.1 Display & Content Formats 90
6.2.2 Audio Formats 92
6.2.3 Image Formats 93
6.2.4 Image Sequence Formats 95
6.2.5 Video Formats 97
6.2.6 Object Formats 100
6.2.7 Encoding and Transcoding 100
6.2.7.1 Adobe Plugin 101
6.2.7.2 coolux Codec 102
6.2.7.3 Encoder Extension 103
6.2.7.4 Quicktime Converter 108
6.3 User Interface - Master 117
6.3.1 Startup Dialog 118
6.3.2 Menu and View Bar 119
6.3.2.1 File Menu 119
6.3.2.2 Edit Menu 121
6.3.2.3 Tabs Menu 122
6.3.2.4 Tools Menu 122
6.3.2.5 Backup Menu 123
6.3.3 Status Bar 124
6.3.4 Tabs Overview 125
6.3.4.1 Active Values 127
6.3.4.2 Aeon FX and FX Explorer 129
6.3.4.3 Assets 131
6.3.4.4 Audio and Video Export 135
6.3.4.5 Audio and Video Recording 137
6.3.4.6 Configuration 139
6.3.4.7 Cues 168
6.3.4.8 Curve Editor 169
6.3.4.9 Device Control 171
6.3.4.10 Device Tree 173
6.3.4.11 Device Types 183
6.3.4.12 Device Viewer 184
6.3.4.13 Encoder Extension 185
6.3.4.14 File Browser 185
6.3.4.15 FireFly Particles 185

6.3.4.16 Groups 189
6.3.4.17 Inspector 190
6.3.4.18 Media Encryption 222
6.3.4.19 Patch 228
6.3.4.20 Multi-User 234
6.3.4.21 Playlist 239
6.3.4.22 Presets 241
6.3.4.23 Preview 243
6.3.4.24 Project 278
6.3.4.25 Sequence 292
6.3.4.26 Sequence Control 303
6.3.4.27 Task Manager 305
6.3.4.28 Text Input Editor 307
6.3.4.29 Time Monitor 309
6.3.4.30 Thumbnails 310
6.3.4.31 Views 310
6.3.4.32 Virtual Site 311
6.3.5 Layout 313
6.3.6 Pandoras Box Keyboard Shortcuts 314
6.4 User Interface - Client 316
6.4.1 Client Live Input Configuration 317
6.4.2 Client Keyboard Shortcuts 317
6.5 Devices and Layer Parameters 318
6.5.1 Video Processing Pipeline 320
6.5.2 Working with Effects 322
6.5.2.1 Adding, Editing and Removing FX 323
6.5.2.2 FX Order 325
6.5.2.3 Copy / Paste of FX 326
6.5.2.4 FX List 327
6.5.2.5 Blacklevel Compensation 646
6.5.3 Video Layer 647
6.5.3.1 Media / Mesh 647
6.5.3.2 Opacity 649
6.5.3.3 Playback 649
6.5.3.4 Audio 650
6.5.3.5 Position 651
6.5.3.6 Rotation 652
6.5.3.7 Scale 653
6.5.3.8 Rotation Pivot 654
6.5.3.9 Scale Pivot 655
6.5.3.10 Blend Mode 657
6.5.4 Audio Tracks 661
6.5.4.1 Media 662
6.5.4.2 Playback Transport 663
6.5.4.3 Volume 663
6.5.4.4 Inpoint / Outpoint 664
6.5.4.5 Pan 665
6.5.5 Pointer Layer 665
6.5.6 Light Layer 666
6.5.6.1 Media 667
6.5.6.2 Intensity 668
6.5.6.3 Position 669
6.5.6.4 Target 669
6.5.6.5 Color 670
6.5.6.6 Settings 670
6.5.7 Notch Layer 671
6.5.8 Camera Layer 675
6.5.8.1 Projection Mode 676
6.5.8.2 Viewpoint 677
6.5.8.3 Target 678
6.5.8.4 Settings 680

6.5.8.5 Lens Shift 680
6.5.8.6 Background Color 681
6.5.9 Output Layer 682
6.5.9.1 Matrix 683
6.5.9.2 Object 684
6.5.9.3 Background Color 686
6.5.9.4 Pixel Warp 686
6.5.9.5 Keystone 687
6.5.9.6 Softedge 688
6.5.9.7 Render State 689
6.5.10 DMX Devices 691
6.5.11 Serial Link Device 692
6.5.12 TCP Device 693
6.5.13 Venue Site 694
6.5.14 Widget Designer Device 696
6.5.15 Spark 699
6.6 External Control (DMX,Midi,...) 706
6.6.1 Input Protocols 706
6.6.1.1 DMX Input 706
6.6.1.2 Midi 716
6.6.1.3 MSC Midi Show Control 716
6.6.1.4 Serial Link 719
6.6.1.5 SMPTE Input 719
6.6.1.6 TCP/IP 720
6.6.1.7 TCP or UDP via SDK 721
6.6.2 Output Protocols 721
6.6.2.1 DMX Output 721
6.6.2.2 Serial Link 722
6.6.2.3 Spark 722
6.6.2.4 SMPTE Output 723
6.6.2.5 TCP/IP 723
6.7 StreamiX Live Input 724
6.7.1 StreamiX Desktop User Interface 725
6.7.1.1 Network Connection 726
6.7.1.2 Desktop Region 726
6.7.1.3 Output Format 727
6.7.1.4 Transmission 727
6.7.1.5 General Information 728
6.7.2 StreamiX Live Input Settings 728
6.8 SDK 730
6.8.1 Getting Started 732
6.8.1.1 Using Visual Basic / C# 733
6.8.1.2 Using JavaScript 734
6.8.1.3 Other Languages 735
6.8.2 Examples 738
6.8.2.1 Connect SDK to Pandoras Box 738
6.8.2.2 Sending Parameter Values 740
6.8.2.3 Receiving Parameter Values 741
6.8.2.4 Error Handling 742
6.8.3 Datatypes and Structures 742
6.8.4 Function Reference 745
6.9 Ports Used by PB and WD 781
6.10 Copyright Pandoras Box 783
7 Widget Designer 785
7.1 Installation and Launching 787
7.2 User Interface 789
7.2.1 File Menu 792
7.2.1.1 Import Settings 793
7.2.1.2 Profile Settings 795
7.2.1.3 Security Settings 796
7.2.2 Edit Menu 797

7.2.2.1 Project Settings 798
7.2.3 View Menu 799
7.2.3.1 Widget Explorer 802
7.2.4 Windows Menu 803
7.2.5 Pages Menu 805
7.2.6 Widgets Menu 808
7.2.7 Nodes Menu 809
7.2.8 Connections Menu 809
7.2.9 Devices Menu 810
7.2.10 Scripting Menu 811
7.2.11 Tools Menu 813
7.2.12 Effects & Animations (Web Styles) 814
7.2.13 Widget Designer Keyboard Shortcuts 816
7.3 Widgets 818
7.3.1 Analog Clock 819
7.3.2 Buttons 821
7.3.2.1 Custom Script Button 822
7.3.2.2 Script Timer 826
7.3.2.3 Media Control Panel 828
7.3.2.4 Cue Control 831
7.3.2.5 Image Loader 833
7.3.2.6 Video Snapshot 837
7.3.2.7 Art-Net Snapshot 839
7.3.2.8 Art-Net Recorder 841
7.3.2.9 Scroller 844
7.3.2.10 Flipswitch 848
7.3.3 Digital Clock 850
7.3.4 Displays 851
7.3.4.1 Bar Graph 852
7.3.4.2 Digital Display 854
7.3.4.3 Gauge 856
7.3.4.4 Graph Display 858
7.3.4.5 PowerPoint Display 860
7.3.4.6 Terra Display Array 863
7.3.4.7 Video Input Display 865
7.3.5 Drop Down List 868
7.3.6 Event Viewer 871
7.3.7 Fader Controls 874
7.3.7.1 Fader 874
7.3.7.2 Encoder 877
7.3.7.3 Color Picker 880
7.3.7.4 Wheel 883
7.3.8 Input Box 886
7.3.9 Label 888
7.3.10 List View 890
7.3.11 Panel 892
7.3.11.1 Drawing Canvas 893
7.3.11.2 VNC Panel 894
7.3.11.3 XY Panel 896
7.3.11.4 Multitouch Panel 899
7.3.12 Picture Box 911
7.3.13 Playlist WD 913
7.3.14 Shape 919
7.3.15 Sticky Note 921
7.3.16 Textbox 923
7.3.17 Text Input 926
7.3.18 Timecode 928
7.3.19 Tree View 929
7.3.20 Video Player 932
7.3.21 Web Browser 934
7.4 Nodes 936

7.4.1 Creating and Setting up Nodes 937
7.4.2 Tutorial: Nodes 939
7.4.3 Syntax TCP- / UDP- / Serial Messages 944
7.4.4 Input Nodes 952
7.4.4.1 Connections Input 952
7.4.4.2 Devices Input 977
7.4.4.3 File System Input 1035
7.4.4.4 Generic Input 1038
7.4.4.5 Pandoras Box Input 1051
7.4.4.6 Tools Input 1067
7.4.4.7 Widgets Input 1075
7.4.5 Filter Nodes 1092
7.4.5.1 Add Relative Filter 1094
7.4.5.2 Angle to Point Filter 1095
7.4.5.3 Angle Distance Filter 1096
7.4.5.4 Angle/Distance to X/Y Filter 1097
7.4.5.5 Compare Filter 1098
7.4.5.6 Condition Filter 1099
7.4.5.7 Damping Filter Nodes 1100
7.4.5.8 Delay Filter 1103
7.4.5.9 Delta Filter 1104
7.4.5.10 Dynamic Trigger Filter 1105
7.4.5.11 If Filter 1106
7.4.5.12 Image Pixel Reader Filter 1107
7.4.5.13 Is In Bounding Box Filter 1108
7.4.5.14 Is In Range Filter 1110
7.4.5.15 Jitter Reduce Filter 1111
7.4.5.16 Math Filter Nodes 1113
7.4.5.17 Max Filter 1138
7.4.5.18 Min Filter 1139
7.4.5.19 Polar>Rectangular Filter 1140
7.4.5.20 Prediction Filter 1141
7.4.5.21 Range Asymetric Filter 1143
7.4.5.22 Range Curve Filter 1144
7.4.5.23 Range Filter 1147
7.4.5.24 Ranges Filter 1148
7.4.5.25 Text Filter Nodes 1150
7.4.5.26 Vector Filter Nodes 1166
7.4.6 Output Nodes 1177
7.4.6.1 Info Output 1177
7.4.6.2 Connections Output 1178
7.4.6.3 Devices Output 1195
7.4.6.4 File System Output 1205
7.4.6.5 Generic Output 1207
7.4.6.6 Pandoras Box Output 1210
7.4.6.7 Widgets Output 1226
7.4.7 Script Nodes 1237
7.4.7.1 Action Script 1237
7.4.7.2 COM Scripts 1239
7.4.7.3 UDP Scripts 1240
7.4.7.4 TCP Scripts 1242
7.4.7.5 Gesture Scripts 1243
7.4.8 Interaction Nodes 1244
7.4.8.1 Layer UV Draw to Canvas 1245
7.4.8.2 Mouse Draw to Canvas 1249
7.4.8.3 Mouse Injection 1250
7.4.8.4 Touch Injection 1251
7.4.8.5 WatchFolder to PB Playlist 1252
7.4.9 Composite Nodes 1254
7.5 Connections 1255
7.5.1 PB Network Configuration 1256

7.5.2 Connection Manager 1258
7.5.2.1 SMPTE Link 1258
7.5.2.2 Art-Net 1259
7.5.2.3 DMX Link 1260
7.5.2.4 MA Net Input 1260
7.5.2.5 Midi 1260
7.5.2.6 WD Remote Touch 1260
7.5.2.7 TUIO Transmitter 1261
7.5.2.8 TUIO Receiver 1261
7.5.2.9 PSN MA Tracking Server 1262
7.5.2.10 Optitrack Camera Manager 1262
7.5.2.11 ID Tag Configuration 1263
7.5.2.12 IP Info and Setup 1264
7.5.2.13 TCP Connections 1265
7.5.2.14 UDP Connections 1267
7.5.2.15 Serial COM Connections 1269
7.5.3 Midi Connections 1271
7.5.4 TCP/UDP/COM Connections 1272
7.5.5 Art-Net Monitor WD 1272
7.5.6 Remoting 1273
7.5.6.1 TCP Server 1274
7.5.6.2 UDP Listener 1274
7.5.6.3 HTTP Listener 1274
7.5.6.4 Network Broadcasting Service 1274
7.5.6.5 IPhone WD Remote 1275
7.5.6.6 Serial Input 1277
7.6 Devices 1277
7.6.1 AirScan 1277
7.6.1.1 General Settings 1278
7.6.1.2 Input Point Processing 1279
7.6.1.3 Calibration 1280
7.6.1.4 Point Tracking 1281
7.6.1.5 Mouse Control 1283
7.6.2 Kinect 1283
7.6.3 Face Tracker 1287
7.6.4 Motion Detector 1289
7.6.5 Camera Tracker 1291
7.6.5.1 General Settings 1292
7.6.5.2 Video Settings 1293
7.6.5.3 Color Processing 1294
7.6.5.4 Image Processing 1294
7.6.5.5 Point Tracker 1296
7.6.5.6 Touch Output Processing 1297
7.6.5.7 Canvas 1297
7.6.6 Modbus 1298
7.6.6.1 Modbus Master Configuration 1299
7.6.6.2 Modbus Patch Configuration 1300
7.6.6.3 Using Modbus 1302
7.6.6.4 Modbus Master List 1304
7.6.7 Configuration 1305
7.6.8 Barco 1306
7.6.8.1 Barco ImagePro2 1306
7.6.8.2 Barco MatrixPro2 1314
7.6.9 Christie 1318
7.6.9.1 Phoenix 1319
7.6.9.2 Spyder 1322
7.6.9.3 Terra 1332
7.6.10 Event Listener 1353
7.6.11 Fader Extension 1354
7.6.11.1 Fader Extension Events 1355
7.6.11.2 Fader Extension Members 1356

7.6.12 Jog/Shuttle 1364
7.6.12.1 Jog/Shuttle Events 1365
7.6.12.2 Jog/Shuttle Members 1366
7.6.13 Lightware Matrix LW2 1378
7.6.13.1 Lightware Matrix LW2 Events 1379
7.6.13.2 Lightware Matrix LW2 Members 1379
7.6.14 Phidgets 1383
7.6.14.1 Phidget Control Panel 1385
7.6.14.2 Phidgets - Digital Input 1388
7.6.14.3 Phidgets - Digital Output 1393
7.6.14.4 Phidgets - InterfaceKits 1399
7.6.14.5 Phidgets - PhidgetIR 1414
7.6.14.6 Phidgets - IR Code Table 1418
7.6.14.7 Phidgets - PhidgetRFID 1422
7.6.14.8 Phidgets - Server 1425
7.6.14.9 Phidgets - Servo 1428
7.6.14.10 Phidgets - Sound Phidget 1431
7.6.14.11 Phidgets - Spatial 0/0/3 1434
7.6.14.12 Phidgets - Spatial 3/3/3 1436
7.6.14.13 Phidgets - Stepper 1439
7.6.14.14 Phidgets - Temperature Phidget 1450
7.6.14.15 Phidgets - Voltage Input 1452
7.6.14.16 Phidgets - Voltage Ratio Input 1460
7.6.15 PowerPoint 1468
7.6.15.1 PowerPoint Events 1470
7.6.15.2 PowerPoint Members 1471
7.6.16 TCP Client 1471
7.6.16.1 TCP Client Events 1472
7.6.16.2 TCP Client Members 1473
7.6.17 UDP Client 1475
7.6.17.1 UDP Client Events 1476
7.6.17.2 UDP Client Members 1477
7.6.18 UDP Server&Client 1479
7.6.18.1 UDP Server&Client Events 1480
7.6.18.2 UDP Server&Client Members 1480
7.7 Tools 1483
7.7.1 Events 1483
7.7.2 Keyboard Input 1483
7.7.2.1 Shortcut Editor 1484
7.7.2.2 Logitech Presenter 1485
7.7.3 Midi Input 1485
7.7.3.1 Midi Notes Editor 1486
7.7.3.2 Midi Message Editor 1487
7.7.3.3 AKAI APC40 1488
7.7.3.4 Behringer BCF 2000 1489
7.7.4 Blacklist 1490
7.7.5 Email Settings 1490
7.7.6 RSS Settings 1493
7.7.7 SMS Settings 1494
7.7.8 Projector Recalibration 1496
7.7.8.1 Recalibration Hardware Setup 1497
7.7.8.2 Recalibration Software Setup 1498
7.7.9 Pandoras Box 1503
7.7.9.1 Add Layers to Pandoras Box 1503
7.7.9.2 Array Align Layers 1504
7.7.9.3 Circle Align Layers 1505
7.7.9.4 Cue Generator 1506
7.7.10 Video Logger 1507
7.7.11 Image Resource Manager 1509
7.7.12 Options 1510
7.8 Scripting 1511

7.8.1 Script Language 1511
7.8.1.1 Parameter List 1514
7.8.1.2 Command List 1520
7.8.1.3 Programming Statements 1886
7.8.1.4 Mathematical Expressions 1894
7.8.1.5 Scripting Cheat Sheet V6 1896
7.8.2 Functions and Macros 1897
7.8.3 Variables 1900
7.8.4 Object and Member Notation (dot syntax) 1904
7.8.4.1 Project and Context Member 1906
7.8.4.2 Session and Session Value 1913
7.8.4.3 Data Type Specific Members 1914
7.8.4.4 Math Object 1924
7.8.4.5 Using JSON 1926
7.8.4.6 Node Commands 1928
7.9 Web Server 1929
7.9.1 Group Values 1933
8 Hardware and Accessories 1937
8.1 Server Hardware 1939
8.1.1 LCD Menu 1942
8.2 Compact Player 1944
8.3 Input and Output Cards 1947
8.3.1 Graphics Card Settings 1948
8.3.1.1 Changing the Resolution 1949
8.3.1.2 Multiple Displays 1952
8.3.1.3 Setting up Frame Lock 1961
8.3.1.4 Graphics Driver 1967
8.3.2 3G-SDI Input Cards 1969
8.3.3 12G-SDI Input Cards 1972
8.3.4 Flex Input Cards 1975
8.3.5 HDMI Input Card 1978
8.3.6 Sync Card 1981
8.3.7 ADAT Sound Cards 1982
8.3.8 Dante Audio Card 1986
8.3.9 Ethernet Card 1987
8.4 AirScan Hardware 1988
8.5 Controller Boards 1991
8.5.1 Jog/Shuttle Control 1991
8.5.2 Fader Extension 1993
8.5.3 Jog/Shuttle Control (Discontinued) 1994
8.5.4 Playback Extension (Discontinued) 1995
8.6 DMX Link 1999
8.7 SMPTE Link 2000
8.8 NET Link 2000
8.8.1 NET Link Hardware 2002
8.8.2 NET Link Boards and Examples 2004
8.8.3 NET Link Software 2008
8.9 Discontinued Accessories 2009
8.9.1 DMX Link 8 2009
8.9.2 DVI Input Cards (Discontinued) 2014
8.9.3 DVI and SDI Processor 2017
8.9.3.1 DVI and SDI Processor Hardware 2018
8.9.3.2 Software - Quad Setup Menu 2019
8.9.3.3 Software - Create New Timing 2024
8.9.4 EDID Link 2026
8.9.4.1 EDID Link Hardware 2027
8.9.4.2 Software - EDID Link Editor 2028
8.9.5 MADI Sound Card (Discontinued) 2031
8.9.6 Sensor Link 2033
8.9.6.1 Technical Features 2033
8.9.6.2 Connecting Encoders and Sensors 2034

8.9.6.3 User Interface 2037
8.9.7 Serial Link 2044
9 Tools and Other Software 2050
9.1 Art-Net Monitor 2051
9.2 CITP Desktop Streamer 2052
9.3 Dome Master 2055
9.3.1 Single Image Conversion 2057
9.3.2 Batch Folder Conversion 2058
9.4 Factory Reset - Acronis True Image 2060
9.5 Image Converter 2069
9.5.1 Single Image Conversion 2070
9.5.2 Batch Folder Conversion 2071
9.5.3 Image Converter Settings 2075
9.6 Leica 3D Disto 2076
9.7 Matrix Patcher 2077
9.7.1 Matrix Patcher User Interface 2077
9.7.1.1 File Menu 2078
9.7.1.2 Edit Menu 2079
9.7.1.3 Toolbar 2081
9.7.1.4 Fixture Library 2082
9.7.1.5 Fixture Navigation Tools 2083
9.7.1.6 Fixture Patch 2083
9.7.1.7 Patch Settings 2085
9.7.2 Fixture Editor 2085
9.7.2.1 File Menu 2087
9.7.2.2 Fixture Settings 2087
9.7.3 Patching Guide 2089
9.7.4 Remapping Guide 2093
9.8 PB Menu 2097
9.8.1 Utilities Menu 2099
9.8.2 Settings Menu 2101
9.9 Server Management Application 2109
9.9.1 Menu Bar 2110
9.9.2 Pages and Workspace 2116
9.9.3 Information Panel 2118
9.10 Splitter 2120
9.10.1 File Menu 2121
9.10.2 Edit Menu 2123
9.10.3 Toolbar and Window 2125
9.10.4 Splitter Workflow 2127
9.11 Warper 2129
9.11.1 General 3D Modeling Terms 2131
9.11.2 Warper User Interface 2136
9.11.2.1 Menu Bar 2137
9.11.2.2 Tools Bar 2149
9.11.2.3 Tabs 2151
9.11.3 Warping Guide 2166
9.11.3.1 General Warping Workflow 2166
9.11.3.2 Using an Object in Pandoras Box 2168
9.11.3.3 Warping with Several Meshes 2170
9.11.3.4 Live Warping 2174
9.11.4 Keyboard Shortcuts 2176
9.12 Third Party Software 2177
9.12.1 gmax 2178
9.12.1.1 Software Setup 2178
9.12.1.2 Object Creation and Export Using gmax 2179
9.12.1.3 Warping with gmax 2189
9.12.1.4 gmax Keyboard Shortcuts 2197
9.12.2 3ds Max 2197
9.12.3 Blender 2198
9.12.3.1 Object Creation and Export Using Blender 2198

Christie
Pandoras Box

Part 1

Hazard Warning

1 Hazard Warning

Dieser Warnhinweis ist Bestandteil des erworbenen Pandoras Box Produktes.

This warning note is part of the Pandoras Box product that you have purchased.

Cet avertissement fait partie du produit Pandoras Box que vous avez acheté.

Gefahrenhinweise:
Betreiben Sie das Gerät nur in Innenräumen und vermeiden Sie den Einfluss von Feuchtigkeit,Staub sowie Sonnen- oder andere Wärmebestrahlung. Öffnen Sie das Gerät nicht, es enthält keine durch den Anwender zu wartenden Teile.

Hazard information:
This device is to be operated indoors only and to be kept away from the influences of humidity,dust and sunshine or other radiating heat sources. Do not open the device. It does not contain any parts to be maintained by the user.

Avertissements de danger:
Installez l'appareil dans un endroit à l'abri de la chaleur, de l'humidité et de la poussière. N'exposez pas l'appareil directement aux rayons solaires ou à des sources de chaleur telles que radiateurs, fours, etc. Ne pas ouvrir l'appareil, il contient aucun composant a maintenir par l'utilisateur.

Vorsicht!
Explosionsgefahr bei unsachgemäßem Austausch der Batterie. Ersatz nur durch den denselben oder einen vom Hersteller empfohlenen gleichwertigen Typ. Entsorgung gebrauchter Batterien nach Angaben des Herstellers.

Caution!
Risk of explosion if battery is replaced by an incorrect type. Dispose of used batteries according to the instructions.

Attention! Il y danger d'explosion s'il y a replacement incorrect de la batterie. Remplacer uniquement avec une batterie du meme type ou d'un type équivalent recommandé par le constructeur. Mettre au rebut les batteries usagées conformément aux instructions du fabricant.

Verbrauchte Batterien gehören nicht in den Hausmüll! Entsorgen Sie diese in Ihrer örtlichen Batteriesammelstelle!

Used batteries are not to be disposed of with the house-hold waste! Please dispose them at your local battery collection point!

Ne pas jeter les piles aux ordures menageres (loi relative aux piles et accu). Veuillez rapporter vos piles usagees dans un centre de collecte agree.

Entsorgungshinweis:
Gerät nicht im Hausmüll entsorgen! Elektronische Geräte sind entsprechend der Richtlinie für Elektro- und Elektronik-Altgeräte über die örtlichen Sammelstellen für Elektronik-Altgeräte zu entsorgen.

Instructions for disposal:
Do not dispose off the device as part of household garbage! Electronic devices are to be disposed of in accordance with the guidelines concerning electrical and electronic devices via the local collecting point for old electronic devices.

Instruction pour l' élimination des dechets:
Ne jetez pas l'appareil en fin de vie avec les ordures ménagères, mais déposez-le à un endroit prévu à cet effet par les pouvoirs publics pour son recyclage.

Dieses Gerät darf nur unterhalb von 2000m Höhe genutzt werden.

This device may only be used lower than 2000m altitude.

Ce produit est seulement utilisable à une altitudes inférieures à 2000m.

Dieses Gerät darf nicht in Tropenregionen genutzt werden.

This device may only be used in non-tropical regions.

Ce produit est seulement utilisable dans les régions non tropicales.

Christie
Pandoras Box

Part 2

Conformity and Electrical Information

2 Conformity and Electrical Information

If you are interested in a declaration of conformity and/or compliance statement for our Christie Pandoras Box hardware, please contact the support team or your local dealer.

Please let us know:
- which hardware you address (e.g. Compact Player or Server, SMPTE Link, NET Link or software dongle) and
- which country you need the document for.

Christie Pandoras Box hardware is certified for all major regions. Depending on the hardware, we might also provide certificates for various specific countries.

Pandoras Box Server and Player Conformity and Electrical Information

The Pandoras Box Server and Player devices are compliant with both safety and EMC regulations.

AC input:
100VAC / 240VAC
12 / 6A (max.)
50/60Hz

Warning! This is a class A device. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Pandoras Box Compact Player Conformity and Electrical Information

The Pandoras Box Compact Player is compliant with both safety and EMC regulations as well as WiFi approvals for various regions.

DC input:
19.5V / 120W with an external power supply unit

Warning! This is a class A device. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Christie
Pandoras Box

Part 3

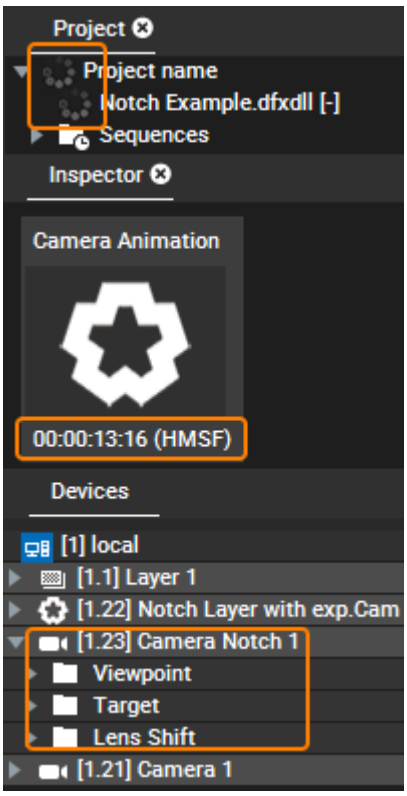
What's New?

3 What's New?

This "What's New" chapter lists updates from Pandoras Box including its tools. For updates regarding Widget Designer [47], please go to the next chapter.

Version 8.3.0 - July '21

++ Notch Updates ++



Pandoras Box version 8.3.0 includes many updates regarding the Notch integration. When working with the Notch feature in Pandoras Box, please install the Notch Builder on the same system.

1) When adding a Notch block to the Project tab in Pandoras Box, an icon let's you know whether the block is still loading.

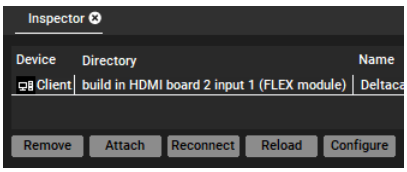
2) The Notch option "Layers as separate Effects" can now be left deactivated as Pandoras Box 8.3.0 can read those blocks too. In this case, all layers in Notch are baked into one scene. The Project tab in Pandoras Box will show only one sub resource. Assign it to a Site in the Device Tree and the resulting Notch Layer will show all parameters that are exposed parameters, no matter which Layer they belong to in Notch. Accordingly, when saving the active value(s) to the timeline, you will have only one Container that combines all scenes from Notch.

3) Pandoras Box retrieves the length of individual clip containers of Notch layers as well as the maximum length of the block. After loading the Notch block to the Project tab, simply select a sub resource to see the duration in the Inspector. After assigning it to a Site in the Device Tree and saving the active values the Containers in PB automatically adopt to that duration. If the Notch block is exported with combined layers (see above), Pandoras Box retrieves the length of the resulting Container. Please use Notch Builder version 0.9.23.195 or above for this feature.

4) We completely overhauled how Notch Cameras are handled. If your scene includes an Exposed Camera (with exposed parameters) you now have the option to add a Notch Camera layer in Pandoras Box. Notch cameras can be controlled like regular PB cameras and can be stored to the timeline of course. Further, you can select a Notch camera in the Preview to interactively move through the Notch scene

More information about Notch in Pandoras Box [671]...

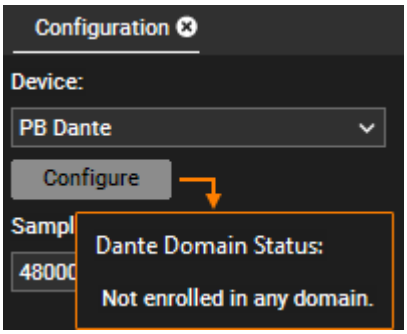
++ Configuration of Live Inputs ++



The Pandoras Box Master can now configure Deltacast video live inputs, including all boards sold by Christie. There is no need to access the Client interface anymore in order to do so. Simply add the video input to the Project tab [278] and click the "Configure" button down in the Inspector tab [195] .

Depending on the type of board, the dialog shows other settings. Currently, DisplayPort and HDMI inputs can toggle the chroma subsampling mode whilst SDI inputs have no configurable settings. If your Client is equipped with another DirectShow board, please configure it through the Client interface [317].

++ Dante Domain Manager ++



With this release Pandoras Box supports the Dante Domain Manager, which can be used to configure Domains for your Dante network. The Configuration tab > ASIO Audio [166] in Pandoras Box displays the Domain in the "Configure" dialog.

Version 6.5.1 - July '21

++ Bug Fix Release ++

For users who have not updated to version 8 yet, this version includes important bug fixes when working with multiple text inputs, 8 channel audio files or older Particle Systems. Further, we added Softedge effects [611] again, which were not included in the previous version.
As always, bug fixes can be tracked in the "changelog.txt" file in the installation folder under `C:\Program Files\Christie\Pandoras Box ...`

Version 8.2.0 - May '21

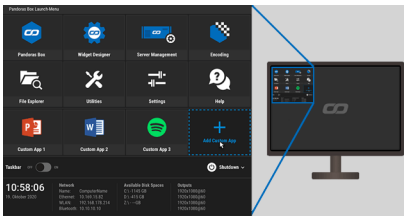
++ Bug Fix Release ++

This release takes care of many bug fixes and smooths its overall usage. It especially removes a performance glitch which was introduced with the new blend modes and resolves the issue where a drag from the scroll bar would be carried on in the Project or Device Tree.

As always, bug fixes can be tracked in the "changelog.txt" file in the installation folder under `C:\Program Files\Christie\Pandoras Box ...`

PB Menu and Server Management

++ New PB Menu ++



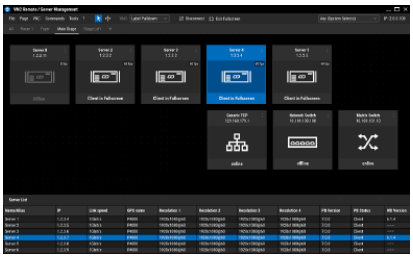
We updated the software "PB Menu" [2097] which is pre-installed on all Pandoras Box hardware but can also be installed on custom hardware. PB Menu allows for quick access to most needed applications, shows system information and publishes those in the network.

"PB Menu 7.0" has an entirely new look and overhauled clear structure. There is a new focus on security settings : We introduce a default password (PandorasBox) for the VNC connection and removed the FTP server. Further, we discontinue deprecated network protocols to better fit in larger network setups. And testpattern now include information like the IP address, Output ID and computer name.

You can find a new button to start the "Server Management" which replaces the former VNC Remote.

The PB Menu can be found in our Download-Center and is described in the chapter PB Menu [2097].

++ New Server Management Application ++



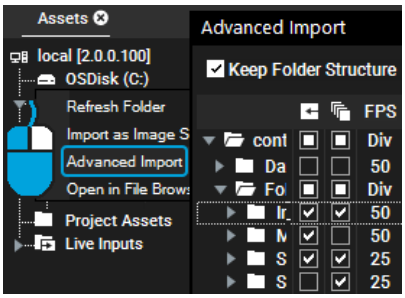
The "Server Management" application replaces the former (VNC) "Remote" software. It is part of the above described "PB Menu" software but can also be installed separately. Its main focus is is to manage multiple hardware systems and to keep track of their status.

We redesigned the Server Management App and overhauled the structure of the user interface. You can now arrange devices on separate pages and their properties (resolution, frame rate, version and status of PB and WD, etc.) are displayed permanently in a table in the lower part of the UI. Of course, you can still easily connect via VNC to appropriate remote systems, however we added a password for security reasons ("PandorasBox", see PB Menu > Settings > VNC Server [2103]).

The Server Management can be found in our Download-Center and is described in the chapter Server Management Application [2109].

Version 8.1.0

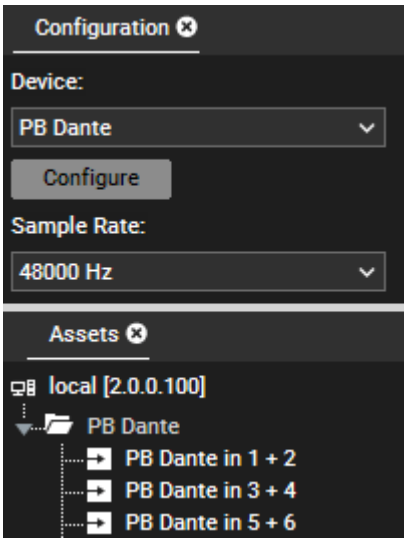
++ Advanced Import Option ++



The new dialog "Advanced Import" facilitates ingesting extensive amounts of videos, image sequences or still images to your project at once. Batch-handling and assigning settings before importing files saves valuable time, especially within content-heavy shows. The automated content handling, including the spread of files to remote systems allows the operator to focus on programming.

In the Assets tab, simply right-click the main folder with all your content and choose "Advanced Import". A dialog opens, analyzes the content and shows all sub folders. Click the first check box "Import" for each folder you would like to import. A second option allows to import the folder as an image sequence or as normal images. Further, you can set the frame rate, threads, and naming option. All changes apply automatically to containing sub folders but can be individualized if necessary. Read more about this feature and its settings in the chapter "Assets" > Advanced Import [132].

++ Audio Input via Dante ++



With the last version we introduced our new Dante card [1986] and the option to output Dante (i.e. digital audio over IP) in Pandoras Box. Now it is also possible to receive Dante audio as a Live Input. Pandoras Box Licenses are now natively supporting 32 input channels and 32 output channels without the need of any hardware, driver configuration or licensing.

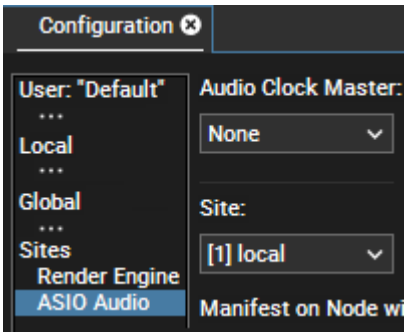
To input Dante streams, first open the Configuration tab and choose the ASIO Audio tab [166] from the new "Sites" category. Select the Site that should be receiving the stream.
Choose "PB Dante" in the "Device" list and click the "Configure" button. In the dialog, choose which "Network Adapter" should be used for Dante and check the number of input and output channels. Lastly, check the "Sample Rate".

Then, go to the Assets tab [131] and open the folder "Live Inputs > PB Dante" and choose a Dante stereo input from the PB system with the Dante device drag it into the Project tab [278]. Per default, you will see 32 inputs (16 stereo channels). If you have entered less channels, only the according inputs will be active. If you have chosen more channels, you might need to refresh the list to see more inputs.
Now you can assign an input to an Audio Track [661] from the PB system with the Dante device.

As demanded by Dante, you need a Dante Master in the network used for transmitting Dante. The Dante Master unlocks the Dante functionality per se, generates a clock and syncs the signals. The Dante Master can be any Dante interface or Dante audio card for example. The "Dante Controller" software can be used to assign the Master functionality, route signals, match sample rates for sources and outputs, etc.

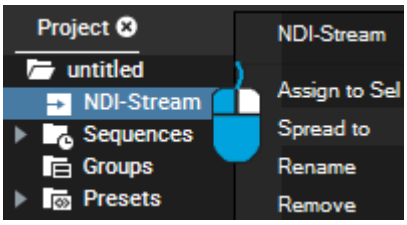
For using more than 32 channels, we recommend to separate the network traffic from the show network and use a dedicated Dante card. The Dante card offered by Christie [1986] allows to use 256 channels in total. As said above, it can be a Dante Master.

++ Restructured Configuration Settings ++



The Configuration tab has a new category "Sites" which shows settings regarding the Render Engine [162] and ASIO Audio [166]. The "Render Engine" tab has not changed, it was just moved. Like this tab, the "ASIO Audio" tab now also offers a drop-down list to select either a local or remote Site.

++ NDI is Spread-able and Supports Dynamic Resolution++



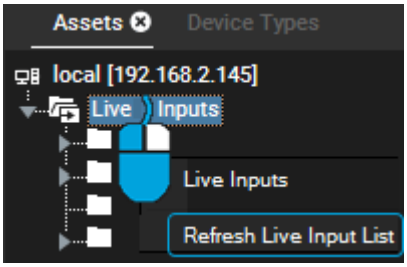
Pandoras Box V8 introduced the support of NDI® streams. With this version it is easier to use the same NDI stream(s) on multiple systems because you can now spread NDI sources after adding them to the project. This applies to Clients sharing a stream but also to a Master previewing a stream. As explained in more detail in the chapter "Assets tab" [133], NDI streams might not appear synchronized on multiple systems because of different transmission times via the network and decoding times.

Of course, all PB systems need to be able to receive the stream directly from your NDI device, mobile app, software application etc. Keep in mind that Pandoras Box just receives the stream, it is not transmitting it to others.

Furthermore, a constant resolution for NDI streams is not required anymore. This allows using applications where the stream's resolution changes dynamically, e.g. due to network issues. The Inspector now offers "Texture Width" and "Height" fields for NDI streams. When adding an NDI stream to a PB project, the texture size adopts to the initial incoming resolution and stays even if the resolution changes later. The texture size adopts only, if the "Reload" button is clicked, or if other values are entered manually.

NDI® is a registered trademark of NewTek, Inc. For more information about NDI itself, NDI tools etc., see http://ndi.tv/

++ Live Inputs: Refresh Option ++



The "Live Inputs" folder in the Assets tab now also offers the option to refresh the list. This is especially of interest when receiving NDI streams as the network adapter needs some time to identify NDI sources.

Version 8.0.0

++ New License Model ++

We have fundamentally simplified the license structure. It has never been so simple and yet powerful as it offers the full feature set of Pandoras Box to everyone. The "Pandoras Box Software License" replaces six products: Server, Player, Compact Player, Software Player, Manager and Educational License. Next to it, there is still the "Offline License" and Demo version. The chapter Product Overview [62] explains the licenses in detail.

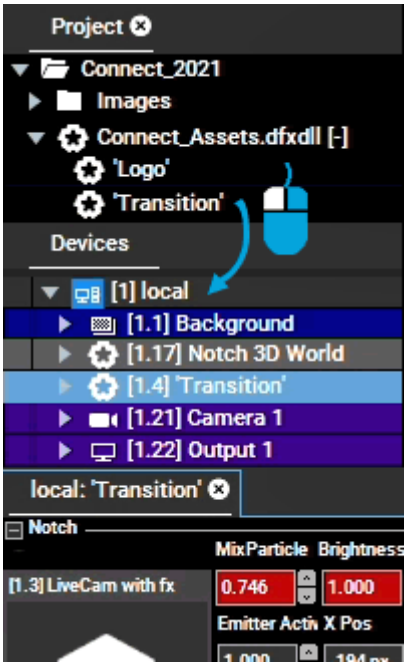
The power of the Server license is now combined with the Manager features. The new license allows 3D rendering, playback beyond 4K and support of 10bit color depth whilst offering unlimited sequences, layers and effects. In addition, this license can run on any hardware system, custom hardware included. In other words, features previously only available with hardware/software combinations now become accessible to everyone.

What's been previously known as the "Manager" (or alternatively the master mode of a Server or Player license) is now the "Master" mode of the new license. To add a Client to it, start a second license in "Client" mode on another system. The Master has two configurations: when running it without Output Layers, it can start a Multi-User session or join one. If you on the other hand add Output Layers, you can use your Output(s) in fullscreen mode.

Multiple licenses can be stacked by combining dongles or they can be applied to a single dongle. The number of licenses is linked to the available number of Output Layers which can be linked with any Camera Layer. If your Site in a project contains four Output Layers, it can only manifest on a PB system with four licenses. If it holds only three, a dialog offers to remove one Output Layer.

Further, we worked on the Layer Structure. As everybody has now access to unlimited Video Layers, we removed Graphics and Text Layers. As described below, there is a new layer type: the Notch Layer.

++ Notch Integration ++



Notch is a real-time content creation tool that brings a combined work flow for creatives and technologists. Notch enables artists and designers to create high-end 3D graphics, live video effects and particles, interactive experiences, virtual production and a lot more in a true real-time environment. The deep and comprehensive integration to Pandoras Box gives you frame accurate playback and full parameter control with zero latency which allows you to edit content directly on stage and over-night renders belong to the past.

In the Notch Builder you first design your scene and then decide which parameters should be exposed in the so called Notch block that will be loaded in Pandoras Box. The exposed parameters can be everything from text, amount, color, to size or position etc. If you program items to the Notch timeline, they will be synchronized to the PB timeline when storing them as a container.

In Pandoras Box, you first select the Notch block, the dfxdll file, in the Assets tab and drag it into the Project tab. The first loading process takes some time. Depending on your Notch composition this can be a few seconds or even minutes when using many textures etc. When finished loading, you can open the Notch entry in the Project tab and see its sub elements which are the layers in Notch. Now, simply drag one on a Site in the Device Tree to add a Notch Layer to it, this is a new layer type. This Site needs to be equipped with a Notch license to be able to render the content.

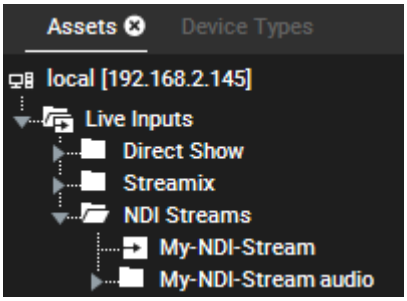
The "Media" parameter of the new Notch Layer becomes an active parameter. When you execute the "Store Active" command, you will see a new Container in the Sequence. When you play it back, this could look like a

2D video or transition or like an entire 3D scene with camera animations and particles reacting on content. Of course you can blend other content and live feeds with the Notch content. Or, go a step further and combine both by assigning PB content to an exposed Notch Media parameter. The new Blend Modes or LUT shaders that come with PB version 8 are very helpful for that as well as the new Share Layer Option, all described below 26.

In the Device Controls tab, you will see other parameters known from a Video Layer but also dedicated Notch parameters that were exposed before using Notch. Just as with PB parameters, they can be changed and stored as key frames on the timeline. They can also be controlled from Widget Designer 786, e.g. through interactive devices like the AirScan 1988 or sensors like Phidgets 1383 and more. Examples can be found in the chapter "Notch Layer 671".

As Notch renderings can demand much GPU performance, we recommend to equip the PB system with the Notch license with one of the newly offered RTX graphics cards.

++ NDI Support ++



Pandoras Box V8 features the support of multiple NDI streams including embedded audio streams. NDI® (Network Device Interface) provides separated video and audio network streams via uni- or multicast. Besides many devices such as production switchers or cameras which are able to output NDI feeds directly, there are numerous applications for computers and apps for mobile devices available.

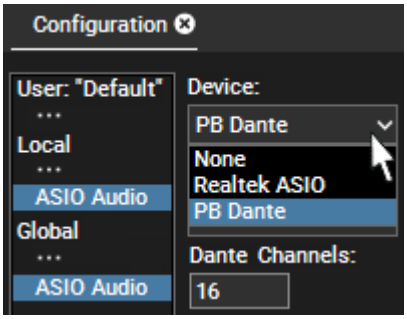
To use NDI in Pandoras Box, go to the Assets tab 131 to open the "Live Input" folder from a PB system and you will find a new folder "NDI Streams". Once an NDI input is added to the project and assigned to a Layer, the according PB system subscribes to the stream and displays the content.

Pandoras Box supports all NDI variations including different compressions (i.e. NDI and NDI HX) with a resolution of up to 4k@60 and the alpha option which needs to be activated in the Inspector 190. All Live Inputs are analyzed once, hence it is not recommended to change the resolution after adding the source to the project.

The NDI feature is described in the chapter "Assets tab" under "... Live Inputs" 133.

NDI® is a registered trademark of NewTek, Inc. For more information about NDI itself, NDI tools etc., see http://ndi.tv/

++ Dante Support and Dante Audio Card++



Dante became the de facto standard for digital audio over IP. Pandoras Box Licenses are now natively supporting 32 output channels without the need of any driver configuration or licensing. Dante can be activated on-the-fly anytime needed and is available right away.
Dante distributes high-quality, low-latency media signals with perfect synchronization over existing IT data networks. Dante is a scalable, reliable and easy to use media network solution which does not require dedicated network infrastructure but runs on off-the-shelf computer networking hardware.

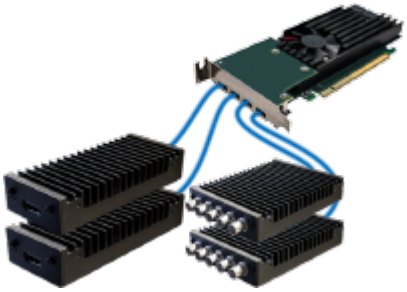
To output Dante streams, open the Configuration tab and the ASIO Audio tab 166 (from Local or Global in case you like to stream from another system than the local one).
Choose "PB Dante" in the "Device" list, check the sample rate and choose which network adapter should be used for Dante audio.

As demanded by Dante, you need a Dante Master in the network used for transmitting Dante. The Dante Master unlocks the Dante functionality per se, generates a clock and syncs the signals. The Dante Master can be any Dante interface or Dante audio card for example. The "Dante Controller" software can be used to assign the Master functionality, route signals, match sample rates for sources and outputs, etc.

Lastly, in the Configuration tab, check the number of Dante channels. Then assign an audio file [92] or live source to an Audio Track [661] and check the channel routing in the Track Inspector [217].

For using more than 32 channels, we recommend to separate the network traffic from the show network and use a dedicated Dante card. The Dante card offered by Christie [1986] allows to use 256 channels in total. As said above, it can be a Dante Master.
Another advantage of a Dante audio card is redundancy. As it provides two distinct network connections and features Audinate's patented Glitch Free redundancy, audio continues uninterrupted in the event of a network connectivity issue with the first connection. The card is explained in the chapter "Dante Audio Card" [1986] in more detail.

++ New Flexible Input Card: Flex Card and Modules ++

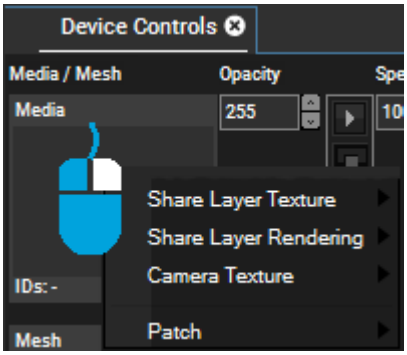


The Flex card is a revolutionary solution for input boards. The new gateway card which is built into your hardware offers four inputs. They can be connected to various external modules which provide the connectors. There are four modules available offering SDI, DiplayPort and HDMI inputs. In total, you can input up to 16 SDI channels (12x 3G or 4x 12G) or alternatively, four 4K60 streams via DisplayPort 1.2 or HDMI 2.0, or a mixture of them.

To get more or different connector types, simply connect another module that offers the needed connectors. This allows to mix various connection types just as your show requires it. There is not need to open the chassis - just shut down the system (never hot plug the modules!) and changing the inputs becomes a matter of seconds. Thus, the FLEX input card is especially of interest for Rental and Staging customers or whenever changing setups are needed. The external modules can be used as floating devices or can be integrated in a 19" rack offered by Deltacast.

Find more information about the Flex card and its modules in the chapter "Flex Input Cards" [1975] or on the website from Deltacast.

++ Share Layer Option Added ++

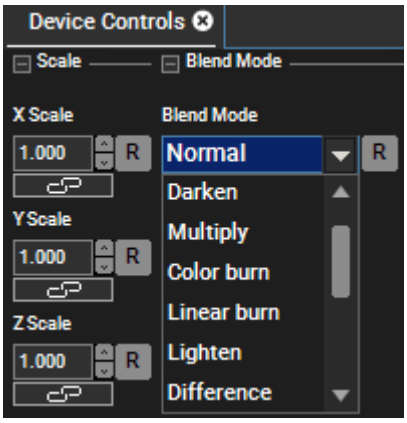


Since a long time Pandoras Box offers the command "Share Layer Texture" (previously only for Server licenses). It loads the same texture which is assigned to the linked layer and has the advantage that a video file (or any other content) does not have to be loaded several times from the hard disk. This allows a better performance and saves programming time.

However, this loads only the texture. If there are effects on the linked layer, they only affect this one layer. For that reason, PB V8 offers a new command **"Share Layer Rendering"** which loads the texture of the chosen Layer and applies its effects.

Right-click the Media field to find both "Share Layer" options.

++ New Blend Modes (also for Particles) ++

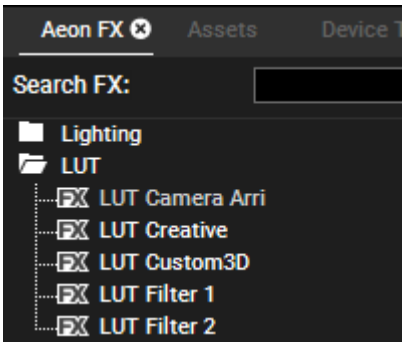


Pandoras Box now offers a total of 20 different blend modes, which allow to blend a layer in all imaginable ways with the background.

Blend modes can be found in all Layers, including Notch Layers and also Particle Layers [185]. In older versions, particles where simply added to the background. With the new blending options, you can now create particles darker than the background and much more.

All blend modes are listed and explained in the chapter "Blend Modes [657]".

++ New Effects: LUT Shader ++

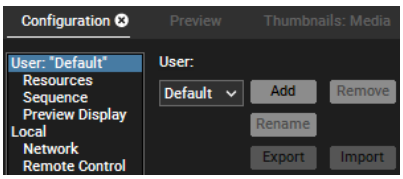


We added new effects which are based on the principle of so called look up tables (short: LUT) and which you might know from Photoshop or other graphic software. LUT shaders are used for technical color correction or artistic color modification.

You can find the new effects in the folder "LUT [544]" from the Aeon FX tab [129]. As with all effects, drag them onto a Layer and set the "Mix" to 255 to see the full effect. Each LUT effect offers a drop-down list from which you can choose an entry. In total, you have access to almost 200 look up tables. However, there is one exception. With the effect called "Custom 3D LUT Filter" you can define the effect by adding a custom image. For this, you can find examples in the Stock Assets folder under Textures > Custom 3D LUTs, they are all 16bit PNG files. Load the "neutral_LUT" file in Adobe Photoshop, apply your color correction and re-import it to Pandoras Box.

++ User Profiles Implemented ++

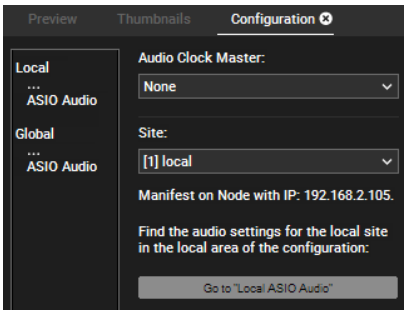
With Pandoras Box V8 you can now save specific settings from the Configuration tab and create one or more User Profiles.



User Profiles are displayed in the Configuration tab [139]. You might already know that the configuration settings are separated into three groups: User, Global and Local. All settings from the first group "User" can now be saved in profiles. The settings influence Resources, Sequences and the Preview Display. Any project has at least one profile named "Default". You can create a new User Profile and save changes in various settings. You can also export and import a profile. An exported profile has the file ending "pup". You can rename all profiles except the Default one; keep in mind that they must have a unique name.

When saving the PB project, the profiles are saved in it. This includes the "Default" profile (for this single project). In addition, profiles are automatically shared with other Users when working in a Multi-User session [234]. Note that the "Default" profile is selected per default, hence changes apply to these instances too. But of course, every User can apply an individual profile. If deleting a profile, it also disappears for other Multi-Users. PB warns you in case you like to delete a profile used by another Mutli-User. The Configuration tab and Multi-User tab display which profile is used by which Multi-User.

++ Restructured Audio Settings ++



The ASIO Audio settings are now separated and located in the "Local" and "Global" category in the Configuration tab. This allows for different audio cards in a Multi-User setup.

On the one hand, the setting regarding the local site can be found in the category "Local" and are also saved locally.

On the other hand, the ASIO Audio [166] settings from the "Global" category allow choosing an audio clock master and shows ASIO audio settings for connected Pandoras Box Clients. When working with other Multi-Users, the global settings are saved within the project and therefore they are also synchronized.

++ v6 Projects Supported / "Export XML" Discontinued ++

As Pandoras Box v8 supports the same project structure and format, there are no limitations in loading older v6 projects.
In those versions PB offered dedicated Graphic and Text Layers. If you used them, they will be loaded as such but you are not able to add more of them. It is recommended to test older projects before running a show with them.
Since version 6 Pandoras Box projects can be saved and loaded faster using the PBB format. This does not change with the introduction of v8. The previous XML format was still supported via the commands "Import project" and "Export XML". The later is now discontinued. However, the Support team can assist you in converting your file for debugging purposes.

R5 Hardware

++ New R5 hardware ++

With Q1/2021 we start shipping our entirely new hardware platform with state-of-the-art components. The R5 hardware replaces both, the Server R4 and Player R4 hardware. Servers are made-to-order and the wide range of performance, drive space and optional cards allow to fit the exact customer's need and budget.
Read more in the chapter "Server Hardware" 1939.

Performance depending on video codec but up to:
4x 4K60 / 12x HD60 uncompressed, 10bit color depth
20x 4K60 / 64x HD60 compressed, 8bit color depth

Available in different performance kits:
PK1: 960GB SSD / 24GB RAM
PK2: 7.5 TB SSD / 48GB RAM
PK3: 32TB SSD / 48GB RAM

Optional graphics cards:
P4000 / RTX 4000 / RTX 6000 (RTX cards are meant for Notch rendering)
Up to 8 physical DisplayPort outputs from a single system

Optional video input cards:
12G/3G SDI, HDMI 2.0, DisplayPort 1.4 Read more... 1947
The FLEX input card or rather gateway card creates a single gateway in the system which can be connected to various boards. Read more... 1975

Optional audio input/output cards:
ADAT 8ch audio
Dante 128x128ch network audio

Version 6.5.1

++ Bug Fix Release ++

This version includes important bug fixes when working with multiple text inputs, 8 channel audio files or older Particle Systems. Further, we added Softedge effects 611 again, which were not included in the previous version.
As always, bug fixes can be tracked in the "changelog.txt" file in the installation folder under `C:\Program Files\Christie\Pandoras Box ...`

Version 6.5.0

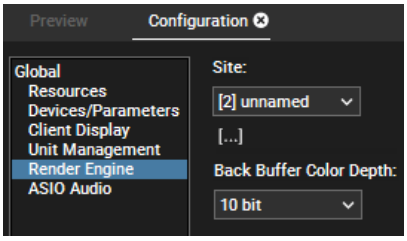
++ New Render Engine ++

With this version, we introduce an absolute new render engine for all Pandoras Box products.

At the first glimpse, you won't notice anything new in the Preview tab or Rendering window but that's just the outside - the inside is a completely new render engine which is based on DirectX 11.

We have reworked, optimized and improved the entire video rendering process from ground up. The advantages you can witness already today are a better performance and a higher color depth (see below) but the future will reveal more.

++ 10bit Color Depth on Servers ++

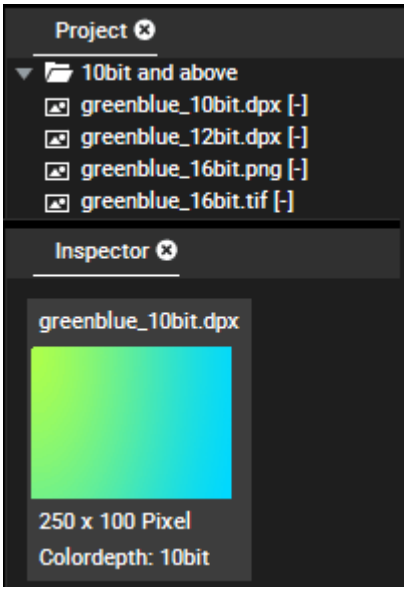


The new render engine sets ground for new features, the first one is the capability of rendering in 10bit color depth. Servers can now be set to this higher bit depth in order to increase the image quality. Uncompressed content can now be shown without any artifacts or color banding. 10bit rendering per color channel allows for a total of 1 billion colors to be displayed, instead of only 16.7 million. More contrast, cleaner color transitions and HDR rendering becomes herewith possible.

In the Configuration dialog > Render Engine [162] you can now find the option "Back Buffer Color Depth" which can be set to 8bit, or 10bit on Servers. As this is available for each Client separately, you can save performance on those Clients which are not connected to 10bit capable displays or projectors.

Christie is now offering 10bit HDR support throughout its entire product offering.

++ Support of DPX File Format and Others Formats with Higher Color Depth ++



In order to render in a higher color depth, the inserted content needs to provide it. Therefore, Pandoras Box supports now PNG and TIFF files with a color depth of 16bit per channel. The color depth information has been added to the File Inspector [191].

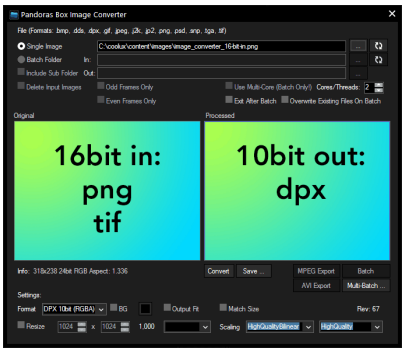
In addition, you can make use of a new image (sequence) format called DPX (Digital Picture Exchange) which is a common format for uncompressed still frames. If you have not heard from DPX, you can think of it as a bitmap format that can be saved with more variety in color depths. Pandoras Box accepts DPX files with a color depth of 8, 10 or 12bit per channel. However, any content will always be rendered in 8 bit or 10bit, according to the render engine. This happens automatically, i.e. no re-rendering is needed.

All image formats are explained in detail in the chapter "Image Formats [93]".

To sum this up: If you render images or image sequences with a color depth of 8bit there is no advantage of using DPX files over BMP files, it is just another possibility.
When rendering in 10bit, you can use 16bit TIFF or PNG files, but the highest efficiency is reached when using 10bit DPX files.

Various QuickTime codecs containing 10bit or more are also supported.

++ Image Converter Tool also Supports DPX ++



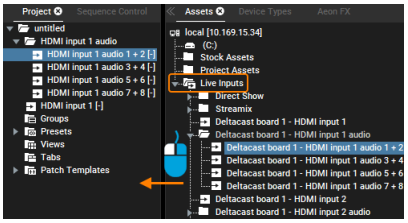
The Image Converter tool is updated to import TIFF and PNG files that where saved with a color depth of 16bit per channel and in addition it can now read DPX files with 8,10,12bit, all formats with or without an alpha channel.

The export functionality has been extended and includes DPX files with 8 or 10bit color depth, with or without transparency.

Thus, the new Image Converter allows to transcode 16bit PNG or TIFF files into 10bit DPX files to be efficiently used in Pandoras Box.

For more information please see the Image Converter chapter 2069.

++ HDMI Audio on Audio Tracks ++



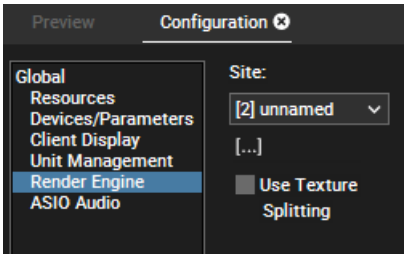
Pandoras Box version 6.5.0 now also allows the usage of all embedded audio channels within the HDMI signal when using the HDMI 2.0 input cards 1978 or 12G-SDI Input Cards 1972.

Similar to SDI audio, the new HDMI audio options are displayed in the Assets tab 131. Simply open the "Live Inputs" folder of the respective Client or Master system and look for the HDMI audio entries. HDMI supports up to eight channels which are displayed as four stereo channels in the Assets tab. Select an entry and drag it to the Project tab 278. From there you can assign it to an Audio Track 661. To playback the audio part you always have to play the video part on a Video Layer simultaneously.

Pandoras Box supports sample rates of 44.1k and 48k. As Inputs are not resampled please make sure that the audio device matches this sample rate (Configuration tab > ASIO Audio) 166. The sample rate of the HDMI audio channel is displayed in the Inspector 191 when selecting it in the Project tab.

More information can be found in the chapter about the HDMI input card 1978.

++ Texture Splitting Option in Configuration Dialog++



This version includes a second new option in the Configuration dialog > Render Engine 162 which is called "Texture Splitting" and is available for each Client separately.

We have seen a higher show performance when we checked this option working ... :
- on our current hardware (Player R4 and Server R4)
- with uncompressed formats
- with high resolutions above HD

As the name suggest, this option splits textures internally which has the effect that the system handles smaller (but more) textures instead of one very large one. As a result, the loading process of large textures gets smoother and thus increases the "show performance", which we state on our Performance Sheets found in the Download-Center. The show performance defines the maximum number of videos which can be started and stopped without interfering other videos.

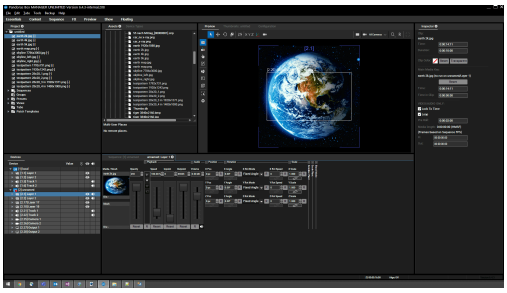
Version 6.4.4

++ Bug Fix Release ++

This version includes important bug fixes for issues which occurred when adding audio files with an unsupported audio format 92 to Pandoras Box.

Version 6.4.3

++ View Bar with New Default Views + Export / Import of Views + New Partial Views ++



With this version a View Bar [119] is added to the top of the user interface showing seven new views. The first, called "Essentials", is our new default view and shows the Inspector on the right side. The others are named "Content, Sequence, FX, Preview, Show, Floating" and emphasize a particular use of the UI. Simply click on a View to change the currently shown tabs and their arrangement.

To change a view to your needs, open the "Views [287]" folder in the Project Tree, right-click the according View and choose "Update". This saves the current layout.

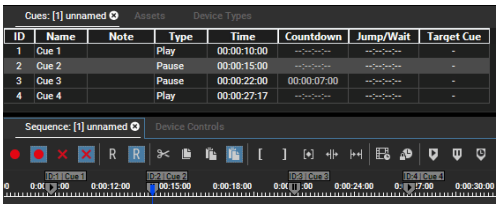
When saving a new layout, you can now choose between two options from the right-click menu of the "Views" folder. With "Create View" you save the entire layout as usual (with a small structural change as explained further down). With "Create Partial View" you are saving only a specific part of the interface. A dialog asks which of the four sub-views should be included:
- "Layout" saves the layout of the UI including the pane arrangement, size, break-out state, which tabs are loaded and what they display. Its pretty much everything except of what the following three partial views save.
- "Preview" saves the layout of the Preview including the mode, number of preview windows and assigned cameras/outputs.
- "Sequence" saves the loaded Sequence ID.
- "Device Tree" saves which devices and sites are hidden or shown.

When creating a View (via "Create View"), all four sub-views can be found as sub entries in the Project tree. They can be applied, updated or removed separately. The seven new default views change only the "Layout" part, so that the currently loaded Sequence, Preview and Device Tree do not change.

You can add a new View to the View Bar to access it faster. You can either drag it there from the Project tab, or right-click it and choose "Add to View Bar" or make a right-click in the View Bar and choose "Add".

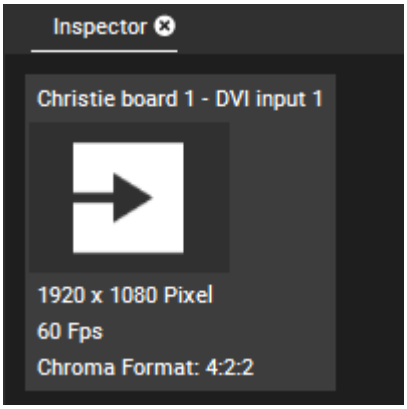
In the View's right-click menu, you can also find the new option to "Export" a view, which saves the "Layout" sub view as a pbv file to the hard disc.
The right-click menu of the main Views folder, offers the corresponding "Import" option. By the way, the default views are saved in the installation path under `data/views`.

++ Revised Cues Tab ++



The behavior of the Cues tab [168] is now more intuitive. When the Sequence is paused at a Cue (manually or via a Pause Cue), the current Cue is highlighted in the Cues tab and the next Cue is listed with a Countdown time. In the example, the Nowpointer is paused at Cue 2 and Cue 3 is 7 seconds away.

++ Inspector Shows Chroma Sub Sampling for Live Inputs ++



The Inspector from version 6.4.3 will now also display the chroma sub sampling for live inputs, e.g. "4:2:2". Since version 6.4.0, you can choose another chroma sub sampling, i.e. RGB 4:4.4, in addition to YUV 4:2:2 when working with HDMI inputs [1978], which was described below [35].

++ New Default Values for Pre Roll and Image Sequence Import ++

The Pre Roll Time, which can be found in the Clip Inspector [206], allows to load content before the Nowpointer enters the Container which is stored in the Sequence. Per default, this time is now 2 seconds. In addition, you can find the Pre Roll option when a Live Input is stored in the Container.

When importing an Image Sequence, the default frame rate is now proposed with 50 FPS.

++ Multi-Selection for Image Sequence Import and Sites ++

From now on, you can multi-select folders in the Assets tab [131], right-click them and choose "Import as Image Sequence". If you don't need to set up individual frame rates, set up your frame rate under Configuration > User > Resources > Image Sequence Initial Values [140] and uncheck the option "Show Popup..."

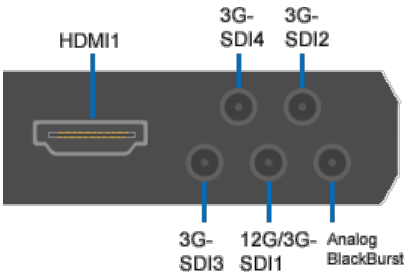
In addition, you can multi-select Sites in the Device Tree [173] and choose "Toggle Preview".

++ QuickTime Supports Embedded Audio ++

From now on, QuickTime files support embedded audio again.

Version 6.4.2 & 6.4.1

++ First Input Card with 12G Solution: 12G/Quad 3G SDI, HDMI/DP Board ++



With this version we introduce a new input card. It is our first 12G input solution and comes with additional inputs to be used as a Quad 3G-SDI board. It is also equipped with an HDMI 2.0 input which can be used for DisplayPort as well. The image shows all inputs from the "12G/Quad 3G SDI, HDMI/DP" board.

Starting with Pandoras Box version 6.4.1 we are supporting the original Deltacast drivers (version 6.14) which we have added to our driver package in the Download-Center.
The 12G-SDI board [1972] can only be used with driver version 6.14 (or later).
The DVI board [2014] (and old HD-SDI board) can only be used with driver version up to 6.01.
The 3G-SDI board [1969] and HDMI board [1978] can be used with both versions.

When having two or more cards in a single computer they have to run with the same driver version. Therefore, a DVI board cannot be used along with the 12G board in a single computer. Using different driver version on separate machines is not a problem.

Furthermore, we have optimized the behavior of live inputs in our software, so that they only use system performance when being assigned to a layer.
As always, bug fixes can be tracked in the "changelog.txt" in the installation folder under `C:\Program Files\Christie\Pandoras Box ...`

Version 6.4.0 - Release

++ PB goes 64bit with more Performance and Better Video Playback ++

Version 6.4 is the first Pandoras Box application in a 64bit environment. This means primarily that it can allocate much more system memory of the operating system, which by itself is not a "visible" but major feature. PB has now more performance in general and especially when it comes to video decoding.

The new environment allows to address more memory so that you can load more layers with large content at a time. Further it can do more processing in parallel. This is why the overall performance increases and is especially visible when playing back various codecs. Obviously, whenever the bandwidth of your hard drive or PCIe bus is the limiting performance factor, this cannot be overcome. But especially with compressed video files (HAP, MPEG-2, etc.), the performance increases substantially on most systems. Depending on the used hardware, even uncompressed videos get a boost as more frames can be simultaneously cached in the system memory now.

We will update our Performance Specs for content playback in the near future in the Download-Center.

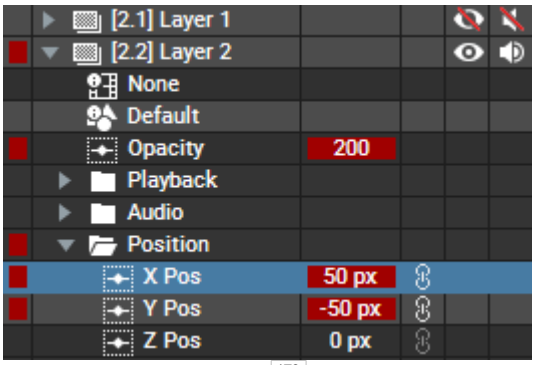
++ New Decoder for QuickTime Videos ++

The QuickTime decoder that was part of the 32bit installations has been replaced with a new decoder. This not only enables you to playback more videos in parallel due to better performance, but also opens up a new world of video codecs. Pandoras Box now easily digests videos encoded with H.264, HEVC, ProRes of multiple flavors or even DNxHD. We will provide a full list of supported codecs and its performance when coming closer to the next release.

In this version, embedded audio in QuickTime files is ignored and the GUI states "no audio" in the Inspector.

Video performance for QuickTime based files now also benefits from hardware acceleration. Depending on your Nvidia GPU, you can hardware decode codecs such as H.264 or HEVC. You will witness a massive CPU usage drop when Pandoras Box automatically activates the hardware acceleration. You can double-check your GPU charts to verify which files can be accelerated, e.g. on https://developer.nvidia.com/video-encode-decode-gpu-support-matrix (the right side of the table).
While the P4000 for example easily accepts H.265 (HEVC) files up to 8K in various chroma sub sampling flavors, the M4000 mainly ceases at H.264 and only supports H.265 (HEVC) in 4K 4:2:0.

++ Redesign of GUI ++



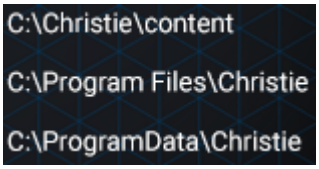
The entire graphical user interface (GUI) comes with a new font, colors and icons and has been cleaned up for a better understanding. Separators as well as headers or frames make it easier to visually gather settings of different topics, for instance in the Inspector.

File and device trees (project tree, asset tree, device tree, patch tree,...) have been rebuilt to include new functionality. The tree item selection now follows Windows defaults and does not maintain selections when single-clicking on an already selected device.

In the Device Tree tab [173], you will now find distinct columns for different information: parameter names, values and various icons are placed separated:
- To maintain readability of layers and parameter names, the active state is now only shown in the value column and on the left side of the layer or parameter name.
- Double-clicking a layer name toggles its parameters. Double-clicking a parameter name (first column) opens the Curve Editor. Double-clicking a parameter value (second column) allows changing the value.
- Click-able icons in designated columns allow to mute the rendering or audio.

Please activate "ClearType" to have a clearer font and, as always, make sure that the "DPI scaling" is set to 100% for a correct scaling of our interface. The chapter "User Interface - Master" [117] explains this in more detail.

++ New File Paths for Installation, Logs and Content ++



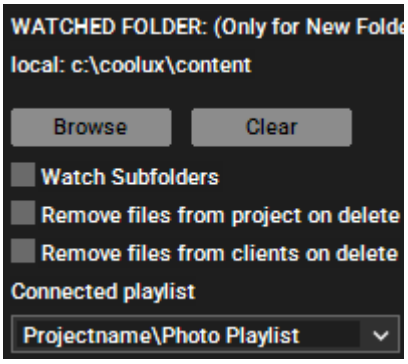
Pandoras Box is now installed into `C:\Program Files\Christie` next to the Widget Designer.

Also, the log files and crash dumps got a new location to prevent the need to navigate into system locked installation folders when grabbing a file for support. You can now find the log files within `C:\ProgramData\Christie`. We added a link to this new place in the former log folder.

A bigger change is the new content folder:

As introduced many years ago, Pandoras Box prefers and defaults into `C:\coolux\content`. Christie hardware always mounts content drives (RAIDs) into this folder to act as a regular folder and all content data is placed into the right location. This way the OS-drive is not overloaded with content or cache files. Pandoras Box 6.4.0 also finally removes all mentioning of coolux and transitions to Christie (Digital Systems Germany GmbH). This requires us to mount content drives into `C:\Christie\content` going forward. No worries, we do care for compatibility! You will not even realize that this has changed as we continue to use the coolux folder if it exists already. Only when installing Pandoras Box 6.4.0 on a brand-new system, the Christie folder will be created. Existing hardware can be used as it is and also a mixture of hardware configurations is no issue at all.

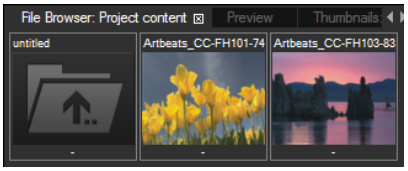
++ Linking a Watch Folder to a Playlist ++



Another 6.4 feature is the functionality to directly link a Playlist in Pandoras Box [239] to a watched Windows folder. This way Pandoras Box automatically adds and removes files to its Playlist. It can therefore run in standalone mode and does not need to rely on another application like Widget Designer [786] performing this task. This makes it easier and faster especially for fixed installs or when dropping photographs into Pandoras Box on the fly. You can find the new options in the (Watch) Folder Inspector [197].

Step-by-step instruction:
- In the project tab, right-click a folder and choose "Add Playlist" and afterwards "Add Folder".
- Select the folder and look at the bottommost options named "Watched Folder" in the Inspector. (Later on, you might need to open the folder attributes by clicking the topmost + icon.)
- Click "Browse" and select the Windows folder you like to watch. Now, additional options appear for the Watch Folder.
- Open the drop-down list and select the Playlist you like to link to the Watch Folder.
- Optionally, select the Playlist and in the Inspector, click the button "Open in Resource Editor". All files from the Windows folder are part of the Playlist. If you are missing files, check whether the format can be read from Pandoras Box [90]. Other Playlist information can be found in the chapters Playlist [239] and Playlist Inspector [202].

++ New "File Browser" Tab ++



Pandoras Box version 6.4.0 introduces a new tab: the File Browser [185]. Right-click on a folder in the Project [278] tab or in the Assets [131] tab, and you will see the option to open this folder in a new File Browser [185]. It opens in a new tab and shows the content from the respective folder as thumbnails. You can open as many File Browsers as you need. If you are interested in changing the layout of the user interface, please see the chapter "Layout [313]". It describes how to add or break-out panes and how to save and load views.

The Thumbnails tab [310] is still available but as always, you can open it only once and the displayed content changes dynamically whenever you select another folder. The File Browser on the other hand is more static as it keeps displaying the same content.

++ New Algorithm for Media Encryption ++

The 64bit version of Pandoras Box uses a new encryption algorithm. Hence, media files that are encrypted with this new version can only be decrypted with another 6.4 (or higher) PB system. In return, media files encrypted earlier can only be decrypted with a PB version below 6.4.

Pandoras Box Media Encryption is a technology that encrypts media in order to protect content from being played back by unauthorized persons or systems without permission. More information... [222]

++ New Color Space Option for HDMI Inputs ++

When working with HDMI inputs [1978], you can now choose another chroma sub sampling, i.e. RGB 4:4.4, in addition to YUV 4:2:2. You can find this option in the configuration window from the input.

Master: add input from Assets tab [131] to Project tab [278] and select it; click "Configure" button at the bottom of the Inspector
Client: click "Configure Live Inputs" in the Client's user interface [316]

++ Discontinued Support of Older Controller Boards ++

With this Pandoras Box version, the older Jog/Shuttle [1994] and Playback Extension [1995] are not supported anymore. Please make sure to use the newer version of the Jog/Shuttle Control [1991] (depicted below) and Fader Extension [1993].

++ Discontinued Support of MA-Net 1 ++

MA-Net version 1 is no longer supported in Pandoras Box. MA-Net 2 (v2.9xx) and other protocols can be activated in the Configuration tab > Remote Control Protocols [148].

++ Discontinued Support of Older Input Card Drivers ++

Please make sure to have Input Card driver version 6.0.1 installed when using video inputs [1947] with Pandoras Box version 6.4.0. Older input card driver versions are no longer supported.

++ Christie USB Driver Added to Installer ++

The installer now includes the USB driver which you need for various Pandoras Box USB devices.

++ New Version of Pandoras Box Menu ++

We also updated the software "PB Menu" [2097] which is pre-installed on all Pandoras Box hardware. "PB Menu 6.4.0" reflects the changes with the new file paths for the installation of Pandoras Box as well as for the Logs and Content folders. Besides, it includes bug fixes. Please download it from the Download-Center.

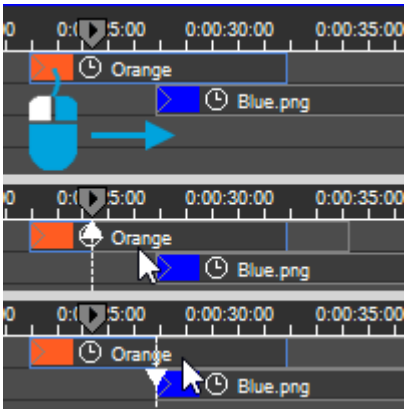
Version 6.1.4

++ Last 32bit Release ++

This version includes bug fixes to prevent possible crashes. Pandoras Box 6.1.4 is the last version in a 32bit environment. Following PB versions are 64bit applications.

Version 6.1.3

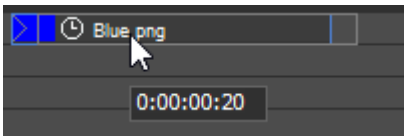
++ New Snapping for Containers and Key Frames ++



This version makes it easier and more intuitive to drag containers and key frames precisely to another element on the timeline.
If you drag one or several containers or key frames in the Sequence [292] tab, you will notice that they snap to cues and to the borders of other containers. Key frames can also snap to other key frames within the same container. Hence, you can align containers and keys much faster without positioning the Nowpointer first (which was always "magnetic"). You can snap to the beginning or end of all containers, might they be on other Layers or Sites. You can only snap to elements in Layers that are shown and not hidden. Whenever a snapping target is active, you will see a white dashed vertical line and arrows which indicate the snapping partners, e.g. the beginning of your dragged container and the end of another container.
Zooming in and out could help finding the required snap target faster. To disable snapping temporarily, hold the [Shift] and [Alt] key whilst dragging. To disable it permanently, go to Configuration tab > User > Sequence [142] and uncheck "Snap Containers" or "Snap Cues".

More information can be found in the chapter Programming in the timeline [296].

++ New Relative Time Pop-up ++

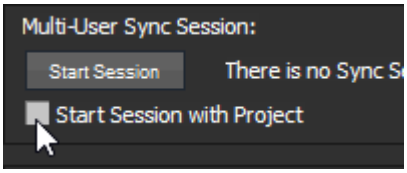


When dragging a container or key frame in the timeline, a new pop-up appears next to your cursor and shows the relative time how far it is moved, e.g. plus 20 frames.

++ Possibility to Recalibrate CMSH-files via Widget Designer ++

Pandoras Box version 6.1.3 introduces the possibility to recalibrate (transform) CMSH-files from Widget Designer (version 6.1.0). You will need the Projector Recalibration [1496] tool from WD and the hardware device "Calibration Link [2000]".

++ Faster Possibility to Start a Sync Session ++

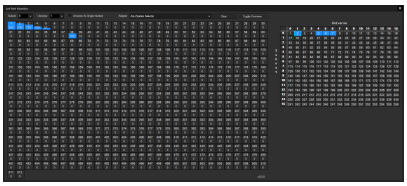


The Startup menu of Pandoras Box version 6.1.3 offers the possibility to start a project and a sync session directly. For more information about Sync Sessions, please see the chapter "Multi-User" [234].

++ Single Touch Events in CEF Browser ++

Touch induced mouse events now work for CEF browser as well. Up until now, touch events only worked in the Awesomium browser engine. More information can be found in the chapter Configuration > Resources(Global) [155].

++ Revised Art-Net Monitor ++



Pandoras BPANDORAS BOX WEB BROWSERox version 6.1.3 includes a new version of the Art-Net Monitor [2051].

The button "Toggle Overview" shows (or hides) a table with all Universes and Subnets. The table can also be clicked to see all channels from a respective universe. Alternatively, you can choose the universe with the drop-down list(s) on the left side. You can either choose a "Universe" and "Subnet" or a single "Universe #" in case the option "Universe As Single Number" is checked.

A new drop-down list lets you select the network adapter that is used for Art-Net. If channel values cannot be received anymore (e.g. because the connection is interrupted) the displayed blue values turn gray. In that case you can erase them with the "Clear" button. Now, "---" is displayed per channel until values are received again.

Version 6.1.1

++ Art-Net Optimization ++

Pandoras Box version offers three optimizations regarding the outgoing Art-Net data when Sites or Layers are patched [228]. From now on, Art-Net output values are only sent when a Channel is patched. Further, channels are reset to 0 when a patch is removed. And lastly, Art-Net universes stop sending when they are not used anymore.

++ New TCP Device ++

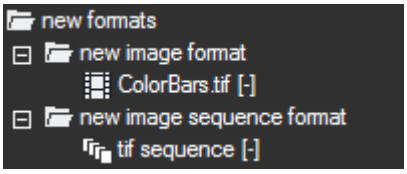
New TCP Device [693] for simple communication to TCP/IP Server (e.g. projectors or video switchers).

Version 6.1.0

++ Localization: User Interface in Different Languages ++

With Pandoras Box version 6.1.0 we introduce a user interface in different languages. To change it, please go to the Configuration tab > Startup [150] and select your preferred language. Close the Pandoras Box master software and start it again. Save your project if you like but this is not mandatory. From now on, Pandoras Box will always start in the selected language. Currently, you can select these languages: English, French, German, Italian, Japanese, Korean, Russian, Simple Chinese and Spanish.

++ New Image (Sequence) Format: TIFF ++



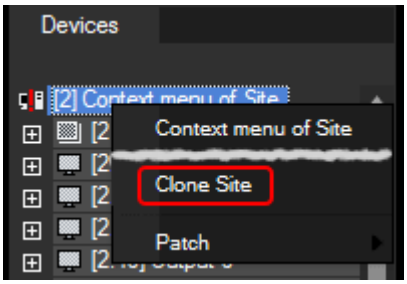
Pandoras Box version 6.1.0 supports now the TIF format for images and image sequences.

The playback performance of TIF image sequences equals the one for BMP or other uncompressed files. The performance specs can be found online in the Download-Center. Note that TIF sequences are only supported when the images are saved uncompressed with a color depth of 8 bit and with or without an alpha (transparency) channel.

For importing TIF still images, you can also choose other TIF saving options like image or layer compression but the above recommendation gives you the best quality and performance. Note that multiple layers are discarded as the file can only be assigned to one Layer in PB, e.g. a Video Layer.

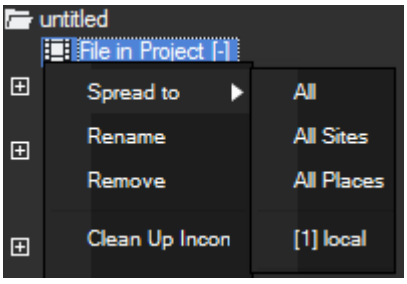
Please see the chapters Image Formats [93] and Image Sequence Formats [95] for more information.

++ New Command in Device Tree: Clone Site ++



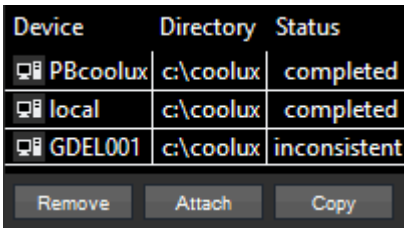
If you right-click a Site (remote Player or Server) in the Device Tree [173] you will find a new command called "Clone Site". It creates a copy of the Site including the entire layer structure (layer names and IDs, render order and all Inspector settings) as well as the programming. You simply need to enter a new IP address. This is especially useful when programming backup scenarios or a show where many Sites equal each other but the Virtual site feature is not possible.

++ New Spreading Commands in Project Tab ++



The context menu of a Resource in the Project tab offers new commands to spread the file. You can choose between these options:
- All available systems including Clients and Multi-User Places
- All Clients (but not Multi-User Places)
- All Places (but not Clients)
- Only to one of the listed systems

++ Inspector Shows Resource Location Including Multi-User Places ++



The Inspector tab in Pandoras Box 6.1 now shows other Multi-User [234] Places too.

The File Location Table [193], e.g. in the File Inspector [191], informs about the location and status of the selected file on each system connected to the Master. In former versions this included only Clients, but now also Multi-User Places are listed to give you a proper overview where the file exists.

++ Assets Tab Shows Multi-User Places and Allows Multi-Selection ++

Up until now, the Assets tab [131] only showed the local computer and remote Clients. From now on, you will also find other Managers as Multi-User Places [234] when using this feature. You can navigate through their folder structure as usual and drag content from their hard disks to the Project tab [278]. When no other Multi-User is online, it will say "No remote places."

Furthermore, it is now possible to multi-select remote Clients and drag them into the Device Tree [173].

++ New Shortcut to Store to Selected Device(s) ++

The command that stores active values only for those Devices that are currently selected, has now the shortcut [Ctrl + Alt + D]. As in previous versions, it is still available in the Edit menu [121] or in the Sequence Button Bar [293] via this icon  .

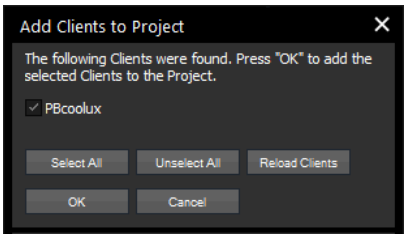
++ Device Viewer Tab Can Be Opened Multiple Times ++

The Device Viewer tab [184] in Pandoras Box 6.1 has a new drop-down menu to select a Site and hence, the tab can now be opened multiple times (via the menu "Tabs"). This feature is very useful when you need an overview of several Sites.

As a reminder, you can right-click a pane and choose to split it horizontally or vertically and then drag the second Device Viewer to the new location. This is explained in the chapter "Layout [313]".

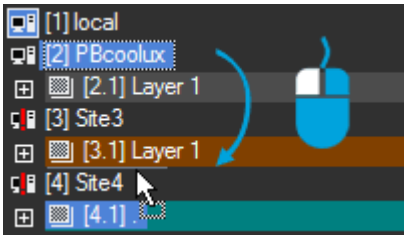
Version 6.0.5

++ Clients Can Be Added to the Project Automatically at Startup ++



When starting a new project on your Master software, Pandoras Box 6.0.3 will offer to add Clients to the Project automatically. Of course your Clients need to be in the same network and run with the same Version and Domain [147] number. All existing Clients are visible in the dialog "Add Clients to Project". The list can be reloaded with the "Reload Clients" button. You can select or unselect individual Clients or all of them by pressing the according button. Press "Ok" to add all selected Clients to the Device Tree [173].

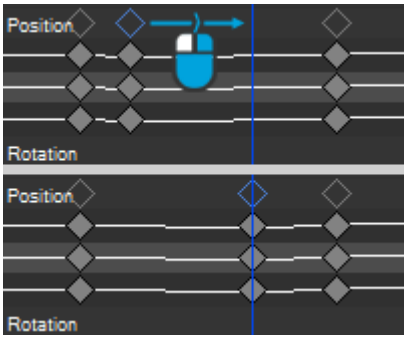
++ Sites Can Be Re-ordered ++



It is now possible to re-structure entire sites in the Device Tree [173] via drag and drop. Simply drag the Site to its new place as depicted in the left image. Note the horizontal line and the cursor that indicates a valid position. Re-ordering Layers to influence the render order works in the same way.

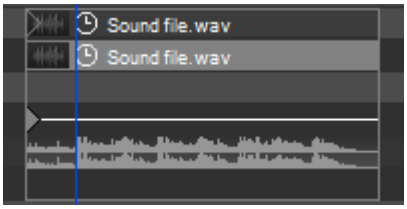
Note that you can also change the Site ID in its Inspector [210].

++ Improved Ghost Key Behavior ++



From now on, if you select a ghost key, all underlying keys will be selected too. Select the ghost key with a single click or by dragging a multi-selection box around one or more ghost keys. You can then drag the ghost key(s) to a new time or enter a value in the Inspector which changes all underlying keys.

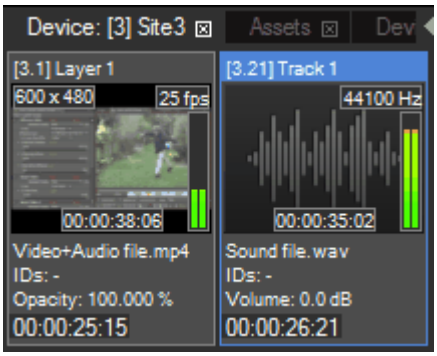
++ Waveform included in Audio Tracks ++



Audio Tracks [661] and Video Layers [647] of Pandoras Box include a new parameter "Waveform" that displays audio peaks. In case you like to align other content on Video or Graphic Layers as accurate as possible it is recommended to zoom in closely. Use the [+] key on the main keyboard or the Zoom-in icon  in the Sequence Button Bar [293] to do so. There, you may also find the drop-down for changing the line height 

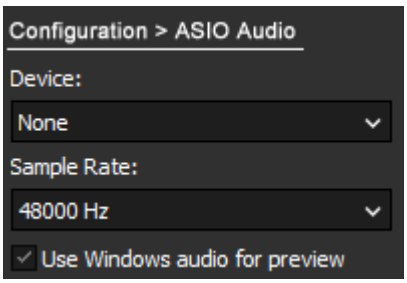
which enlarges the waveform accordingly. You can choose between two waveforms, a standard one with positive and negative peaks or a a rectified waveform (depicted left) with only positive peaks. You can find this setting in the Configuration tab > User > Sequence [142] > Clip settings.

++ Audiometer Included in the Device Viewer ++



The Device Viewer tab [184] in Pandoras Box now shows a thumbnail with an Audiometer for each file that is playing. The Audiometer depicts the Audio level of a resource containing audio, that might be an audio file [92] playing on an Audio Track [661] or a video file containing audio information playing on a Video Layer. Note that the following should apply to see information in the Audiometer:
- a Site must be previewed [244] and selected,
- the (audio) files must be spread to the Master in order to play them there in case you add them from a remote location to the Project,
- the Master needs to have a sound card, but not necessarily an ASIO one.

++ Windows Audio Device Can Listen to Audio Tracks ++

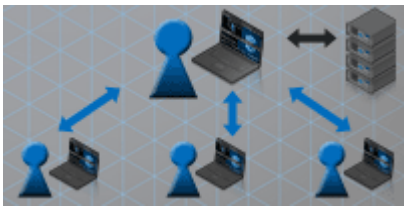


This feature is of interest if the computer that runs the Master has no dedicated ASIO sound card or driver installed but offers a Windows audio device. In previous versions this hardware setup allowed only to play audio (containing) files on a Video Layer but not from an Audio Track [661]. With version 6.0.3 you can now choose to play all audio with the audio device that is setup as the standard audio device in the Sound card settings of Windows. Please also check the sample rate (e.g. 48 or 96kHz) and bit depth (16bit) there, mostly available in the advanced properties tab.
Then simply activate the check box "Use Windows audio for preview" in the ASIO section of the Configuration [166] tab in Pandoras Box and you can preview and listen to your programming of the local or remote node. When using Windows Audio all ASIO channels are mixed down to two channels, all odd ones to the left channel and the even ones to the right channel.
Please also note that a normal sound card does not synchronize audio information in the same way as an ASIO interface. For audio playback during a show, an ASIO interface is recommended.

++ Audio Channels Can Be Routed Individually. ++

The left and right channel of an Audio Track [661] can now be routed individually in the Track Inspector [217]. If you like to use one channel only for playing mono files, you can simply set the second channel to "0".

++ Multi-User ++



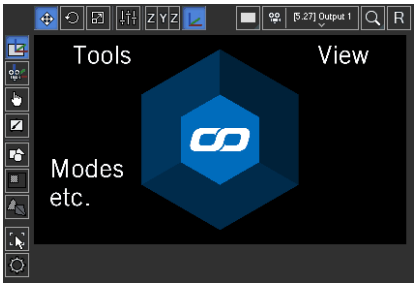
Pandoras Box version 6 introduces the Multi-User feature allowing several operators to work on the same Master file loaded with a Manager license. As the ways to build shows with Pandoras Box are so versatile, Multi-User editing is a flexible feature. This means that you can choose what exactly you would like to synchronize and how often or when you like to do so.
As soon as another Manager is in the same LAN network (and Domain) it participates as another Multi-User. The new tab Multi-User [234] allows to setup the Multi-User environment and gives an overview on all Users (also called Places) online.

The first possibility for Multi-Users is to work in a "Sync Session" where all changes are synced continuously and seamlessly between all Users. However, you can also activate the "Blind Mode" within the Sync Session for working with an independent Nowpointer and Active Values. This allows you for example to jump to "your" time in the Sequence to store content or to view content in the Preview.
You can leave and (re-) join the Sync Session at any time. When joining a Sync Session you are asked to load the session project.

Alternative to (fully or partly) synchronized programming, you can work asynchronously in a separate project and send ("push") your changes at a certain time, e.g. checked or newly encoded content, warp and softedge settings, parts of a Sequence, etc. The "Run Operation" button in the Multi-User tab opens a dialog to define these Pull and Push Operations.

Note also that the Configuration is newly structured. All settings that are listed in the Global category apply to all Multi-Users whilst the other ones apply only locally.

++ New Structure for the Preview Tab ++



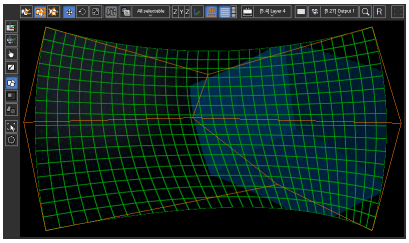
As you will see when reading on, many v6 features are associated with the Preview making it as powerful as never before. Together with the features, we enhanced the structure of the Preview by adding buttons and drop-down lists around the main window. Some buttons are fixed and they either set a view option or toggle to one of the new modes (the image to the left shows where to find these buttons). Tools on the other hand vary because they depend on the chosen mode.

Mostly, a "Parameter Floater"  is contained. The Floater refers to certain parameters and allows very fast and accurate programming. The Floater dialog can be undocked from its position.

In case you maximize the Preview to full screen (shortcut [Ctrl + Shift + F]) a new window "Button Bar" opens and shows all modes and associated tools. Its shortcut is [T] to show and hide it.

See the Preview chapter [243] for full and accurate information about all the new options or read on for a short description of the new Preview features.

++ Editable Meshes Allow Warping in the Preview ++



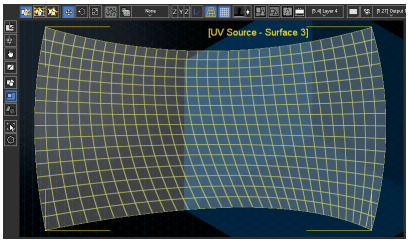
One of the biggest v6 features is the Editable Mesh which allows to warp in the Preview.
To create one, right-click in the Project tab and choose "Add Editable Mesh". Per default the newly created Editable Mesh folder lists one Sub Mesh called "Surface" being a planar 2D mesh in fullscreen size. The Surface's Inspector lets you change the Segment count (known as Mesh points in the Warper), the Control Point count (aka FFD), mesh size and segments and other parameters.
Alternatively, you can import an existing object and convert it to an Editable Mesh using the right-click menu. An Editable Mesh can be assigned to a Layer or an Output. When you set the Preview to the new "Mesh

Editing Mode" the Segments and Control Points become visible. Above the Preview you may find according buttons to select and move either the orange FFD points or gray vertex points. You can warp using the mouse or the keyboard. The above mentioned "Parameter Floater" shows the exact position in pixels.

In the Output layer [682] you may find a new "Edit Warp Mesh" button. It either selects the Output Mesh or creates a new one for you and enters the "Editable Mesh" mode.

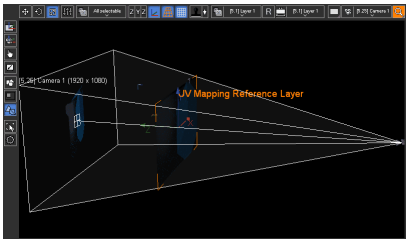
All this and more is explained in the chapter about the Mesh Editing Mode [257].

++ UV Texture Mapping in the Preview ++



In addition to the introduction to warp inside Pandoras Box, we also included texture mapping to the Preview. In short, texture mapping is the way how the texture (i.e. the media file) fits onto the mesh or in other words, which pixel from the 2D (UV) texture belongs to which 3D (XYZ) coordinate. The Pandoras Box Preview now offers two different modes to apply and modify the UV source: the planar and the perspective UV Mapping Mode.

The planar one probably reminds you of how you worked in the Warper. You have the choice between two differently sized planar maps and the default 1:1 map.
The perspective mapping is totally new. Depending on the Layer and Camera position a so called Reference Layer positions itself and "projects" the UV source onto the 2D or 3D mesh.



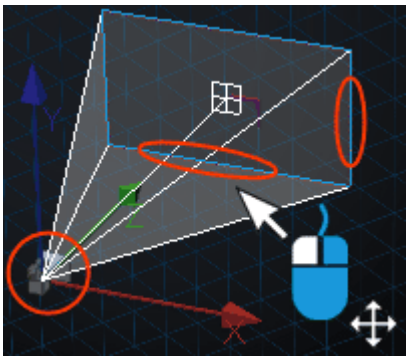
Either way, after choosing the method, you click the "Imprint" icon to calculate the UV coordinates and toggle back to another Preview mode. Of course you can always go back and modify the UV source, e.g. you can fine-tune the mapping by moving selected vertices.
The Sub Mesh Inspector [201] offers the option to reset all changes.

Read more in the chapter about the UV Mapping Modes [269] in the Preview...

++ Enhanced Canvas ++

Version 6 comes with a significantly improved Canvas. It is now possible to draw directly in Pandoras Box with tools provided in the Preview tab. In addition, the Canvas Asset can be saved, exported and (re-)imported. All together the Canvas feature now also allows to mask directly in Pandoras Box.
Right-click in the Project tab and choose "Add Canvas". Assign the Canvas to a Layer overlaying your other Layers. Until now nothing changes as the default Canvas is still empty. To draw on it select the "Canvas Editing Mode" in the Preview, leave the Brush tool selected but change the Color to black. You can draw (and erase) freehand by clicking and dragging. Straight lines can be done by clicking once on the beginning and holding the [Shift] key when clicking on the end point of the line.
To export the mask as a PNG file, right-click on the Canvas and choose "Export". After editing the image for example with another graphics software you can import it with another right-click on the same Canvas. Alternatively, you can add an image file to the project, right-click it and say "Create Canvas from Image".
Read more about the Canvas Editing Mode... [256]

++ Camera Interaction in the Preview ++

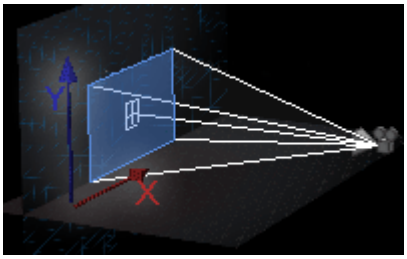


With this version you can interact with Cameras in the Preview in the same way as you are used to position Layers.
First, set the Preview to a Camera View (incl. the "All Cameras" view) and switch to the new mode "Camera Transformation Mode".

If you now click on the Camera rectangle it highlights blue and you can drag it around. This changes the X and Y Offset parameters.

Cameras of a Server have more parameters available. First, click the "Camera Visibility" button to toggle the Camera icon and frustum as seen in the left picture. Now you can either drag the Camera icon around which influences the Camera Viewpoint and Target simultaneously, or you can drag the Viewpoint and Target separately. Hold the [Alt] key down to move the Camera / Viewpoint only and the [Alt] and [Shift] key for the Target.

++ Enforced 2D Interaction in the Preview ++



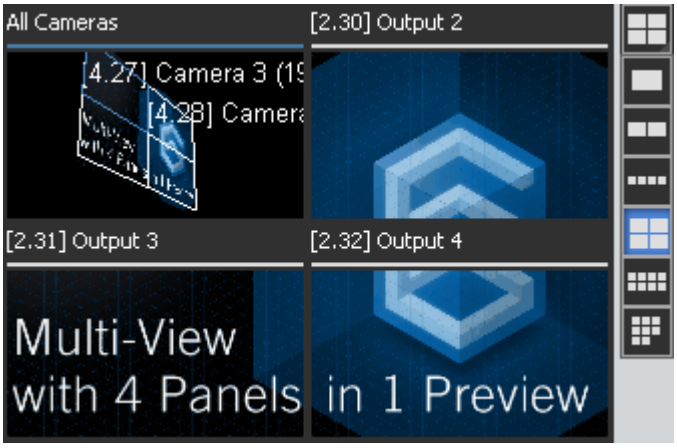
The continuous development to interact with more and more items in the Preview directly now allows to change Layers, Cameras, Mesh Points etc. To optimize this workflow and reduce errors, version 6 introduces a slightly changed interaction. Eventhough you still work with 3D parameters, each interaction happens in a 2D plane which depends on your viewpoint. This results in those changes you aimed for and prevents you from dragging items to an unwanted position, e.g. a distant Z position.

++ Pixel-Oriented Workflow ++

The pixel-oriented workflow, introduced in version 5.7 to allow pixel accurate settings is now the new default when creating a project.

In short, this involves:
- most units displayed (e.g. in the Device Parameter tab) are in pixel units
- the size of a media on a Layer is pixel accurate due to the Layer Sizing Mode [211] in the Inspector
- the size of a Camera window is pixel accurate
- in the Configuration tab > "Unit Management [160]" the "Unit Translation Mode" is set to "Fixed relationship" and a translation factor is automatically set (this factor transfers 3D Units to pixel units within one or multiple systems)

++ Preview Window Allows Multi-View and Shows Tools in Fullscreen ++



The Preview of version 6 can be toggled to a Multi-View window. Per default you see one view port as usual, but now you can also choose to work with a split view with 2, 4 or 8 view ports in various arrays. If you have loaded the "All Cameras" view before toggling to more view ports, the newly created ones will show "None"; in case you have loaded a Cameras (or Output) view, the newly created windows will automatically show other Camera (or Output) views if they are available. Of course you can change each view port individually using the drop-down menu. Please note that for the time being it is not allowed to view a Camera and the corresponding Output at the same time. Another restriction in this version is that one view can be loaded into one view port only. In other words, the "All Cameras" view or "Camera 1" can not be depicted twice.

Independent from the number of view ports, it is possible to maximize the Preview to full screen. This is especially of interest in case you warp several Outputs in the Manager using the Editable Mesh feature.

++ New Project Data Format ++

A Pandoras Box project can now be saved as a binary file. The "Pandoras Box Binary" format with the extension "pbb" is the new default but the XML format is still supported. The advantages of a binary code are a smaller memory space and thus faster saving and loading.

Christie
Pandoras Box

Part 4

What's New in Widget Designer

4 What's New in Widget Designer

This "What's New" chapter starts with updates for the Widget Designer [786] version 6. The change log lists also earlier changes, minor ones and fixed issues. You may find it in the installation path, e.g. `C:\Program Files\Christie\Widget Designer...`

Version 6.5.1

++ Bug Fix Release ++

This release is a bug fix release fixing e.g. issues when changing Widget IDs.

As always, bug fixes can be tracked in the "changelog.html" file in the installation folder under `C:\Program Files\Christie\Widget Designer...`

Version 6.5

++ NDI and RTSP Support ++

Widget Designer now also supports NDI® as well as RTSP streams and hence allows to monitor systems and camera feeds. Using camera feeds directly in the Widget dashboard provides control and monitoring in the same interface. The Camera Tracker [1291], Face Tracker [1287], Motion Detector [1289] and Video Input Display [865] all feature a "Devices" drop-down list to which NDI and RTSP streams have been added.

NDI (Network Device Interface) provides separated video and audio network streams via uni- or multicast. Once an NDI input is chosen, Widget Designer subscribes to the stream and displays the content. All NDI variations including different compressions are supported. When using NDI streams, keep an eye on the system performance and network traffic. The more streams you like to use or the higher the resolution or frame rate is, the more it becomes necessary to separate the show network from the streaming network. It is very simple to work with multiple network adapters. Simply choose one for the show traffic (Connection Manager [1264] or PB Network Configuration [1256]) and use another for your NDI streams. The system receives the streams automatically without the need to setup the NIC explicitly.

NDI® is a registered trademark of NewTek, Inc. For more information about NDI itself, NDI tools etc., see http://ndi.tv/

RTSP (Real Time Streaming Protocol) is another way of streaming video over the network (often based on H.264). Many web cams and CCTV cameras support this protocol to distribute their live stream to other systems. When choosing "RTSP" in the above mentioned tools or widget, you also need to enter the URL into the according text field. This obviously depends on the device you like to receive the stream from, e.g. your web cam. You can also connect to a public stream.

++ New Viewer Mode ++

Widget Designer version 6.5 features a new Viewer Mode. In this mode the project cannot be change anymore, it can only be used as intended and hence is perfect for end customers in museum-like scenarios. In difference to the former functionality, the Viewer Mode is not an independent edition but based on the WD license which means that it has access to the same features and tools etc. First, prepare a project, e.g. using a licensed Widget Designer. Then you save it and load it in the Viewer Mode. The File Menu [792] offers the option "Create Viewer Shortcut" to saves a shortcut which opens the current project in Viewer Mode.

In short, the project cannot be toggled from "Run Mode" to the "Create" or "Edit Mode" [789] and there is no option to change the user interface or save the project if it was not prepared beforehand with scripts. All scripts, Nodes, Widgets, connection etc. are working as they do in the "Run mode". Read more about the differences from the Viewer Mode to regular editing... [787]

If you are interested in a fullscreen view of your project, you can also activate the "Kiosk View [801]" from the View Menu or allow the shortcut [F11] to toggle it. This can be done in the Security Settings dialog [796], where you can also setup a password that needs to be entered when a project is loaded, saved or closed.

++ Password Protection ++

In this version you can secure your project with passwords. Widget Designer will ask for it when the project is opened, saved or closed. In the "Security Settings [796]" dialog that can be opened from the File Menu [792], you can enable the protection and choose a password. You can use as many characters as you wish. Click "OK" to close the dialog and save your project.

You can choose two different passwords, one for the above described Viewer Mode and one for general editing. You can use only one protection or both with different passwords. Further, the dialog offers to en-/disable the shortcut key [F11] when the project is started in the Viewer Mode. F11 toggles the Kiosk View which shows the interface in fullscreen and hides all other bars.

++ PowerPoint Plugin, Device and Display ++

The PowerPoint feature consists out of two elements which communicate with each other: a PowerPoint plugin (i.e. add-in) and the newly implemented "PowerPoint device" in Widget Designer's Configuration dialog. Widget Designer can remote control a PowerPoint presentation and in return it receives feedback that can be displayed and used via scripting. This involves the current and total slide number, speaker notes and even thumbnails from the previous, current and next slide.

The plugin can be downloaded from our Download-Center. Install it on the Windows PC that runs the PowerPoint presentation. In PowerPoint, open the tab "WD PPT Remote" and start the TCP Server which will connect via network. Lastly, start the slide show mode for your presentation.

In Widget Designer, add a PowerPoint device to the Configuration dialog and establish the TCP connection. You can remote control the PowerPoint presentation using the scripting language. For example, you can trigger the next action (which is your specified animation or text) or call a specific slide. You can also receive feedback via the scripting language or use the Event Listener for it. The PowerPoint feature is described in full detail in the chapter "PowerPoint" [1468].

Widget Designer 6.5 now also offers a PowerPoint display [860]. First you choose, which slide it should display and then whether a script is executed when the image is clicked. This is perfect for displaying the current slide, or calling the next one.

Of course you can combine the PowerPoint feature with our "web server [1929]" feature and publish WD pages as web pages in order to access them via tablets or other mobile devices. Or, you can schedule actions using an "Event Viewer [871]", or make use of the "Widget Designer Device [696]" in Pandoras Box to program commands to the Pandoras Box timeline.

++ New TCP and UDP Devices ++

The Configuration dialog [1305] now also offers a TCP Client, UDP Client and UDP Server & Client device.

The TCP Client [1471] allows to easily connect to another TCP server.
The UDP Client [1475] in the Configuration dialog allows to easily create a UDP connection. The UDP Client is perfect for scenarios where you like to send UDP messages to a fixed address and optionally also receive messages.
The UDP Server&Client [1479] connection however is more useful if you like to listen on a certain port and optionally send messages to various addresses.

For all devices, the IP address and ports are scriptable which allows you to use the them as an interface that connects to various devices on the fly if needed. Incoming messages can be received via Event Listeners. Messages can be sent out via scripting through members. Each of the linked chapters contain a sub chapter listing all available events and members.

++ Custom Shapes for Custom Script Buttons ++

For a long time, Custom Script Buttons [822] needed to be rectangular but now they can have any shape you wish. Simply create an according image with transparency and assign it to the Release, Click or Highlight state

of the CS button. When you click on a fully transparent area in Run mode, the button does not execute any script. In case you placed other Widget underneath that area, it will receive that click.

++ Line Shape ++

The Shape widget [919] can now also be a simple Line.

++ Controlling Notch Layers in Pandoras Box ++

Pandoras Box version 8 introduces Notch Layers [671] which can render and control Notch scenes. As any other layer in PB it can be controlled using Widget Designer, e.g. through interactive devices like the AirScan [1988] or sensors like Phidgets [1383] and more.

Remote controlling a Notch parameter is very similar to controlling an effect, which means that you simply need to know the exact parameter name. The name is case-sensitive, all spaces count and the hierarchy needs to be represented using the pipe "|" character, e.g. `Notch|Star|Sides`. (See more examples [671])

The parameter name is needed for controls like the Fader [874], or nodes like the Device Control Output [1212]. As Notch offers parameters controlling text, Widget Designer 6.5 introduces a new node that can send text values. You can find it under:
Nodes > Output > Pandoras Box >Text Output [1223]

Further, you need the exact parameter name for the scripting language. Commands like `DeviceSetParam` [1536] simply change an integer or double value. If you like to assign a different media file to a media or mesh parameter, search for a command like `DeviceSetMediaParamByID` [1534]. And for text parameters, WD now offers the command `DeviceSetTextParam` [1537]. By the way, parameters showing a drop-down list (even if it shows text) should be controlled via `DeviceSetParam` [1536] and an index number; value 0 would assign the first entry for example.

Note that position values are displayed in Pandoras Box in pixels but the input via the SDK has a different range per default. The chapter Unit Management [160] explains more about this and the check box "Interpret Automation Param Input as Pixel Values" which can be found in the Configuration tab > Unit Management.

++ New Commands ++

The scripting language offers the following new commands:
- `DeviceSetTextParam` [1537] as mentioned above to influence text parameters in PB
- `VGetFolderPathFromFileDialog` [1632] to return a selectable path
- `WDDropDownListGetTitle` [1706] and `WDDropDownListSetTitle` [1709] to work with the title via scripting

- `WDPowerPointDisplay` [1827]... and its members `PowerPointDisplay1.` which refer to the new display widget "PowerPoint Display [860]"
- [1852] `WDVideoInputDisplay` [1852]... and its members `VideoInputDisplay1.` which refer to the display widget "Video Input Display [865]"; the source [1854] and format [1852] can now be scripted too and the current image [1855] can be saved

Version 6.1.3

++ New Lightware Matrix and Barco Devices ++

Lightware Matrix and Barco devices can now also be added to the Configuration [1305] dialog in Widget Designer 6.1.3. Their members [1323] allow to use the scripting language in order to perform actions on them as well as retrieve information. In addition, you can choose actions in the new Event Listener. For more information, see the following chapters:
- Barco ImagePro2 [1306]
- Barco MatrixPro2 [1314]
- Lightware Matrix LW2 [1378]

++ New Phidgets Devices ++

With the last version [52] we added many Phidgets [1383] types to the Configuration [1305] dialog. With this Widget Designer 6.1.3 you have access to five more of them. Again, you can work with them using the scripting language. Firstly, you can access events raised by the devices and secondly, you can interact them with their members. If you like to add or prefer to use the node system, head to the next paragraph.

For more information, see the following chapters:
- Spatial 0/0/3 Phidget [1434]
- Spatial 3/3/3 Phidget [1436]
- Servo Phidget [1428]
- Stepper Phidget [1439]
- Phidget IR [1414]:

In addition we completely rethought the way of learning and sending IR codes. In the Configuration dialog, you can now find the "IR Code Table". You can store and manage IR codes in Code Tables [1418] and assign a script to be triggered for each received code, no matter which connected PhidgetIR device received it.
In return, you can simply control IR devices by transmitting stored codes over the integrated IR diode of any connected PhidgetIR device via scripting.

++ Nodes Added for Servo Phidget and Spatial Phidgets ++

In addition to the above mentioned implementation of the Servo and Spatial Phidgets to the Configuration dialog, Widget Designer also offers four new nodes. Please find more information in the according chapter:

The input nodes provide the sensor values to the node system to control something else with them.
- Phidgets Servo Input [1020]
- Phidgets Spatial 0/0/3 Input [1021]
- Phidgets Spatial 3/3/3 Input [1022]

In return, the output node sends data from the node system to the connected Phidget.
- Phidgets Servo Output [1201]

++ Input Nodes for Jog/Shuttle and Fader Extension Re-Implemented ++

The fader and button boards "Jog/Shuttle" and "Fader Extension" are now also available as Input Nodes providing fader values and button states in the node system. Please add the devices first to the Configuration [1305] dialog to enable them.
Please see the Widget Designer chapters "JogShuttle Input [1009]" and "FaderExtension Input [983]" for more information about the input nodes and the hardware chapters "Jog/Shuttle [1991]" and "Fader Extension" [1993] for information about the boards.

++ New Spyder Node ++

The Spyder Output node [1204] allows to remote control a connected Spyder device in terms of scaling and positioning its layers. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Spyder" [1322] which also shows how to work with the device using the scripting language.

++ New Widget: Terra Display Array ++

The TerraDisplayArray widget [863] shows the display arrangement of a connected Terra device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Terra" [1332] which also shows how to work with the device using the scripting language. As for all Display widgets, the TerraDisplayArray is not available in the free Widget Designer version.

++ New Commands ++

The scripting language offers the following new commands:
- WindowFocus [1884] to activate a certain Windows application.
- EmailSendHTML [1543] to send HTML formatted emails
- WDTerraDisplayArray [1827]... and its members `TerraDisplayArray1.` which refer to the new display widget "TerraDisplayArray"
- various commands (and members) to set the background and text color for the Widgets: DropDown List [868], Inputbox [886] and Treeview [929]

Version 6.1.2

++ Jog/Shuttle and Fader Extension Re-Implemented ++

The hardware fader and button boards "Jog/Shuttle" and "Fader Extension", can now also be added as new devices to the Configuration [1305]. You can assign custom scripts to buttons and the Sequence opacity or other Parameters to faders. With this version, we added events and members for both devices, which you might know from the previous integration with Terra and Phoenix devices. Please see the Widget Designer chapters "Jog/Shuttle [1364]" and "Fader Extension [1354]" for more information about that and the hardware chapters "Jog/Shuttle [1991]" and "Fader Extension" [1993] for information about the boards and how to use them directly in Pandoras Box.

++ New Spyder Device ++

A Christie Spyder device can now be added to the Configuration [1305] dialog in Widget Designer 6.1.2. Spyder members [1323] are offered to control it. For more information, see the chapter Spyder [1322].

++ New Phidgets Devices ++

We have totally overhauled the usage of Phidgets [1383] within Widget Designer. It now uses the latest features and can connect to most recent devices. Currently we support twelve different Phidget types. Please add your desired Phidget first to the Configuration [1305] dialog. Then you can access events raised by the device or control it with members.

Another big feature is that we support the Phidget network. Attach and setup your Phidget on a remote computer and activate the option in the driver that the data should be streamed via network. On your local computer that runs Widget Designer, add a Phidgets Server [1425] to the Configuration dialog and enter the remote IP address to receive the streamed data.

++ New JSON command ++

With two new JSON commands [1546] you can now load the entire content of a JSON or XML formatted text file to a variable:

`JsonLoadFileToVar(filePath,varName)` [1546] loads JSON formatted text from a file
`JsonLoadXmlFileToVar(filePath,varName)` [1546] loads a file containing XML code and re-formats it to JSON

The variable can be either of **JSON** or of **String** type.

Version 6.1.1

++ Native Christie Terra and Christie Phoenix Support ++

Christie Terra and Christie Phoenix devices can now easily be connected to Widget Designer.
With a new Configuration [1305] dialog, WD version 6.1.1 introduces a completely new way of working with external Christie devices.

Commands can be sent and data received without the need for a sophisticated message evaluation algorithm or node system, all you need is to configure the device and use it either via scripting or as Event Listener [1353] in the Configuration.
No need to go through the whole device manual and search for the remote control syntax, or learn the device's own control UI.

This feature enables you to easily create an object of this kind in WD and connect it to the hardware, which then provides you with all necessary tools for interfacing with the device. All commands are there, ready-to-go for one line of code everyone can write. Actions and parameter changes of those devices can be monitored with an Event Listener and used as triggers for scripts.

Please see the following chapters for more information about Terra and Phoenix:
Phoenix [1319] and Phoenix Events [1320]
Terra [1332] and Terra Events [1333], Terra Members [1335], Terra Tutorial [1345] (the description for the members will soon be migrated to the Script Assist in Widget Designer)

++ New File Extension: ".WD" ++

Starting with version 6.1.1, your project files are no longer saved as "wdpj", but with the new extension "wd". Coming with this, the file type itself changes from a JSON based, human readable file, to binary encoded file.
Files you save in versions starting with 6.1.1 cannot be opened with older revisions!

++ Improved JSON Handling ++

A couple of small but helpful members were added to the JSON data type [1922]:
`Json.Copy` - Creates an independent copy of the JSON, similar to the the List.Copy member
`Json.ToFile(fileName)` - Writes the content of the JSON directly into the dedicated text file
`Json.ToPrettyString` - Formats the JSON elements with line breaks and indentation for better readability
`Json.Remove(location)` - Removes the specified location and all child elements from the JSON

++ Pull HTTP Headers in Client Session ++

With this WD version you can write custom HTTP header values to the client's Session [1913] storage.

Example:
HTTP Client sends "X-Widget-Name: Fred" as header
`DebugMessage(Context.Session.Value("name"))`
-> prints `Fred`

Version 6.1.0

++ New Implementation of Projector Recalibration ++

Widget Designer version 6.1.0 offers a new implementation of the Projector Recalibration Tool. [1496] It is used to set up the connection between a Calibration Link [2000] and other Pandoras Box software components for achieving an automated recalibration of a warped projector output in case a projector or screen moved and the projected image is not perfectly warped and blended anymore. The Projector Recalibration Tool [1496] chapter offers more information about the principle, hardware and software setup.

Version 6.0.8

++ Web Server Features ++

The web server in Widget Designer version 6.0.8 now connects to the default page when no page is specified. In addition, clients now always connect to the specified page in the URL upon connection and reloading. Follow the link for more information about the web server [1929] in Widget Designer.

++ New commands: Midi SysEx Messages ++

Widget Designer now supports sending Midi SysEx Messages via the scripting language [1511]. The command are: `MidiSendSysEx(Data)` [1559] and `MidiSendSysExByID(ID,Data)` [1559].

++ Re-introduced Support for Phidget IR Device ++

With this version we re-introduce the support of the Phidget IR device. As this has changed again in version 6.1.2, please refer now to the chapter "Phidgets" [1383].

Version 6.0.6

++ Edited Picture Box ++

Starting with Widget Designer version 6.0.6 the property "Web Link URL" in the PictureBox [911] widget is now editable from any version or license.

++ Web Server Features ++

From now on, you can configure an interval when Clients should reconnect to the Web Server [1929]. You may find the settings under Edit > WebServer Settings > Global Settings.

Version 6.0.5

++ Modbus Features ++

Widget Designer version 6.0.5 introduces support for FHPP (Festo Motor Controllers).

In addition we added the possibility to set many values at once via "`SetMany(<name/index>, <values>)`", for example:
`ModbusDevice.ModbusPatch.SetMany("MyElement", listVariable)`

For more information please see the chapter Modbus [1298].

++ AirScan Features ++

The AirScan setup dialog [1277] offers a new check box "Enable GUI" to choose if touch points should be sent to the current page or not. Further, the command "`WDAirscanConnect(IP)`" is improved as the argument "IP"

is now optional and the already provided IP from the setup dialog will be used.

Version 6.0.4

++ Revised Macros and Functions ++

This version comes with an updated behavior of Macros and Functions, which also means they can be clearly distinguished again. The chapter "Functions and Macros" [1897] explains both scripting techniques in detail.

Functions can have an optional return value, that can be used when calling a Function. Functions are always called in a blocking / synchronous way. This means that another script that includes a command to call a function will wait until the function has finished before processing the next command. In the same way, calling a function three times, will execute the three instances one after another.

Macros do not have a return value. Macros are always called in a non-blocking / asynchronous / simultaneous way. This means that another script that includes a command to call a macro does not wait until the macro has finished but continues to process the next command immediately. In the same way, calling a macro three times, will execute the three instances at once.

Both, functions and macros, support optional input parameters. You can toggle the type (function or macro) in the "Function and Macro Editor" [1897].

++ Page Password Protection ++

The password protection for pages has been updated and re-implemented. You may find the respective settings under Pages > Edit Page [805].

++ New Commands for Event Viewer ++

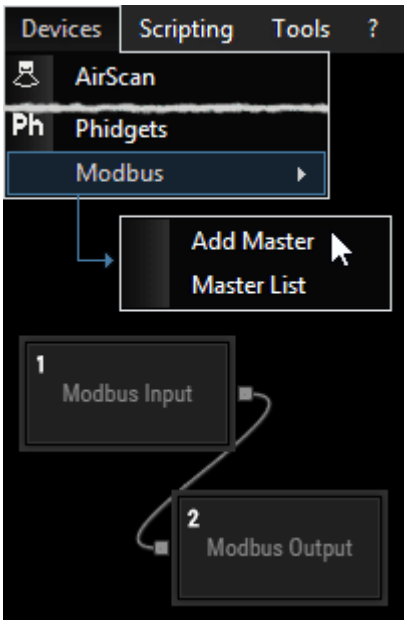
The scripting language now offers new commands for the Event Viewer widget: `WDEventModeMaxCount` [1717], `WDEventRepeatInterval` [1718] and `WDEventRepeatMaximum` [1718].

++ Edited ListView ++

The ListView Widget [890] can now also be loaded in the Free Edition.

Version 6.0.3

++ Modbus Implementation ++



Widget Designer 6.0.3 offers to setup Modbus devices which can then be used in the Modbus input and output node.

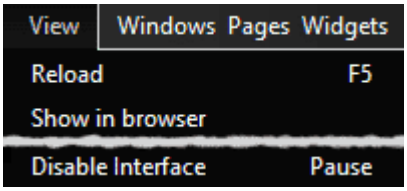
First, open the Devices menu [810] and choose "Modbus" > "Add Master [1299]". You can choose an internal name and settings for the Modbus TCP connection. Next, click "Add" under the "Patches" field. The patch [1300] defines the type of input or output device, so here you can choose an internal patch name and the Modbus type (e.g. Discrete Output Coil), Data Type (e.g. 16Int) and other device settings. You can add other patches, edit, duplicate or remove them.

Now, choose "Modbus" > "Master List [1304]" in the Devices menu. This shows you a list with all Modbus Master devices and options to dis-, re- and connect or to edit them.

Lastly, you can create the according nodes to input or output data:
- Nodes > Input > Devices > Modbus Input [1015] or Modbus FHPP [1013]
- Nodes > Output > Devices > Modbus Output [1196]

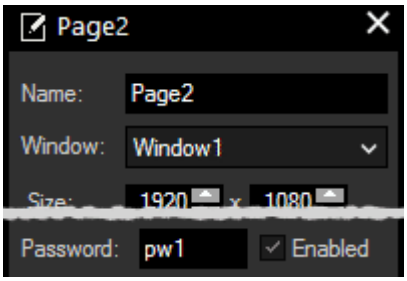
In both nodes the "Device Name", "Patch Name" and "Element Index" (or name) need to be setup whereas Widget Designer offers all Devices and Patches within a drop-down if you have set them up already as described above.

++ Disable Interface ++



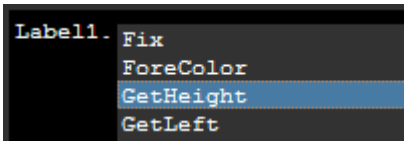
The View menu [799] from Widget Designer 6.0.3 offers a new option. With "Disable Interface" you can disable all (!) mouse, touch and keyboard input and nothing can be clicked or entered anymore. When an input is detected, a Windows error sound is played. This mode is for example perfect for programmers who want to make sure, nobody changes the interface or clicks anything. You can enter and leave this mode with the shortcut [Pause]. Note that the [Pause] key differs from the [Play/Pause] key for the Windows Media Player and cannot be found on all (hardware) keyboards anymore but it is available in the Windows onscreen keyboard.

++ Page Passwords are Back ++



In addition to the feature to disable the entire interface Widget Designer 6.0.3 offers to setup a password in the Page properties [805]. If somebody likes to access a password protected page in Widget Designer itself or in the Web Browser, he first needs to enter the password.

++ Widgets Offer More Members ++



In Widget Designer 6.0.3 many Widgets offer new and more members [1904], i.e properties and methods. The Custom Script Button offers for example new ButtonStyle properties like `CustomScript1.ButtonStyle.FontColor` or methods to return its own height or location which can be found in other Widgets too, e.g. `Fader1.GetHeight` or `Label1.GetLeft`.

Simply, enter the Widgets name and ID into a script field. As soon as you type a dot "." the script assistant will offer all available members. For more information, please see the chapter Object and Member Notation [1904].

[Rev 6077 and 4216] - Version 6.0

++ New Widget Designer Licenses ++

The Widget Designer licenses are newly structured. Earlier, there was a free STD version, a PRO version and an ULT version. The free version, that you can simply download, install and run without a dongle (or another key) is now called Widget Designer Free edition. The PRO version is unchanged but is simply called the Widget Designer or the licensed Widget Designer. It can be extended with the Unlimited Web Clients Option to enable the Web Server to have multiple-sessions and session-based values.

++ Run and Edit Mode Have New Shortcuts ++

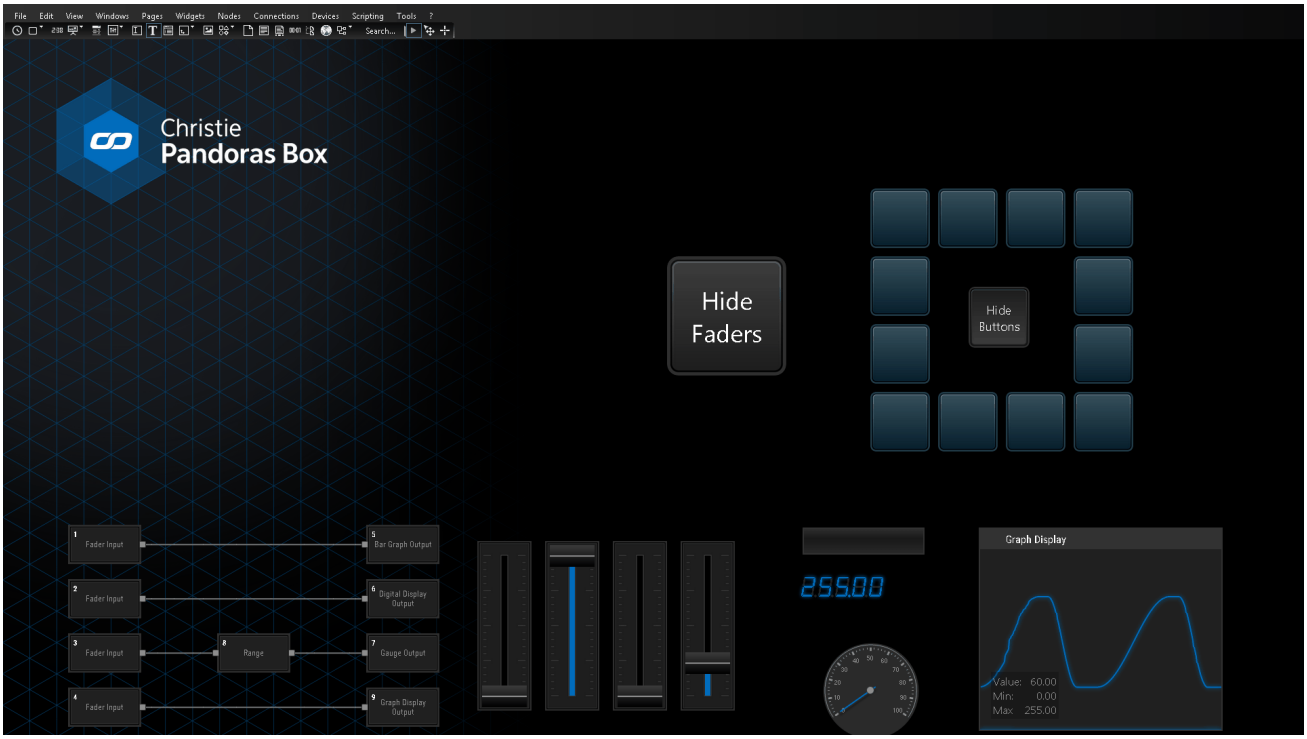
Surely a minor change with great effect. The shortcuts for the Run and Edit Mode are changed.

Go into the Run Mode with [F8] and into the Move / Edit Mode with [F9].
[Ctrl + Shift] lets you create the last Widget. By the way, the right-click menu from any Widget gives you the command to create this Widget again as a new control.

All shortcuts can be found in the Keyboard Shortcuts [816] chapter.

Alternatively, use the icons from the new toolbar to toggle between the Run and Edit Mode or choose the Last-Created icon. 

++ New Interface Behavior and Additional Toolbar ++



One of the first things you might notice is that Widget Designer and all its Widgets have a new look. When you create a larger project you will also soon notice that its performance is better and in general everything goes faster.

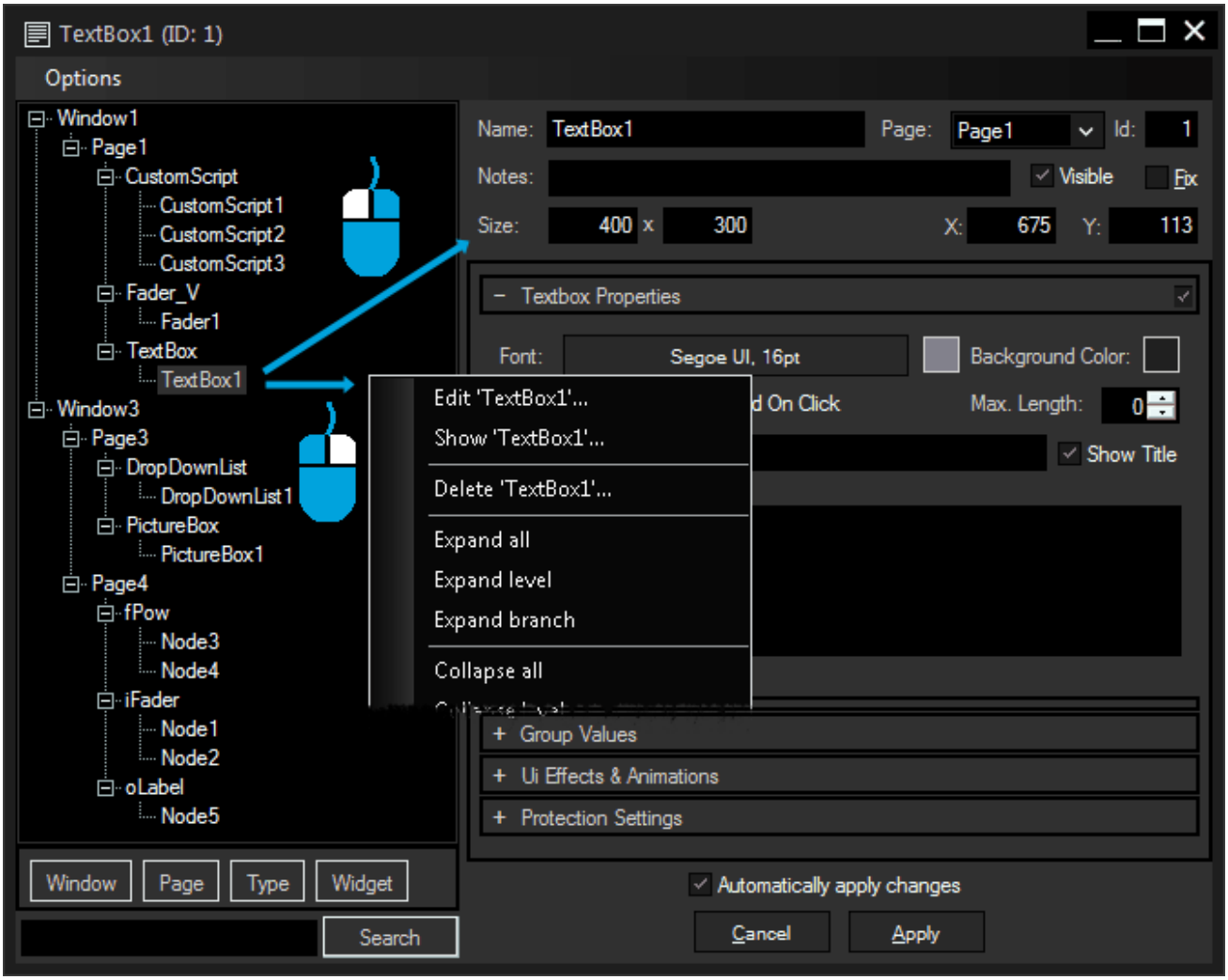
It also starts within a default window, not in fullscreen mode any more. One WD application can have several windows, whilst each window can be subdivided into pages (that are known from older versions already).

To set up the project, a window or the page go to the Project Settings [798], Window Settings [803] or Page Settings [805].

In general all menus and dialogs are newly structured and designed. Starting at the top, the menu bar ("Controls" or "Create" is now "Widgets" and the Tools were separated) adjoins the new toolbar. It contains all available widgets...

Further down, you will notice that the right-click menu is cleaner and "smarter", e.g. right-clicking on a Widget like a Fader offers the commands to either generate relating nodes or to create this Widget again. The item property dialog and others are subdivided into extendable sections. How to work with them and why it matters how you draw a selection box is explained in the chapter User Interface [789]. At the bottom you will see a new Status bar.

++ Widget Explorer, User Profiles and Import ++



In addition to the above and below mentioned enhancements, there are further workflow optimizations:

The completely new Widget Explorer [802] found in the View menu or via [Ctrl + W] is depicted to the left. It displays a tree view of all items of your project. They are sorted by windows [803], pages [805], types (of widgets [818] or nodes [936]) and individual items. The four buttons below the tree view allow you to blank out the corresponding type. You can also search for a special item. Once clicked on an item its Item Properties are loaded to allow very fast programming.

Widget Designer supports user profiles [795] [File Menu > Profile Settings] where you can specify some general settings concerning the usage of the WD application itself.

Eventhough working in Widget Designer was never as easy and fast, the fastest way of course is to re-use projects and solutions you have already created. For this, we have designed an Import tool that helps you choosing and loading the required widgets, nodes, pages etc. The Import Settings [793] can be found in the File menu.

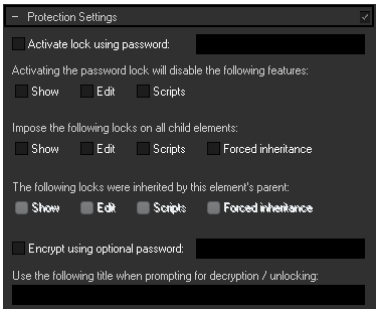
++ CSS Styles ++

Since Widget Designer 6, the Widget Designer interface is based on HTML-5 and features the possibility to style and manipulate complete pages or single widgets due to internally used CSS3 styles (Cascading Style Sheets). Those can be imported and applied to the local user interface. Web designers can easily style customized control panels for a variety of different users and applications. But you can also utilize the full power of CSS in WD6, with or without any technical knowledge of CSS specifics.

Most widgets (and pages and windows) can be modified using web styles. This includes layout, design, filters, images, animations - basically anything you can do with CSS.
In the widgets' properties dialog, these styles can be found in the panel "Effects & Animations". There you can add and apply several styles that you can choose from a large selection of out-of-the-box styles, but you can also add your own styles, of course. Any Window, Page and Widget

See more in the chapter Effects & Animations (Web Styles) [814]

++ Protection and Encryption ++



Widget Designer allows you to lock and/or encrypt single or multiple widgets, nodes, pages, windows or the whole project file. Hence the Protection Settings are part of many dialogs:
To protect the project, open the Project Settings [798] from the Edit menu [797].
To protect a page, open its Edit Page dialog either with the right-click menu or from the Pages menu [805].
To protect a widget, open its Item Properties dialog with the right-click menu or the shortcut [Alt + P].

All passwords can be remembered in Widget Designer so that you do not have to enter them every time. However, this applies only to the current session. After closing the Widget Designer, the passwords will be expired.
You have the possibility to reset all passwords during an open session with Edit [797] > Reset protection passwords...

Please note that the protection settings are rather designed for protecting your data in case of distribution. If you like to set passwords for windows, pages and widgets in order to prevent other people from using them, please use the command WDPasswordDialog. A common example would be a page containing only nodes that should not be accessed by customers using the project.

++ Web Server and Group Values++

The integrated web server [1929] now also drives the native Widget Designer interface. In addition it is completely redesigned to work more efficiently so that viewing pages and using controls is much more fluent than before. The web server can host multiple sessions of Widget Designer. This means that multiple pages can be controlled independently of one another from an unlimited amount of clients. This is perfect for installations where several remote controls are needed with a centralized control station. The later requires the optional feature "Unlimited Web Clients".

If you are familiar with Session values from version 4.7, you will understand Group Values very fast. By adding Widgets to a certain Group, they are not synchronized on all interfaces, the main Widget Designer interface and browser interfaces, anymore. Each interface can either have individual Widgets, whereas some Widgets even

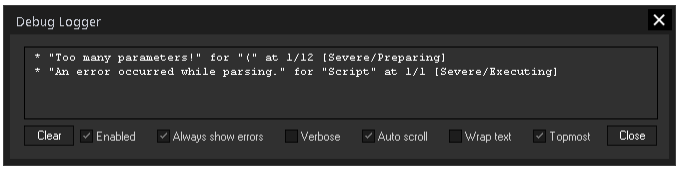
support different styles, or Widgets are synchronized for certain Group participants. Read more in the chapter Group Values [1933].

++ Major Changes in the Scripting and Member notation ++

The Script Language is improved. The Script Assistant offers commands only, when an associated Widget is part of the interface already. In addition, it helps you filling out the commands by telling you the format, e.g. ID needs to be an integer. It also offers an example for the chosen command. The Debug Logger (explained below) also helps you with scripting.
In addition, there are some new rules when it comes to scripting. The Scripting Cheat Sheet [1896] sums up all changed and is a fast guide line for all who are familiar with scripting in WD.

Member and Session Values are now referred to as Member and Group Values (which were explained above), to work with them please read the topic Object and Member Notation [1904].

++ Debug Logger ++



In any Script Field you can choose from the right-click menu to "Test" the entire script or to "Test Selected Lines". This will execute your commands and in case there is an error or warning, the Debug Logger opens and shows it.

++ Composite Nodes ++

A compilation of numerous nodes can be integrated in a re-usable custom node for further use. This drastically reduces the amount of required nodes, makes changes to the system much more efficient and enhances the overview when it comes to extensive node systems. See the chapter Composite Nodes [1254]. (Examples will follow shortly...)

++ Copying Nodes is Faster ++

Node dependencies are now also copied. For example, if you copy a Label and a Label output node that refers to it, the pasted node will refer to the pasted Label

++ New Widget InputBox and Added Features to Existing Widgets ++

The InputBox can be best described as a Textbox with a single line only.

A Label has now the ability to execute a script when being clicked on.

The TreeView is now usable to upload from external web browsers.

++ 64bit and New Project Data Format++

Widget Designer has transitioned to a 64bit application, so please make sure to run the Widget on an appropriate operating system. Due to this you will find the data in the 64bit application folder "C:\Program Files\Christie".

Additionally, the data format of the project has changed to the json format (JavaScript Object Notation).

Christie
Pandoras Box

Part 5

Product Overview

5 Product Overview

The Pandoras Box product family consists of hardware and software for Windows 8.1 / 10 Professional 64bit systems.

Introducing the Pandoras Box License Model

With version 8 we have fundamentally simplified the license structure. The "Pandoras Box Software License" replaces six products, the Server, Player, Compact Player, Software Player, Manager and Educational License. The new license allows real-time 3D rendering, 8K video playback and support of 10bit color depth whilst offering unlimited Sequences, Video Layers and Effects. In addition, this license can run on any hardware system, custom hardware included. In other words, the full feature set of PB is accessible to everyone.
The Pandoras Box Software license can be started in two ways. The "Client" or "Master" mode.

The **Client** [316] simply shows a (full screen) rendering window and connects via network to the Master from which it receives all commands and media files. Connect your display, projector, LED wall etc. to the Client. The number and type of physical outputs depend on the graphics card whilst the number of software outputs depend on the number of licenses. For each license you get another Output Layer that you can link to a physical output. Of course, you can always add more Clients to the system. Pandoras Box is a scalable playback system. All Clients communicate directly via TCP/IP with the Master control system. A synchronization process takes care of all connected Client devices to be controlled in sync. Read more about the Master / Client remote setup [67].

The user interface of the [117] **Master** [117] is divided into various tabs used for file management, timeline programming and configuration in general. It is a timeline based show control solution, allowing fast and easy on the fly programming and show setup. As everything happens in real-time, interactive show and media control is always possible.
Each Client offers various Layers including Video Layers and Output Layers for example. Read more [318] about the different layer types and their parameters. You can either work with Clients that are connected via network or pre-programm your show and connect later.
The Master has two configurations. Per default it starts without optional Output Layers which enables it to be part of a Multi-User [234] session. This feature allows several Masters to work on the same project. As soon as you add outputs to the local machine, you can use them in fullscreen mode instead of the preview rendering. As above, more licenses enable more outputs. In this configuration you cannot connect as a Multi-User to other Masters anymore, but the connection to other Clients is still possible. However, keep in mind that rendering needs performance. For extensive shows, we recommend to use the Master only for show control.

The **Offline License** is used for show pre-programming and video editing and looks just like the Master mode of the Software License described above. The major difference is, that it cannot connect via network to any system, hence the name "Offline". Other differences are listed in the product structure table [63] at the end of this chapter.

With the free **Demo version**, which is valid for 90 days, you have access to the same feature-set as the Offline License. Note that the Preview and audio output contains a water mark.

Project files can be opened and saved with either license / version.

Widget Designer

Widget Designer offers an advanced control surface creation framework, that lets you create dedicated user interfaces and interaction logic. The great number of features, including a dedicated script language [1511], are all based around the idea, that even non-programming specialists should be given the tools to create truly immersive interactive experiences for their customers.

You can add Widgets to your interface, e.g. buttons and faders. The Widget Designer interface is based on HTML-5 and CSS3 styles can be applied optionally. The integrated Web Server can publish all created pages to external browsers on any kind of device which allows to remotely control shows with customized self-designed interfaces from a tablet, smart phone or computer.

In addition, visual node based programming is available to create customized show-control scenarios. With the embedded node programming environment users can route and set up almost any possible control scenario. Easily interact with sensors and data sources to route input data to any other output protocol such as Art-Net, Midi/MSC, TCP/UDP, DMX or RS-232/-422 devices.

Widget Designer comes in three editions: a free version with a reduced feature-set, a regular edition and one offering unlimited Web Clients. The below table gives a rough overview of the differences, the introductory page of the chapter Widget Designer [786] provides a more detailed description.

Hardware and Accessories

As said above, Pandoras Box and Widget Designer can be installed on your custom hardware or be purchased with our hardware. We offer two options. The Server hardware and Compact Player.

The **Server hardware** is 19" wide and 4 rack units high and includes server grade components built for 24/7 use. There are three Server hardware models, called performance kits PK1 - PK3, which define the amount of hard drives (SSD) and RAM. The number of physical outputs is defined through the graphics card, for which we offer three choices from NVIDIA. A maximum of four outputs can be used per card simultaneously. An LCD interface allows for basic setup of the server such as network, output configuration or to select a test pattern without the need of setting up a network before. The hardware can be additionally equipped with optional boards [1947], e.g. video input cards, Dante audio cards, etc.

The **Compact Player hardware** is only one rack unit high and roughly half 19" wide. Three performance kits (PK1 - PK3) define the amount of hard drives (SSD). The Compact Player cannot be configured with optional boards.

The playback performance depends on the content format, its resolution and frame rate. The Download-Center includes a performance sheet listing many examples.

Pandoras Box is designed to interact with multiple protocols and output devices. Beside Pandoras Box products, other SMPTE, DMX, Art-Net, serial RS232 / 422, or any TCP/IP devices might be remote controlled from any timeline as well. This allows perfectly synchronized show control.
We offer specialized **interfaces** for SMPTE I/O, DMX, serial and sensor control.

All accessories and optional features are described in Hardware and Accessories [1938] whilst the programming is detailed in Devices and Layer Parameters [318] and External Control [706].

Additional Tools

The chapter "Tools and Other Software" [2051] explains all software solutions offered by Christie Pandoras Box. Amongst others, you may read about the PB Menu [2097] which is pre-installed on Pandoras Box Servers or Compact Players. It starts automatically when booting and gives access to the most needed actions. The PB Menu includes another software called the Server Management Application [2109]. With that software you can manage various hardware systems in your network and establish a VNC connection to them.

Product Structure

The number of software outputs for the "Software License" is defined with the number of licenses. Multiple licenses can be stacked by combining dongles or they can be applied to a single dongle. The number of physical outputs is defined through the graphics card.
For the hardware we offer hardware editions called "Performance Kits" (PK) which define properties like size of memory and RAM.

In the Download-Center on our web site (under Technical Documents) additional hardware information is offered in the "Technical Specs" and "Performance Specs".

Pandoras Box	Software License	Offline License	Demo Version runs 90 days
Outputs	1 per license	No	No
Preview	Yes	Yes	Yes (watermark)
Network Sync	Yes	No	No
Multi-User	Yes	No	No
Base Parameters	3D	3D	3D
Video Playback	max. 16K	max. 16K	max. 16K
Video Layer	Unlimited	Unlimited	Unlimited
Audio Tracks	Unlimited	Unlimited	Unlimited
Notch Layer	Yes	Yes	Yes
Effects, Particles	Unlimited	Unlimited	Unlimited
Warping, Keystone, Softedge, Matrix	Yes	Yes	Yes
Sequences *	Unlimited	Unlimited	Unlimited
Input Recording *	Yes	No	No
Media Encryption	De- and Encryption	Decryption	Decryption
Playlists	Yes	Yes	Yes
NDI Inputs	Yes	Yes	Yes
Dante Support	Yes	No	No
Web Browser	Yes	Yes	Yes

Widget Designer	Free	Regular	Unlimited Web Clients
Number Web Clients	1	Unlimited	Unlimited
Asynchronous Web Sessions	No	No	Yes
Built-in Script Language	Yes	Yes	Yes
- Variables, Macros, Functions	No Editing	Yes	Yes
- Debug Logger	No	Yes	Yes
Nodes	No Editing	Yes	Yes
Connections (e.g. TCP, UDP, Art-Net)	Via "Conn.Mgr"	Yes	Yes
- MIDI Connections	1	8	8
Devices (e.g. AirScan, Camera, Phidgets)	No	Yes	Yes
Widgets (e.g. Buttons, Displays)	Limited	Unlimited	Unlimited
Widget Explorer	No	Yes	Yes
Viewer Mode	Yes	Yes	Yes

Hardware Solution	Server Hardware	Compact Player
Form Factor	4U 19" Rack	1U ½ 19" Rack
Performance Kits - SSD storage	PK1 - 1 TB PK2 - 7,5 TB PK3 - 32 TB	PK1 - 0.5 TB PK2 - 1.0 TB PK3 - 1.9 TB
RAID	RAID-0	
RAM	PK1 - 24 GB PK2&3 - 48 GB	8 GB
Operating System	Win 10	Win 10
Separ. Storage OS	480GB SSD	-
Processor	XEON	Intel i5
Graphics Card (NVIDIA)	Quadro P4000 RTX 4000 RTX 6000	Quadro P1000
Outputs	max. 4 DisplayPort 1.4	4x HDMI 2.0
LAN	1x 10GbE 1x 1GbE 1x IPMI port	2x 1GbE
Power	100-240V AC, 15-7.5A integ. PSU with 850W	DC 19.5V exter. PSU with 120W
Optional Boards	see below	None
Video Input	3G SDI, 12G SDI, HDMI, DisplayPort	DirectShow only
Audio Output	ADAT, Dante	
Framelock / Sync	Optional	
10Gb Ethernet	Optional	

* Available when running as a Master

Date: Jan, 2021. All information contained in this document is subject to change without prior notice.

Christie
Pandoras Box

Part 6
Pandoras Box

6 Pandoras Box

The Pandoras Box section is divided into these themes:
First Steps and Troubleshooting 67
Content 90
User Interface - Master 117
User Interface - Client 316
Devices and Layer Parameters 318
External Control (DMX, 706 Midi,...) 706
StreamiX 724 Live Input 724
SDK 730
Ports Used by PB and WD 781
Copyright Pandoras Box 783

6.1 First Steps and Troubleshooting

This chapter is for all who are new to the Pandora Box product family, it covers:

- how to troubleshoot 69 (e.g. for technicians who have to maintain a Pandoras Box installation but did not attend a training)
- the most important tabs and buttons of the user interface to do basic changes 86

6.1.1 Master / Client Remote Setup

General explanation: PB Master, PB Client

If you have only one Pandoras Box system, you start the Master mode of the Pandoras Box Software License. This configuration is also called "stand-alone mode".

If you like to link multiple Pandoras Box systems configure them as a Master-Client setup. In that case you have one Master and one or more Clients (where the so called Client mode of the Pandoras Box license is started). A Master may connect to multiple Client systems whilst a Client may only connect to one single Master system. An exception to this rule is when you are working in a Multi-User environment 234 which allows multiple Masters to work simultaneously on one project.

The Master acts in general as the main controller for all attached devices, whilst a Client system is "only" receiving commands and media to playback in fullscreen.
On the Master, you create a project which holds all information about the resource media files and connected Clients (and other devices 318). The Master takes care of the remote file management, meaning the media upload called "spreading" to all Clients; and of course the video and audio synchronization throughout the entire network. In addition, the Master sends all control parameters to the Clients and organizes the protocol patch and routing, e.g. it sends all incoming control-data like DMX or Art-Net in sync to the Clients.

Network Setup

In a Master-Client setup, the Master communicates over Ethernet to the Clients. As sad above it remote controls them, synchronizes the video and audio playback on all units and shares content. But also in a stand-alone setup you might need to send or receive data over Ethernet, e.g. Art-Net. Therefore, the network setup of your system is essential.

For a proper network communication and bandwidth, we recommend setting up standalone local area networks (LAN). In most cases, regular network cables connect the LAN port of your system to an Ethernet router or switch, but of course you may also use straight peer-to-peer connection with a cross-link cable.

To setup your IP address, you can either make use of the PB Menu 2097 (Settings > Network 2107) or directly the Network Settings of Windows, i.e. the Properties dialog of your network adapter. Even if your router supports DHCP, we recommend to use fixed IP addresses. Make sure to choose the same IP range for all systems that should be able to communicate with each other and an according Subnet mask.

Make sure that the firewall and anti-virus software does not block any communication from Pandoras Box. For advanced network administration, e.g. if routers include filtering or an additional firewall, please see the list of all TCP and UDP ports 781 Pandoras Box uses.

In the Pandoras Box Master, the IP is shown in the Assets tab [131] and the network adapter can be chosen in the Configuration tab > Network [147]. The Client [316] shows the IP and network setting next to the Preview window when the fullscreen mode is off.

Software Version and Domain

Next to the network setup, Master and Client systems also need to use the identical software version and the so called Domain channel.

The version of the Master is shown in the Title Bar and bottom right corner [125] of the interface. In the Client GUI [316] it is displayed next to the Preview window when the fullscreen mode is off.

The domain channel is set to 0 per default and should be changed only if multiple Master-Client networks are needed which should not interact with each other. In the Master, it can be found in the Configuration tab > Network [147] and the Client displays it next to the Preview window when the fullscreen mode is off.

Controlling Clients in the Master Interface

When the network, the version and the Domain are set up correctly, all Clients appear in the Assets tab [131] of the Master. The Clients in return display the Master's IP instead of "not connected".

To start controlling the Client, for instance assigning media files to it, drag the unit from the Assets tab into the Device Tree tab [173]. Now the Client is in the project and files can be transferred across the network.

To include media files to your project, drag the file from the Assets tab to the Project tab [278]. Per default, all added media files are auto-spread to the available Clients (that are part of the Device Tree). To share files or entire folders manually, you right-click on the entry in the Project tab and choose the according "Spread" command.

Now you can start assigning the sources (media files) to the outputs (Layers on the Clients). This is explained in the chapter Programming on the Timeline [296].

Please note:
When re-starting the project on the Master with unconnected Clients, they will show up with a red"!". Once the Clients are started, they will automatically reconnect to the Master

Programming with Clients that are not connected yet

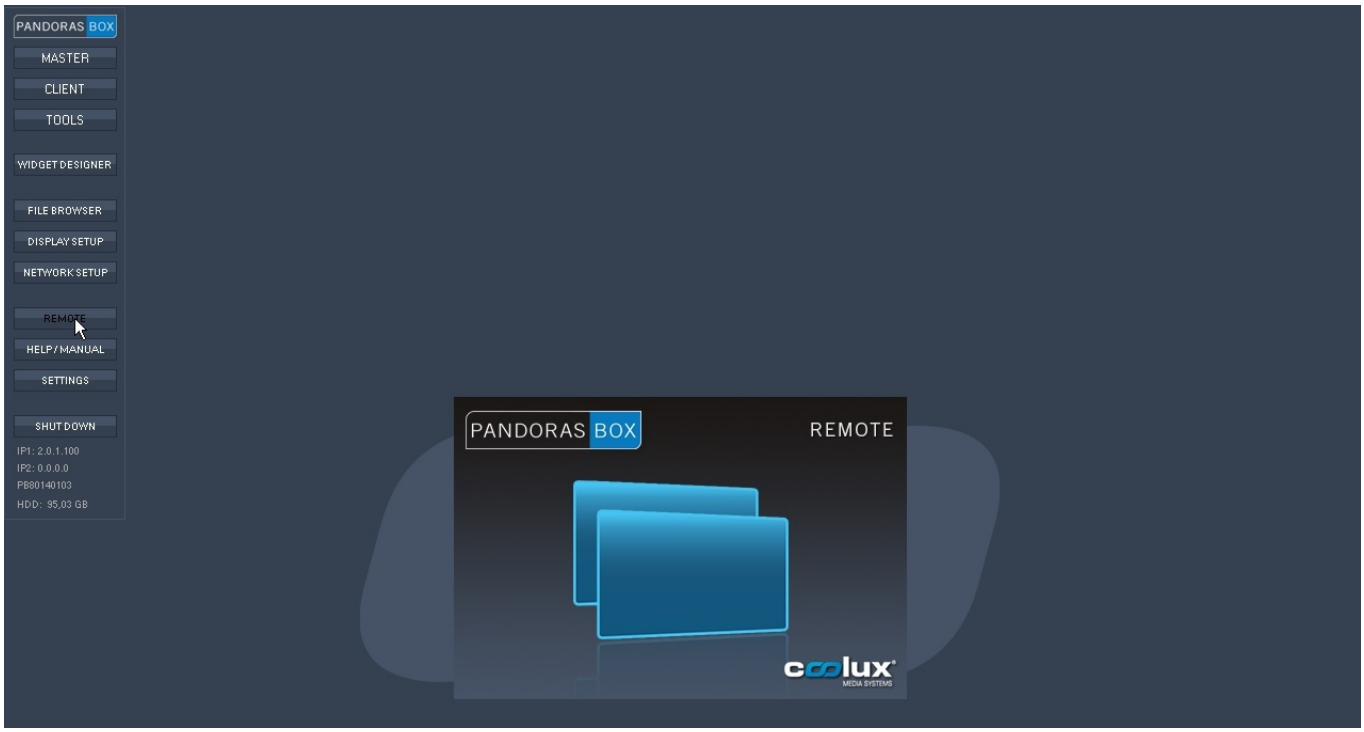
If you like to prepare a show for a Client that is not connected to the network yet, you can simply drag a Client from the Device Types tab [183] to the Device Tree tab [173]. Once you added a Pandoras Box Software License, PB asks, how many Outputs should be applied. Once the unit will be present on the network, make sure that you set the correct IP address in the Client's Inspector [210] in order to be connected to the Master. If your Site in a project contains four Output Layers, it can only manifest on a PB system with four licenses. Multiple licenses can be stacked by combining dongles or they can be applied to a single dongle.

Once the Client is connected, spread all the content using the context menu from the Project tab [278].

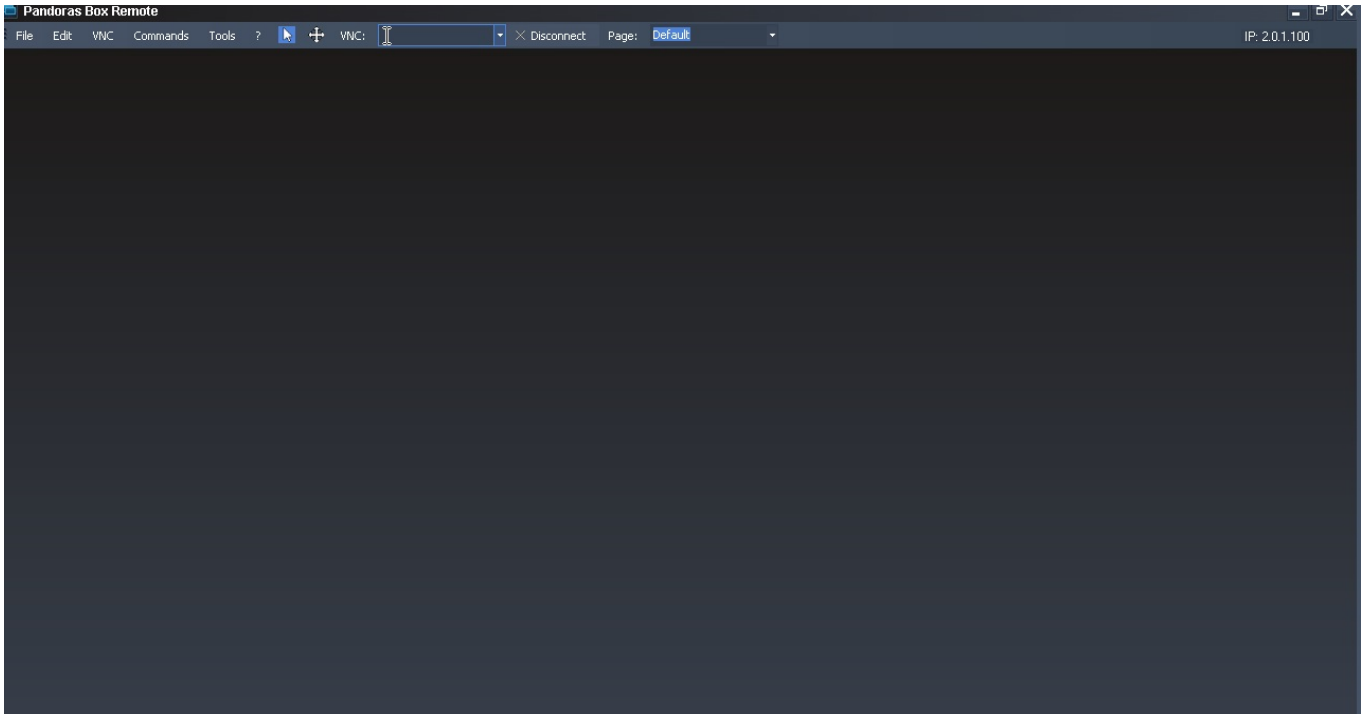
6.1.2 How to Connect via VNC Remote

This is the screen you start with. What you see is called the PB Menu. If you like to access another computer, click the "Remote" button.

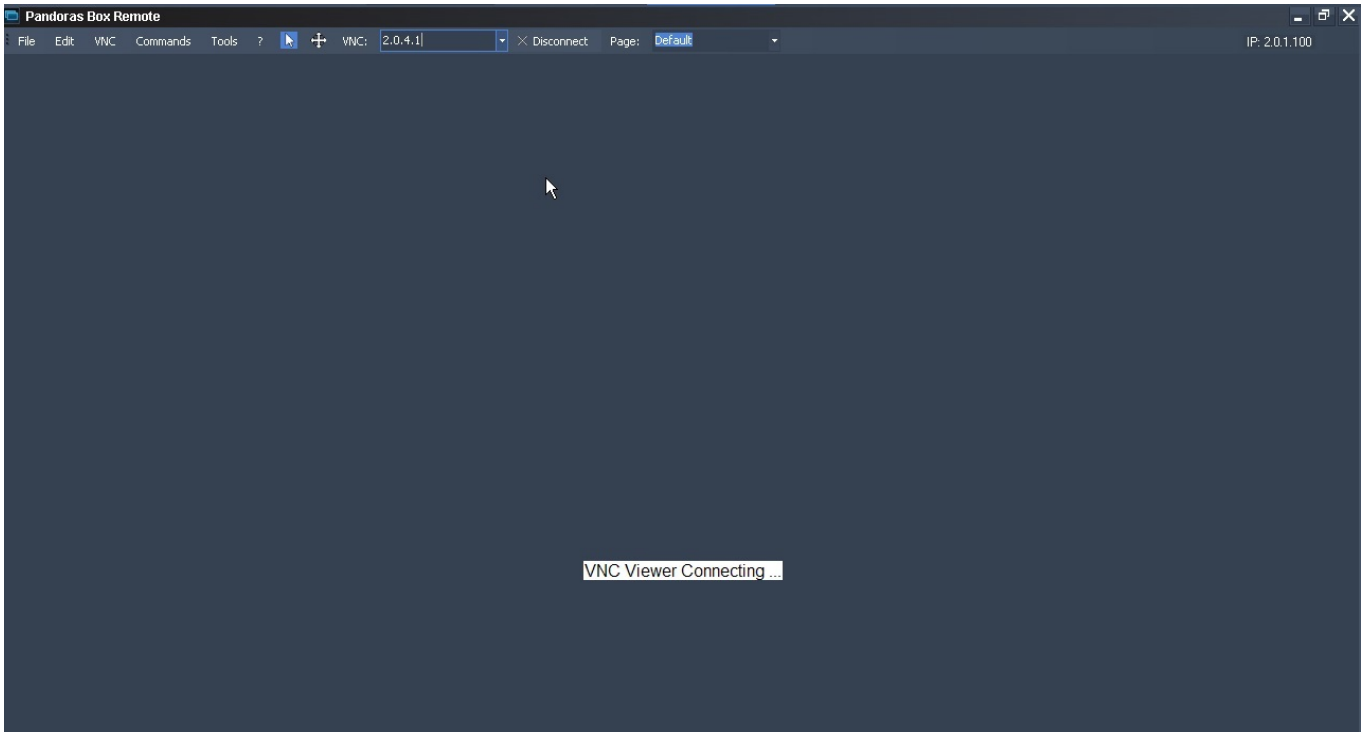
You will see a an image, while you wait for the Remote window to open.



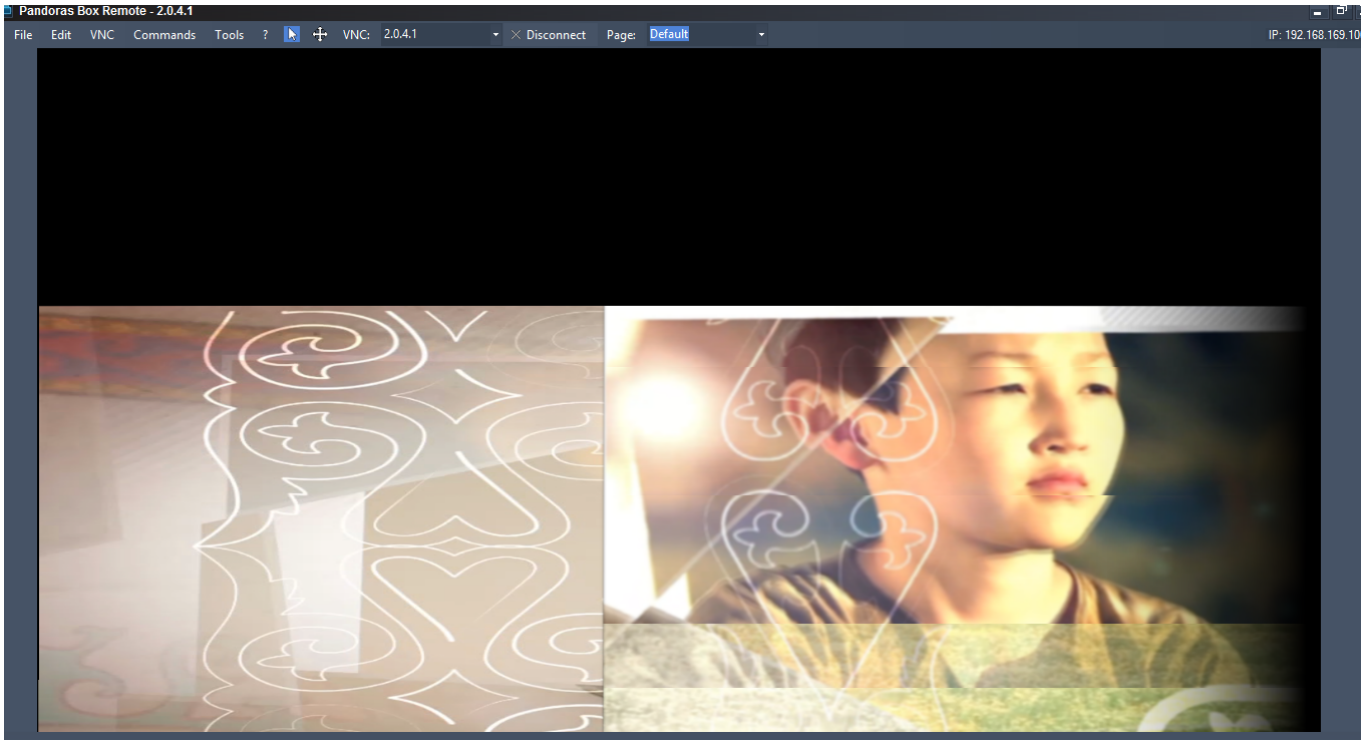
Now, you see the interface from the Remote tool. With that software you can establish a VNC connection to another computer that is in your network. The other computer needs to have a VNC client running to pick up the connection. If PB Menu is started on that computer, it automatically started one for you. Now, you can enter the according IP address in the "VNC" text field.



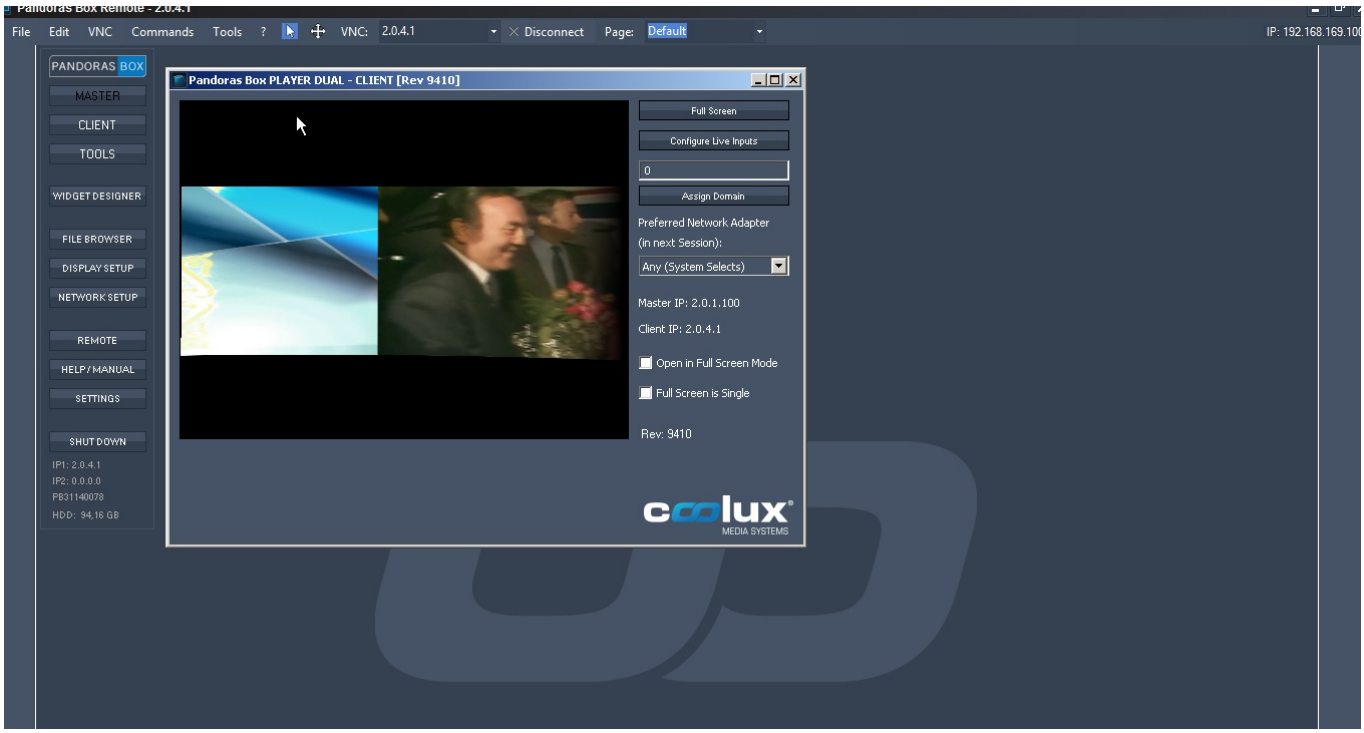
Now VNC Remote tries connect. If the connection cannot be established, check:
- if you entered the correct IP address
- if the computer is in the network
- if the network is OK, e.g. by pinging the computer



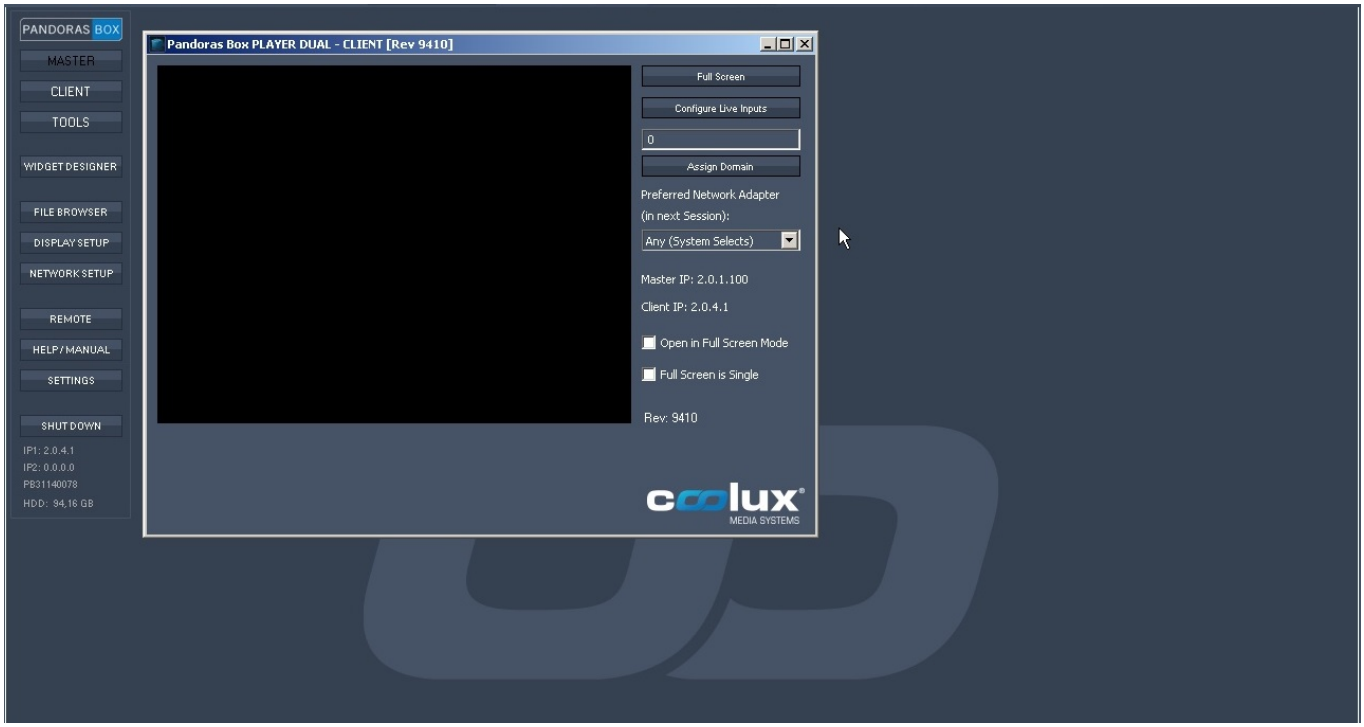
When the connection is established, you see the desktop of that computer. Click your local mouse and use the local keyboard to interact with the remote desktop.
If you connected to client computer that renders in fullscreen, you might want to leave fullscreen: click into the window and then use the shortcut [Ctrl + F]
If you connected to a Win7 client that renders in fullscreen, you do not see the fullscreen, even though it is there. Instead you see the desktop underneath it. You will notice that you cannot click something in the desktop. Do the above steps, click in the Remote window and use [Ctrl + F] and you will see the small rendering window.



Above you see the fullscreen rendering and after [Ctrl + F] the small rendering window.

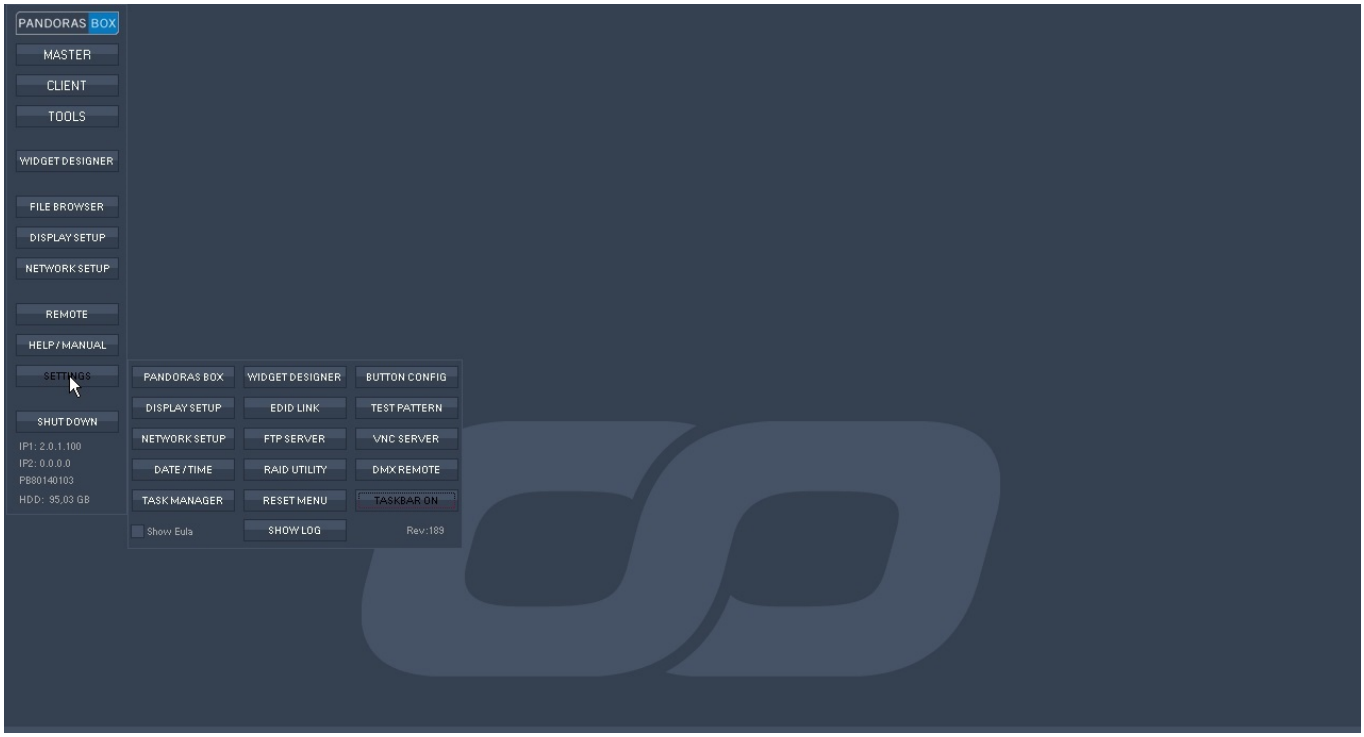


If the Client does not render anything, for example because the content was not spread, it looks like this. How to spread content is explained in the chapter "Basic steps" 86.
If there are problems, also check, that "Master IP" displays the correct IP and not the words "Not connected"


PANDORAS BOX
MASTER
CLIENT
TOOLS
WIDGET DESIGNER
FILE BROWSER
DISPLAY SETUP
NETWORK SETUP
REMOTE
HELP / MANUAL
SETTINGS
SHUT DOWN
IP1: 2.0.4.1
IP2: 0.0.0.0
PB31140078
HDD: 94,16 GB
Pandoras Box PLAYER DUAL - CLIENT [Rev 9410]
Full Screen
Configure Live Inputs
0
Assign Domain
Preferred Network Adapter
(in next Session):
Any (System Selects)
Master IP: 2.0.1.100
Client IP: 2.0.4.1
Open in Full Screen Mode
Full Screen is Single
Rev: 9410
coolux
MEDIA SYSTEMS

6.1.3 How to See the Taskbar

This is the screen you start with. What you see is called the PB Menu. It hides the taskbar. If you like to see it, click the "Settings" button.

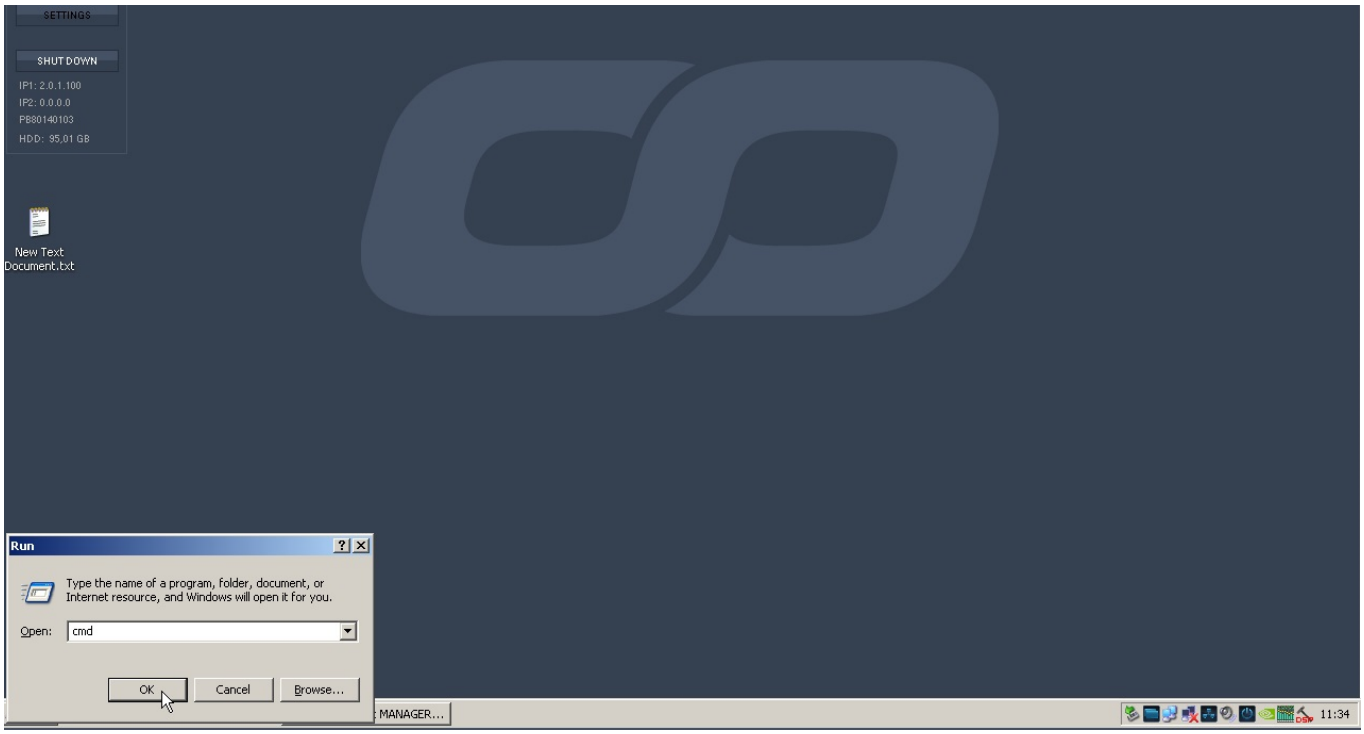


Then click on "Taskbar On".

6.1.4 How to Ping and Check Your IP Adress

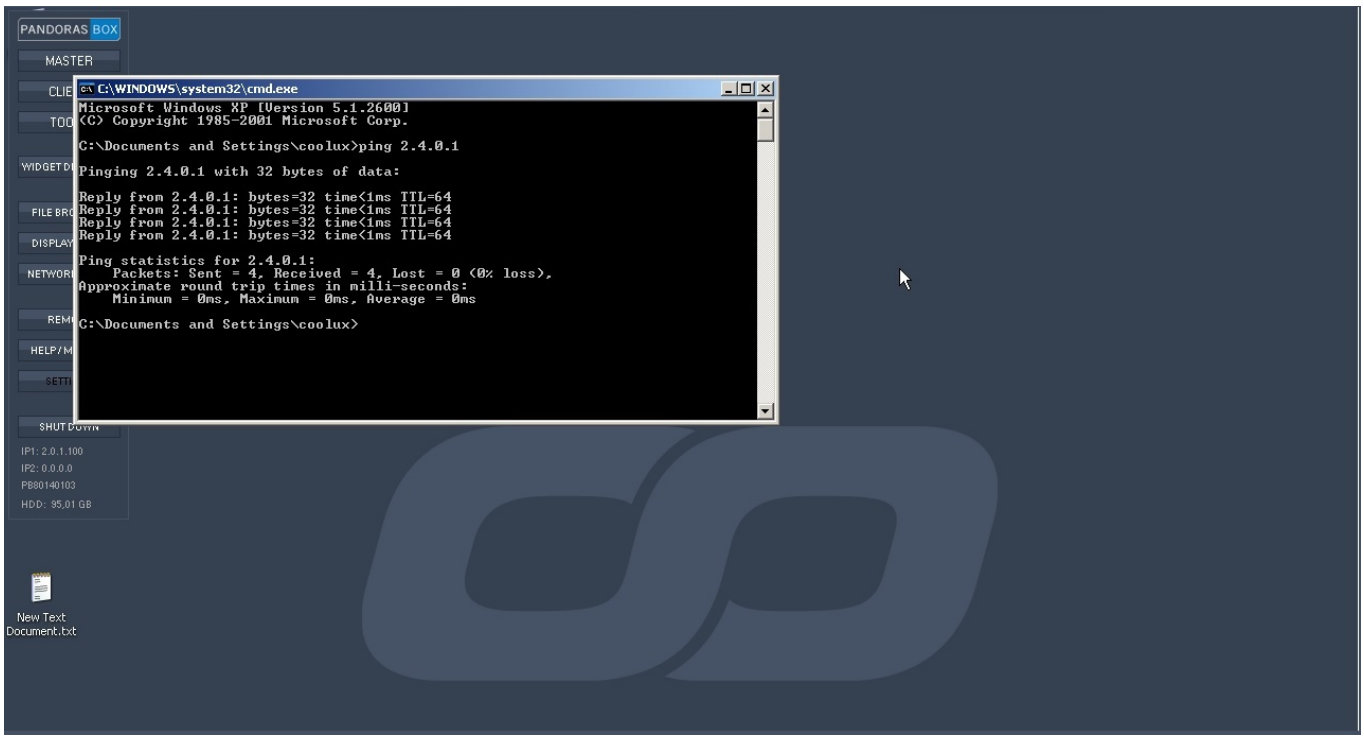
This is the screen you start with. What you see is called the PB Menu. It hides the taskbar. To see it, click the "Settings" button and then the "Taskbar On".
Now click the Windows Start button and choose "Run...".

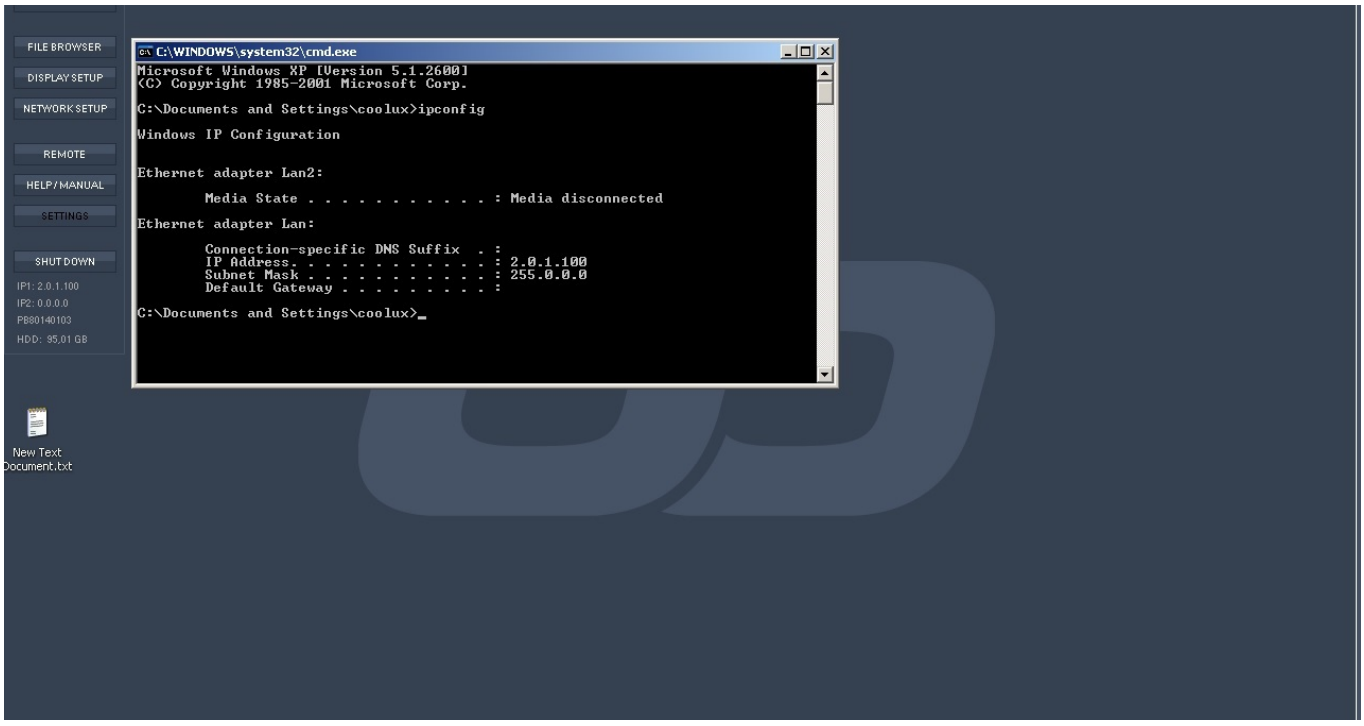
Enter the below code and click OK:
cmd



The Command Prompt window opens. Simply write (behind the already existing entry) the below codes and hit the [Enter] key.
In order to ping another computer in the network, write "ping " and the IP address. If the ping goes through it looks like the below image.

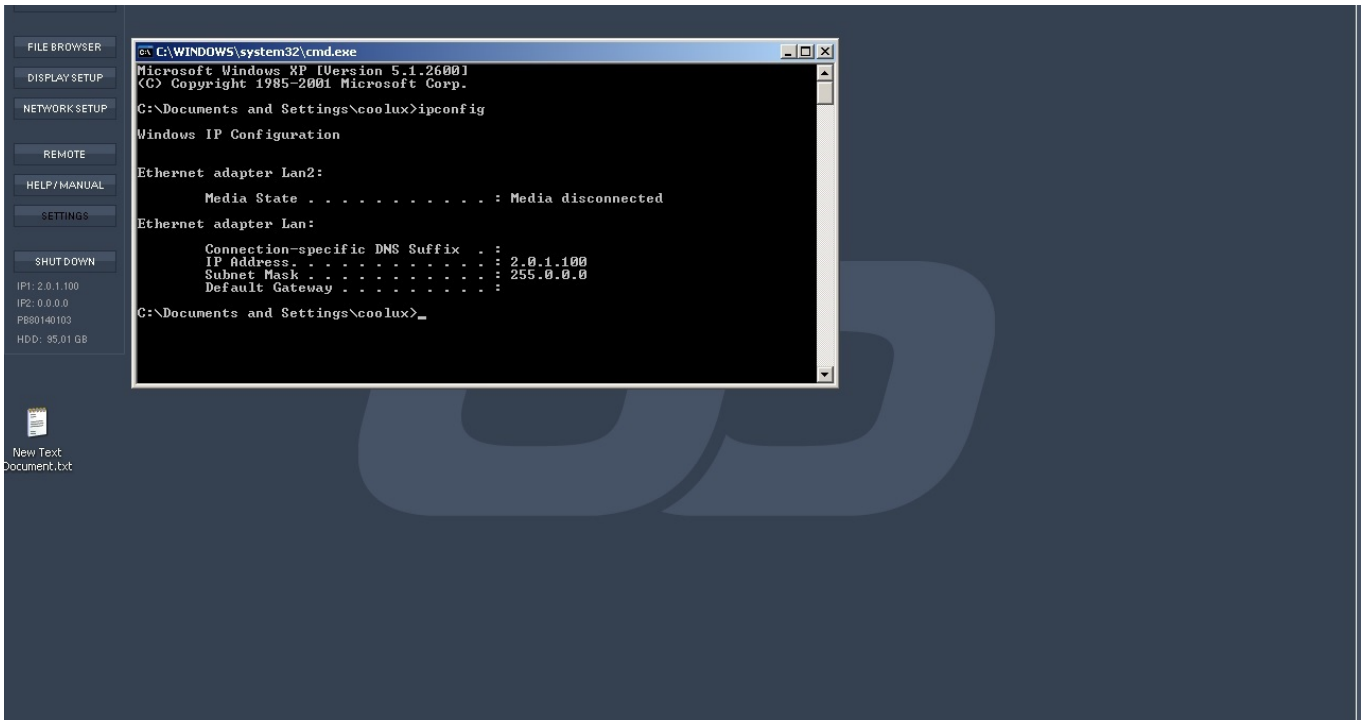
Enter for example:
ping 2.0.4.1





If the ping request is not successful, there is either:
- a fundamental network problem (loose or broken cables, overloaded or broken switches or other connected devices)
- an IP conflict with other devices, for example because other devices have the same IP address (check all IP addresses within the Windows settings or with the PB Menu button "Network Setup" or the command in the below image)
- the entered IP address is wrong. The answer takes then longer and says for example "host not available"
- entered text is wrong, check that you have not forgotten to write "ping", that there is a space and that the numbers are separated with dots

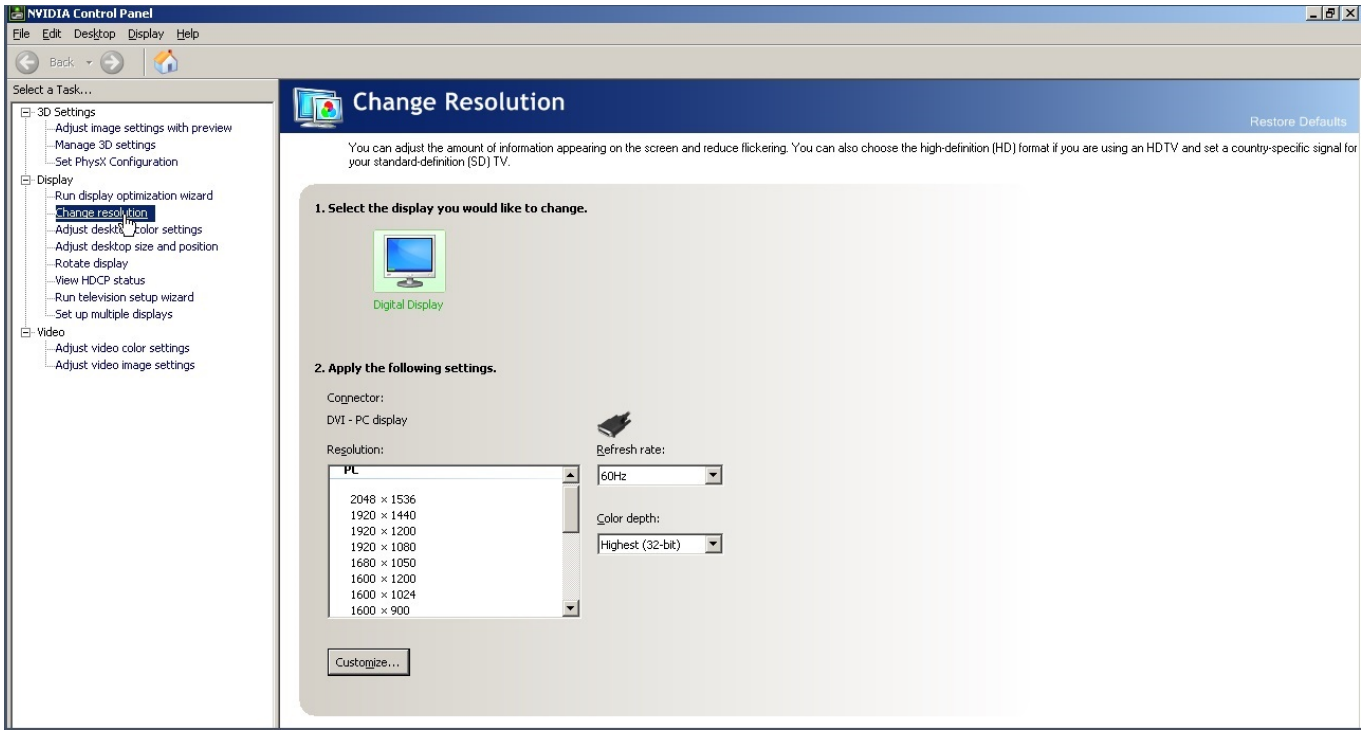
In order to check what your own IP address and the MAC address, enter:
ipconfig
For more advanced results:
ipconfig /all



6.1.5 How to Check and Change the Resolution

This is the screen you start with. What you see is called the PB Menu. If you like to check the resolution or need to change it, click the "Display Setup" button.

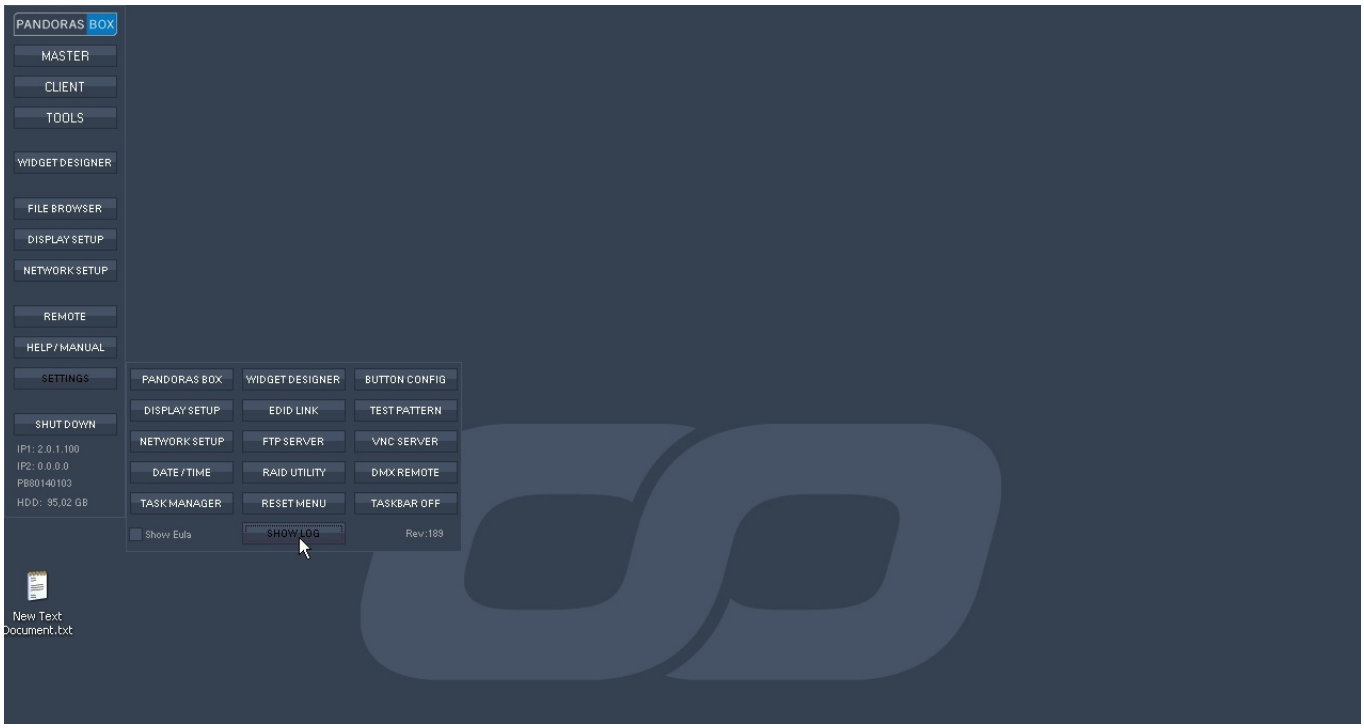
Now you see the NVIDIA Control Panel (or another window if another graphics card is used). To check the current resolution, or in order to change the resolution, click on "Change resolution". For other settings, see this chapter [1948].



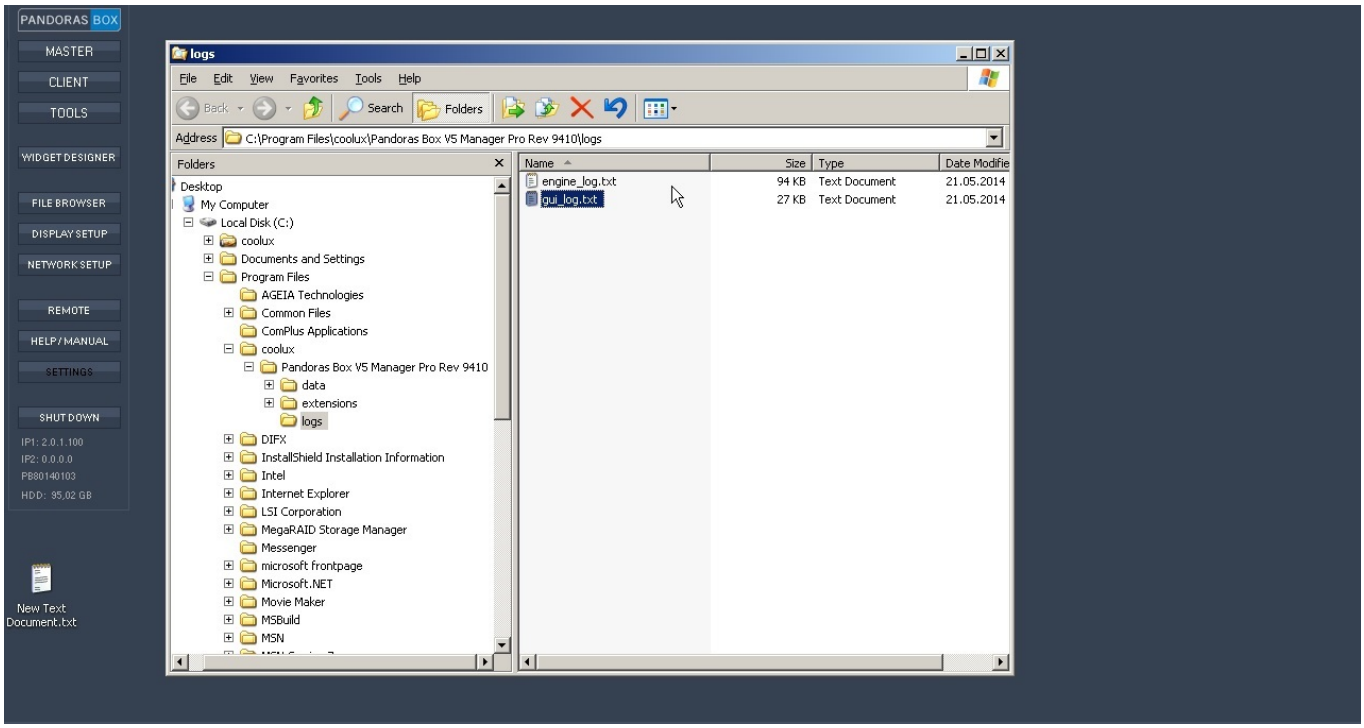
6.1.6 How to Access Log Files

This is the screen you start with. What you see is called the PB Menu. If you had problems and the technical support asked to receive your log files, first of all you need to connect to the computer you had problems with. That means, if it was a Client, first connect to it. These steps are described in the chapter "How to connect via VNC Remote" 69.

Now, click the "Settings" button and then the "Show log" button.

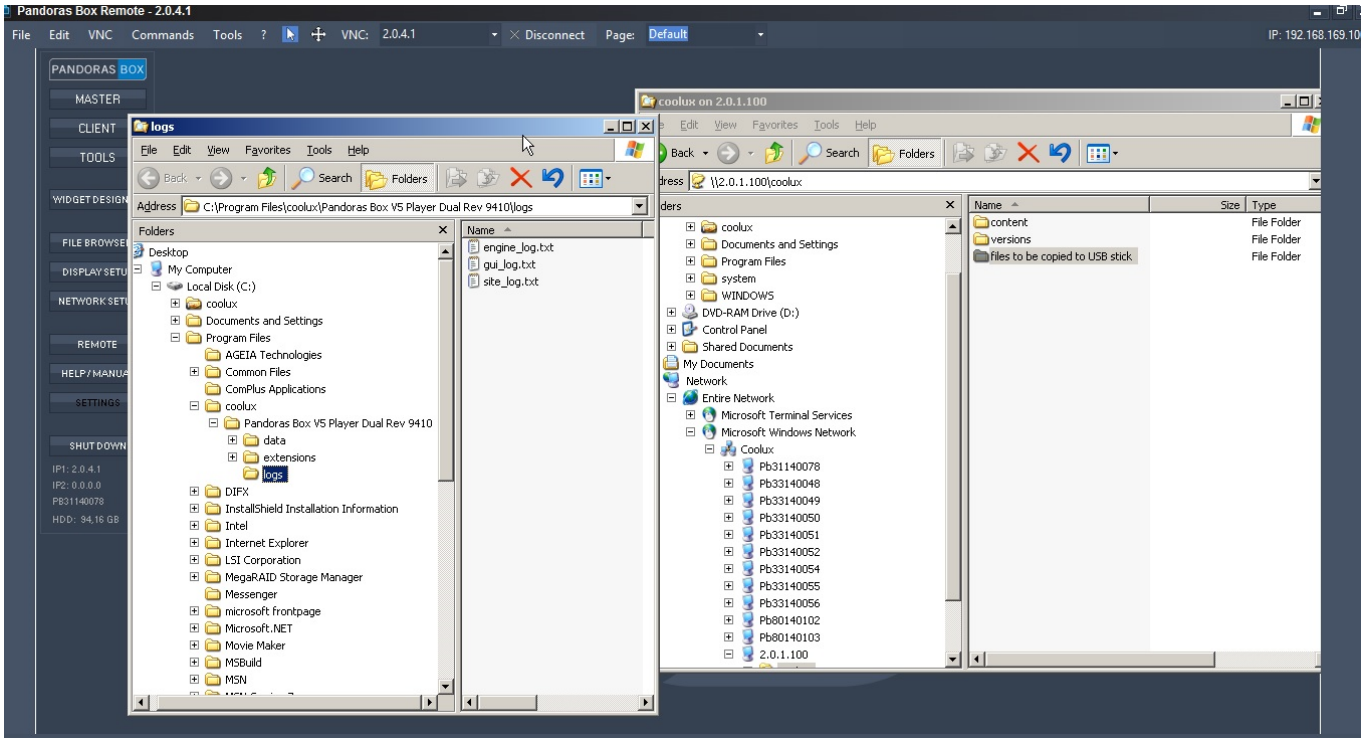


You see the Windows Explorer with the according Log folder `(C:\ProgramData\Christie)`.



Take all files that you see and mail them all to the technical support.

You might need to do that for the Master as well as for the Clients. If yes, enable a VNC connection to the Client as described in the chapter "How to connect via VNC Remote" 69. Then do the above steps to enter the logs folder. In order to copy them through the network to your local machine, open a second browser. In there, you enter two backslashes and the IP address, e.g.
\\2.0.1.100
Then you can make a new folder, in our example it is called "files to be copied to USB stick" and copy the files from the Client to that folder.



Afterwards you close the VNC Remote tool again, and back on your local PC you can then copy the files to an attached USB stick and transfer them to a PC that is online and send a mail to the Pandoras Box support.

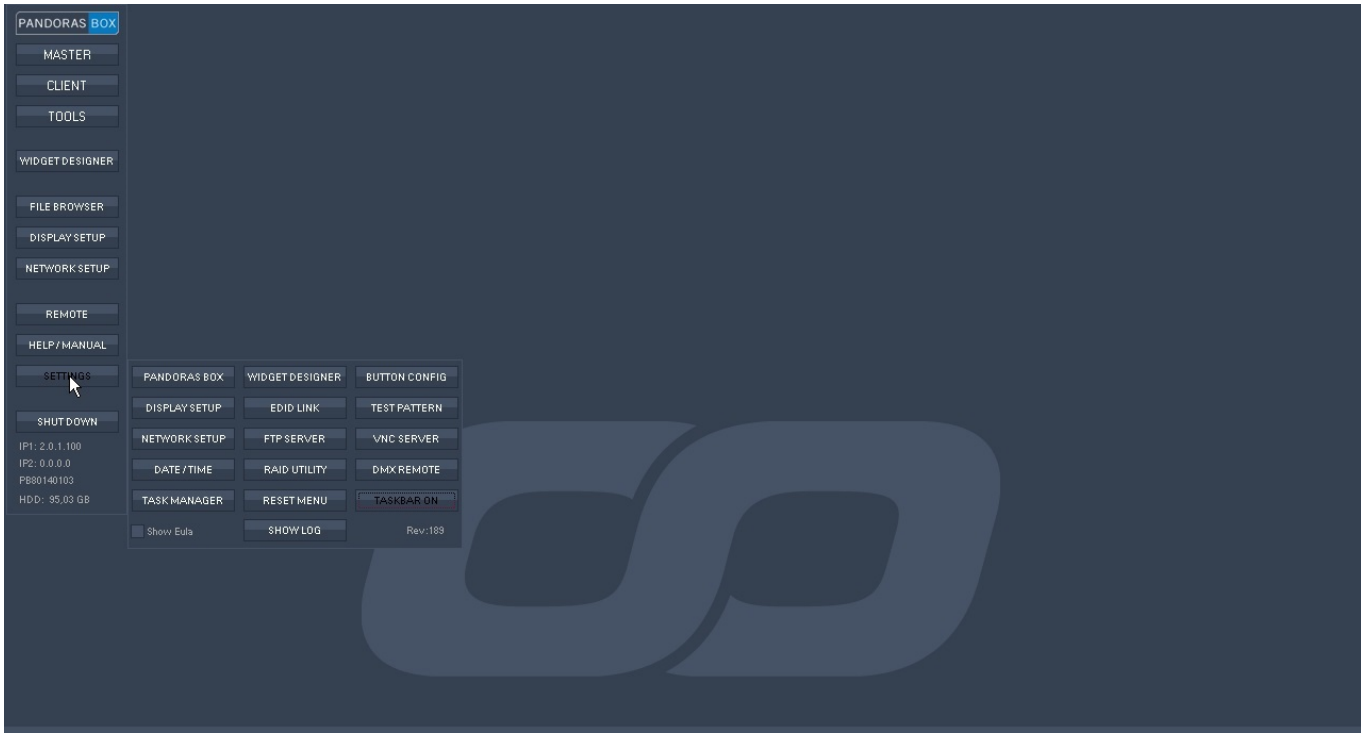
6.1.7 How to Install a New Pandoras Box Version

If you need or wish to update your current Pandoras Box version, you first of all need to download the new installer. To do so, go to christiepandorasbox.com > Support > Download-Center and log in with your user account settings. If you have not registered, please do so, it will only take a minute. Click on the link for the new version to start the download.

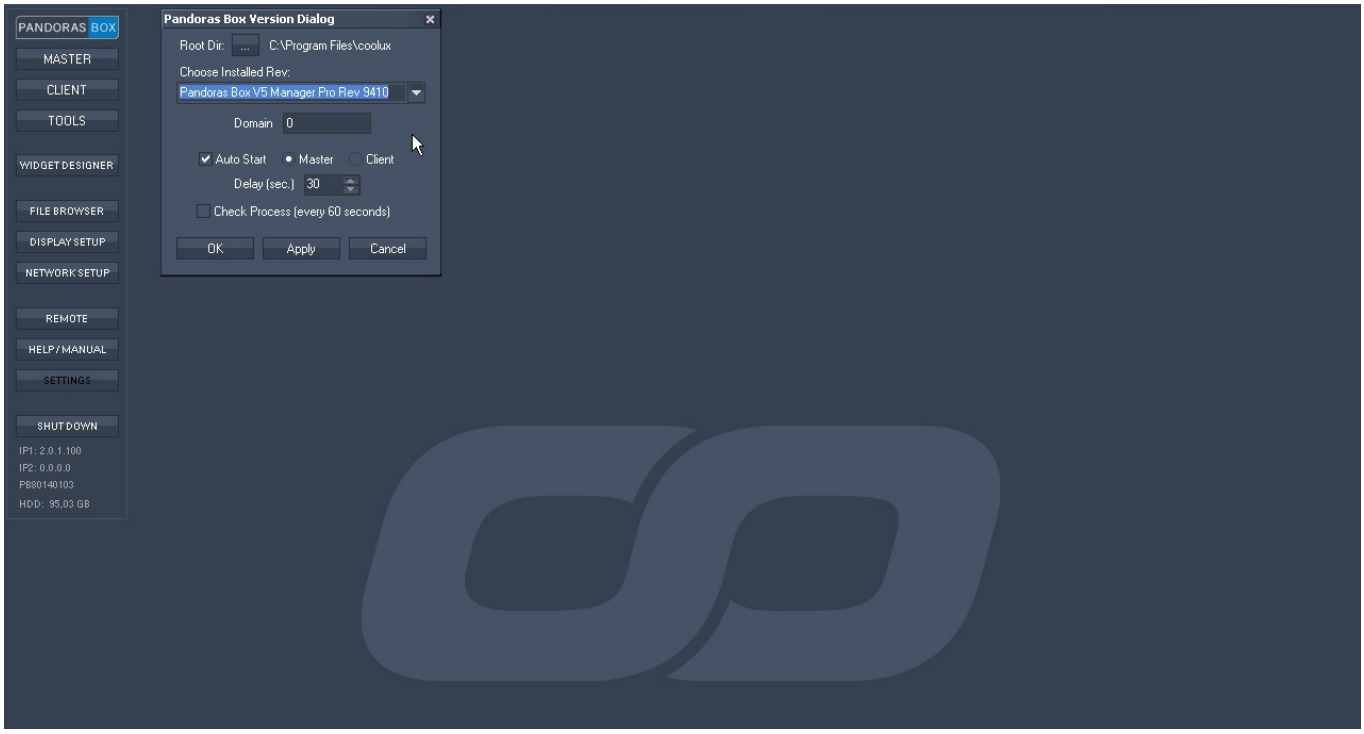
Copy the installer file to all computers that run Pandoras Box as a Master or Client. Please read the chapter "How to copy files from and to a remote PC" 80.

Run the installer on each computer.

Then, you need to tell the PB Menu to start the newly installed version when clicking the Master and Client button. If you do not do that, the buttons will start the old version. So please click on "Settings" and "Pandoras Box".



A dialog opens where you can see all already installed versions of Pandoras Box. Pick the newest version and press "OK".



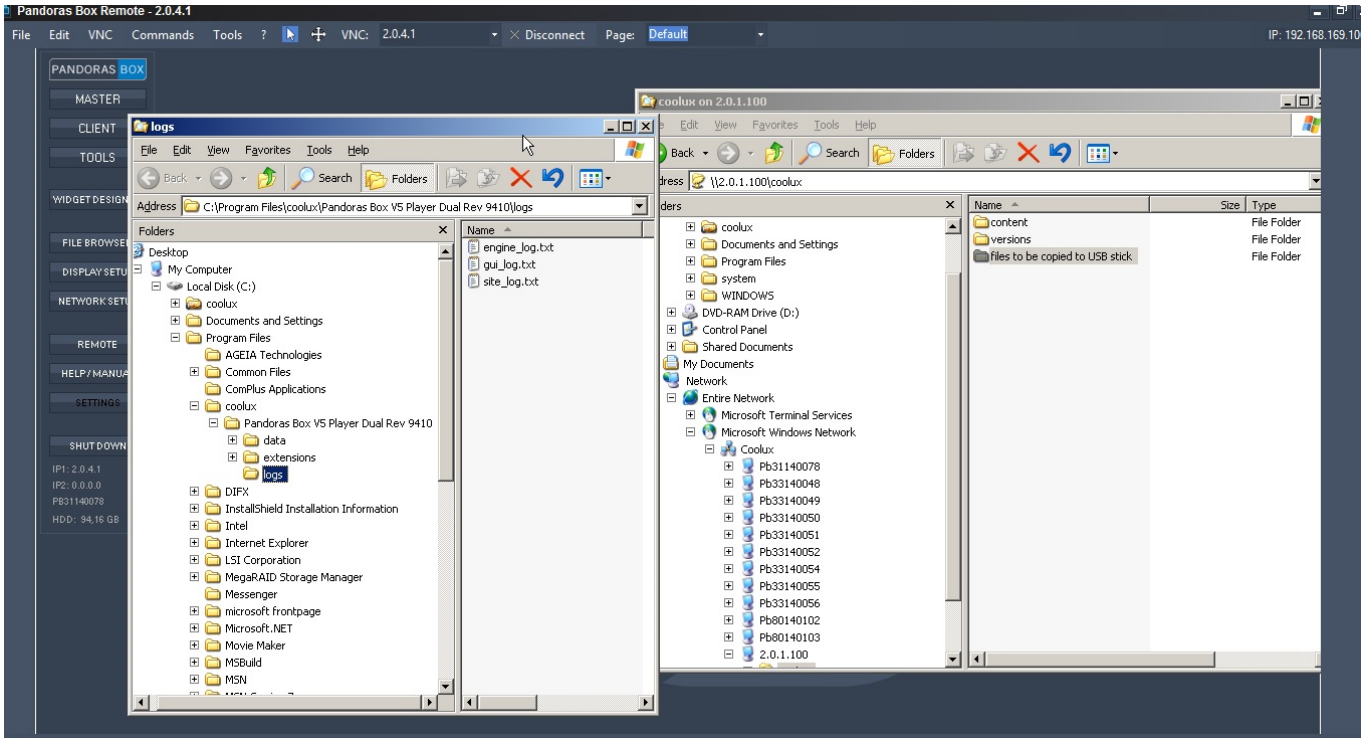
6.1.8 How to Copy Files to a Remote PC

This is the screen you start with. What you see is called the PB Menu.

This explains how to copy files to a remote PC (e.g. a new installer). Important notice: Do not follow these steps if you simply want to copy content to all your Clients. The content spreading and management is done via the PB Master.

Open two Windows Explorers by clicking twice on the button "File Browser".

In order to copy files through the network to a remote computer, enter two backslashes and its IP address. In the example below, the browser on PC 2.0.4.1 copies files to the remote 2.0.1.100, so this was entered in the second browser:
\\2.0.1.100



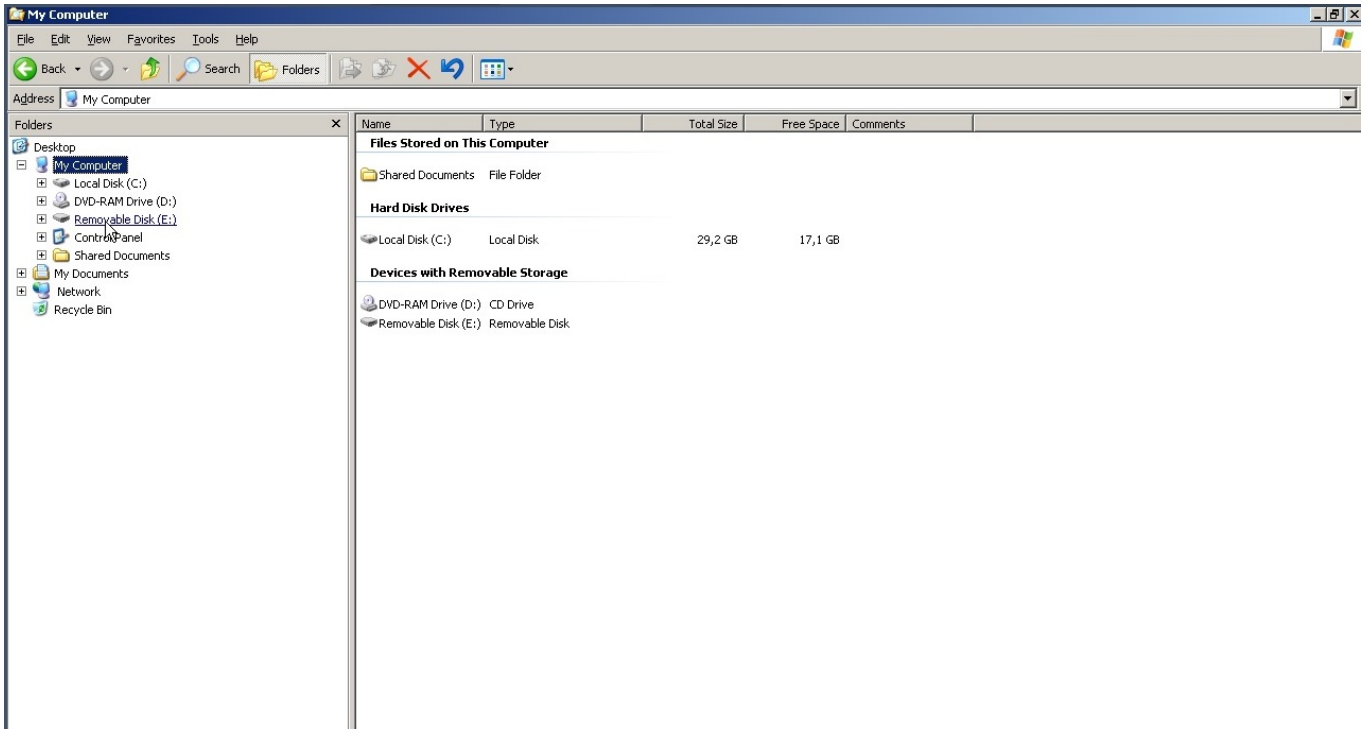
Afterwards you close the VNC Remote tool again, and back on your local PC you can then copy the files to an attached USB stick and transfer them to a PC that is online and send a mail to the Pandoras Box support.

6.1.9 How to Copy Content to Your Hard Drive

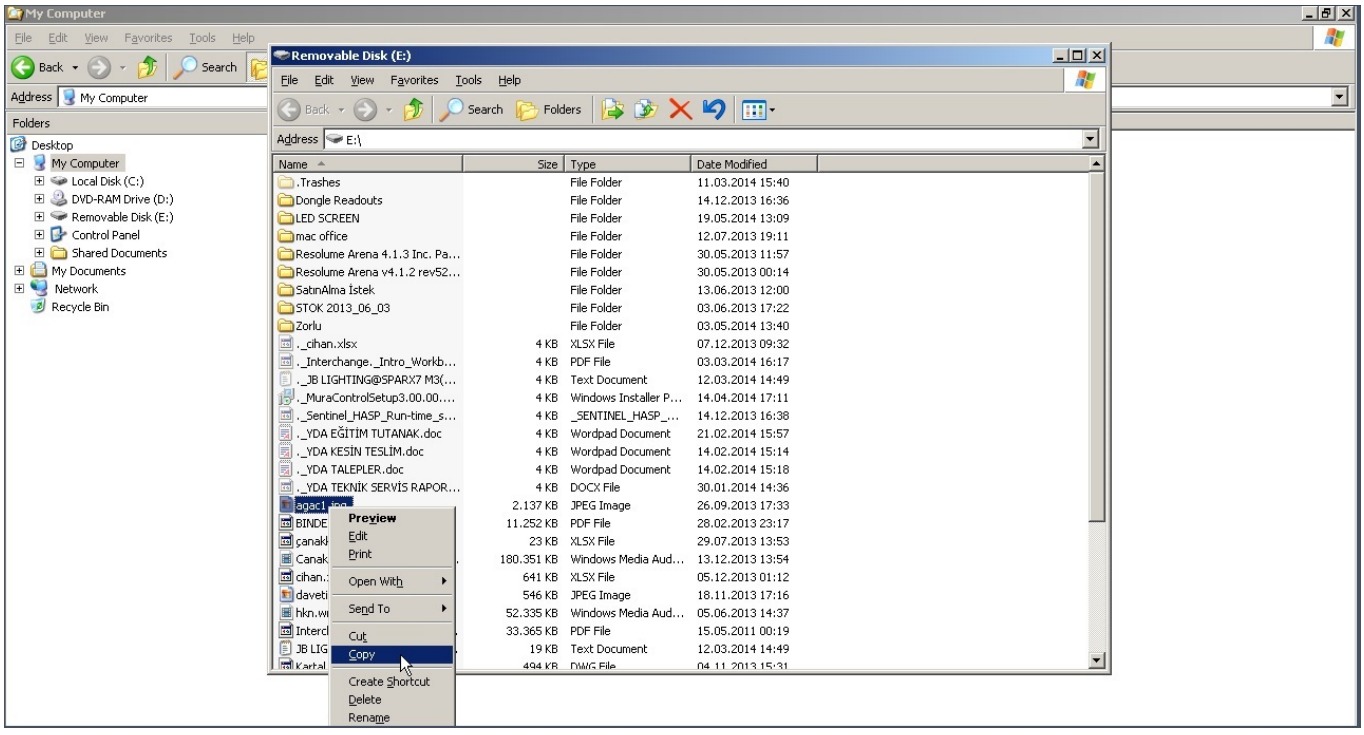
This explains how to copy files from an external hard disk, e.g a USB stick, to Pandoras Box hardware. Important Note: There is no anti-virus software running and protecting Pandoras Box hardware!! It is highly recommended to check ANY hard drive for viruses BEFORE plugging it in.

This is the screen you start with. What you see is called the PB Menu. To copy files we need the Windows Explorer, so click the "File Browser" button.

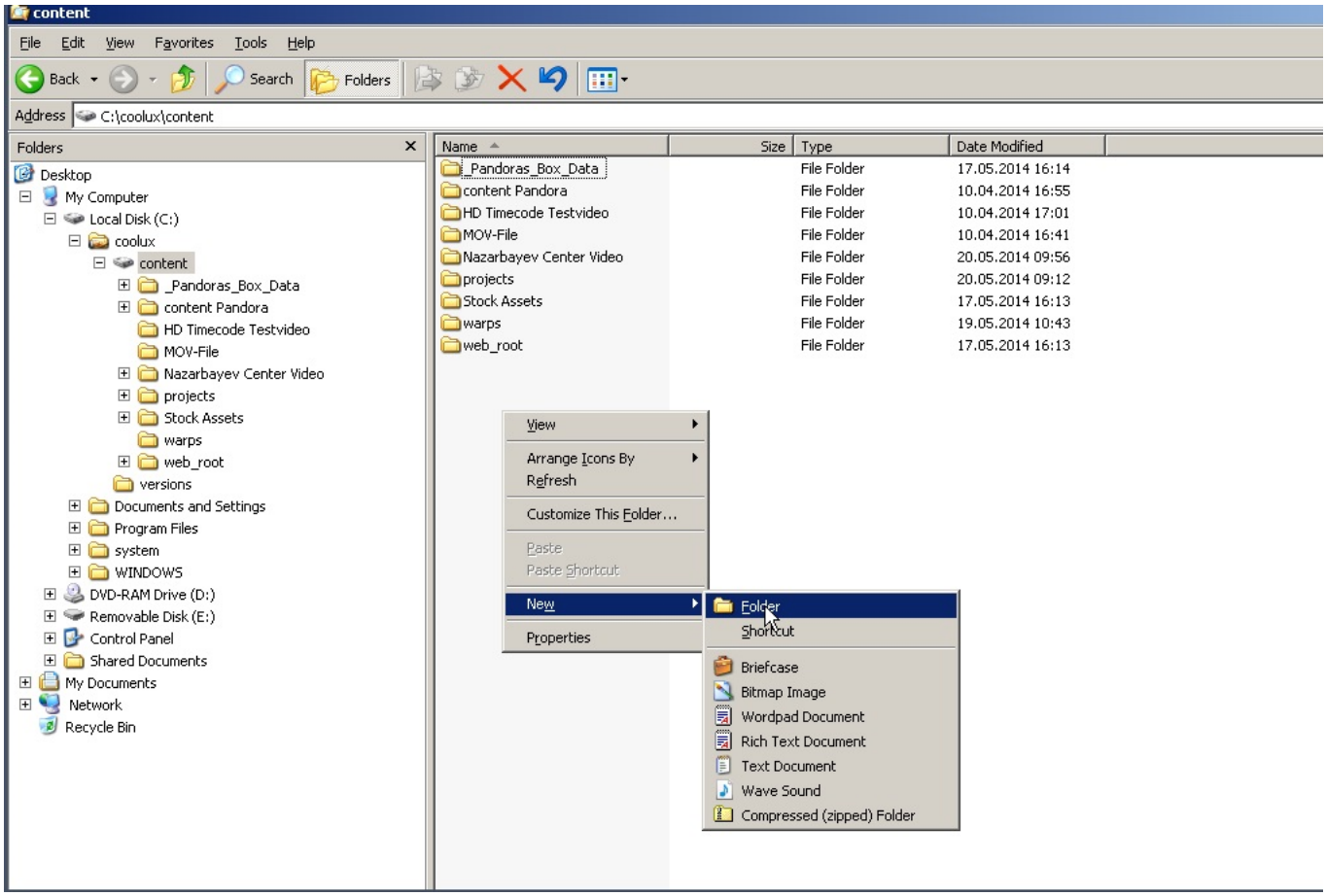
In the explorer, go to the drive "Removable Disk".



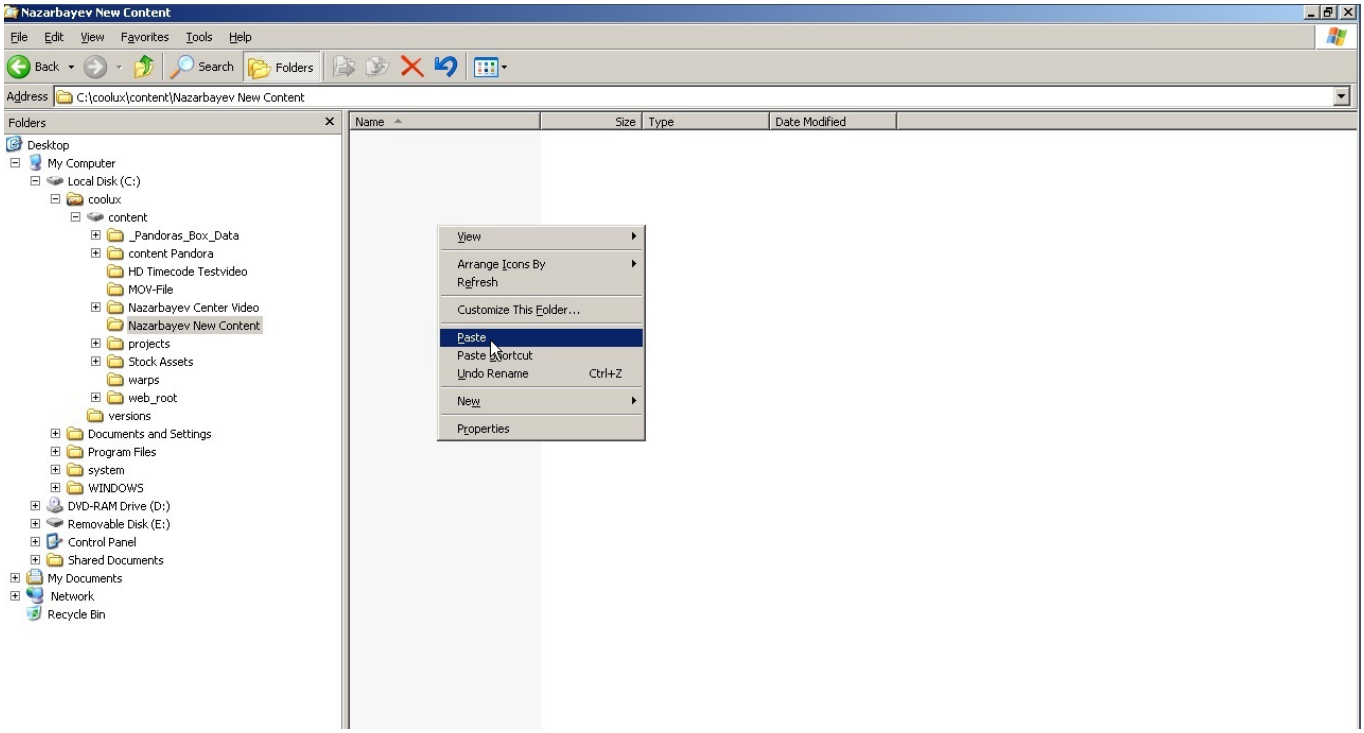
Search for the files you need. Make a right-click on them, and say "Copy". You can also use the shortcut [Ctrl + C].



Go back to the "Local Disk (C:)" > coolux > content and the folder that already exists.
OR
Go back to the "Local Disk (C:)" > coolux > content and create a new folder. Choose a good name.



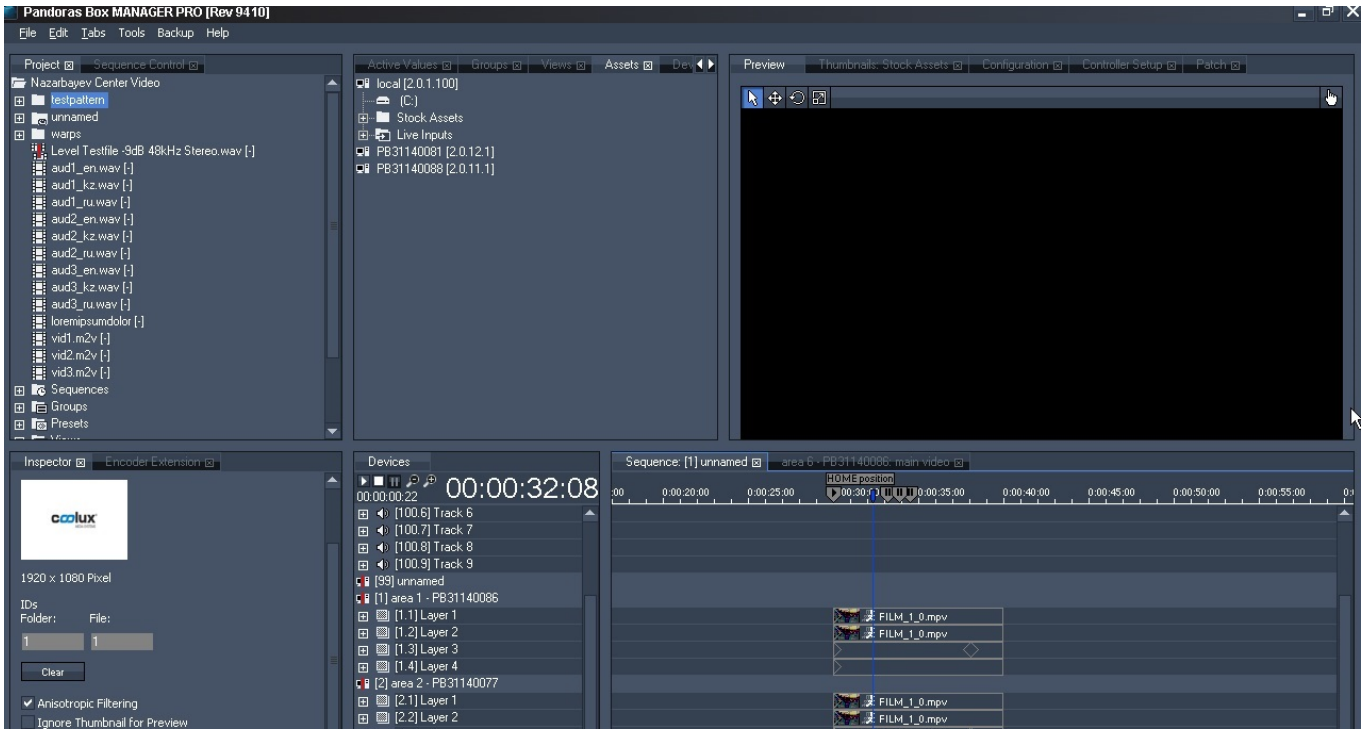
Paste the content into the folder with a right-click and "Paste" and the shortcut [Ctrl + V].



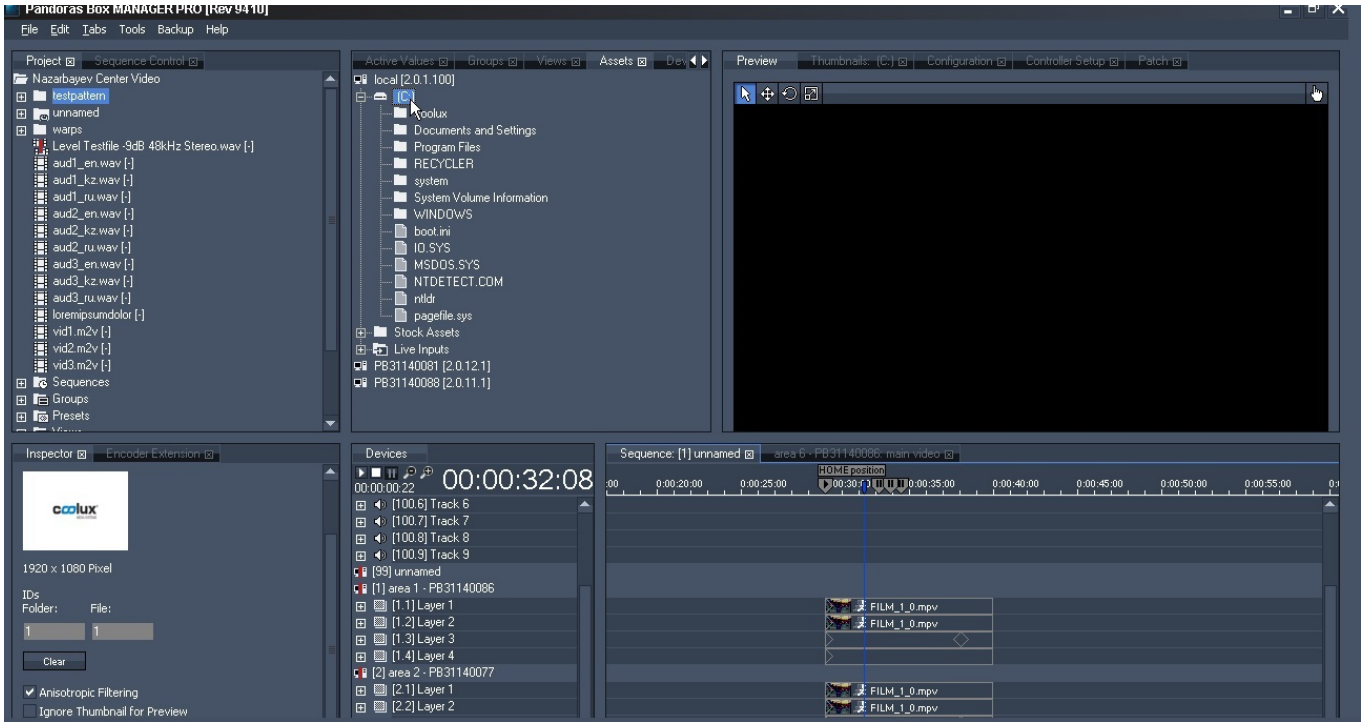
6.1.10 How to Copy Content to the Project

The last chapter explained how to copy content from an external hard disk, e.g a USB stick, to Pandoras Box hardware. Now, it needs to be included in the Pandoras Box software.

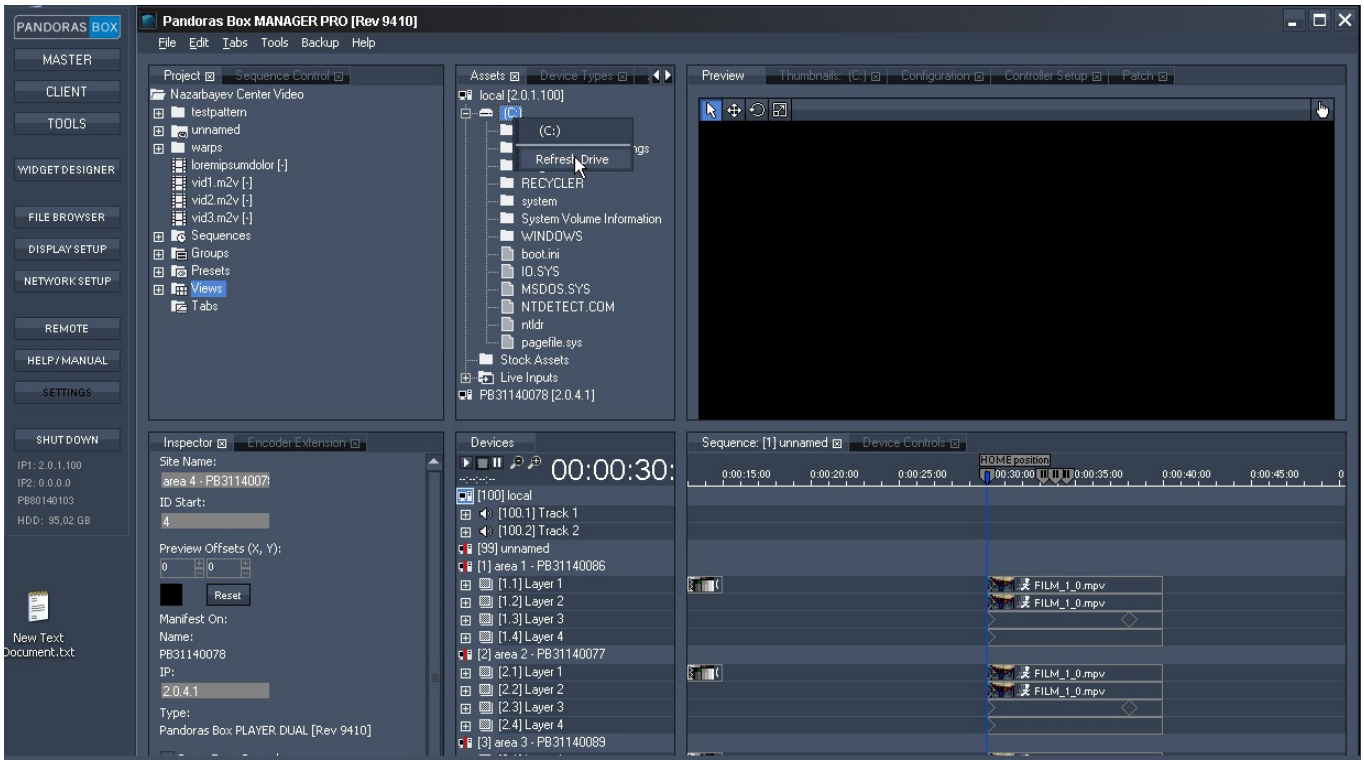
This is the screen you start with. What you see is the Pandoras Box Master.



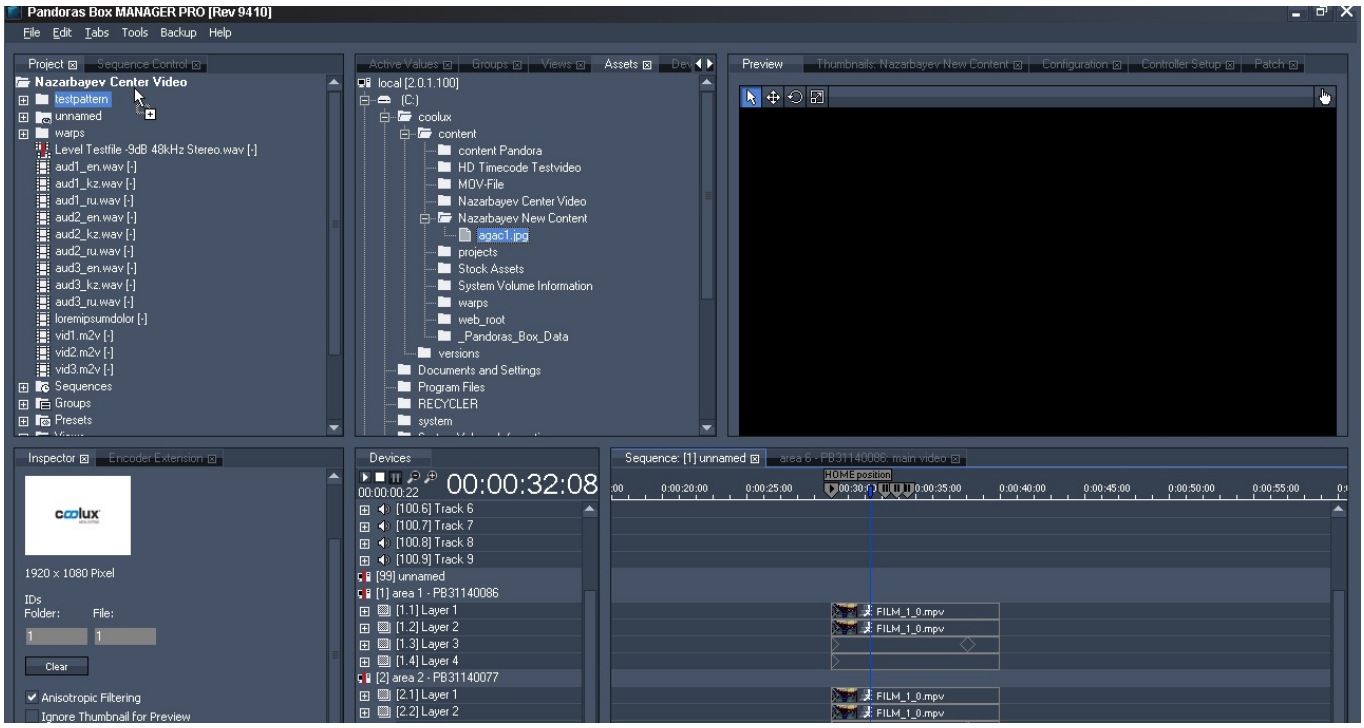
In the "Assets tab" open the C drive from the "local" and then go to the folder you have copied the files to. Mostly, that is C:/coolux/content/...



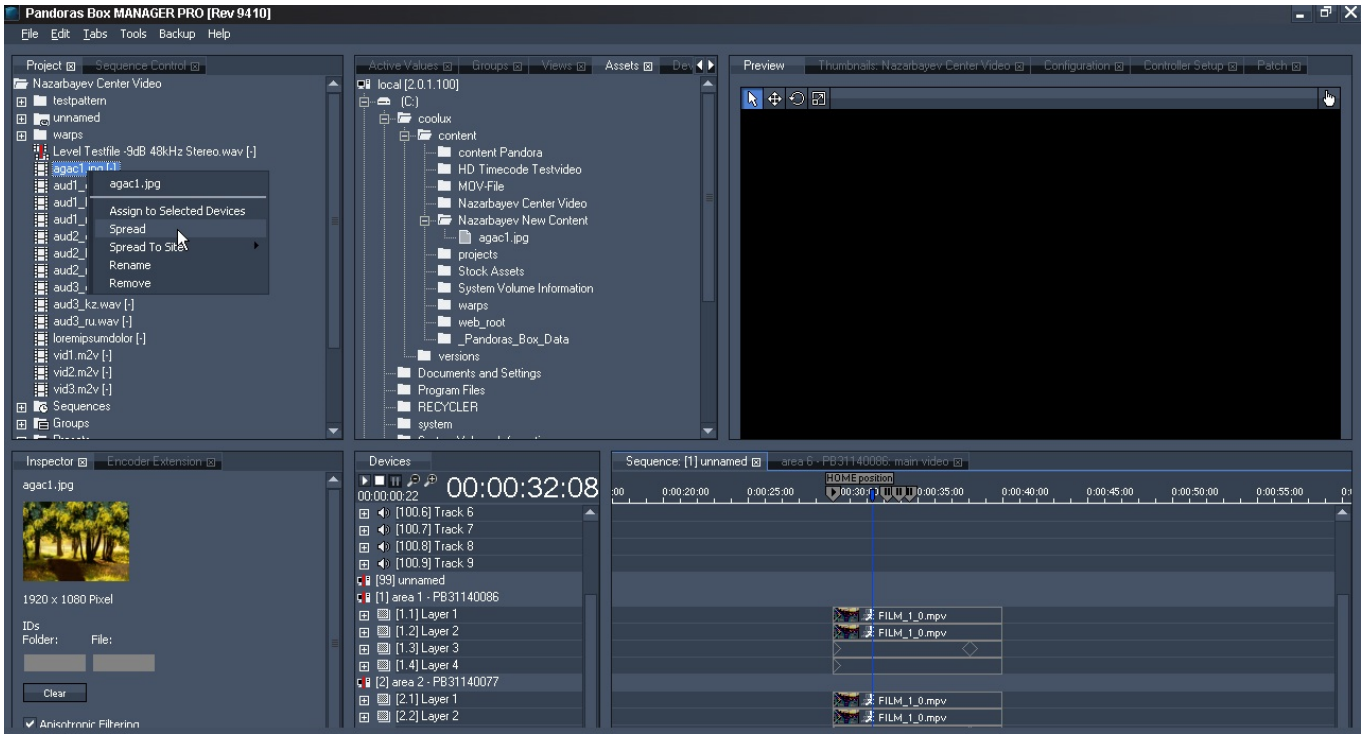
If you do not see the folder, make a right-click and choose "Refresh Drive".



Then drag the folder or single files to the "Project tab" to the project folder you want to have the files in.



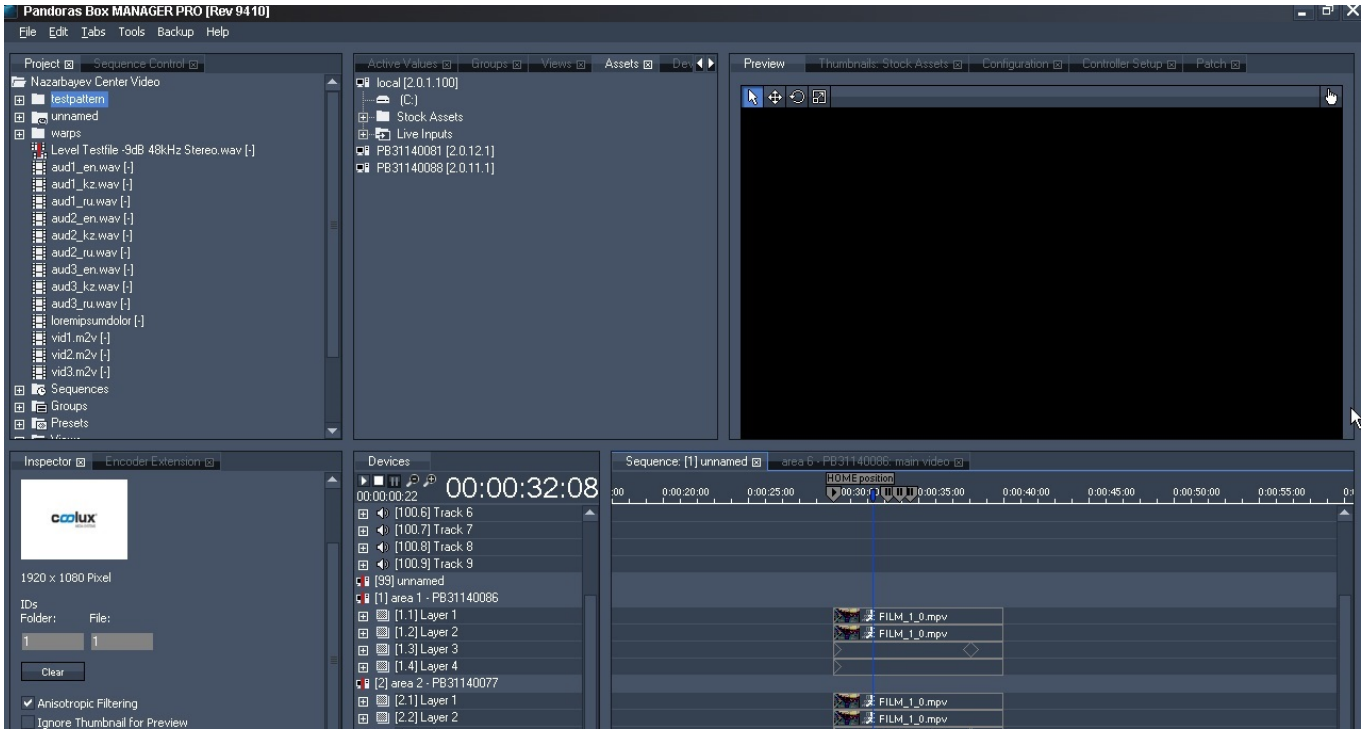
Sometimes you need to Spread the file manually. Spreading means that the Master sends this file to ALL of its Clients (e.g. Servers).
Right-click on the file and choose "Spread".



6.1.11 Basic Steps in the User Interface

This chapter covers the most important tabs and buttons of the user interface to do basic changes.

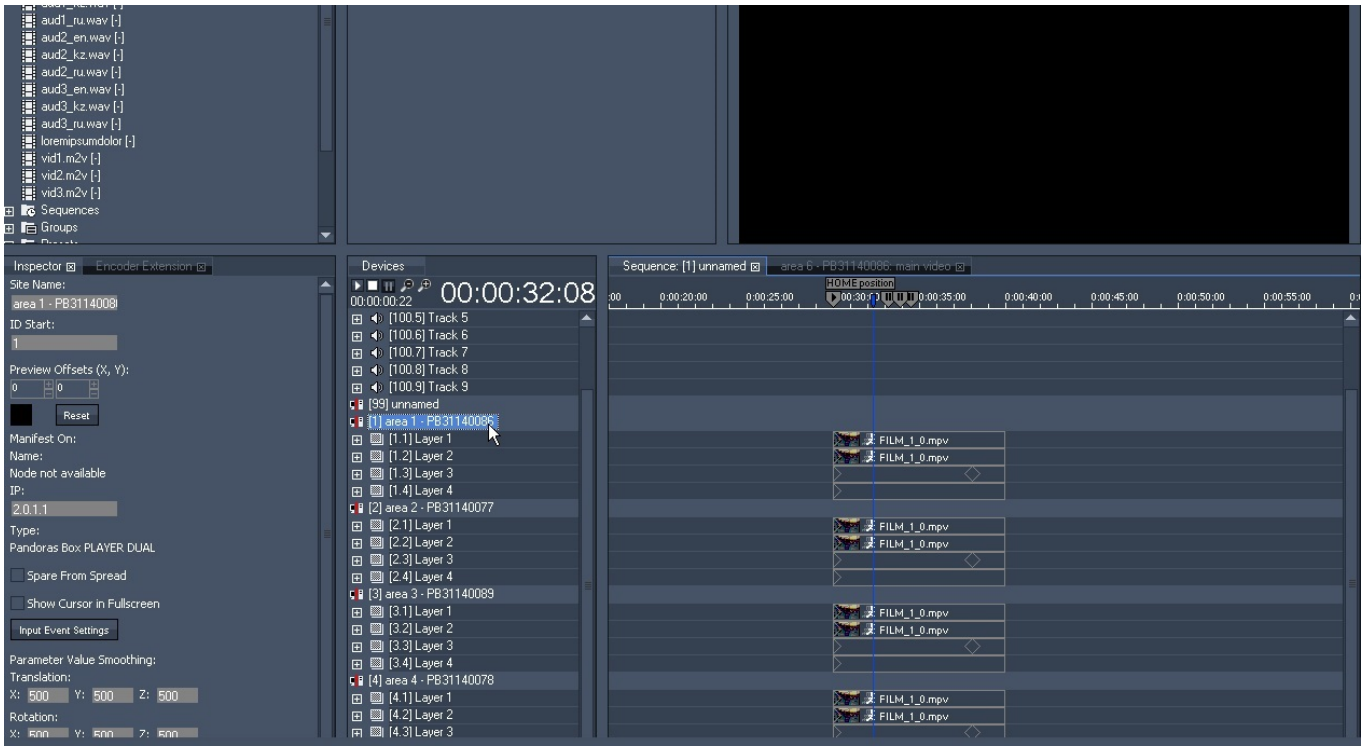
This is the screen you start with. What you see is called the PB Master. In the chapter "How to copy content ... to the project" 84 you have already met the "Assets tab" and the "Project tab".



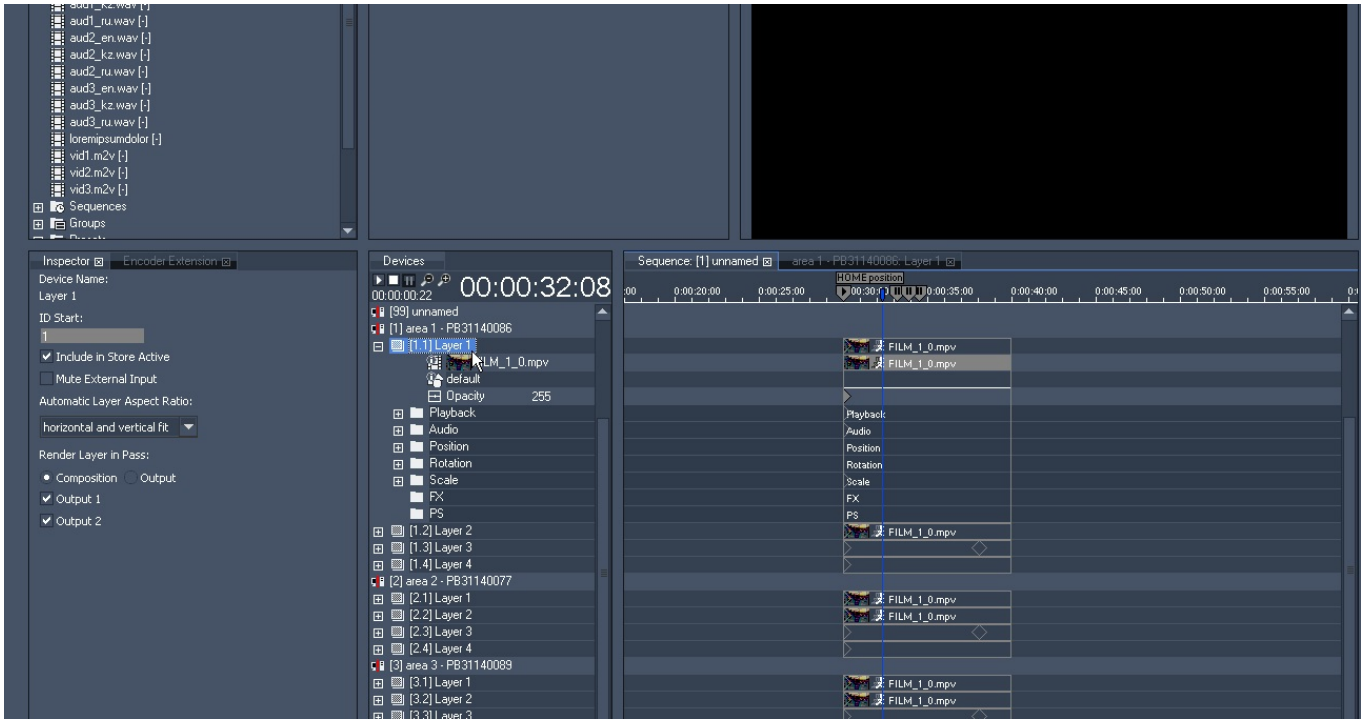
This is the "Devices tab" where you see all Clients. The icon displays a red exclamation mark that warns you that the Client is not properly connected, e.g. because
- network problems
- wrong IP address (in below picture, the "Inspector tab" shows IP 2.0.1.1. for the selected Client "[1] area 1 - PB..."

- Client not turned on
- Client software not started (no Auto-Start)

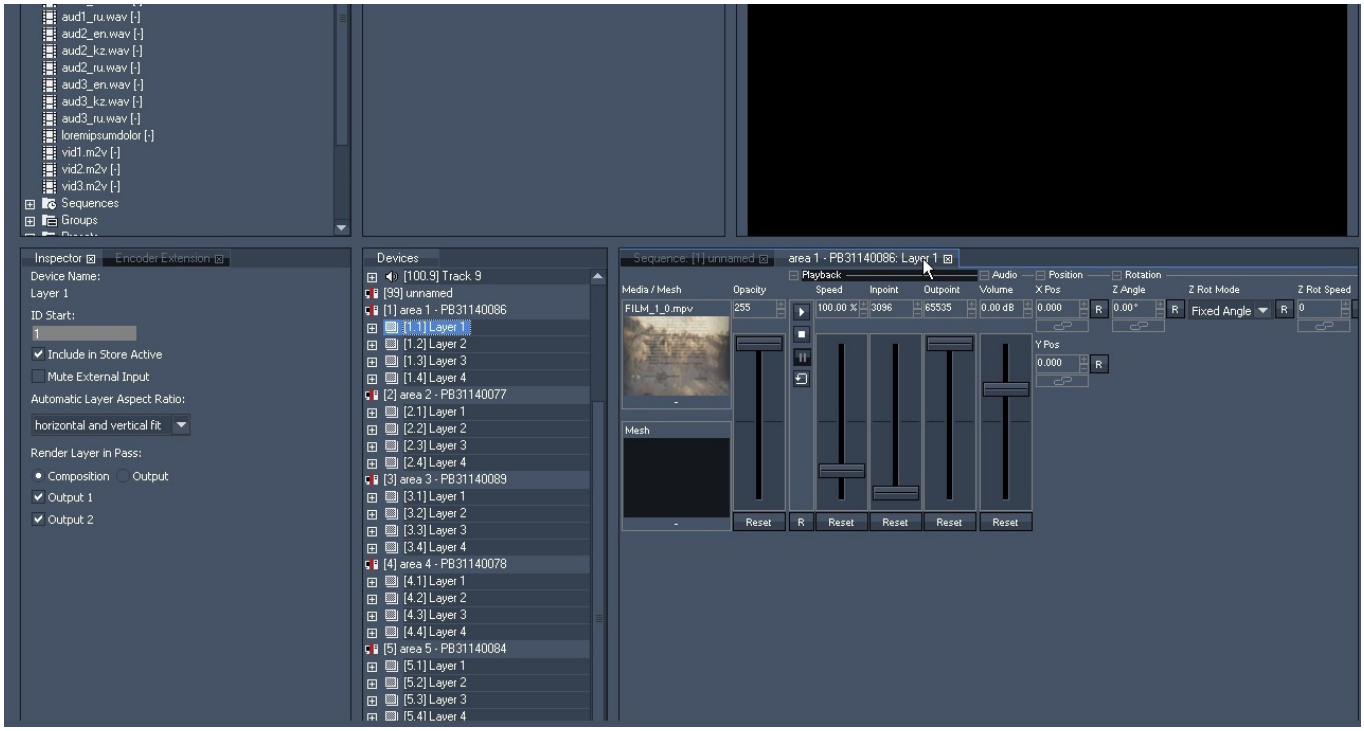
If that happens, try to ping the Client [74] or try to connect to it via VNC [69].



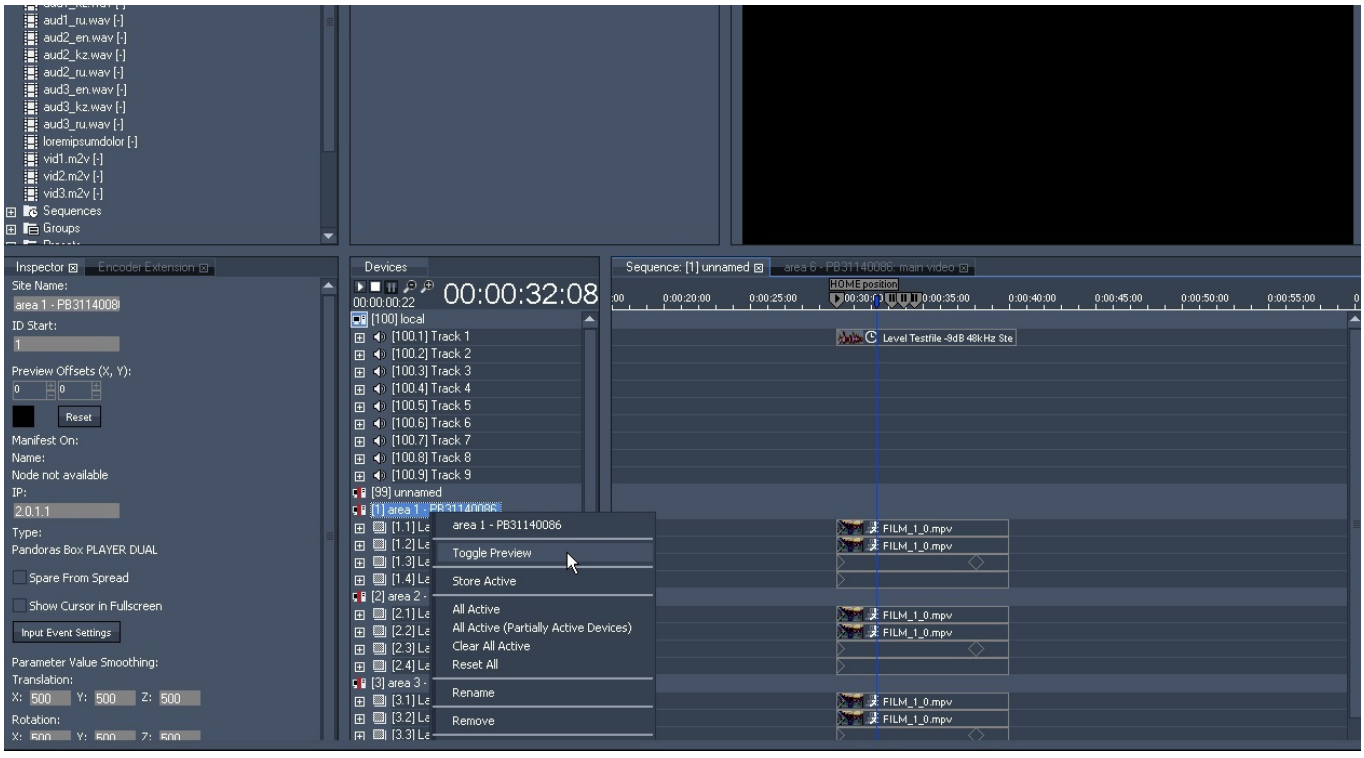
The Clients in the "Devices tab" have different layer types. Below, is an example of a Video Layer [1.1] of the selected Client "[1] area 1 - PB...". To the right side you see the "Sequence tab" with the according container. The container has stored parameters for the Video Layer. Stored parameters are called keys and are displayed as diamonds.
On top of the sequence there is the time bar with play or pause "cues". On of them is called in the example "HOME position".



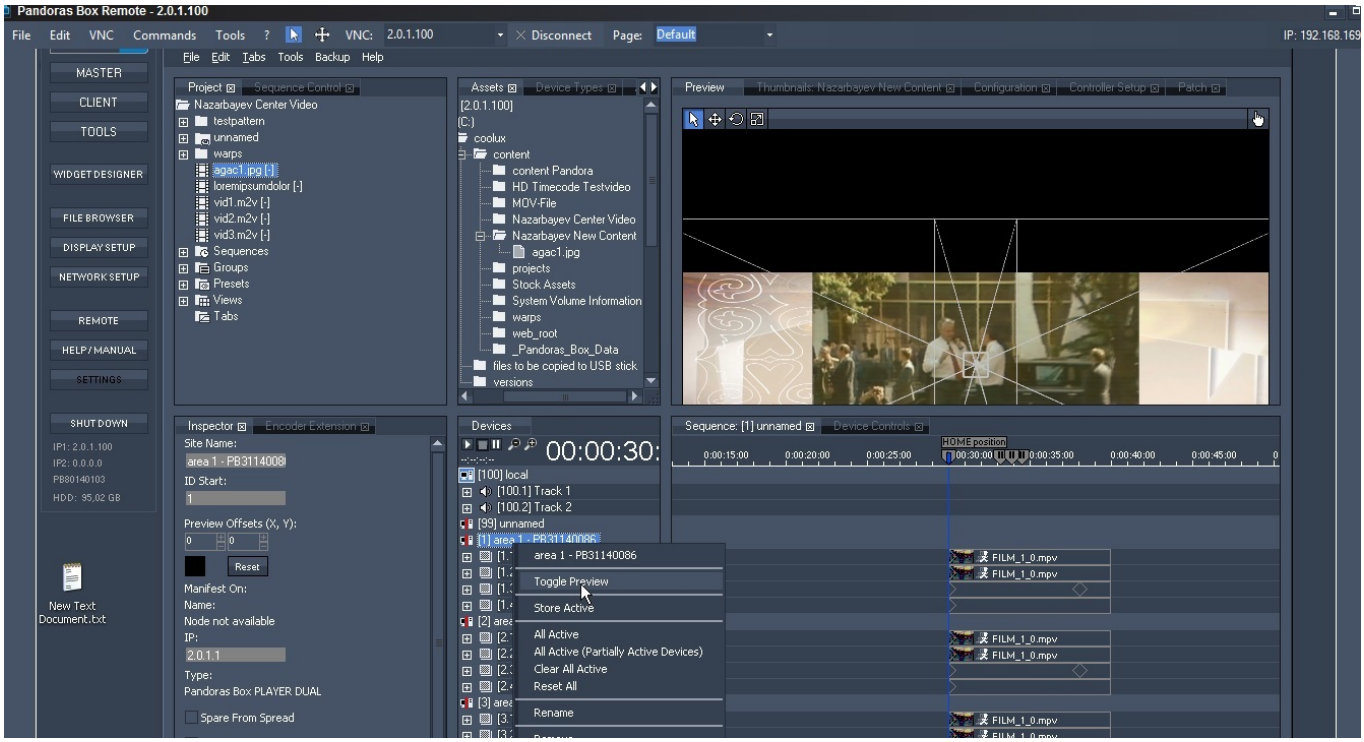
If you need to change the parameters of a Layer, select it and then go to the "Device Control tab" which is right next to the "Sequence tab". Here you see for example the parameter "Media / Mesh". If you need to exchange content, simply drag the file from the "Project tab" onto the media thumbnail.



To see what the Client renders and what your changes look like, you have the option, to use the "Preview tab" on the top right side. If your Client is not seen in there, right-click on it and choose "Toggle Preview". All previewed Clients show a blue icon in the "Devices tab".

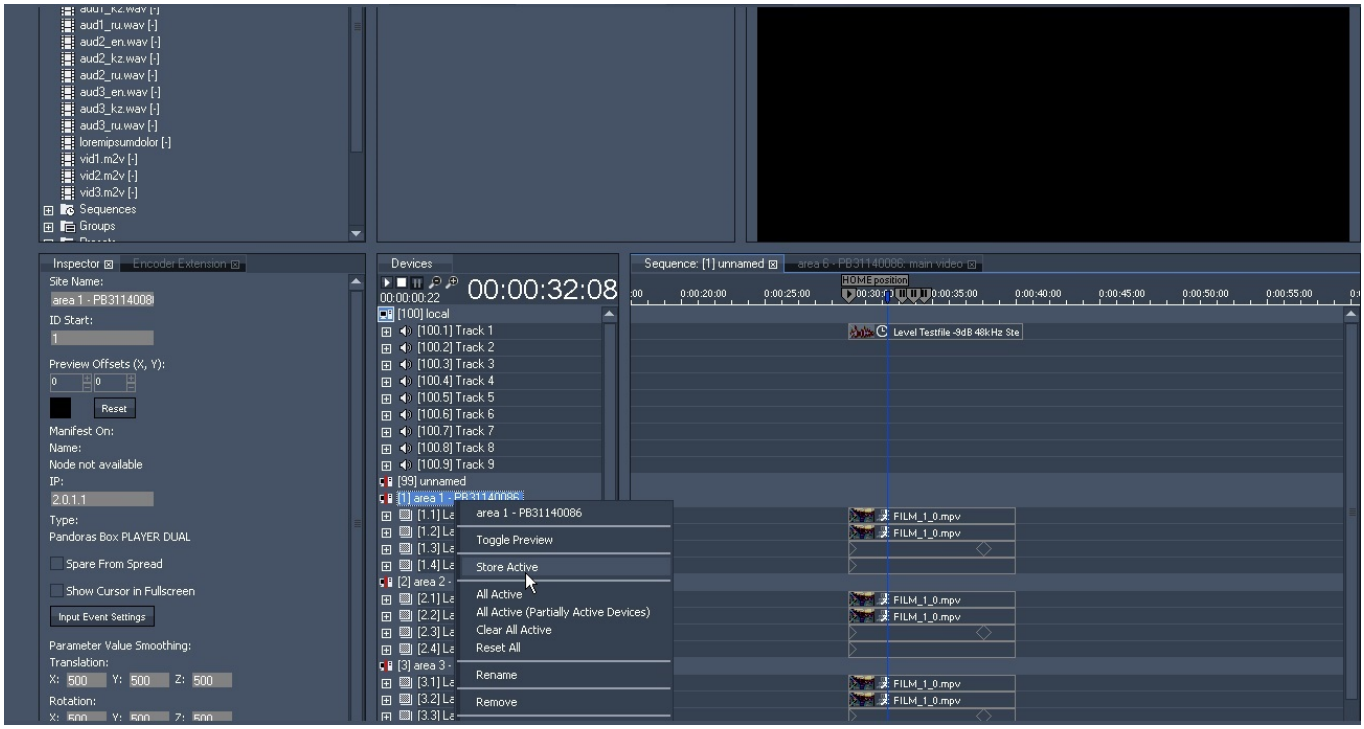


If you stand in front of the display and want to see your changes but the Client is only running in the smaller window, you can enter fullscreen in the same menu.

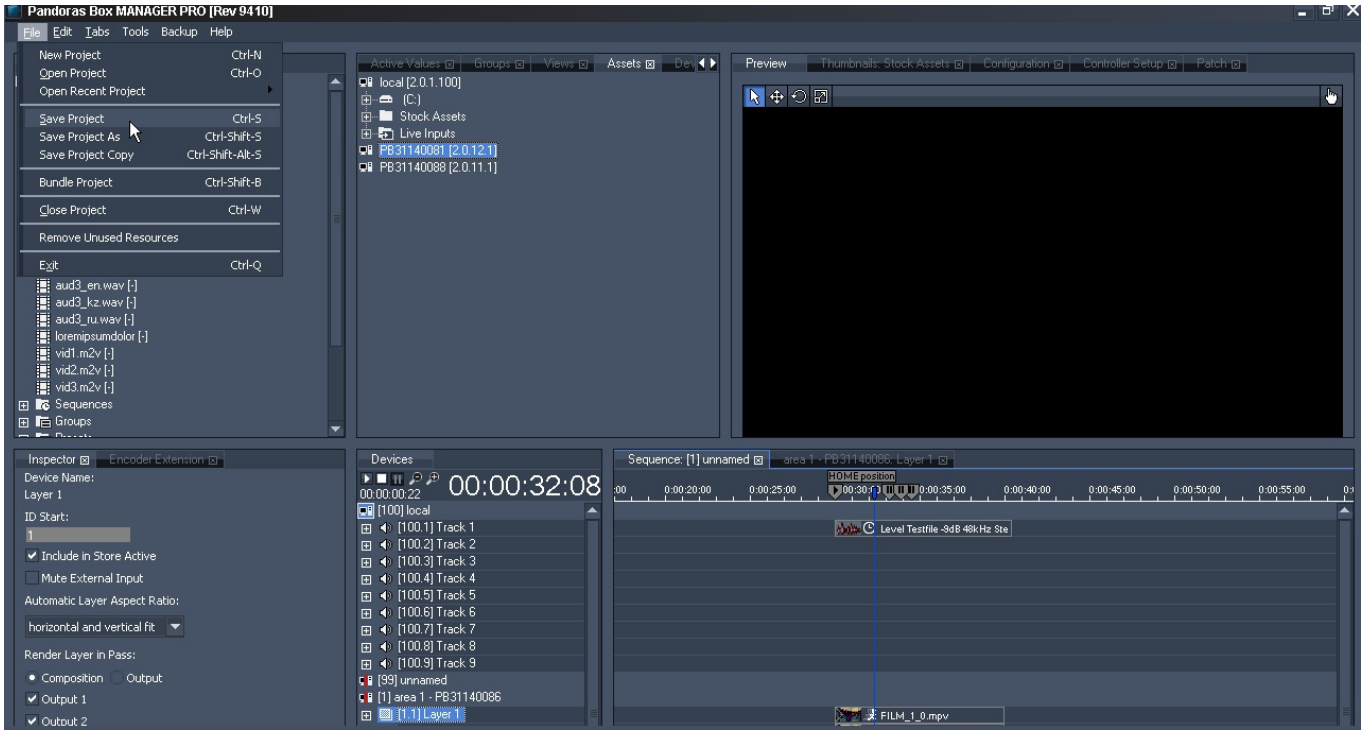


To store your changes, go back to the sequence and choose the time:
- click in the timecode and enter a new number (500 means 5 seconds and 0 frames)
- drag the blue line (now pointer) by clicking in the time bar
- use the shortcut [Ctrl + Alt + left arrow] to move in front of a container

Right-click and say "Store active". You can do the right-click on:
- the time bar
- a Client
- a layer
- a parameter



When you make changes like these, you might want to save your project. To do that, click "File" > "Save project".



6.2 Content

Pandoras Box systems allow you to play back multiple audio, video and image formats.
Since real-time rendering requires specific encoding settings, you will find detailed information about the supported file types in the following chapters.

Display and Content Formats 90
Audio Formats 92
Image Formats 93
Image Sequence Formats 95
Video Formats 97
Object Formats 100
Encoding and Transcoding 100

6.2.1 Display & Content Formats

This chapter provides background knowledge on terms like content format, aspect ratio and resolution.

Different Image Formats

Since the invention of film and TV, a lot of image formats have been around and are still in use. Three different fields have their specific image formats:

Film	8mm, 16mm, 35mm, IMAX
Television *	SD (PAL, PAL+, NTSC), HD (720i/p, 1080i/p),UHD
Digital Displays	VGA, SVGA, XGA, SXGA, SXGA+, QXGA, 4k

Many television formats are subdivided into progressive scan (= p) and interlaced image (=i). Most digital devices are optimized for progressive scan. In Pandoras Box, the setting "Deinterlacing" can be found in the content's Inspector tab 191.

All these different image formats have different aspect ratios and different resolutions as explained below.

Resolution and Aspect Ratio

The resolution describes the width and height dimensions of content or outputs. The above mentioned TV format "1080p" for example means 1920 x 1080 pixels and "XGA" means 1024 x 768 pixels.
The aspect ratio describes the relationship of the width and height. Common aspect ratios range are for example: 4:3 or 16: 9 or 1:1,85.

Both, aspect ratio and resolution are very important figures to consider during content production. It goes without saying that content can only fit on screens or displays if both aspect ratios are the same. Wrong settings lead to distortions in your images, i.e. a circle may not look round anymore. Likewise the resolutions should match. If content is produced to small, it must be enlarged which leads to blurry images at some point. Too large content on the other hand might need too much rendering performance.

Pandoras Box, per default, renders content with its native aspect ratio and native resolution. However you have many possibilities to influence it:
- Layer Inspector [211]
- X and Y scaling [653]
- Mesh objects [647]
- for more advanced users, the Camera [675] settings might be of interest.

These settings are regarding the content / input side. The output settings need to be done on the displays or projectors and can be influenced in the Configuration tab > Render Engine. [162]

Resolution Optimization

Resolution wise, the trend towards higher resolutions is unbroken since years. This does not only impact content production and display technology but of course also rendering. When using high resolutions or when rendering a lot of content at the same time you should always consider that there is a performance limit which defines what is possible to render on one Pandoras Box system. For Pandoras Box hardware, there are Performance Sheets available in our Download-Center.

Depending on the screen design, you may not always need the highest resolution for playback. If your setup consists of a large background image with a smaller video insert window, it does not necessarily need to be fullHD resolution. Playing it back with an originally smaller size, will save you performance!
Resizing content to a smaller output resolution may lead to a worse image quality. Therefore it can be more effective if the content has been created for the desired image size.

It is recommended to work with high resolution files if you need the largest flexibility. If the flexibility is not required or if the performance limit is reached, it is better to decide which content must have a smaller resolution.
For a maximum of performance and image quality, the optimum case would be to have all video files in several resolutions.

But even when your screen has large dimensions, this does not necessarily create the need for a high definition video projection. Creating content in 4k or HD is more expensive than SD and the decision should also be taken upon criteria like the distance from the audience to the screen or whether the nature of the projected image is more a graphical background animation or real film footage or readable text. Based on this you should decide, in which resolution your content needs to be created, respectively what resolutions are needed for your display or projector and how your Pandoras Box setup then looks like.

The next topics explain different content formats in detail: audio [92], single images [93], image sequences [95] and videos [97].

6.2.2 Audio Formats

To playback audio with Pandoras Box, you have two options.

ASIO Audio Device

Firstly, you may use dedicated ASIO devices or native Dante to play audio files from Audio Tracks [661].

Pandoras Box supports all common ASIO sound cards. The chapter "Input and Output Cards" [1947] lists all boards we offer. Additionally, PB supports Dante natively since version 8. Pandoras Box Licenses are natively supporting 32 input and 32 output channels without the need of any hardware, driver configuration or licensing.
PB offers audio and video playback synchronization to the audio clock via dedicated Audio Tracks. The Pandoras Box Software License offers unlimited Audio Tracks.

Audio tracks can play and synchronize:
- **PCM Wave files**
  16, 24 or 32bit
  mono and stereo
  sample rate should be supported by sound card

- **audio channels from input cards**
  embedded audio channels from HDMI 2.0 input cards [1978] or 12G-SDI Input Cards [1972] or dedicated audio channels from audio inputs cards [1947]
  in- and output sample rate should match

More details regarding the setup and use of ASIO or Dante in Pandoras Box can be found in the chapters "Audio Tracks [661]" and "Configuration tab > ASIO Audio [166]".

Audio delay settings are available in the Inspector [190] per Resource, Layer or on Clip Level. That enables you to set up different delays for different output signal chains.

Next to the feature to input, route and delay audio inputs, it is possible to record ASIO inputs via the Audio and Video Recording tab [137].

Windows Audio Device

Secondly, you may use Video Layers to playback:
- video files with an embedded audio stream (such as **MPEG2** or **MP4** but also some **MOV** or **AVI** files (depending on the codec used inside))
- audio without a video part (**WAV**, **MP3**, **WMA**).

The audio device from which you will hear sound depends on your Windows settings where you can define the default audio device. Note that you can also "preview" sound with your Master if you enable to "Preview" the respective Client.

Important: Please be aware that audio files and videos with embedded audio playing on a Video Layer cannot be synchronized as with the ASIO option mentioned above, simply due to the used technology. For applications where video and audio files need to perfectly synchronized at any time of the show it is mandatory to use proper video formats and the ASIO option. However, Video Layers still provide a decent alternative for all other applications.

To convert audio formats or transcode video files with embedded audio, there are a numerous of converter tools available. It is also worth to check out the optional Encoder Extension [103] for Pandoras Box.

The next topics describe other content formats, such as single images [93], image sequences [95] and videos [97].

6.2.3 Image Formats

Pandoras Box supports the most common file types. Version 6.5.0 can import:
BMP, DDS, DPX, JPG, TGA, TIF, PNG, SNP

Image Resolution

There is no resolution limit for images set by Pandoras Box. However, if images exceed the maximum texture size of your graphics card they would be scaled down automatically.

When using graphic (as well as video) content with more than 2048 x 2048 pixels you might need to alter the texture size in the Configuration > Render Engine [162]. Some FX and features like deinterlacing and FluidFrame need to buffer the media file. The memory space for that purpose is limited to 2k per default. Choosing another texture size will enlarge the buffer for all textures, which increases the used memory space. Thus it will consume plenty of graphics card memory. Please use the option only if needed. This is the case when the rendered image is displayed perfectly until the last pixel line is repeated to the right or bottom edge.

Alpha Channel

Alpha is the transparent part of an image where you can see through. Pandoras Box will play both alpha file types: Straight Alpha and Premultiplied Alpha.

If you like to create logos or images with transparency, choose an appropriate format that supports the alpha channel in addition to the RGB color information. From the supported formats from Pandoras Box, you can choose: BMP, DDS, DPX, TGA, TIF or PNG.
Some formats, like JPG, are designed to save only the RGB color information.

Color Profile and Color Depth

Please make sure, that the images are saved with an RGB color profile as CMYK is not supported.

The Pandoras Box Software License offers two options for the color depth. A color depth of 8bit per channel results in 16.7 million colors whilst the higher color depth of 10bit offers a total of 1 billion colors. More contrast, cleaner color transitions and HDR rendering become herewith possible and uncompressed content can be displayed without artifacts or color banding.

In order to render in a higher color depth, the inserted content needs to provide it too. You can either import PNG and TIFF with a color depth of 16bit per channel or DPX images with a color depth of 10 or 12bit per channel. The information about the color depth has been added to the File Inspector [191]. 10bit rendering can be enabled in the Configuration tab > Render Engine [162]. In other words, content will always be rendered in 8bit or 10bit, according to the render engine even if the imported file format provides a higher color depth. Hence, when rendering images (or image sequences) in 10bit, 10bit DPX files are the most efficient ones.

When working with different file versions pay attention how the color depth is named: per pixel, per channel or per file. In Pandoras Box we talk about 8 or 10bit per channel, of course referring to the RGB(A) channels.

Image Compression, Quality and File Size

Bitmaps and **Targa** images provide uncompressed images; this means that there will be no loss of color depth or resolution. In general, uncompressed files have a larger file size than compressed data, thus they are using more memory on the hard drive. However, Pandoras Box converts any image to an uncompressed format anyway to play it back in real-time. Both formats support alpha.

The **TIF** format is a bit more complicated as it supports many different things. To make a long story short: the TIF format for image sequences is only supported when the images are saved uncompressed, with a color depth of 8 or 16bit and with or without transparency. The playback performance of TIF image sequences equals the one for BMP or other uncompressed files. The performance specs can be found online in the Download-Center. For importing TIF still images (i.e. single images, not imported as a sequence), you can also choose other TIF saving options like image or layer compression but the above recommendation gives you the best quality and performance. Note that multiple layers are discarded as the file can only be assigned to one Layer in PB, e.g. a Video Layer.

The **DPX** (Digital Picture Exchange) image format is a common format for uncompressed still frames. If you have not heard from DPX, you can think of it as a bitmap format that can be saved with more variety in color depths. Pandoras Box accepts DPX files with a color depth of 8, 10 or 12bit per channel. When rendering with a color

depth of 10bit per channel, the 10bit DPX option becomes very interesting due to its performance. All other formats support only 8 or 16bit.

The **JPG** image format does an image compression that leads to a lower image quality. The conversion of JPG to BMP does not enhance its quality, it only maintains the quality. If you save a JPG as JPG again, the quality will decrease even more due to the re-compression.

The **PNG** image format does an image compression too, but other than JPG it is a loss-less data compression. PNGs can be imported with a color depth of 8 or 16bit per channel.

The **DDS** format is a special format and many tools cannot edit or display this format natively. Adobe's Photoshop for example offers dedicated plug-ins to support the format. For Adobe AfterFX, Media Encoder and Premiere Pro you can download our Adobe Plugin [101] if you like to encode DDS image sequences (or other formats). Further, the Pandoras Box Image Converter [2069] supports the DDS format as well, thus it is possible to convert images and provide them for Pandoras Box systems. You may use them as single images or for image sequences [95].
The DDS format claims to perform better than other formats like JPG, PNG or BMP. In comparison to these formats DDS saves playback performance because it is a texture format that can be interpreted by the graphics card directly. In other words, the CPU and the bus have no load regarding the decompression. Similar to the JPG format, DDS compression is lossy and does reduce the image quality. Depending on the content it is more or barely noticeable. But especially if used in a playing image sequence the much higher performance makes up for the loss of image quality. In fact, the DDS format is nowadays the first choice (for most content) when looking for optimized playback. The equivalent video format is the well known HAP codec. There is also the option to include transparency using the **DDSA** format (same file ending "dds"). **YCoCg** (file ending "yds") aims for a higher image quality and is especially recommended when the content shows gradients as for example computer generated (CG) renderings do often. But even then, you might notice artifacts and color banding.
Regarding the file size, DDS images with the same resolution have always the same size which in other words mean, that the file size is independent from the image content. DDSA and YCoCg files with the same resolution have both a doubled file size which affects directly the band width and hence playback performance.

To be accurate, the **SNP** (snappy) format is actually not an image format itself but a compression library offered from Google. So if you have a JPG, PNG, BMP or DDS image, snappy compresses the file size but keeps the available quality, as it simply reduces the file data. If you work with SNP files, the saving or reading process does not need more time, as the (de-)compression works in highspeed and real-time. In other words, Pandoras Box can decompress snappy image sequences on the fly.
The result is a much smaller file size whereas the size difference depends on redundancy in your visual content, such as alpha channels or same colors in motion graphics. This way you gain a lot more drive space which is especially interesting for SSD systems as their drives normally offer less space.
The tools Dome Master [2055], Splitter [2120], Image Converter [2069], the Quicktime Converter [108] and the new Adobe Plugin [101] offer to save images using the snappy format.

The first topic described the display and content formats [90] in general. Please click these links, if you are interested in other content formats, such as audio [92], image sequences [95] and videos [97].

6.2.4 Image Sequence Formats

An image sequence is a series of sequential still images that represent frames of an animation. Commonly, the images are saved within one folder and are labeled with an incrementing file name in order to preserve the chronological order. They have the same pixel resolution, size and file format, whereas Pandoras Box supports:
BMP, DDS, DPX, JPG, TIF, PNG and SNP

As you can see from our Performance Sheets, we recommend the use of DDS image sequences as they provide a good balance between image quality and playback performance.

That said, please keep in mind, just as with video formats, there is no "the one and only" image format that is perfect for any use. When rendering content, you choose a specific content format and each has its strengths and weaknesses. Some factors can limit the number of possible formats and are easy to decide on, for example the need for transparency or a higher color depth. Other factors are not so easy to decide on or need to be balanced with other needs, e.g. high image quality versus short file transfers or high playback performance. For that reason, Pandoras Box supports all the above formats and you can choose which is best for your show. The differences between the formats are explained in the previous chapter about Image Formats [93].

Most high-level programs like Adobe After Effects, Premiere Pro or 3D Studio Max are able to render an image sequence. For Adobe products, Christie offers a plugin [101] which can be downloaded for free from our Download-Center. The plugin allows you to render content as an image sequence in three DDS flavors or videos using the HAP codec.

Advantages of Image Sequences

+ Image Quality
When rendering a BMP, DPX, TIF or PNG image sequence, no image information, e.g. color depth or resolution are discarded and thus the best possible quality is preserved. The downside is the file size and more playback performance.
A good balance is the DDS format, even though it compresses the file size the quality does not suffer much. The DDS format is a special texture format send directly to the graphics card without drawing performance from the bus and the processor. This way you may play back much more DDS image sequences than with the other image formats. Please refer to the previous chapter, "Image Formats" [93] for more information.
In addition, always depending on the kind of content, it is possible to achieve better results regarding the quality than the MPEG video foremat can provide.
If you like to render in a higher color depth e.g. HDR, image sequences are a good choice too, please check the paragraph "Color depth" in the previous chapter [93].

+ Alpha channel
Only few codecs support the alpha (transparency) channel. When rendering image sequences, you have a choice or image formats that support it, see previous chapter > Alpha [93].

+ Codec Independence
When playing back a video you need to have the same codec installed that was used to render the video. When working with special / unusual codecs you have to keep in mind that they need to work platform independently and with good performance. When rendering and playing back an image sequence, no codecs are needed.

+ Flexibility
Most of the times content production starts before all hardware or technical decisions are made. For example, for large high-resolution softedge projections, the total number of projectors might not be known, or the final projector positions are still in question. Cropping images or scaling them down with a batch program is far easier than editing a video, especially when the quality should be preserved. Please see the other PB tools Dome Master [2055], Image Converter [2069], and Splitter [2120] that are free of charge. Other image adjustments can be made e.g. with Adobe Photoshop.
Furthermore there might be limitations and misunderstandings regarding the communication between companies that participate in the same project. Sometimes there is no direct contact between the content creator and the operator or other companies within the production chain. In some cases it is a safer or more convenient workflow for them to receive an image sequence and either do the final rendering process with one of our above mentioned tools or to playback the image sequence directly in Pandoras Box.

Disadvantages of Image Sequences

- File Size
Uncompressed image sequences (BMP etc.) preserve the best image quality possible thus no information is

compressed or discarded. That increases the entire system requirements regarding file transfer and memory. The critical question for a smooth playback is definitely the number of pixels per frame and the number of file accesses per second.

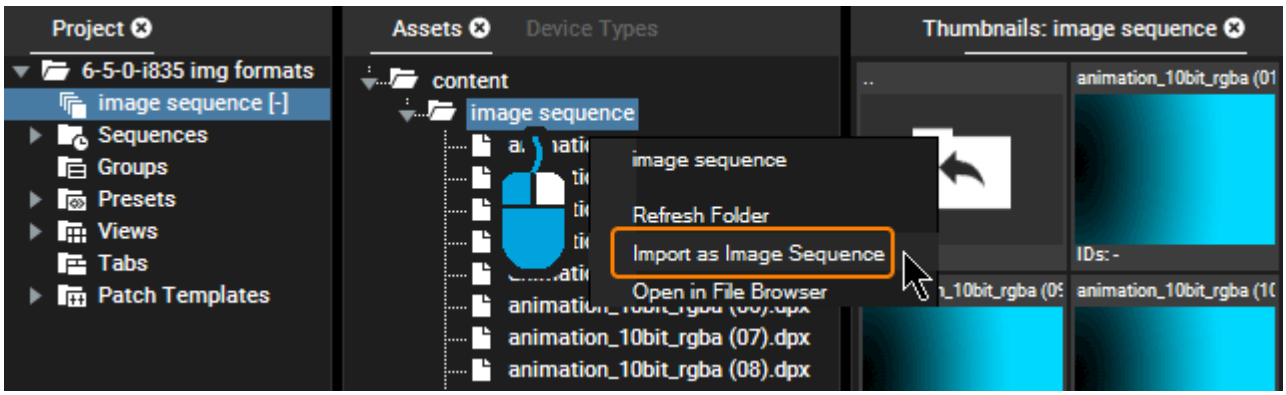
- Decompression
PNG, JPG and some TIF sequences compress the image's information in order to get a smaller file size. When being played back, the system need to decompress each single frame. Here not only the number of file accesses per second is critical, but also the processor's performance consumption due to decompression. For that reason we recommend to use BMP sequences, optionally compressed to the snappy format [94]. If you like to playback PNG or JPG sequences you might want to alter the number of threads used, this is described below.

It is recommended to use image sequences in Pandoras Box with SSD drives as normal hard drives are slower regarding the number of file accesses per second. If you do not necessarily need to change individual frames during playback it is a very good and recommended workflow to render an uncompressed AVI at the very end of the content production; you may use the Image Converter [2069] for example. The AVI format combines all advantages but as it is only one file, the system will be able to load it faster and the playback requires less performance.

Using Image Sequences in Pandoras Box

Just like any other file an image sequence must be first added from the Asset tab into the Project tab. It is recommended to save all images belonging to one sequence within one folder as well as to remove all other images (and other files, e.g. .dsstore, indexing files added automatically from MAC operating systems) not belonging to the sequence from this folder. In order to add the image sequence you cannot simply drag the folder, as this would generate a folder with multiple files. Right-click on the folder within the Assets tab and choose "Import As Image Sequence". A dialog opens and asks for the frame rate; depending on how the content was created enter for example 25fps, 30fps, 50fps etc. As seen below, at this time a new resource with the sequence icon was added to the Project tab.
You can also multi-select folders and import them as separate sequences. In the Configuration tab > User > Resources > Image Sequence [140] there are some initial values you can adopt to your need.
Alternatively, you can use one of the many Widget Designer commands ResourceAddImageSeq [1581]...
Another way to add a sequence to the project would be to right-click in your Project tab, choose "Add Image Sequence" and attach a sequence using the Inspector.

If you like to import a lot of image sequences, the Advanced Import [132] dialog is of interest. It is explained in full detail in the linked chapter about the Assets tab.



Now you can start programming with the image sequence by dragging the sequence from the Project tab into the timeline or by assigning it to a layer via double-click after selecting one in the Device Tree tab. If the sequence does not play back fluently, you can activate "FluidFrame" in the Inspector [191] (select the sequence in the Project tab to see its information in the Inspector). As well you can choose another setting in the text field "Number of Threads". The other options are explained in the topic "Image Sequence Inspector".

The number of threads influence how many threads are called by the Pandoras Box application from the operating system in order to run the sequence. There is no rule of thumb how many threads are the best. A higher number of threads can make the sequence run more smoothly and at the same time a too large number can make the playback more stuttery. There is no definite minimum or maximum. It depends on your hardware, e.g. the processor and type of hard drives. Regarding uncompressed sequences the default setting of 5 will give you good results. Other sequences require more processor performance, another thread number can give you better results.

The optimal number depends on their compression rate.
Please make sure that the threat number is smaller than the total number of images in the sequence.

Workflow Tips

Before using image sequences in Pandoras Box...:

- Check the file names:
  best is to name all files with the same number of characters, i.e if you have 100 files, the first one should be named image001 instead of "image" or "image1"

- Delete additional files from the folder:
  best is to have a folder including the same number of files as images (there should be no thumbs file, notes or administration data)

- Name the folder with the frame rate:
  best is to name the folder with the frame rate the image sequence was rendered out for, e.g. "Stars_25fps"; ask the content creator to do that for you; this way you will later know for sure with which frame rate the image sequence should be imported

The first topic described the display and content formats [90] in general. Please click these links, if you are interested in other content formats, such as audio [92], single images [93], videos [97] or objects [100].

6.2.5 Video Formats

Pandoras Box supports many video formats. For optimized playback, we recommend using the **HAP** or **MPEG2** codec. Please see the Performance Sheets how many files can be played depending on your Christie hardware.

As described below, **image sequences** are a recommended alternative too, and are of interest especially when looking for uncompressed playback. Other codecs might be used, but it is strongly recommended to test them before using them on a show. That said, please keep in mind, just as with image formats, there is no "the one and only" video format that is perfect for any use. When rendering content, you choose a specific content format and each has its strengths and weaknesses. Some factors can limit the number of possible formats and are easy to decide on, for example the need for transparency or a higher color depth. Other factors are not so easy to decide on or need to be balanced with other needs, e.g. high image quality versus short file transfers or high playback performance. For that reason, Pandoras Box supports many video formats and you can choose which is the best one for your show.

Please note:
When using video content with more than 2048 x 2048 pixels you might need to alter the texture size in the Configuration > Render Engine [162]. Some FX (check their description) and features like Deinterlacing and FluidFrame need to buffer the media file. The memory space for that purpose is limited to 2k per default. Choosing another texture size will enlarge the buffer for all textures, which increases the used memory space. Thus it will consume plenty of graphics card memory. Please use the option only if needed.

Keep in mind that former PB Player versions were limited to a maximum file resolution of 4K regarding video playback.

Codecs

A video codec is a compression / decompression method based on a specific algorithm, optimized for different uses. Please keep in mind that different codecs have different advantages, depending on the exact nature of the content being used. Not all codecs are optimized for synchronized playback.

An important thing about the AVI, Quicktime (MOV) and MXF file format is its kind of container format. This format may contain different video codecs, it is not clearly determined by its file extension. The recommended HAP codec, for example, is used inside a MOV container. Another MOV file could be encoded using the ProRes format. This one can also be played by Pandoras Box, but uses more performance. The File Inspector [191] shows the so called "Frame Type", e.g. HAP or ProRes.

Pandoras Box uses all available codecs that are currently installed on your Windows system.
If a video file may not play in Pandoras Box, please check if the codec is installed. Checking with other media

players is not recommended as they might use their own codecs which are installed on the system but cannot be used by other applications.

You might need to re-encode (i.e. transcode) videos before using them in Pandoras Box. The chapter Encoding and Transcoding [100] shows helpful tools for that.

HAP Formats

As said above, the first recommended video codec for optimized video playback with synchronization is the HAP codec. Compared to the MPEG2 format, the HAP format can achieve a higher image quality for most content.

Next to the HAP codec, Pandoras Box can play files encoded in **HAP Alpha** or **HAP Q**. All formats are encoded in a MOV container.

In general, HAP is based on the same compression algorithm as the DDS image format. Likewise, HAP Alpha can be compared to DDSA and HAP Q to the YCoCg format. For that reason content encoded as an DDS image sequence (with snappy compression [94]) provides an extremely similar file size as a HAP file does and shows the same playback performance.

For Adobe AfterFX, Media Encoder and Premiere Pro you can download our Adobe Plugin [101] if you like to encode HAP videos or DDS image sequences. This chapter also lists dis-/advantages of using an DDS image sequence over a HAP file.

DDS is a special texture format that can be interpreted by the graphics card directly and thus saves playback performance. In other words, the CPU and the bus have no load regarding the decompression. DDS / HAP compression reduces the image quality. Depending on the content it is more or barely noticeable. But especially if used in a playing image sequence the much higher performance makes up for the loss of image quality.

There is also the option to include transparency using the **HAP Alpha** format.
HAP Q aims for a higher image quality and is especially recommended when the content shows gradients as for example computer generated (CG) renderings do often. But even then, you might notice artifacts and color banding. See below for uncompressed playback [99].

Regarding the file size, DDS images with the same resolution have always the same size which in other words mean, that each frame of a HAP file has a constant size independent from its content. HAP Alpha and HAP Q files with the same resolution and length have both a doubled file size which affects directly the band width and hence playback performance.

Optimal Settings for MPEG Files

The second recommended video codec for proper video playback with synchronization is the MPEG2 codec.

Each codec must meet the two demands to encode a file with low file size but high image quality. The MPEG codec still strikes a good balance between the two demands. At the same time it supports the possibility to playback the file with frame accuracy and synchronization.
In general it does not support to have an alpha channel included (transparent information) as it saves RGB information. Transparent parts must be keyed out, for example with a masking effect and an according mask which can be generated using the Image Converter [2069] tool.

The **MXL** format is a proprietary Pandoras Box file type that can be used on video resolutions up to 4K. Basically it is the same as the MPEG2 format. The implemented Encoder Extension [103] enables you to transcode code files to this format and some optional Tools (Splitter [2120], Image Converter [2069]) allow you to render MXL files too.

To provide a synchronized playback it is mandatory to encode the MPEG2 files with a constant bit rate (CBR) and as elementary video only! That means that the audio information needs to be split or discarded. Even if the audio level is at 0db, it is still included in the file and will destroy the possibility to play it back synchronized.

The bit rate itself depends on the content. The more pixels each frames has, the higher the bit rate needs to be. The more different color information the frame has, the higher the bit rate should be to preserve quality. Statement in our Performance Charts, e.g "a Server may play four HD files smoothly" refers to the recommended bit rates seen below. If you encoded HD files with a higher bit rate you would not be able to play back four files at the same time!

Recommended bit rates:
PAL or NTSC resolution: up to 8.000 kbit/s with a progressive scan
HD resolution: up to 20-25.000 kbit/s
4k resolution: up to 80-120.000 kbit

Playback with Uncompressed Formats

If you are interested in playing uncompressed formats please keep in mind that those codecs which preserve the best possible image quality also lead to very large files. They draw plenty of performance during playback. We recommend to use uncompressed formats with SSD drives.
Alternative to uncompressed video formats, there is the possibility to playback image sequences [95] in Pandoras Box. You have the choice between compressed and uncompressed image formats. Depending on the image format you can save performance during playback, hence play larger or more files than possible with uncompressed videos. In addition, you can achieve a much higher quality compared to compressed video formats. The chapter Encoding and Transcoding [100] shows examples how to render image sequences.

The first topic described the display and content formats [90] in general. Please click these links, if you are interested in other content formats, such as audio [92], single images [93] and image sequences [95].

6.2.6 Object Formats

2D or 3D objects can be assigned to the "Mesh" media parameter of Video Layer, Output Layers etc. Typically they represent a two- and three-dimensional object with a certain shape, position and UV texture mapping. In many cases, objects are used for warping purposes . As an alternative you can also create so called Editable Meshes in Pandoras Box which allow for warping in the Preview using the Mesh Editing Mode [257].

Pandoras Box supports four formats for two- and three-dimensional objects:
3DS, FBX, OBJ and the **X format.**

As usual, simply drag the file from the Assets tab [131] to the Project tab [278] and from there either on a certain Video Layer or an Output Layer in order to affect all rendered layers.

If your FBX or 3DS file contains sub objects (groups of objects), you may set up the hierarchy level in the automatically opened dialog. "1" results in one file in the Project tab whereas "2" results in a folder containing sub files and so on. In that case, you may drag any hierarchy level, single file or folder onto a layer (Video Layer, Output Device, Venue Site Layer etc.). If you assign the first level entry to a Layer, all included sub objects are assigned, i.e. the Layer texture is shared on all sub objects. If you rather like to assign all sub obejcts to individual layers, right-click the first entry and choose the command "Assign Sub Meshes to Devices Consecutively" after selecting the "starting" layer in the Device Tree.

You may adjust the global appearance of 3D files in the Configuration tab [139] > Preview Display [145] and for each file separately in the File Inspector [191].

The topic Third Party Software [2177] shows programs which can be used to create custom 2D/3D objects and the chapter General 3D Modeling Terms [2131] explains what you should know about coordinate systems, UV mapping etc.

6.2.7 Encoding and Transcoding

For optimized video playback, please use recommended image sequence [95] and video formats [97]. These tools are of interest when encoding or transcoding content.

Encoding

- Adobe Plugin [101]
The plugin can be used for content creation. Load the plugin into Adobe After Effects, Media Encoder or Premiere Pro and it allows you to encode your content either as DDS image sequences (three flavors) or as a single MOV file with the according HAP flavor.

- coolux codec / Quicktime component [102]
Update: Adobe dropped support of Quicktime 7 in 2018. Unless you are using an extra plugin to generate Quicktime files, the following solution is not possible with current software versions any more. Please refer to the above described Adobe Plugin.

The codec can be used for content creation. Load the codec into third-party rendering software solutions like AfterFX; or third-party content converters like the Quicktime Player.
The codec allows you to encode your content as DDS, PNG and BMP image sequences (folders containing sequential images), or MOV files (containing sequential frames).

Transcoding

- Pandoras Box Encoder Extension [103]
The Encoder Extension is implemented in Pandoras Box since version 4.5 and can be found in the "Tabs" menu of the PB Master if the Encoder Extension is activated on your dongle. Since 2020 the Encoder license is discontinued.
It transcodes AVI, MOV, or WMV files to MPEG1 or 2.

- Quicktime Converter [108]
This tool comes with every Pandoras Box installation and can be found in the "Tools" menu of the PB Master.
It transcodes any Quicktime codec to a DDS, PNG and BMP image sequence (a folder containing sequential images), or a MOV file (containing sequental frames).

6.2.7.1 Adobe Plugin

Christie has developed a plugin for Adobe After Effects, Media Encoder or Premiere Pro running on a PC and Mac platform. With this plugin, content can be produced directly for optimized playback in Pandoras Box without any further transcoding step.

The content can be saved either as DDS image sequences (three flavors) or as a single MOV file with the according HAP flavor. Keep in mind that image sequence playback is mainly aimed for SSD based systems.

Download and Installation

You can download the Adobe Plugin from our web site's "Download Center" and install it on your Windows 8.1/10 platform or macOS 10.15 "Catalina" and above.

The installer checks which version of Adobe After Effects, Media Encoder or Premiere Pro is installed. As soon as you start one of these applications the plugin is automatically loaded. When setting up the final rendering, you can choose from six "Christie" formats which are added to the format list.

In case you are not seeing the codecs show up there (especially after upgrading to Big Sur), hold down the [Shift] key while launching the application. This resets the plugin cache.

Which Formats are Available?

First of all, the plugin allows to create image sequences [95]:
- DDS
- DDSA and
- YCoCg.

These formats are described in detail in the chapter "Image Formats [94]". In short: in difference to the DDS format with the smallest file size (and thus best performance), DDSA supports transparency and YCoCg results in a higher image quality for most content (especially of interest when the content shows gradients). Both formats, DDSA and YCoCg, result in a doubled file size compared to DDS.

If you looking for an uncompressed content format, the current recommendation is to export an image sequence with the DPX or TIF format which are natively supported by the mentioned Adobe products.

Secondly, you can render content in a single file (MOV container):
- HAP
- HAP Alpha
- HAP Q

For the differences, simply see above description as DDS and HAP are essentially the same; or the chapter "Video Formats [97]". Thus the main difference is whether you like to export DDS images as single files to a folder, or combine them in a container format resulting in a single file.

Each format has only one further render setting. To save disk space you can choose to directly compress the image files using the SNP format. Per definition, the HAP format always uses the snappy compression but the plugin gives you the option to deactivate it. In Pandoras Box Snappy content is decompressed on the fly.

Up and Downsides from an Image Sequence versus a Single File

As said above the image quality and playback performance of DDS flavors can be compared to the HAP flavors. Essentially, whether you export the content as images to a folder, or as frames in a MOV container even results in almost the same file size. So what's the difference you might ask. This table gives an overview of advantages and disadvantages that should be taken into account.

Image Sequence	Single MOV Container
- many files > longer file transfer (even though they add up to almost the same file size!)	+ single file > faster file transfer
+ single files > file transfer can continue at any time if connection drops	- large file > bad for download over a bad connection as file transfer must be restarted in case of interrupted connection
- can only be played back with applications (MAC or PC) that support image sequences	+ can be played back with many applications (MAC or PC) with installed codec
+ easy to replace frames (specially of interest if content is not finalized and updated frequently; further, sending single images for review is possible)	- no partial content replacement
+ can be played back forward, backwards or bounced	
+ can be rendered by multiple computers and put together	
- many files can get confusing very fast	

6.2.7.2 coolux Codec

coolux has developed a codec for use on PC and Mac. With this codec, content artists can directly produce MOV files or image sequences for Pandoras Box without any further transcoding step. Keep in mind that image sequence playback is mainly aimed for SSD based systems. Pandoras Box is able to playback uncompressed image sequences. So if the sequence was rendered in an according lossless file format, you can play it back without any quality loss as Pandoras Box never compresses files internally!

In what programs can the codec be used?

Obviously, Pandoras Box software itself can play back videos rendered with the coolux codec. No further steps are necessary, simply install Pandoras Box (Rev 10386 or newer) and the latest Quicktime Player.

The codec can be used for content creation, especially when there must not happen quality reducing and time consuming format conversion. Load the codec into third-party rendering software solutions like AfterFX.
It can be also used in (third-party) content converters like the Quicktime Player or the Quicktime Converter from coolux.
In both workflows, the codec renders content by using the Quicktime API. This is, why the codec is also called the "Pandoras Box Quicktime Component".

As soon as you have installed the coolux codec you can access it. Simply start or execute the according encoding / transcoding command which then offers a drop-down list to choose a custom format. The coolux codec is listed as "coolux".

Where can the codec be downloaded?

If you have installed the latest Pandoras Box version, the codec is already available on your system.

If you like to use the component without PB, download it from our web site's "Download Center" and install it on your PC or Mac system. Note that Mac OS 10.7 or newer is the minimum required version for the coolux QT components to work, older OS version will not be supported. On Mac and Windows PC systems, the codec requires the latest Quicktime Player for rendering as well as playback.

In what formats does the codec render?

First of all, the component allows to create .DDS, .PNG and .BMP image sequences [95] (RGB or RGBA, i.e. with transparency). To save disk space you can choose to directly compress the image files using the SNP format. No matter what format you chose, the result from the rendering process is a folder with separate single (image) files. The folder can be played back from Pandoras Box by choosing "Import as Image Sequence" in the PB Assets tab [131]. Snappy image sequences are decompressed on the fly.
To play the image sequence on another system than Pandoras Box, you need to choose a media player that supports image sequences.

Secondly, the component allows to render a MOV file. By using our codec in the MOV container, you have the advantage of the same quality as an image sequence offers. That is because the container simply contains the

single (image / frames) files, there is no additional intra-frame compression. On the other hand, you gain the advantage of a single file, that is a more convenient file handling and management. The advantage of using a MOV file is, that everybody can play the content with a Quicktime Player.
To play the MOV file in Pandoras Box, simply drag it into your Project tab [278].
Please keep in mind that eventhough PB can playback these MOV files, MOV files containing other codecs are not recommended playback formats!

The chapters "Image Formats" [93] and "Image Sequence Formats" [95] mention many up and downsides per each format. This post in our forum gives a helpful overview too.

How to convert other MOV files? What is the coolux Converter?

As mentioned above, MOV files containing other codecs are not recommended playback formats for Pandoras Box.
However, you can convert other MOV files using either a Quicktime PRO Player or the new coolux tool called the Quicktime Converter.

The Quicktime PRO Player allows to access the coolux codec as soon as it is installed on the system.

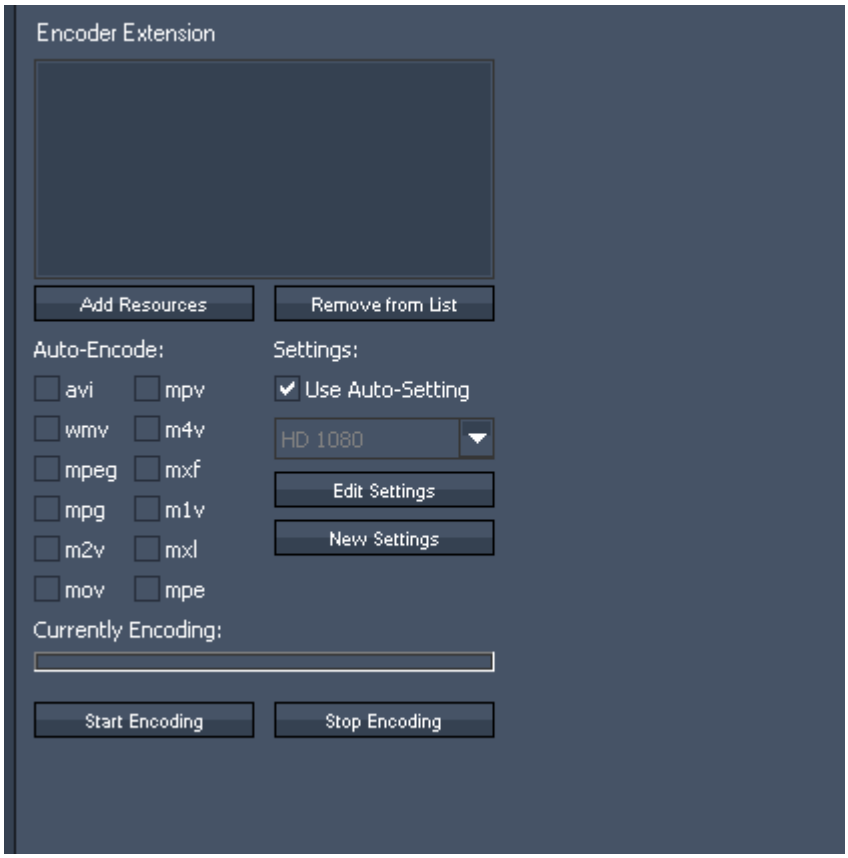
The Converter allows to read, i.e. decode any MOV file as long as the codec used in its container is installed on the system. Again, the latest Quicktime Player needs to be installed.
Afterwards it renders i.e. encode the file into an image sequence or single MOV using the new coolux codec instead. You may find the Converter along with all other coolux tools (Warper; Matrix Patcher,...) in the "Tools" menu in Pandoras Box.

The next chapter covers the use of the Quicktime Converter and the codec settings [108].

6.2.7.3 Encoder Extension

The built-in Pandoras Box Encoder Extension allows to transcode most AVI -and Quicktime- as well as WMV files to Mpeg1 or 2 up to a 4k resolution. The files will be encoded to MPEG Elementary Video Streams. If the original file contains audio, a separate wav-file will be created
The resolution can be set to SD, HD or a custom one. The Encoder Extension allows you to transcode files larger than 1080p to our proprietary .mxl format. The maximum resolution is 4080x2800 px and the encoding process depends on available graphics card RAM.
Please read the topic "Video Formats [97]" if you are interested in knowing more about video formats and codecs that are supported by Pandoras Box and about optimum video playback.

To open the Encoder Extension Tab, please click on the Tabs Menu and choose Extensions - Encoder Extension. The Encoder Extension tab will appear next to the Inspector tab.
Please note that the Encoder Extension needs to be activated on your dongle, depending on your license this will be fee required.



Auto-Encode

To use the Auto-Encode function please set a check in front of all file formats that should be automatically encoded when added into your project.

The Encoder Extension supports the following file formats:
avi
wmv (Windows Media Video)
mpeg (Mpeg)
mpg (Mpeg)
m2v (Mpeg 2 Video only)
mov (Quicktime Video)
mpv (Mpeg)
m4v (Mpeg 4 Video)
mxf (File Exchange Format e.g. used by Avid and Sony XDCam)
m1v (Mpeg 1 Video only)
mxl (proprietary Pandoras Box file type)
mpe (Mpeg)

The added files will appear in the Encoder Extension Window, the encoding starts automatically. The status bar shows the progress of the currently encoded file. Once a file is completely encoded, the file extension in the project tab changes to .mpv. The original file is discarded from the project but kept in the windows folder from where you added the file into the project.
To stop encoding, please press "Stop Encoding". Pressing "Start Encoding" will continue the encoding process.

Manual encoding

To encode files manually please select the files in the project tab.
In the Encoder Extension press "Add Resources", the selected files will be added into the Encoder Extension window. The drag and drop function does not work here. To start the encoding process press "Start Encoding". Now everything behaves as described for the Auto-Encoding.

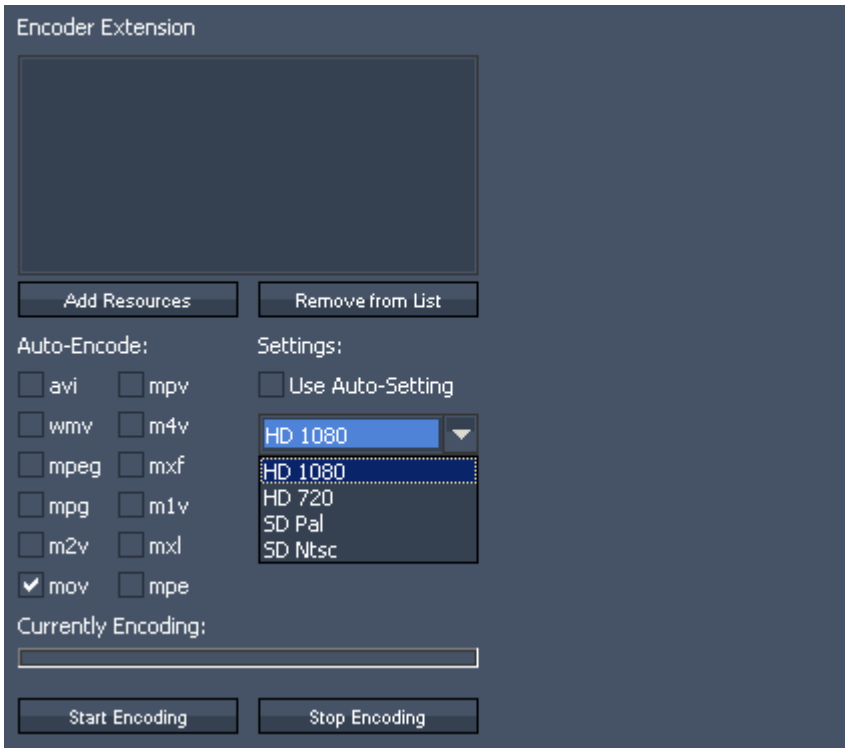
REMOVE FROM LIST:
To clear files from the encoder extension window - no matter if new encoded or not, select the files and press "Remove from List".

Settings

Auto-Settings
Using Auto-Settings will encode all your files to MPEG2 Elementary Video Stream.

The bit rate will be set to a constant bitrate: CBR 8.000 kbit/s (SD) or CBR 20.000 kbit/s (HD).
The field encoding will be set to "Progressive Frame".
All other parameters like video mode (PAL or NTSC), resolution and frame rate will be adopted from the original file and are not affected through encoding.

Choose Profiles manually
By unchecking "Auto-Settings" you will be able to choose one of the default profiles manually.

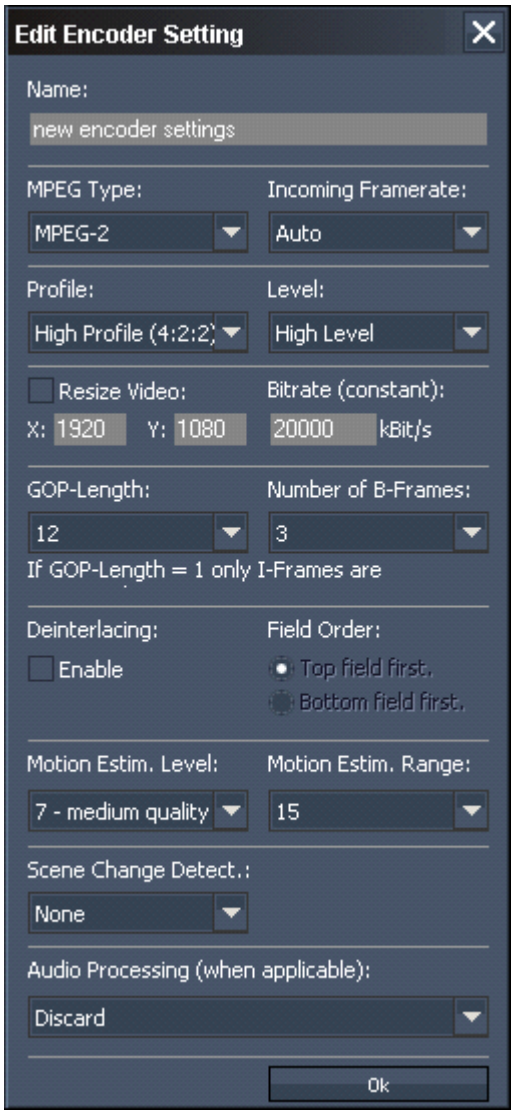


Using one of the default profiles will not only affect the bitrate and field encoding but might change the files resolution: e.g. encoding an HD-file (resolution 1920x1080 px) using the SD PAL Profile will resize the video to 720x576 px.

Please note: the default profiles can not be edited.

Creating new profiles

Please uncheck "Auto-Settings" and press "New Settings" to create a new encoding profile.



The new dialog allows you to:

[Name]
Name the profile

[MPEG Type]
Choose the MPEG Type (MPEG-1 or MPEG-2)

[MPEG Profile / Level]
Set MPEG Profile and Level. Please note that you will need High Profile (Profile 4:2:2:) and High Level for HD 1920x1080 px resolution. The Level automatically changes to MXL if you encode files to a format bigger then 1920 x 1080 px.

[Resize Video]
To resize your video, check the box and insert the resolution your video file should be resized to. Note that the "MPEG Type" and "Level" influence the maximum allowed resolution. In general, the highest resolution is 4080x2800 px for the MPEG2 format and MXL level.

[Bitrate]
Choose an amount of kbit/s (recommended are about 8.000 kbit/s for SD, 20-25.000 for HD and 80-120000 kbit/s for 4k)

[GOP-Length]
The length of the Group of Pictures is editable from 1 to 100. The higher the GOP-Length the better the compression rate gets at the expense of quality.

[Number of B-Frames]
Choose a value between 0 and 7.

[Deinterlacing]
To enable Deinterlacing, please check the box and also choose a Field Order.

[Motion Estimation Level]
Set a value between 0 (no motion search) and 15 (high quality). The higher the value, the better the quality, the longer the encoding process.

[Motion Estimation Range]
Set a value between 0 (no motion search) and 31 (high quality).

[Scene Change Detection]
Choose between None, Fast and Refined.

[Audio Processing]
- Discard: discards the Audio Part, the video file extension
- Split to separate wav: splits the Audio part to a separate wav-File. This file will be added into the project. automatically. On the hard-drive it will be created in the same folder where the related video source is located.
- Include: includes the Audio Part into the encoded video (the video file extension will be "mpeg").

The Audio Part is discarded by default.
If there is no Audio Part included in the encoded video file, the file extension will be ".mpv" (Video Elementary Only). Including an Audio Part, the video file gets the extension ".mpg"

Click OK when the new setting is finished. The created profile will appear in the drop down menu.

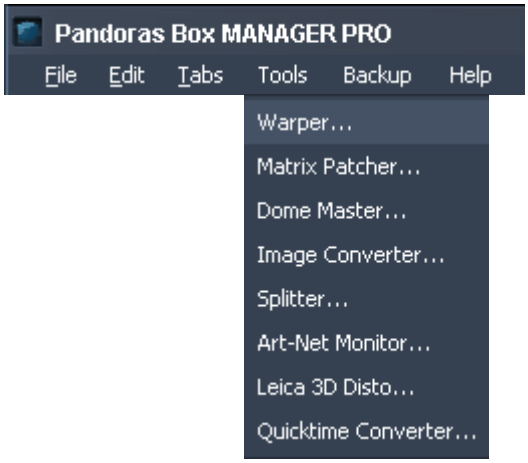
EDIT SETTINGS:
Select your profile from the drop-down list and click "Edit Settings". All settings are described above in "Create New Settings".
Please note that the default profiles are not editable!

6.2.7.4 Quicktime Converter

This topic explains how to use the Pandoras Box Quicktime Converter using the coolux codec, and the codec settings itself. Please refer to the previous chapter for general information about the coolux codec 102 also called the coolux Quicktime Component.
If you use the coolux codec in third-party software like AfterFX and the Quicktime Player PRO, please refer to their documentation to learn how to start the encoding process itself. There should be somewhere a choice what format you like to render in. The coolux codec is listed as "coolux". The first of the below steps do not apply to you, the others do eventhough the dialogs probably look different.

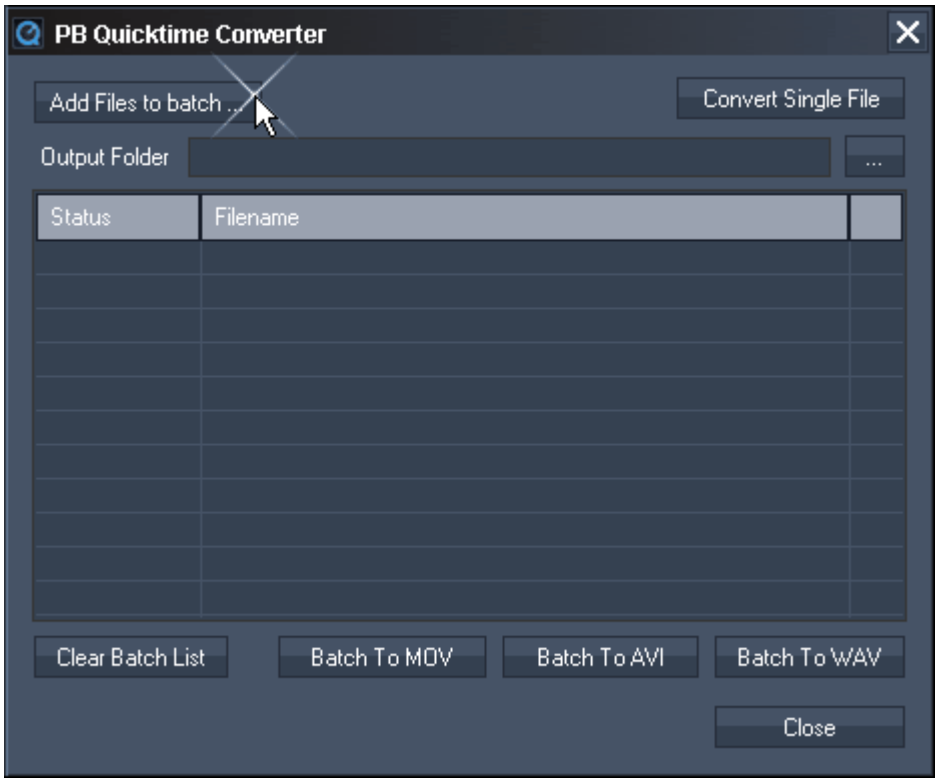
As mentioned in the last topic, the Quicktime Converter allows to transcode other MOV files into a format that is recommended for Pandoras Box. The coolux codec renders i.e. encodes image sequences (a folder containing sequential images), or MOV files (containing sequential frames).
Keep in mind that image sequence playback is mainly aimed for SSD based systems.
To convert other MOV files, install the latest Quicktime Player and start the Pandoras Box Quicktime Converter.

How to start the Pandoras Box Quicktime Converter?



You may find the QuicktimeConverter along with all other Pandoras Box tools (Warper; Matrix Patcher,...) in the "Tools" menu in Pandoras Box. Alternatively, you can open the Quicktime Converter without Pandoras Box by executing the "Pandoras Box Quicktime Converter.exe" from the installation path of Pandoras Box under data > tools.

How to choose a source file?

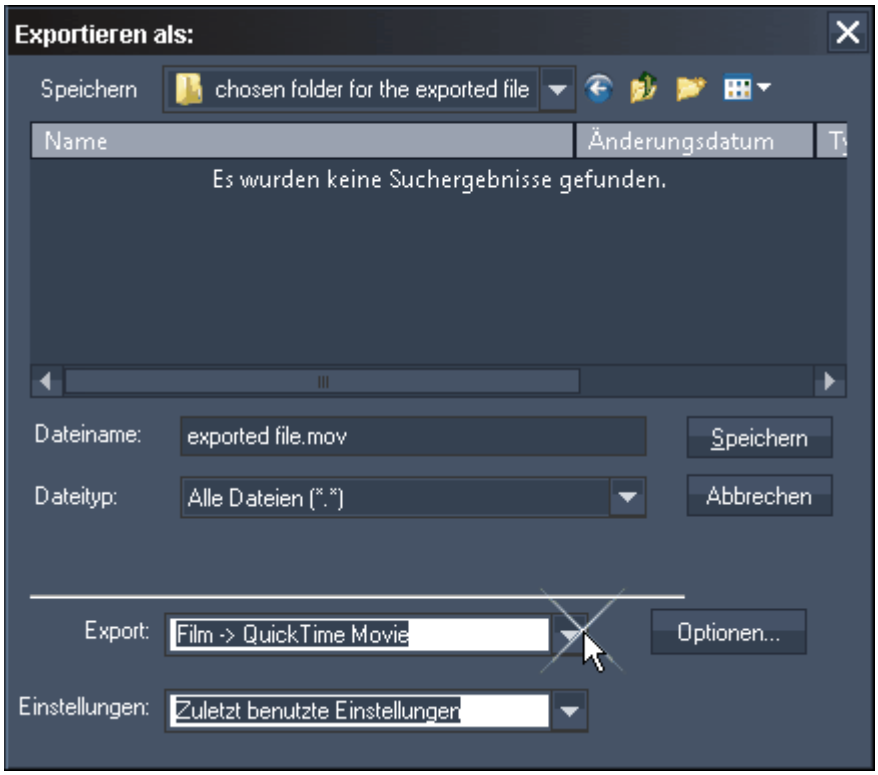


Single File:
The "Convert Single File" button opens a dialog where you can choose a MOV file as a source. As soon as you have picked one, the "Export As" dialog opens automatically.

Batching:
With the button "Add Files to batch" you may choose several source files that will be added to the batch list. Click the button again to (multi-)select additional files. Enter a output folder yourself or click the [...] button. In this folder all new files will be exported with the same name as the source files. Click one of the "Batch To..." buttons to convert all files.

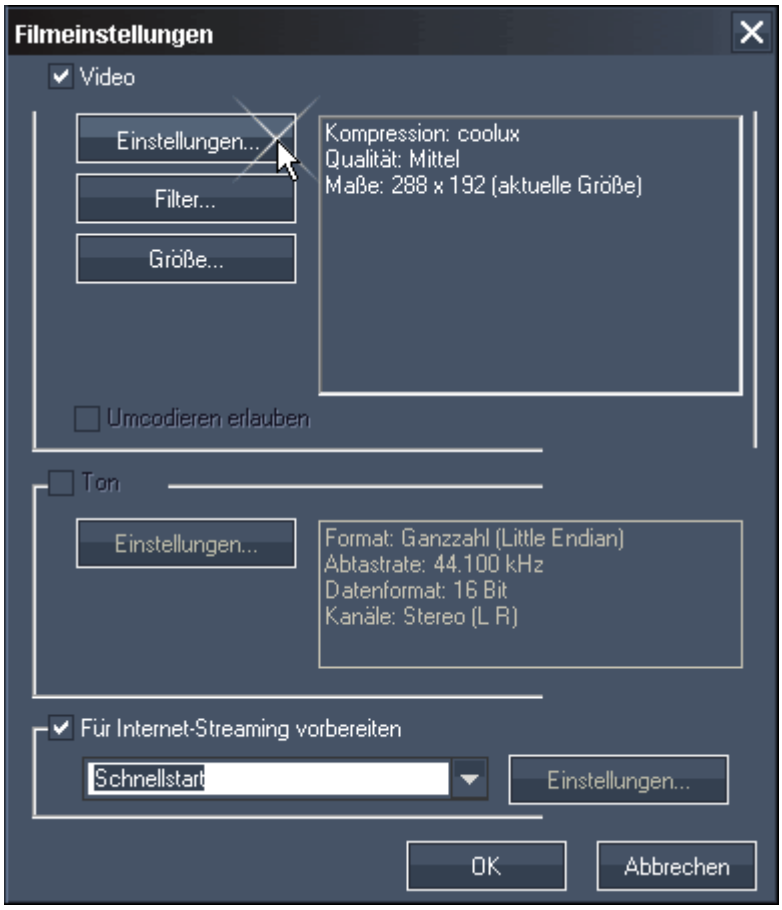
Please note that a MOV file is a so called container format, which means that it can contain other codecs. If the Pandoras Box Converter fails to read your source file(s), make sure that the source codec is installed on your system.

How to choose the export format?



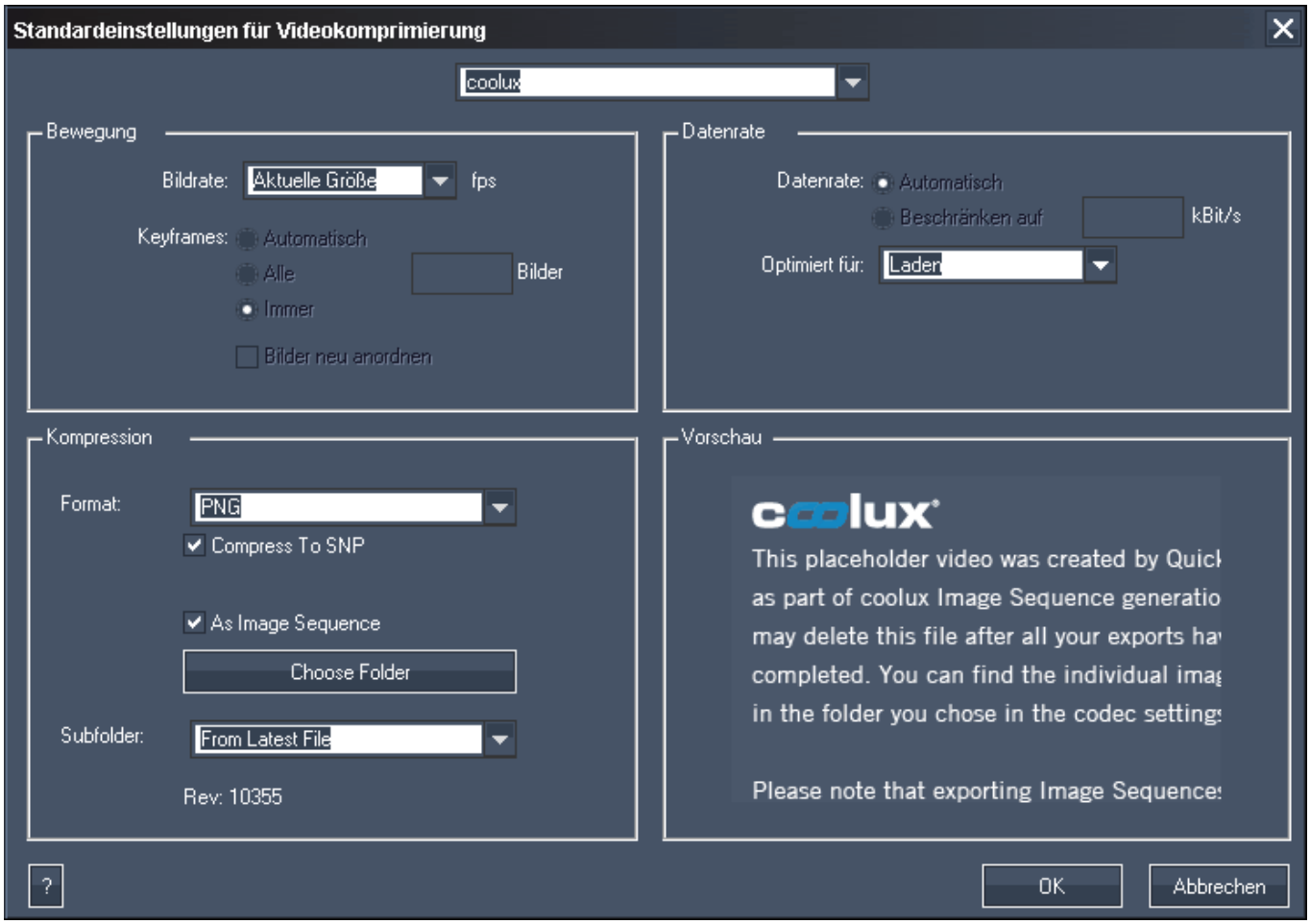
Single File:
The "Export As" dialog opens automatically. Enter a name and choose the entry "Film -> QuickTime Movie" in the "Export" drop-down list.
Click the "Options" button to the right, to set up the output format in more detail. If you use the Converter for the first time, its strongly recommended to set up the format according to your needs. Every following rendering will call the settings previously chosen as a new default.

Batching:
This dialog is skipped when you clicked a "Batch To..." button.



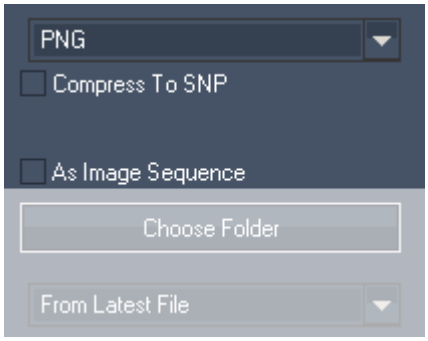
In the "Movie Settings" dialog, click "Settings".

Click "Filter" only if you like to add or adjust additional video filters. If you wish to resize your movie, click "Size".
If your original MOV file contains audio information, you can choose to export a separate wave file. Click "Settings" if you like to change the default settings regarding sample rate etc. For ASIO playback please ensure that "PCM" is chosen.
Internet-Streaming is not needed.



In the "Standard Video Compression Setting" dialog there is a drop-down list to select a codec. All the standard as well as third-party video codecs installed on your system are available. Select "coolux". Now, three sections "Motion", "Data Rate" and "Compression" offer adjustments influencing the quality of your export. Define the "Frame Rate" of your exported movie only if you like to change the original one. The "Data Rate" should be set to be optimized for Upload. The Compression settings are explained in the next paragraph.

How to export a MOV file based on the coolux codec?



Follow the above steps until you reach the section "Compression" in the "Standard Video Compression Setting" dialog. To convert your source MOV file into a MOV that uses the coolux codec (recommended for Pandoras Box playback), follow these steps:

1) Choose the image format [93]: DDS, PNG, or BMP
(in short: DDS compresses the most and allows the best playback performance, PNG compresses the file size but not the image quality but generally needs more performance whilst BMP is uncompressed but demands most performance). Choose the alpha version if your source file consists of transparent parts.
The image format mainly influences the quality of your content. The coolux codec simply renders single frames and bundles them into a MOV container. There is no additional intra-frame compression!

2) Decide whether the chosen image format should be compressed (in real-time) using the snappy compression library [93]. Snappy reduces the file size but keeps the available quality. The result is a much smaller file size which is especially interesting for SSD systems as their drives normally offer less space.

3) Uncheck the option "As Image Sequence".

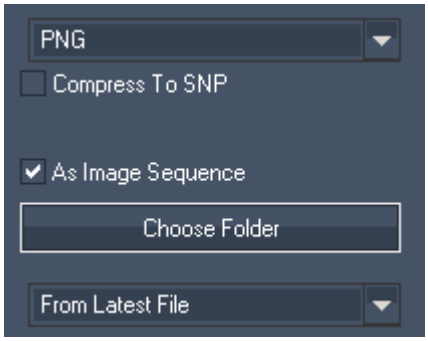
Single File:
Click "OK" twice to return to the "Export As" dialog. The button "Save" starts the conversion. The result is a MOV file with the name and path chosen in the "Export As" dialog.

Batching:
Click "OK" twice to return to the "PB Quicktime Converter" dialog that starts the conversion automatically. The result are several MOV files with the original source name in the folder chosen in the "Output Folder" text field in the "PB Quicktime Converter" dialog.

drop-down in "Standard Video Compression Setting" dialog	-
"Choose Folder" button	-
Single File) "Export As" dialog: C:\folder A\newname.mov	
Result for single converted file	C:\folder A\newname.mov
Batching) "PB Quicktime Converter" dialog: C:\folder A\	
Result for batch list item #1	C:\folder A\sourcename1.mov
Result for batch list item #2	C:\folder A\sourcename2.mov

To play the exported MOV in Pandoras Box, simply drag the file into your Project tab [278].
To play it back without using Pandoras Box, use the Quicktime Player and install the coolux codec.

How to export an image sequence - a folder with separate image files?



Follow the above steps until you reach the section "Compression" in the "Standard Video Compression Setting" dialog. To convert your source MOV file into an image sequence, that is a folder containing separate, sequential images, follow these steps:

1) Choose the image format [93]: DDS, PNG, or BMP
(in short: DDS compresses the most and allows the best playback performance, PNG compresses the file size but not the image quality but generally needs more performance whilst BMP is uncompressed but demands most performance). Choose the alpha version if your source file consists of transparent parts.
The image format mainly influences the quality of your content. The coolux codec simply renders single frames and bundles them into a MOV container. There is no additional intra-frame compression!

2) Decide whether the chosen image format should be compressed (in real-time) using the snappy compression library [93]. Snappy reduces the file size but keeps the available quality. The result is a much smaller file size which is especially interesting for SSD systems as their drives normally offer less space.

3) Check the option "As Image Sequence". Now, the images are not bundled into a MOV container but will be rendered as separate image files into a folder.

4) Choose a folder where the folder(s) should be saved that contains the separate images. It is a good workflow to choose the folder whereto the MOV file is exported as well. That was the step in the "Export As" dialog or "Output Folder" text field in the first "PB Quicktime Converter" dialog. In both cases, the image sequence folder will be named with the same name as the MOV.

Single File:
Click "OK" twice to return to the "Export As" dialog. The button "Save" starts the conversion. The result is a image sequence folder with the name and path chosen in the "Export As" dialog.

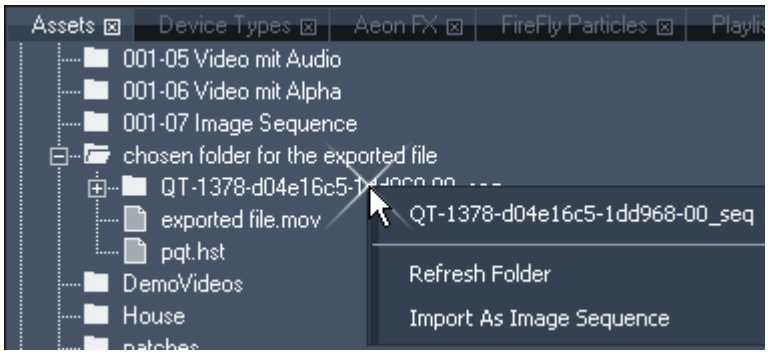
Batching:
Click "OK" twice to return to the "PB Quicktime Converter" dialog that starts the conversion automatically. The result are several image sequence folders with the original source names, all in the folder chosen in the "Output Folder" text field in the "PB Quicktime Converter" dialog.

drop-down in "Standard Video Compression Setting" dialog	From Latest file	
"Choose Folder" button	choose where you saved the MOV file (e.g. C:\folder A)	
Single File) "Export As" dialog: C:\folder A\newname.mov		
Result for single converted file	C:\folder A\newname.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\newname\newname_[00000001].png C:\folder A\newname\newname_[00000002].png etc.	
Batching) "PB Quicktime Converter" dialog: C:\folder A\		
Result for batch list item #1	C:\folder A\sourcename1.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\sourcename1\sourcename1_[00000001].png C:\folder A\sourcename1\sourcename1_[00000002].png etc.	
Result for batch list item #2	C:\folder A\sourcename2.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\sourcename2\sourcename2_[00000001].png C:\folder A\sourcename2\sourcename2_[00000002].png etc.	

drop-down in "Standard Video Compression Setting" dialog	Unique Number	
"Choose Folder" button	choose where you saved the MOV file (e.g. C:\folder A)	
Single File) "Export As" dialog: C:\folder A\newname.mov		
Result for single converted file	C:\folder A\newname.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\folder A_[001]\folder A_[001_00000001].png C:\folder A\folder A_[001]\folder A_[001_00000002].png etc.	
Batching) "PB Quicktime Converter" dialog: C:\folder A\		
Result for batch list item #1	C:\folder A\sourcename1.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\folder A_[001]\folder A_[001_00000001].png C:\folder A\folder A_[001]\folder A_[001_00000002].png etc.	
Result for batch list item #2	C:\folder A\sourcename2.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\folder A_[002]\folder A_[002_00000001].png C:\folder A\folder A_[002]\folder A_[002_00000002].png etc.	

drop-down in "Standard Video Compression Setting" dialog	None (All Files in Main)	
"Choose Folder" button	choose where you saved the MOV file (e.g. C:\folder A)	
Single File) "Export As" dialog: C:\folder A\newname.mov		
Result for single converted file	C:\folder A\newname.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\folder A_[00000001].png C:\folder A\folder A_[00000002].png etc.	
Batching) "PB Quicktime Converter" dialog: C:\folder A\		
Result for batch list item #1	C:\folder A\sourcename1.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\folder A_[00000001].png C:\folder A\folder A_[00000002].png etc.	
Result for batch list item #2	C:\folder A\sourcename2.mov C:\folder A\...hst	(these files can be deleted)
	C:\folder A\folder A_[00000001].png C:\folder A\folder A_[00000002].png etc. Note that this overwrites the files from the first batch item!	

To playback the image sequence in Pandoras Box, right-click on the folder in the Assets tab, and choose "Import As Image Sequence".
To play it back without using Pandoras Box, use a media player that supports image sequences.


Assets
Device Types
Aeon FX
FireFly Particles
001-05 Video mit Audio
001-06 Video mit Alpha
001-07 Image Sequence
chosen folder for the exported file
exported file.mov
pqt.hst
DemoVideos
House
QT-1378-d04e16c5-1dd968-00_seq
Refresh Folder
Import As Image Sequence

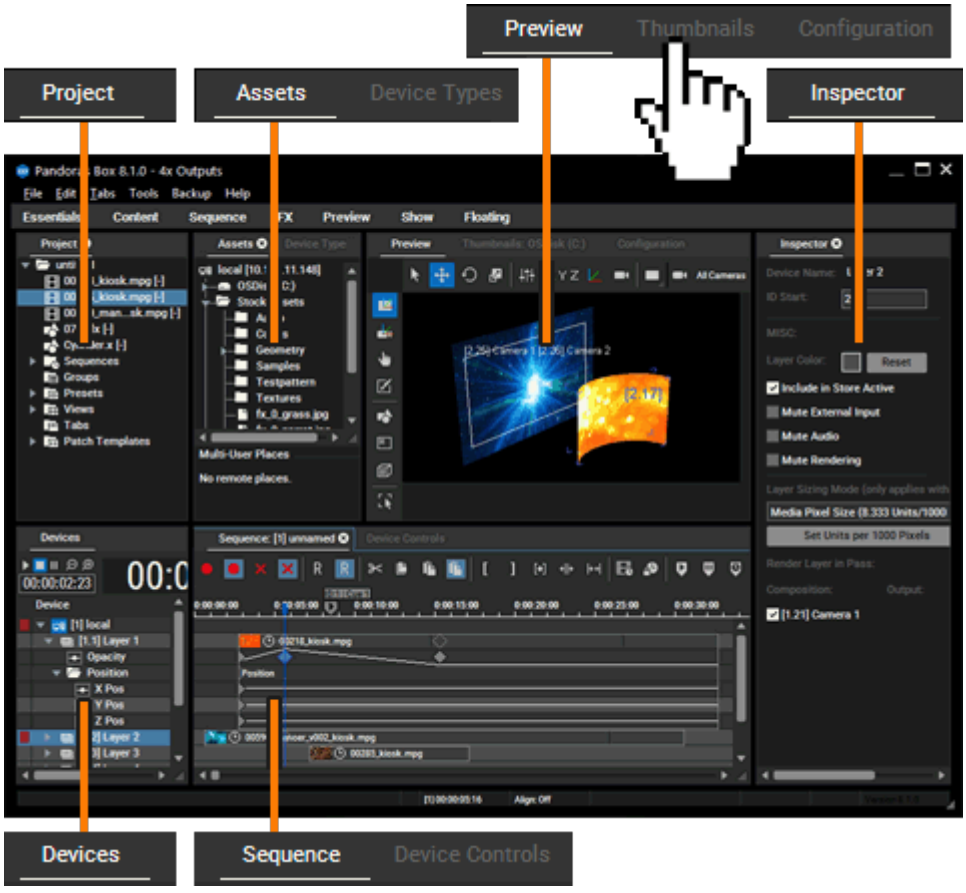
6.3 User Interface - Master

This chapter explains the user interface (UI) of a Pandoras Box Master system. Please follow these links if you are interested in the Client's interface [316] or Master / Client Remote Setup [67].

This chapter is divided into the following sub chapters. Below, you will find general information about the UI:
Startup Dialog [119]: The menu that is displayed when the master software is started
Menu and View Bar [119]: The menus File, Edit, Tabs, Tools and Backup seen in the top and quick access to different views of the interface
Layout [313]: Influence and rearrange the look of the user interface
Tabs Overview [125]: Overview of all available tabs
Keyboard Shortcuts [314]: Lists all shortcuts existing in the Master version of Pandoras Box

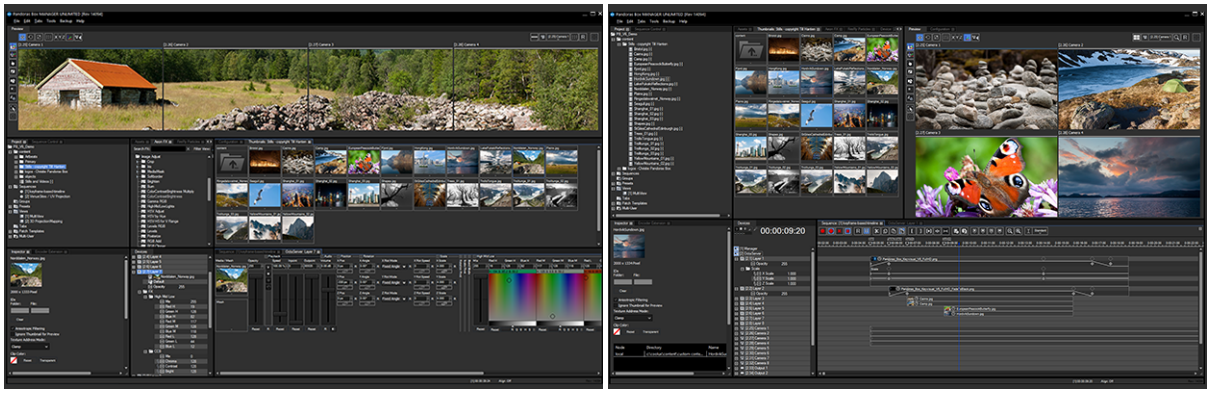
To display the UI correctly, please check the following two Windows settings. Firstly, we recommend to activate "ClearType". Secondly, the "DPI scaling" should be set "100%".
For Windows 8, both settings can be found under "Control Panel > (Appearance and Personalization >) Display".
For Windows 10, please go to "Settings > System > Display" and you will find the option "Scale and layout". Set the scaling to "100%", and even if it says so already, click on "Advanced scaling options" and enter "100" into the field which is populated with 100-500% by default. Further down in the "Display" menu you will find the "ClearType Tuner".
Follow the Window instructions to apply the changed settings.

This image from a master system includes links to different chapters, please click on a tab header to read its respective description. The tabs not included in the default views can be found in the topic "Tabs Overview [125]".

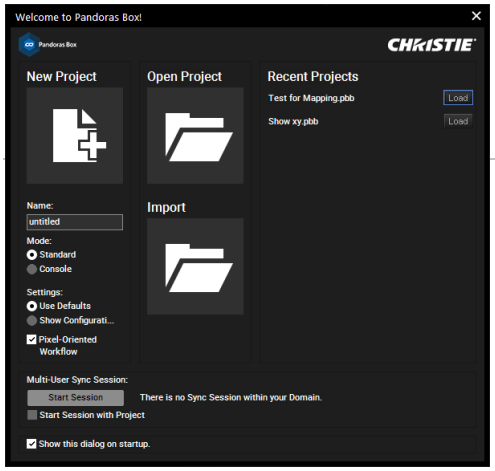


The user interface is designed for real-time playback and operation; therefore all important functions are accessible directly through a tab and view oriented workflow.
Pandoras Box supports a dynamic context-menu structure. A right-click on a particular item opens a context sensitive menu and shows the corresponding commands.

Of course the entire user interface layout can be customized [313] to meet your needs. The tabs size and location can be adjusted either within the main frame or independently on the entire desktop area as a so called breakout pane. You may store several views [310] and recall them with one click at any time during operation.



6.3.1 Startup Dialog



The Startup Dialog (or Welcome screen) shows all options to start right-away with the project of your choice. You may load a new project (with your favorite settings) a recent project or one from the directory.

New Project and its Settings

On the left side you may choose to open a new project. If necessary, change the default settings and then click the button.

- Name: Enter a name for your project. It will appear in the Project tab [278] and will be the file name when you later on save the project to a directory of your choice. Until you do so, the project is temporarily saved under "C:\Christie\content\projects\temp_projects\temp" (or C:\coolux\...).

- Mode: Decide to load a standard project or one that has optimized settings for being remote controlled by a lighting console [183].

- Settings: Choose whether the default settings should be applied or whether a pop-up should show the Configuration tab [139] first to let you decide particular settings. Since version 6 the pixel-oriented workflow is the new default. Untick the option to work with generic units as in version 5. All unit settings can be found in the Configuration tab > Unit Management [160].

Open Project and Import

The button "Open Project" in the middle section opens a dialog where you can choose the directory from where to load an existing project saved in the PBB format. In case you like to open a project in the older XML format, please click the "Import" button.

Recent Projects

The right side shows a list of recently saved or opened projects. Simply click on the "Load" button next to a name.

Multi-User

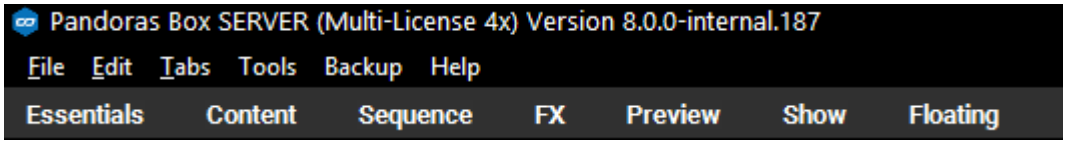
In case other Masters are found in the network, you will see the Multi-User [234] options to either start a new session or join an existing one. You can also check the option "Start Session with Project" if you always like to start a Session when creating a new project.

Show Dialog

Click the check box at the bottom if you do not like to display this dialog on the next startup. The same option can be found in the Configuration tab > Startup [150].

6.3.2 Menu and View Bar

This chapter describes the View Bar and the Menu bar on top of your Pandoras Box Master Software



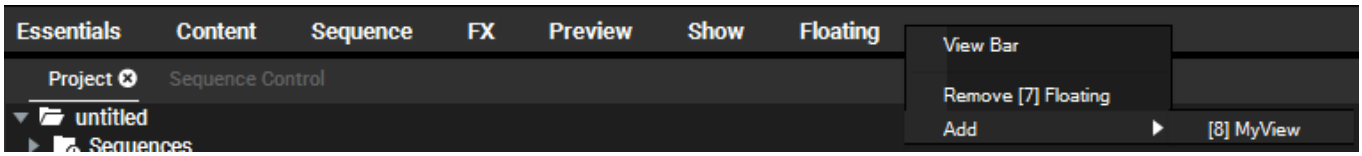
Menu Bar

The Menu Bar contains the following application menus:

- File Menu [119]
- Edit Menu [121]
- Tabs Menu [122]
- Tools Menu [122]
- Backup [123]
- Help (?)

View Bar

The View Bar shows seven views. The first, called "Essentials", is the default view. The others are named "Content, Sequence, FX, Preview, Show, Floating" and emphasize a particular use of the UI. Simply click on a View to change the currently shown tabs and their arrangement.



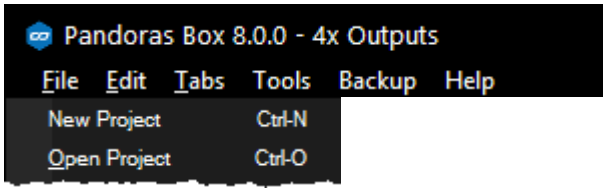
To change a view to your needs, open the "Views [287]" folder in the Project Tree, right-click the according View and choose "Update". This saves the current layout. See the "Views [287]" chapter for more information how to update or save new views. It also explains the "Export" and "Import" option.

You can add a new View to the View Bar to access it faster. You can either drag it there from the Project tab, or right-click it and choose "Add to View Bar" or make a right-click in the View Bar and choose "**Add**".

You can drag a view within the View Bar to another place. To remove it from there, right-click it and choose "**Remove**".

6.3.2.1 File Menu

The file menu allows you to achieve all project related actions.



New Project
Click here to create a new project. Before leaving the current project you will be asked if you want to store the current project. You will have to choose if you want to create a project in Standard Mode (shortcut [Ctrl + N]) or in Lighting Console Mode [183].

Open Project
Click here (or use the shortcut [Ctrl + O]) to open a dialog where you can choose the directory and name of a project saved in the PBB format. In general you can only open files that were saved in the same Pandoras Box version or an older one.
You can also open projects that were saved in a version before v8. In those versions PB offered dedicated Graphic and Text Layers. If you used them, they will be loaded as such but you are not able to add more of them. It is

recommended to test older projects before running a show with them.
If you like to open a project saved in the discontinued XML format, please contact support.

Open Recent Project
This opens a list with recently used projects.

Save Project
Click here (or use the shortcut [Ctrl + S]) to save the current project. This also saves the project for all Multi-Users [234].
Please note: There is no auto-save function!

If you save the project the first time, a dialog opens where you can choose the directory and name for your project. Note that the "Pandoras Box Binary" format with the extension "pbb" is the new default and saves the project as a binary file. The advantages of a binary code are a smaller memory space and thus faster saving and loading. To use the older XML format, you need to "Export" the project.

Save Project As
Click here (or use the shortcut [Ctrl + Shift + S]) to save the current project under a different name and go on working in the renamed project. This also saves the project for all Multi-Users [234] under the new name and switches their project to the new one.

Save Project Copy
Click here (or use the shortcut [Ctrl + Alt + Shift + S) to save a copy of the current project and go on working in the current project. This is an ideal option for saving backup versions of your project. A nice workflow is for example to create a "Backup" folder in your project folder and choose this folder when saving with the "Save Project Copy" option. For a nice overview, you could name your copies with the date or with a description, e.g. "projectname_20170531_0815" or projectname_v10_before warping". If you like to load a project copy (or backup) and in case you work with content saved in the "assets" folder, it makes sense to copy or paste the backup file into the folder with the "assets" folder, i.e. next to the original project file as this ensures that your content is consistent. The benefit of the "assets" folder is explained in the next paragraph "Bundle Project".

Please note, that this saving option does not work in a Multi-User [234] environment.

Export Project
This opens a dialog where you can choose the directory and name to save your project in the older XML format. The above options to save the project use the newer PBB format. The advantages of the "Pandoras Box Binary" format are a smaller memory space and thus faster saving and loading.

Bundle Project
Click here (or use the shortcut [Ctrl + Shift + B]) to bundle your project under a new name and location. All content used in the timeline will be copied to the new location in a folder called "assets" next to the show file. As a result you will have one folder containing the show file and all timeline content which you can then share with others or archive on an external drive for example (of course you could also save the bundled project there directly).

Info: Since version 5.5 the "assets" folder serves as a relative path in Pandoras Box. If you add content from this folder to your PB project, PB saves this as a relative path. That means that you can move the project folder (including the show file and asset folder) to a new location without the need of re-linking the content.

Close Project
Click here (or use the shortcut [Ctrl + W]) to close the current project.

Remove Unused Resources
Click here to remove all content files from your project that are not used in the timeline.

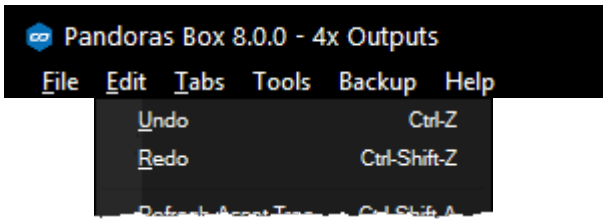
Clean up Inconsistent Resources
Use this command if you like to remove inconsistent resources from the Project tab [278].
Inconsistent resources are marked with a red exclamation mark. Click on one and check the Inspector [190]. At the very bottom you see the file location table [193] with at least one entry that says "inconsistent". This means that the file does not exist in that location any more. Under circumstances this might be alright and you could select the entry and click "Remove" to delete the entry from the table. If the table does not list inconsistent entries anymore, the file status changes to "consistent" again and the red exclamation mark is removed. In case you have many inconsistent files and you checked that this is "alright", you can simply use the command "Clean up Inconsistent Resources". This removes ALL "inconsistent" entries from ALL files. Bare in mind, that a file that is not present on

any Site anymore will be removed from the Project tab and containers in the Sequence will be named "None" instead of the old file name.

Exit
Click here (or use the shortcut [Ctrl + Q]) to Exit Pandoras Box. You will be asked if you want to store your project if there are not stored changes.

6.3.2.2 Edit Menu

The edit menu allows you to execute the following commands. Most commands influence the Sequence tab [292] and the Device Tree tab [173] and can be found in the respective context menus as well.



Undo
Undoes a step of sequence editing as well as changes in project tab. Actions that are not stored to the timeline like changing a value on fader cannot be undone.

Redo
Redoes a step of sequence editing as well as changes in project tab.

Refresh Asset Tree
If any changes have been made on the local hard disc system, you have to refresh the File Tree in the Asset Tab to see these changes inside Pandoras Box. Please note that refreshing the whole tree will close all sub folders and you have to open them again. See the Asset tab [131] to get information about how to refresh single sub folders.

All Active
Sets all parameters to the active status.

Clear All Active
This command clears all active parameters.

Reset All
Resets all parameters to their default values and removes their active status.

Store Active
Stores all active parameters as containers to the sequence.

Store Active (Selected Devices
In contrast to the above command, this one filters active values for those devices (layers) that are selected. The values are stored at the current time at the blue Nowpointer. All active values influencing other layers will stay active.

Store Active To Time
A small dialog opens and asks for the time whereto all active values will be stored. The time can be entered in the format H:MM:SS:FF or shortened to SS:FF or even SFF. So, for example 3 seconds can be 0:00:03:00 or 03:00 or 300. Click Enter or the button "OK" to close the dialog and save the active values.

Copy
Saves an selected item, for example a key or a clip container from the sequence.

Paste
Pastes the copied selection to the device where it was copied from in the Sequence Tab.

Paste to Selected Devices
Pastes the copied selection (keys, clip containers) to other devices in the Sequence Tab. To do this, select one or more devices holding down the [Shift] or [Ctrl] key. Please see Sequence [292] for detailed selection information.

Cut
Cuts out an selected item, for example a key or a clip container from the sequence.

Split Clip
This applies to a selected container and splits it into two separate containers at the position of the Nowpointer. If needed, parameter keys are automatically inserted at the end of the first and the beginning of the second container.

Merge Clips
This applies to two selected containers and merges them to one container. Note that both containers need to be on the same Layer and they should be adjacent, i.e. no other container must lie between them. Parameter curves are automatically created in the space where no container was before and they are based on the parameter keys from both containers. Bear in mind that parameters like "Media" or "Mesh" can only exist once per container, hence the first, earlier container overrides that information from the second, later one.

Trim Left
This applies to a selected container, it erases everything between the Nowpointer and the left clip boarder. The last key of a parameter before the Nowpointer will be moved to the new clip boarder time. The video clip will now be shortened at the beginning.

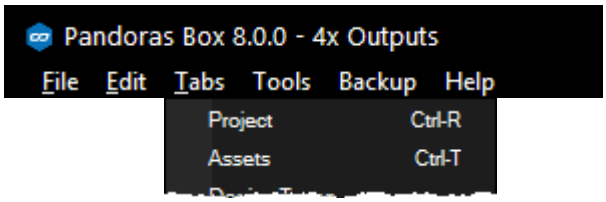
Trim Right
This applies to a selected container, it erases everything between the Nowpointer and the right clip boarder. There will be a new key set at the right clip boarder containing the value a parameter had at the Nowpointer before.

Trim To Clip Borders
If a clip container was scaled down, there could be hidden keys beyond the clip borders. This command removes these hidden keys and sets new keys at the left and right clip border to ensure the wanted behavior in the visible part of the clip.

Add Cue at Current Time
Adds a cue at the current time, indicated by the blue Nowpointer. Click onto the cue to select it and see its properties in the Inspector tab [208].

6.3.2.3 Tabs Menu

The tabs menu allows you to open / reopen tabs in the user interface. Each tab is explained in detail in the topic "Tabs Overview" [125] and "User Interface [117]".

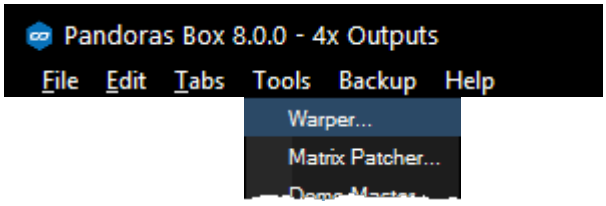


Please note:
There is no option to open a preview tab, because you can not close this tab.

The Extension drop-out includes all extensions that are available like the optional Encoder Extension [103].

6.3.2.4 Tools Menu

The Tools menu allows you to open additional tools that are installed with your Pandoras Box installation.

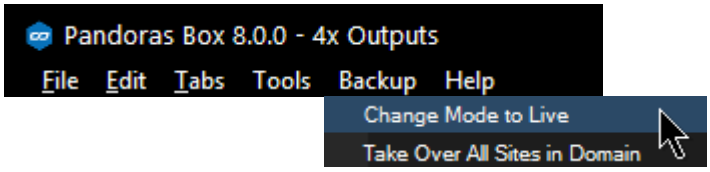


Warper [2129],
Matrix Patcher, [2077]

Dome Master, 2055
Image Converter, 2069
Splitter, 2120
Art-Net Monitor 2051
Leica 3D Disto 2076
Quicktime Converter 108

6.3.2.5 Backup Menu

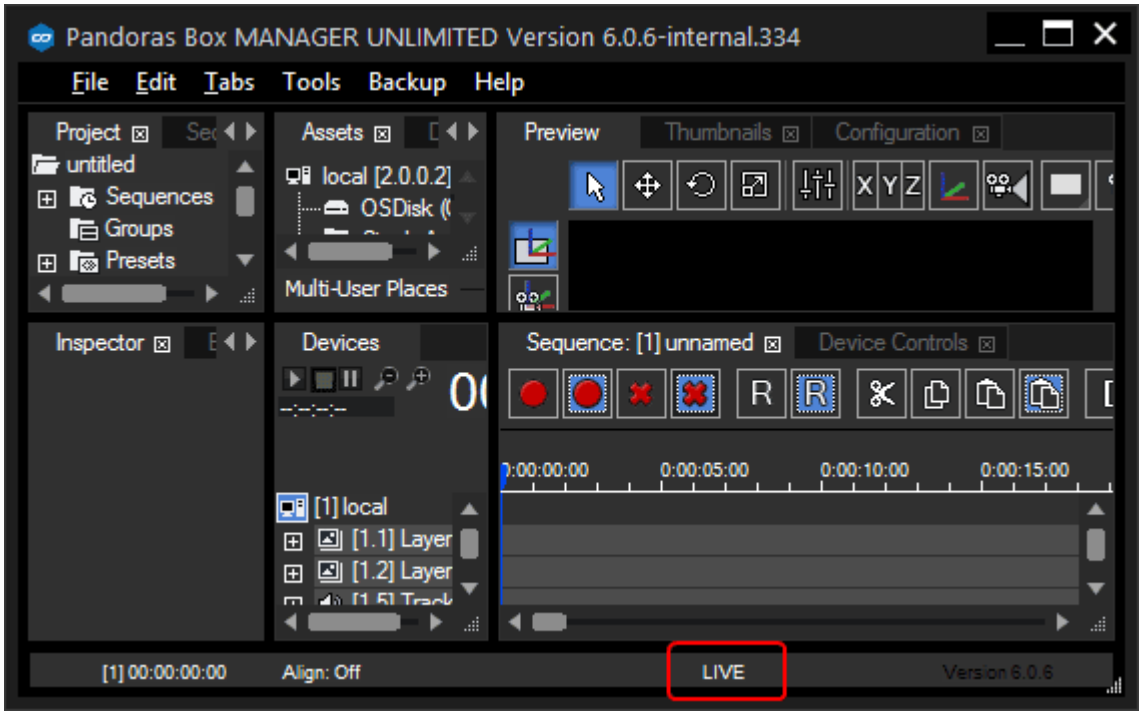
The backup menu is of interest if you run a second Master for backup reasons or if you are in a Multi-User environment with several Masters. In general, the command "Take Over All Sites In Domain" helps you (re-) connecting Clients in case of problems. It simply tells all available Clients to connect to this Master device.



Backup

You can run a "Show" Master and a second "Backup" Master in the same domain without causing any conflict. As long as one device is in the Backup mode, it won't give out any control data.

Per default, a Master is started as the Show Master and displays "Live" in its status bar, in the lower right corner of the user interface.



To change a Master into backup mode, go to the Backup menu and choose "Change Mode to Backup". The status line at the bottom of the user interface displays"Backup" highlighted in orange.

If it becomes necessary to toggle this device to Show Mode, choose "Change Mode to Live" or "Take Over All Sites In Domain". Both commands quit the Backup mode and tell the Clients to connect to this Master device.

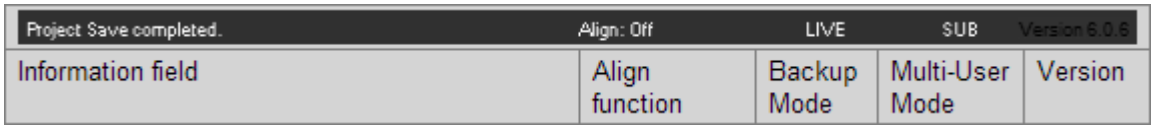
Multi-User

As explained in the chapter Multi-User [234], there can only be one Main Master but several Sub Masters. Per definition, the Main Master hosts the session and the connection to the Client(s). If a Sub Master calls the command "Take Over All Sites In Domain" he will become the new Main Master and will be connected to the Clients directly. The previous Main, becomes a Sub Master.

Again, the "Main" or "Sub" status is displayed in the status bar, in the lower right corner of the user interface.

6.3.3 Status Bar

The Status Bar at the bottom of the Pandoras Box Master user interface includes four sections.



Information Field

This field informs you about specific actions, like loading or saving the project file or spreading media files. If there is a show critical or very important information this field is highlighted red and the information stays. In that circumstance you can click onto the field to open a dialog containing a description.

Align Function

If you multi-select Layers or Cameras, you can align their parameters following a certain pattern. Per default the Align function is switched off. You can click onto the "Align" label to switch through the available options: <, >, ><, <>, Off.

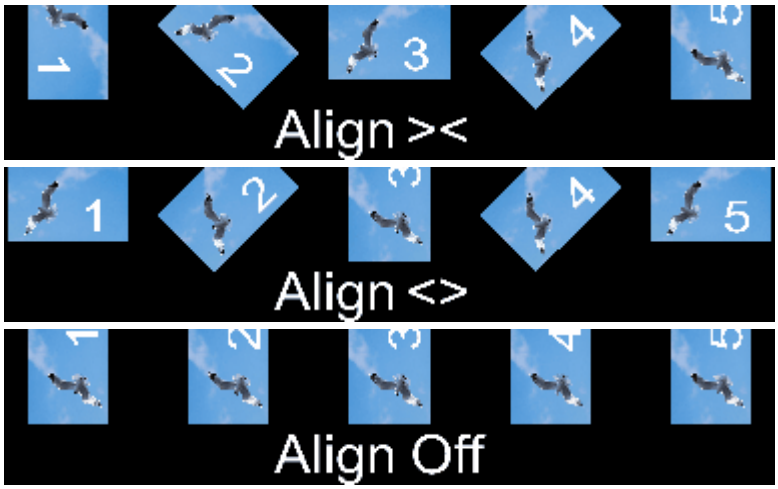
Example: Assign five different media files to Layer 1-5,then scale and position them like depicted in the image.

To select all Layers, click on Layer 1 in the Device tree [173] and hold [Shift] whilst clicking on Layer 5. The Align function depends on the order of selection, in our case we have selected Layer 1 first, then 2,3,4 and 5. If you now open the Device Controls tab [171] you will see the parameters of Layer 5, this is visualized in the Device Tree with a white dotted border. You can click on the other Layers to see their parameters in the Device Control tab without loosing the selection. Re-load Layer 5 into the tab. In case you made an error with the selection, press [Esc] to deselect and then select the Layers again.

With this selection, click on the Align label in the Status Bar to switch to another Align pattern. Change the Z Rotation Angle parameter of Layer 5 to 90°. The value range of 0°-90° is applied to the Layers in different patterns but the change from one layer to another is always equal. See the result in the Preview.

The applied rotation values increase.
Smallest value = Layer 1.
Highest value = Layer 5.

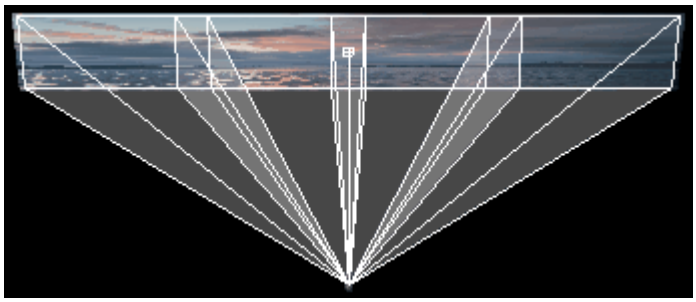
The applied rotation values decrease.
Smallest value = Layer 5.
Highest value = Layer 1.



The applied rotation values increase from the middle but in opposite directions.
Smallest value = Layer 3.
Highest value = Layer 1 and 5.

The applied rotation values decrease from the middle.
Smallest value = Layer 1 and 5.
Highest value = Layer 3.

The applied rotation values are constant.



The Align function can be applied to almost all parameters like position, rotation, scaling, effects etc. You can align Layer parameters but also Camera parameters and others. For the left image the Lens Shift "X Offset" parameter of four Cameras was aligned.

Backup Mode

This field shows whether your Master is in "Live" or "Backup" Mode. For more information please see the chapter "Backup Menu" [123].

Multi-User Mode

This field shows whether your Master is a "MAIN" or "SUB" Master within a Multi-User session [234].

Version

This field informs you about the version you are using. The version from the Master and Clients must be the same and is depicted in the lower right corner of each interface.

6.3.4 Tabs Overview

This is a list of all available tabs in the PB Master software. If you are not familiar with the tab names, the chapter "User Interface [117]" might be of interest for you, as it includes a picture with integrated links, showing all tabs available in the default view.

Active Values [127]
Aeon FX [129] (FX Explorer)
Assets [131]
Audio and Video Export [135]
Audio and Video Recording [137]
Button Bar [293] (Sequence)
Configuration [139]
Cues [168]
Curve Editor [169]
Device Control [171]
Device Tree [173]
Device Types [183]
Device Viewer [184]
Encoder Extension [185]
File Browser [185]
FireFly Particles [185]

Groups 189
Inspector 190
Media Encryption 222
Multi-User 234
Patch 228
Playlist Editor 239
Presets 241
Preview 243
Project 278
Sequence 292
Sequence Control 303
Task Manager 305
Text Input Editor 307
Thumbnails 310
Time Monitor 309
Views 310
Virtual Site 311

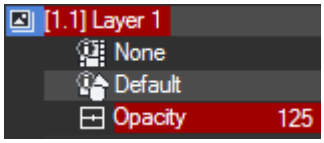
6.3.4.1 Active Values

The Active Values tab in Pandoras Box gives you an overview of all Layers and parameters with active values.

The Active Values tab is not part of any default View [287]. To open it, please click on "Tabs" in the Toolbar and choose "Active Values" or use the shortcut [Ctrl + D]. Even if it was already open and simply overlaid by another tab, it is now the visible tab within the pane [313].

What Is an Active Value?

In Pandoras Box device parameter values are designed to have different states: active and normal.



Changing a parameter in the Device Tree [173] or Device Control tab [171] for example, highlights it in red (everywhere in the user interface) and marks it as an active values. Active values stay active until they are stored to a Preset [283] or in a Sequence, as explained in the chapter "Programming on the Timeline" [296]. Alternatively, you can discard the active state with the commands "Clear" or "Reset". For example, if you change the parameter "Opacity" to 125, you can either store "125" as a Preset or as a key in the Sequence. Or, you can clear it, which means that "Opacity" is not active anymore, but holds "125". Lastly, you can reset it, which means that "Opacity" is not active and holds the default value "255".

Now, there are two common ways to work and program in Pandoras Box. The first workflow is to store an active value as a key in the Sequence. If you store several keys over a time span and play the Sequence, you will see how the parameter changes according to the stored values.

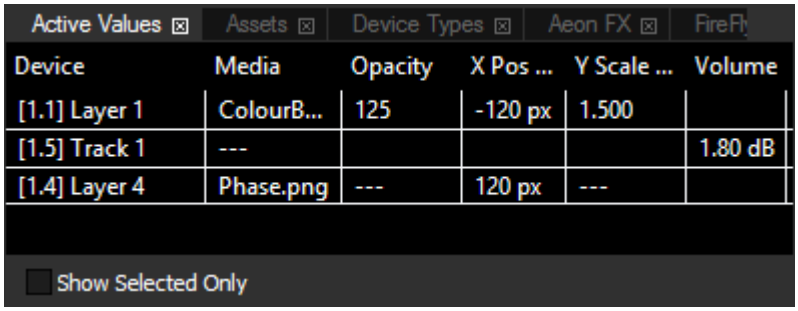
However, if you now create an active value again, you will see that it overrides the stored values. This is the purpose of active values! The active state indicates a change you have not stored yet and all values that have previously been stored in the timeline will be ignored until you store or reset the active value. As soon as there is no active value anymore, Pandoras Box automatically calls the values from the (playing) Sequence again.
There are different commands to "store" in the Sequence. You can either store a single active parameter, all active values from one or several Layer(s) or Site(s), or you can simple store ALL active values. Any store operation is always based on the active values.

Another common workflow is, to "ignore" the Sequence and possibility to store values but to work with active values only. Any time you like to change your parameters and Layers, you simply change the active values. You can do this manually, by calling Presets, or by remote controlling Pandoras Box via a lighting console, Widget Designer or another device or program.

In both workflows active values are very important for programming and this explains why you might need a dedicated tab to see all active values.

In case you are working in a Multi-User [234] scenario: The tab displays local and remote active values without distinction whilst the Device Tree [173] or Sequence [292] show remote active values colored in beige / yellow. Any store command will store only local active values. Remote active values can be turned into local ones with the command "Take over Activity" from the right-click menu from the respective parameter, Layer or Site.

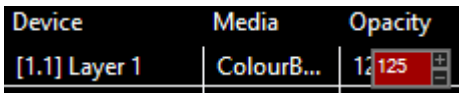
The Active Values Tab



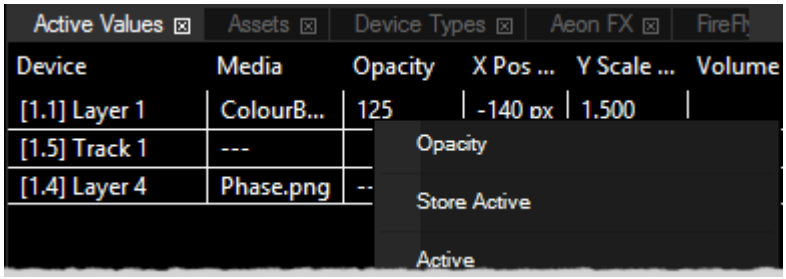
The Active Values tab gives an overview of all current active values. In case you run a store operation all listed values will be stored and the list is cleared.
To see all active values of the selected devices only, check the option "Show Selected Only" at the bottom of the tab. This might be especially helpful if you like working with the "Store active (Selected Devices)" command.

In case the list is not cleared after calling the store command, check the following:
- Is the check box "Include in Store Active" in the Layer Inspector [211] deactivated?
- Are you remote controlling Pandoras Box, i.e. constantly receiving (new) values from a lighting desk, Widget

Designer etc.?
- Are the values remote active values as you work in a Multi-User environment?



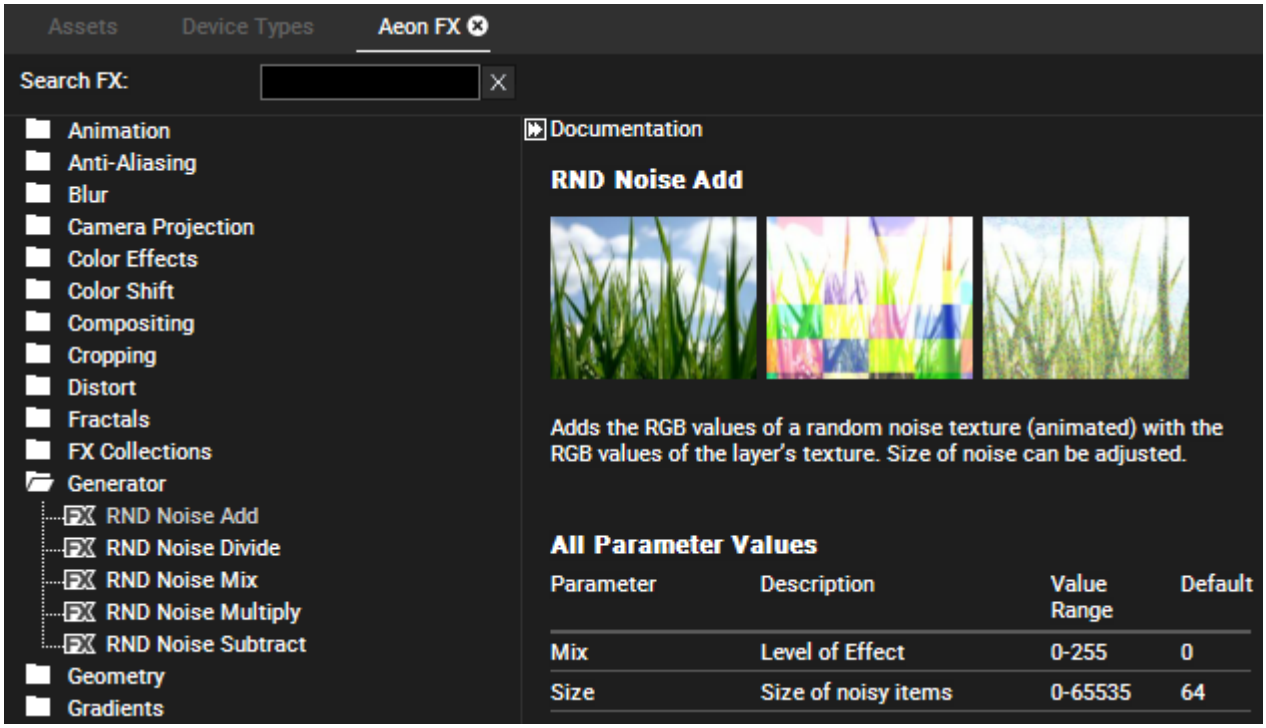
You can apply changes to a parameter value directly in the tab. You can either left-click and drag the value in one move. Or you can left-click it once to open the parameter box and then click the value in the box to edit it, e.g. double-click it and enter a new value with the keyboard or use the +/- icons.



To open the context menu of a value, simply right-click it. The first line simply tells you the parameter name, the value belongs to. The commands are explained in the chapter "Context Menus" in the paragraph "The Context Menu of a Parameter [181]". In other words, all the commands you can see in this tab can also be called within the Device Tree. In addition you can find the most important commands in the Sequence Button Bar [293].

6.3.4.2 Aeon FX and FX Explorer

The Aeon FX tab in Pandoras Box allows to browse through all available effects to add them to a Layer, e.g. Video or Output Layer.
The Aeon FX tab is part of the View [287] called "FX". In case you closed it, please click on "Tabs" in the Toolbar and choose "Aeon FX". Even if it was already open and simply overlaid by another tab, it is now the visible tab within the pane [313].



The effects are sorted by themes into FX folders. Open for example the "Cropping" folder to see all effects that crop a texture (graphic or video) somehow. To see what an effect does without applying it to a layer, you can load the documentation. Simply click on an effect and the right pane will display exemplary images, a short description what the effect does and a table with all effect parameters. The same information can be found in the manual, chapter FX list [327].

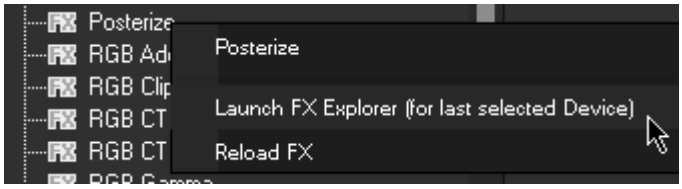
The effect may be assigned to a Video or Output Layer. There are multiple work flows available:
- select a layer first (multi-selection is possible with the [Ctrl] key) and then double-click the desired effect
- choose the effect and drag and drop it onto the Layer in the Device Tree tab
- choose the effect and drag and drop it onto the Device Controls tab, into an empty area right to the default parameter faders.

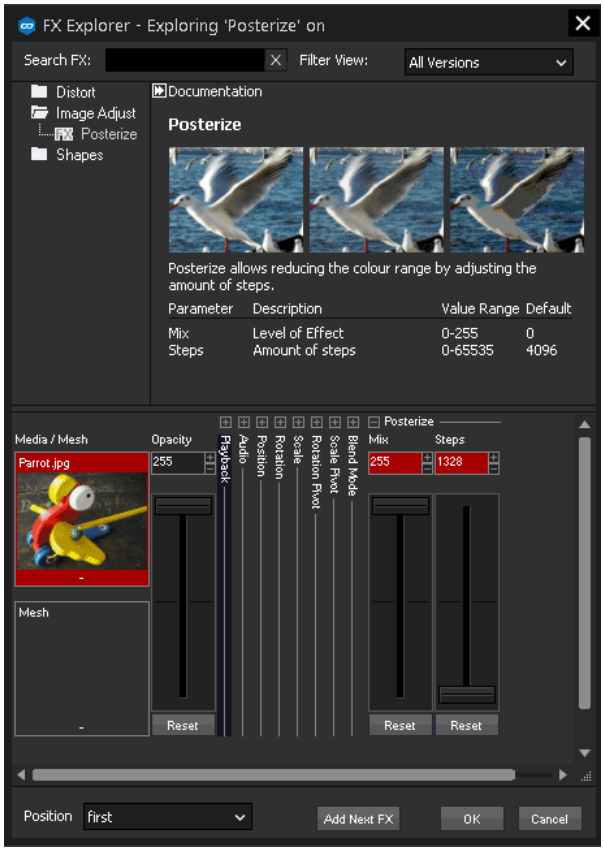
The chapter "Adding, Editing and Removing FX" [323] explains in further detail how to work with effects on a layer.

There is a search filter available: enter e.g. "Glow" into the "Search FX" field in the top left corner, and the list will only display those effects that contain "glow" anywhere in their name.

The FX Explorer

You can also work with the FX Explorer to try out an effect (or effect chain) on one or multiple layers. You may find it under Tabs > Launch FX Explorer or within the right-click menu from an effect in the Aeon tab itself. Select a layer first whereto you like to add an effect and open the FX Explorer, for this example we start the FX Explorer with layer 2.1 and the effect "Posterize". Make sure that the Preview tab is visible before opening the FX Explorer.

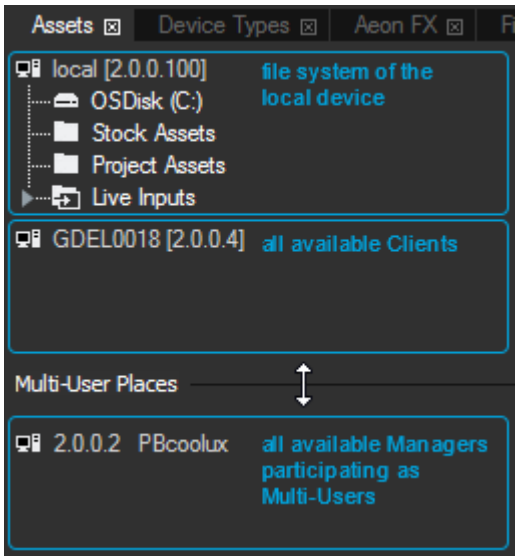




The FX Explorer starts in a new window. At its top you can see all effects and their documentation, just like in the Aeon FX tab. At the bottom you see the media fields and all other parameters including the effect parameters from the selected layer, just like in the Device Control tab. Note that the "Mix" parameter is already at its full value so that the effect's influence can be seen in the Preview tab right away.

In case you do not like the current effect, simply choose a new one by clicking once on another effect in the upper part. It instantly removes the first effect and replaces it. In this way you may try out very fast how the effect interacts with the layer's media file and how changing the parameters influences this.
If you like to keep that effect but add another one on top, click the "Add Next FX" button and choose another effect. With this second effect, you can use the drop-down list in the left bottom corner to influence its render order. For many effects it makes a difference in which order they are applied to the layer, for more information please see the chapter "FX Order" 325 that also explains how to change the order in the Device Tree tab later.
If you found the effect or effect chain you like to use, click the "Ok" button underneath the controls and the chosen effect(s) are added to the layer, the current parameters are marked as active values and the FX Explorer closes. The "Cancel" button would simply close the FX Explorer without adding the last effect.

6.3.4.3 Assets



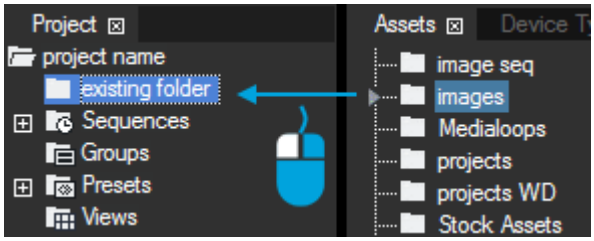
The Asset Browser (or shorter Assets) tab in Pandoras Box has two functions: file import (to the Project tab) [278] and loading available Client devices (to the Device Tree tab) [173].

The Assets tab is part of the default View [287]. In case you closed it, please click on "Tabs" in the Toolbar and choose "Assets" or use the shortcut [Ctrl + T]. Even if it was already open and simply overlaid by another tab, it is now the visible tab within the pane [313].

The Assets tab itself is divided into two sections. The upper part shows the local system and then all Clients in the network. The lower part lists all other Masters in the network that participate as Multi-Users [234] in your session. You can change the heights of both parts, if you position your mouse cursor on top the black line and drag it. Just like in any other file browser, you can "open" and access a listed system, drive or folder by double-clicking on it. If you open a system, you will see all available drives in alphabetic order and then three special folders which are explained further down: Stock Assets, Project Assets and Live Inputs.

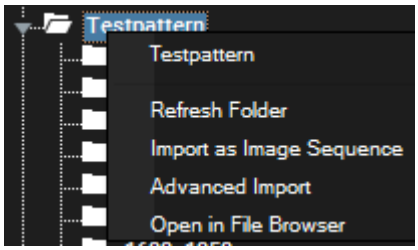
Troubleshooting: In case you miss a system in the Assets tab, make sure it is started in the same version [125] and Domain channel [147] as the Master is and that it is in the same IP address range.
In case, you cannot double-click or drag something in or from the Assets tab, check if there is a virus software or firewall blocking these actions.

File Import in General



You can only import supported file types into Pandoras Box. An overview of valid files can be found in the chapter Content [90] whilst the chapter Context Menu-Project [279] explains further media types such as Text inputs etc.

Files are imported from the Asset tree by drag and drop into the Project tab. As said above, you can (un-)fold a drive or folder by double-clicking it. You can also click on the small triangle displays in front a folder or on the vertical line. You can select a single file, multiple files (hold down the [Ctrl] key or [Shift] key) or folders and drag them into the Project tab. There, you can either drop the files into an existing folder including the very first project folder or simply into an empty space. Note that the content is listed in a slightly different order than known from the Windows File Explorer. E.g.: `0,1,10,9,99,AB,abc,de,DEF, _underscore`



If you right-click on a drive or folder, the context menu opens and offers a "**Refresh**" Tree / Drive / Folder command. Alternatively, you can use the shortcut [F5]. This will load the selected entry once more which is of interest if you have moved folders, created or renamed files whilst Pandoras Box was already running. Note that all included sub folders will be closed and you have to open them again. Thus it is a good idea to reset only the desired folder.

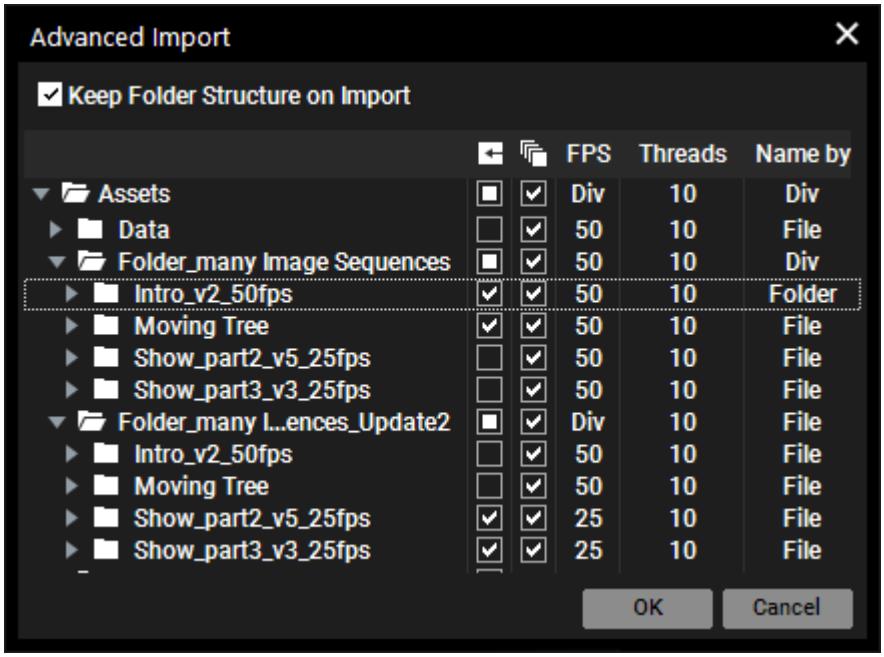
Another available command in the context menu is "**Import as Image Sequence**". If you drag and drop an image sequence folder, Pandoras Box would import all images separately. To import the folder in terms of one single file that can be played back, use this command. A dialog opens and asks for the frame rate; depending on how the content was created enter for example 25fps or 30fps. Find more information about this in the chapter "Image Sequence Formats" [95]. The next option, "**Advanced Import**", is described below. The last command from the context menu is "**Open in File Browser**" and it opens a new File Browser tab [185] which shows the content of the respective folder from the Assets (or Project [278]) tab as thumbnails. You can open as many File Browsers as you like. If you are interested in changing the layout of the user interface, please see the chapter "Layout [313]". It describes how to add or break-out panes and how to save and load views.

Alternatively, if you like to see the content of a folder as thumbnails, select it and open the Thumbnails tab [310] (in the right pane that also contains the Preview) [243]. From there or the File Browser tab, you can also drag single files

into the Project tab. After adding files to the project you can view its properties, e.g. width or length, in the Inspector tab [190]. Simply select the file in the Project tab to see it there. At the top you will see the most important properties. The following options depend on the file type and are explained in the chapter covering the File Inspector [191]. At the bottom, you will see a table listing all file locations.

It is very important to understand, that importing a file to Pandoras Box, simply creates a link to its location on the hard drive. In case you work in a network with other remote Clients and Multi-User Places and they are part of your project, imported content is automatically spread to them too, which creates a hard copy of the file on the respective system(s). A system can only access a file, if it is available on its own hard drive. In Pandoras Box, however, only the link for each system is saved and listed in the file location table [193]. For more information about Spreading please see the chapter Project [278].
If you like to see files only locally, and spread them later for example, you can deactivate the automatic Spread function in the tab Configuration > Resources (Global) [155]. However, if you do so, make sure that all files are available on the local system. This might require a manual spread in case you import content from a remote system. This is especially important, if you want to transfer show files to other systems later on for backup scenarios or other purposes or if you like to bundle [120] the project. For a long time the recommended workflow in Pandoras Box was to import files exclusively from the local hard disc system of the Master system. You do not need to follow this recommendation if you understand manual spreading and the file location table.

Advanced Import



The dialog "Advanced Import" facilitates content handling when dealing with many folders and /or image sequences.

In the Assets tab, simply right-click the main folder with all your content and choose "Advanced Import". A dialog opens, analyzes the content and shows all sub folders. All settings apply to containing sub folders but can be individualized if necessary. At the end, all content will be imported to the folder that is currently selected in the Project tab. Per default, the folder structure from the hard drive is adopted. However, if you uncheck the option "Keep Folder Structure on Import" all items will be imported directly without further sub folders.

Click the first check box "Import" for each folder you would like to import.

The second option "As Image Sequence" allows to import the folder as an image sequence instead of individual images. If Pandoras Box recognizes that the folder contains an image sequence, this option is activated automatically.

Further, you can set the frame rate ("FPS") and threads. The frame rate depends on how the content was created, for example with 25, 30 or 50 images per second. "Threads" is a playback option which depends on the chosen image format. It can also be changed later in the Image Sequence Inspector [199], e.g. if the playback is not fluent. The number alters how many threads are called by the Pandoras Box application from the operating system in order to run the sequence. The number of threads must be smaller than the total number of images in the sequence. Find more information about this in the chapter "Image Sequence Formats" [95].

The option "Name by" is set to "File" per default which means that Pandoras Box names the image sequence in the Project tab after the first file in the folder. If you like to name it according to the folder's name, choose the second option "Folder".

Stock Assets, Project Assets and Live Inputs

As mentioned above, if you unfold the local or a remote system, you will see all available drives in alphabetic order and then three special folders: Stock Assets, Project Assets and Live Inputs.

The **Stock Assets** folder is a shortcut to the location `C:\Christie\content\Stock Assets` (or `C:\coolux\...`). If you install Pandoras Box a few content folders are generated here, e.g. "Testpattern" with test pattern images in various resolutions. The standard Stock Assets are only a few examples but they can help you in case you do not have much content yourself. Mainly, it is very "typical" content that can be used in various and frequent programming tasks and thus it makes sense to have fast access to it. Of course you can create your own sub folders and content in the Stock Assets folder.
You can also download the Stock Assets Installer from our Download-Center which gives you more test pattern and over 4000 images supplied by the companies Apollo, GAM and LEE who use them for their gobos in moving lights.

The **Project Assets** folder is a shortcut to the folder "assets" that is automatically generated next to your project file as soon as you save it. Thus it is not a fixed path but leads to your project path. If you copy content files to that folder and drag them from here into your Pandoras Box project the content links are relative. That means that you can move the project folder (including the *.pbb file and asset folder) to every path wanted without any inconsistency effect and the need of re-linking the content. This is especially of interest for users that use the "Bundle Project" feature.

The **Live Inputs** folder lists all available live inputs, i.e. video and audio input cards or devices devices as well as network streams (e.g. NDI or StreamiX feeds) 724. The chapter Input and Output Cards 1947 explains all available cards for Pandoras Box hardware. If you like to use a graphical or audio input, please drag it to the Project tab in the same way as you would do it with regular content. All Live Inputs are analyzed once, hence it is not recommended to change the resolution after adding the source to the project. An exception to this are NDI streams. Note that you can setup a Pre Roll 207 time for Live Inputs, which is 2 seconds per default, when using them as Containers in the Sequence to reduce or compensate for the time the loading process takes.
Bare in mind, that each Client will list its own live inputs and that live inputs are not shared by Pandoras Box over the network. In other words, if only your Client but not the local system offer a video card, you will not be able to preview the source on the local system, but if you assign it to a layer on the Client, it will display it. If your local system does have a video card connected to the same video signal, you can attach it to the Client's live input. The attaching feature is explained in the topic File Location Table 193.

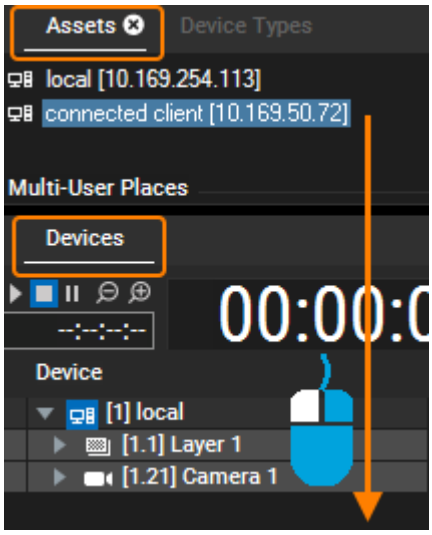
NDI® provides separated video and audio network streams via uni- or multicast. Besides many devices like CCTV cameras which are able to output NDI feeds directly, there are numerous applications for computers and apps for mobile devices available. Once an NDI input is added to the project and assigned to a Layer, the according PB system subscribes to the stream and displays the content. In difference to other live inputs, there is no need to use the "Attaching" feature if you like to use the stream on multiple systems. Simply spread the NDI stream after you added it to the Project tab. However, all PB systems need to be able to receive the stream directly, no matter on which network adapter. So keep in mind that Pandoras Box just receives the stream, it is not transmitting it to others.
Another difference is, that PB does not require a constant resolution for NDI streams. This allows using applications where the stream's resolution changes dynamically, e.g. due to network issues. The Inspector 190 offers "Maximum Texture Width" and "Height" fields for NDI streams. When adding an NDI stream to a PB project, the texture size adopts to the initial incoming resolution. You can change that by entering new values manually, or, in case the stream size has changed in the meanwhile by clicking the "Reload" button at the bottom of the Inspector. Some NDI applications change the stream resolution e.g. due to network capacity. In that case, PB strives to keep the texture size constant, which means that it resizes the incoming resolution to the "Maximum Texture Width/Height" but at the same time, PB adopts to the (new) incoming aspect ratio.

Pandoras Box supports all NDI variations including different compressions (i.e. NDI and NDI HX = H.264 version with higher compression / lower data rate) with a resolution of up to 4k@60 and the alpha option which needs to be activated in the Inspector 190.
NDI HX feeds have a higher latency because they have to be compressed as they are based on a 1G network. Technically, NDI has a latency of 16 lines but in practice it will usually be one frame. NDI HX has 30 to 50 milliseconds of delay which means depending on the framerate it will be 2 to 3 frames delay. NDI streams might not appear synchronized on multiple systems because of different transmission times via the network and decoding times. Keep an eye on the system performance and network traffic. The more streams you like to use or the higher the resolution or frame rate is, the more it becomes necessary to separate the show network from the streaming network. It is very simple to work with multiple network adapters. Simply choose one for the show traffic (Configuration > Network 147) and use another for your NDI streams. The system receives the streams

automatically without the need to setup the NIC explicitly. So just make sure, that the device or application that streams NDI uses a network that is also shared by the system that receives the stream.

NDI® is a registered trademark of NewTek, Inc. For more information about NDI itself, NDI tools etc., see http://ndi.tv/

Client Import

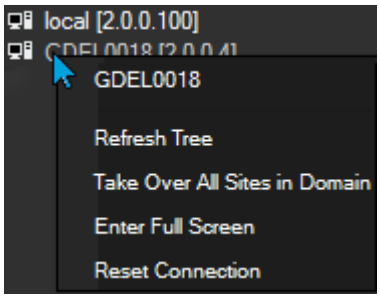


The Assets tab in the Master list all available Clients in the network. As written above, if you miss one system, make sure it is started in the same version [125] and Domain channel [147] as the Master is and that it is in the same IP address range.

Do add a Client to the Device Tree in order to access its Layers and program content on them for example, simply select it in the Assets tab and drag it in the empty space at the bottom of the Device Tree. You can also multi-select several Clients with the [Ctrl] or [Shift] key and drag them at once.

Note that the symbol for the Client indicates that it is "consistent" or simply connected. In case you see the symbol later on with a red exclamation mark, it would be inconsistent because it is not started for example. The topic Device Tree [173] with its sub chapters explain the further steps.

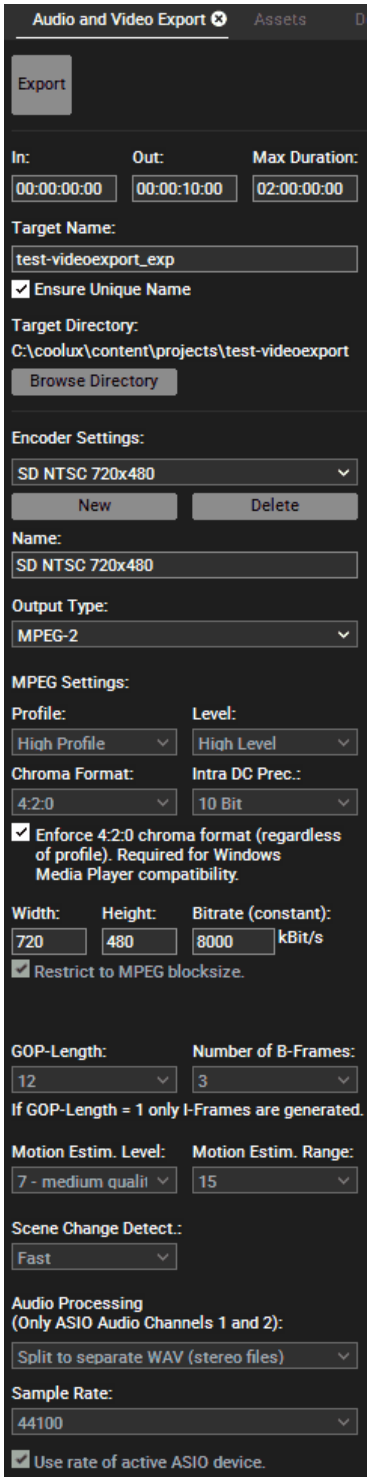
Similar to the File Inspector, the Site Inspector [210] shows more information about the added Client.



If you right-click on a Client in the Assets tab, the context menu opens and offers some commands.

"Refresh" or the shortcut [F5] load the selected entry once more which is of interest if you have moved folders, created or renamed files whilst Pandoras Box was already running.
The "Take Over" command is the same as in the Backup menu [123] and will tell the Client to connect to the current Master which helps in case it is connected to another Master.
"Enter Full Screen" and "Leave Full Screen" toggle the render window of the Client between full screen and normal window size.
"Reset Connection" simply reconnects to the Client in case you faced network problems.

6.3.4.4 Audio and Video Export



To open the Audio and Video Export tab, please click on "Tabs" in the Toolbar and choose "Audio and Video Export". Please note that this feature requires an activated Encoding option on the PB Master.

The Audio and Video Export tab allows you to define all settings for exporting audio and video from a sequence.
Set the Preview tab [243] to preview the specific output that should be exported. Note that not the Preview itself is exported. It is rather the image the camera sees, not the one from the output. As a result all changes made in the output are not included in the exported video. That includes effects on the output layer and parameters like softedge and keystoning.
If you like to change the aspect ratio of the exported video select a camera layer and change the camera aspect ratio [218].
If you like to change the resolution of the exported video make the according changes in the encoding settings.
If you export an uncompressed image sequence the resolution is adopted automatically to the current screen resolution the local system is set to.

Please note that incoming active values will be ignored, e.g. Widget Designer and Art-Net data. This is because the export is not running in real time, thus the incoming data cannot be allocated to a specific frame. Same applies to containers that are not locked to the timeline [206]. Free-run containers behave differently than during normal play back as the time reference is missing.

The Export feature is designed to work with MPEG2 videos, images and ASIO WAV tracks. Exporting other formats such as MOV and AVI might lead to unexpected behavior. Please see the File Inspector [191] for adjusting colorspace settings.

Also encrypted media [222] cannot be exported. On the other hand you may encrypt your exported video afterwards.

[Export]:
Click [Export] to start the video export. Please not that the user interface is locked while the export is in process. To abort the export, press the shortcut [Ctrl + Q]. Please note that the video export won't be done in real-time. The higher the quality and resolution set in the encoder the export will take a longer time to be finished.

[In]:
Define the timecode at which the export should start in the format hh:mm:ss:ff.

[Out]:
Define the timecode at which the export should end in the format hh:mm:ss:ff.

[Max. Duration]:
You may define the maximum duration of the exported video file in the format hh:mm:ss:ff. By default the max. duration is set to 2 hours.
Example: Having the inpoint set to 00:00:00:00 and the output set to 03:00:00:00, the video export will stop after 2 hours.

[Target Name]:
Enter the name for the video file you are going to export.
By default the video will be named "'Project name'_video".
If the option "Ensure Unique Name" is checked, the video file will be enumerated serially if the chosen name already exists.

[Target Directory]:
By default the video will be saved to the directory of your project file. Click [Browse Directory] to browse to a new directory if you want to change this path.

[Encoder Settings]:
In this section you may choose the settings of the file you're going to export.
There are several predefined profiles:

Profile Name	Output Type	Mpeg Settings		Resolution, px	Bitrate (constant), kbit/s	Audio Processing
		Profile	Level			
SD NTSC 720x480	MPEG-2	High	High	720x480	8000	separate wav
SD PAL 768x576	MPEG-2	High	High	720x576	8000	separate wav
XGA 1024x768	MPEG-2	High	High	1024x768	10000	separate wav
HD 1280x720	MPEG-2	High	High	1280x720	12000	separate wav
HD 1408x1056	MPEG-2	High	High	1408x1056	15000	separate wav
HD 1920x1080	MPEG-2	High	High	1920x1080	20000	separate wav
2048x1080	MPEG-2	High	MXL	2048x1080	80000	separate wav
4080x2160	MPEG-2	High	MXL	4080x2160	80000	separate wav
Uncompressed	Uncompressed Frame Images (.png)			Computer's resolution		none

These profiles are not editable. You may choose one of these profiles from the drop-down list or create a new profile by clicking on [New].

[New]:
Click [New] to create a custom encoding profile and name it.

[Output Type]:
Choose between MPEG-1, MPEG-2 and an image format to export an image sequence.
When using MPEG-1 or MPEG-2 go on with the MPEG settings. Pandoras Box Encoder allows to encode files larger than 1080p to our proprietary MXL format. The maximum resolution is 4094 x 2800 px, though it is recommend set 4080 x 2800 px as this conforms with the MPEG standard. The encoding process depends on available graphics card RAM.
If you like to export an image sequence, you may choose between three image formats: PNG as an lossless compression format, BMP as an uncompressed format and dds as an lossy compression format. For more information please see the chapter "Image Sequences" 95.
Now, simply enter the resolution for the final image sequence and tick the "Export Alpha" check box if you need transparent image parts. Adjust the audio processing as described below.

[Profile]
Set the MPEG Profile and Level.
Please note that you will need High Profile (Profile 4:2:2:) and High Level for HD 1920x1080 px resolution. The Level automatically changes to MXL if you encode files to a format bigger than 1920 x 1080 px.

[Resize Video]:
As long this option is not checked the video resolution will have the size of your windows desktop. Check [Resize Video] and insert a custom resolution for X and Y (in px) if necessary.

[Bitrate (constant)]:
Choose the bitrate (in kbit/s). Recommended are about 8.000 kbit/s for SD and 20.000 or higher for HD.

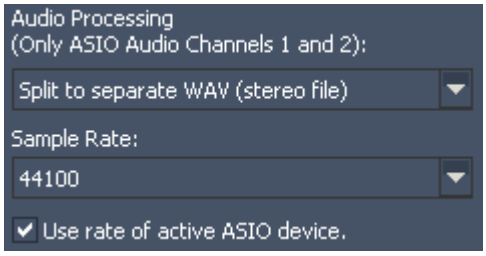
[GOP-Length]:
The length of the Group Of Pictures is editable from 1 to 100. The higher the GOP-Length the better the compression rate gets at the expense of quality. GOP set to 1 will use I-frames only.

[Number of B-Frames]:
Choose a value between 0 and 7.

[Motion Estimation Level]:
Set a value between 0 (no motion search) and 15 (high quality). The higher the value, the better the quality, the longer the encoding process.

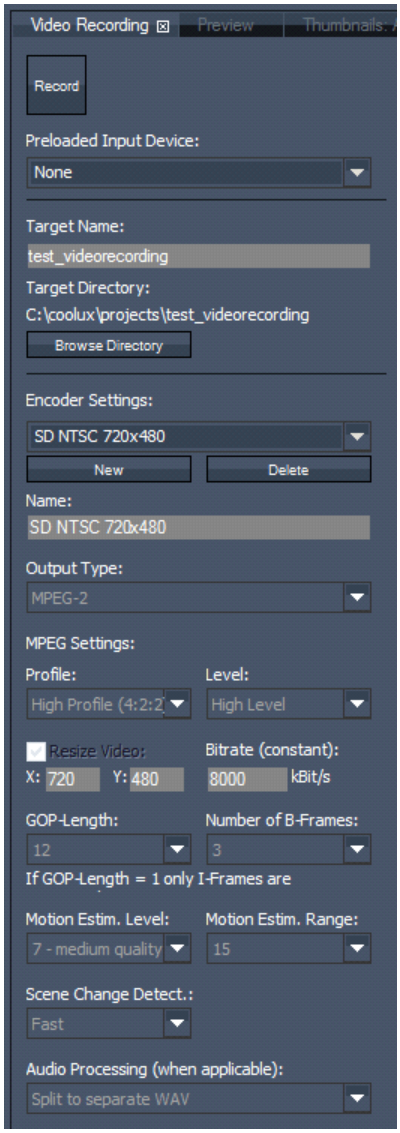
[Motion Estimation Range]:
Set a value between 0 (no motion search) and 15 (high quality).

[Scene Change Detection]:
Choose between None, Fast and Refined.



[Audio Processing]:
Please note that only the first and second ASIO channel are processed.
Choose between the options to...
- discard the audio information, it is simply not present in the resulting export files
- split to separate WAV (stereo file) to render one WAV file with two channels; the file is created in the same export folder
- split to separate WAV (mono files) to render two WAV files each with one channel; the files are created in the same export folder
- include the audio as a Audio Track; this is only possible for MPEG videos. Please see further information and why it is sometimes not recommended to include Audio Tracks in the MPEG format [97].

6.3.4.5 Audio and Video Recording



The Audio and Video Recording tab allows you to define all settings for recording a local video and / or audio live input source. Please note, that the recording tab is only available in the Master interface, and that the feature is restricted to local inputs excluding remote inputs from Clients.

Record:
Click the "Record" button to start the video and audio recording. Press this button again to stop the recording.

Video / Audio Input:
Choose the video and audio source you would like to record. Note that these sources must be added to the project first.

Limit Record Duration:
Enable this function and enter a time when you would like to record the sources for a defined time.

Target Name:
Enter the name for the video file you are going to record.
By default the video will be named "'Project name'_video".
Please note: if the file name already exists and you start a new recording with the same name, the existing file will be overwritten without warning.

[Target Directory]:
By default the video will be saved to the directory of your project file. Click [Browse Directory] to browse to a new directory if you want to change this path.

[Encoder Settings]:
In this section you may choose the settings of the file you're going to record.

There are several predefined profiles:

Profile Name	Output Type	Mpeg Settings		Resolution, px	Bitrate (constant), kbit/s	Audio Processing
		Profile	Level			
SD NTSC 720x480	MPEG-2	High	High	720x480	8000	separate wav
SD PAL 768x576	MPEG-2	High	High	720x576	8000	separate wav
XGA 1024x768	MPEG-2	High	High	1024x768	10000	separate wav
HD 1280x720	MPEG-2	High	High	1280x720	12000	separate wav
HD 1408x1056	MPEG-2	High	High	1408x1056	15000	separate wav
HD 1920x1080	MPEG-2	High	High	1920x1080	20000	separate wav
2048x1080	MPEG-2	High	MXL	2048x1080	80000	separate wav
4080x2160	MPEG-2	High	MXL	4080x2160	80000	separate wav

These profiles are not editable. You may choose one of these profiles from the drop-down list or create a new profile by clicking on [New].

[New]:
Click [New] to create a custom encoding profile and name it.

[Output Type]:
Choose between MPEG-1, MPEG-2 and Uncompressed.
When using MPEG-1 or MPEG-2 go on with the MPEG settings.
Using the uncompressed output type there is nothing more to set up. This output type will render uncompressed frame images (.PNG) in the resolution your local computer is set to. There will be no audio processing.

[Profile]:
Set the MPEG Profile and Level.
Please note that you will need High Profile (Profile 4:2:2:) and High Level for HD 1920x1080 px resolution. The Level automatically changes to MXL if you encode files to a format bigger than 1920 x 1080 px.

[Resize Video]:
As long this option is not checked the video resolution will have the size of your windows desktop. Check [Resize Video] and insert a custom resolution for X and Y (in px) if necessary.

[Bitrate (constant)]:
Choose the bitrate (in kbit/s). Recommended are about 8.000 kbit/s for SD and 20.000 or higher for HD.

[GOP-Length]:
The length of the Group Of Pictures is editable from 1 to 100. The higher the GOP-Length the better the compression rate gets at the expense of quality. GOP set to 1 will use I-frames only.

[Number of B-Frames]:
Choose a value between 0 and 7.

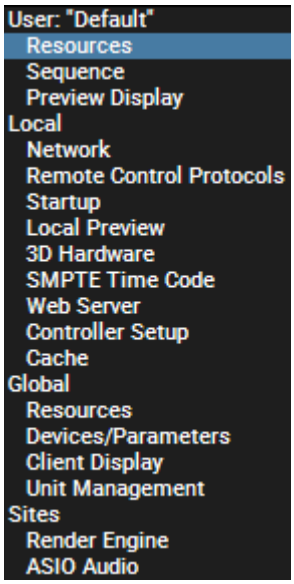
[Motion Estimation Level]:
Set a value between 0 (no motion search) and 15 (high quality). The higher the value, the better the quality, the longer the encoding process.

[Motion Estimation Range]:
Set a value between 0 (no motion search) and 15 (high quality).

[Scene Change Detection]:
Choose between None, Fast and Refined.

[Audio Processing]:
Discard the Audio Part (if applicable), split it to a separate wave-file (will be created in the same folder) or choose to include it. The Audio Part is discarded by default.

6.3.4.6 Configuration



The Configuration tab in Pandoras Box is divided into four categories: User, Local, Global and Sites. Each category offers several sections which are described in detail on the following pages. Please note that in the Pandoras Box interface itself, every section offers a question mark icon. When you click on it, the installed help file will open and display the according page.

The Configuration tab is part of the default View [287]. In case you closed it, please click on "Tabs" in the Toolbar and choose "Configuration" or use the shortcut [Ctrl + K]. Even if it was already open and simply overlaid by another tab, it is now the visible tab within the pane [313].

User
All User settings can be stored in profiles (see below) and apply to the local instance only. In case you work in a Multi-User environment, changes in profiles are shared with others.

- Resources [140]
- Sequence [142]
- Preview Display [145]

Since Pandoras Box V8 you can save all settings from this category in so called User Profiles. Any new project offers the User Profile called "Default". If you change a setting and save your project, it saves the changes for the "Default" profile. If you create another User Profile with different settings, it is saved in the project too. You can also export and import a profile. An exported profile has the file ending "pup". You can rename all profiles except the Default one; keep in mind that they must have a unique name.
In addition, profiles are automatically shared with other Users when working in a Multi-User session [234]. Note that the "Default" profile is selected per default, hence changes apply to these instances too. But of course, every User can apply an individual profile and as mentioned above it only influences the local instance. If deleting a profile, it also disappears for other Multi-Users. PB warns you in case you like to delete a profile used by another Mutli-User. The Configuration tab and Multi-User tab display which profile is used by which Multi-User.

Local
The Local settings refer to the local computer setup. Mostly they are saved in a local setup file and not in the project file!

- Network [147]
- Remote Control Protocols [148]: e.g. DMX, Art-Net, Midi...
- Startup [150]
- Local Preview [150]
- 3D Hardware [151]
- SMPTE Time Code [152]
- Web Server [153]
- Controller Setup [154]
- Cache

Global
The Global settings refer to all instances at the same time. In case you work in a Multi-User environment, these settings are shared (i.e. synced) with others.

- Resources [155]
- Devices / Parameters [156]
- Client Display [158]
- Unit Management

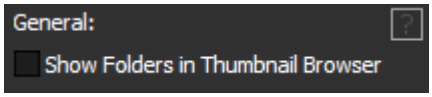
Sites
The Sites settings can be adjusted for each Site individually. In case you work in a Multi-User environment, these settings are shared (i.e. synced) with others.

- Render Engine [162]
- ASIO Audio [166]

6.3.4.6.1 Resources (User)

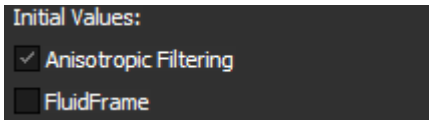
The section "Resources" in the User category [139] from the Configuration tab enables you to set up the general settings for resources to improve your workflow. In addition you can influence the (initial) properties of specific resources like videos, image sequences and others. Note that User Resource settings influence the local instance only while Global Resource settings [155] influence also remote instances. In a Multi-User environment they are shared with other Places.

General



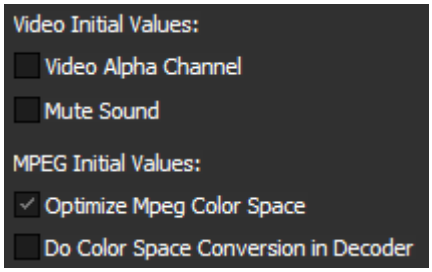
- **Show Folders in Thumbnail Browser**
If you click on a folder in the Project tab or Assets tab, its contained files are displayed as individual thumbnails in the Thumbnail tab. If the folder includes sub folders, they are depicted with a folder icon. If you do not like that and only wish to see the files itself, untick the check box.

Initial Values (for Resources)



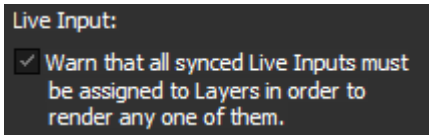
These options apply to newly added images, videos, playlists, etc. For more information about the settings see the File Inspector [191].

Video and MPEG Initial Values



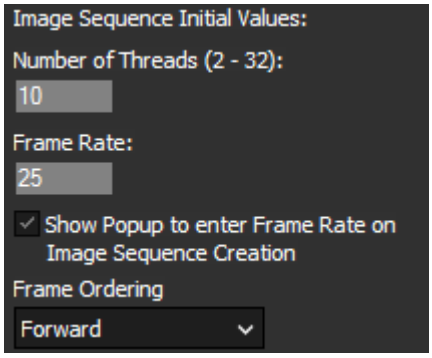
These options apply to newly added videos. For more information about the settings see the File Inspector [191].

Live Input



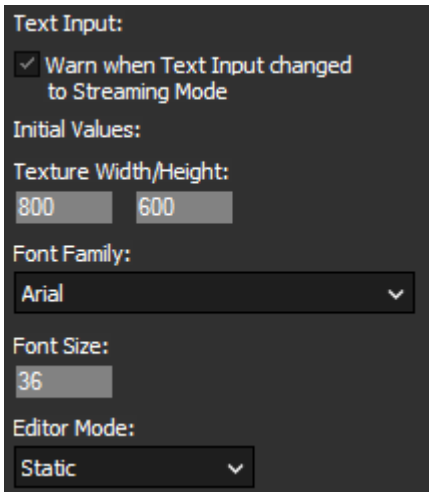
Choose whether you like to be warned with a message box when only one of the synced Live Inputs is assigned to a Layer.
If your hardware is equipped with a SDI input card which has genlocked input feeds with the same resolution and framerates, Pandoras Box can synchronize them. More information can be found in the chapters about the SDI input cards, e.g. 12G-SDI Input Cards > Synchronized [1974]....

Image Sequence Initial Values



These options apply to newly added image sequences. For more information about the settings see the Image Sequence Inspector [199].

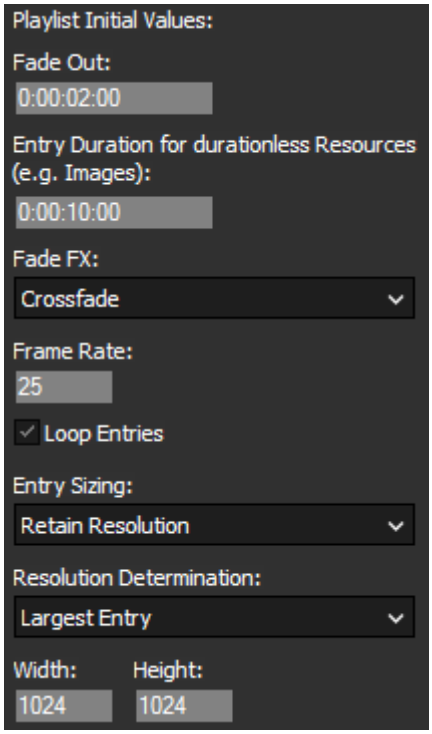
Text Input



Choose whether you like to be warned with a message box when a Text Input is changed from "Static" or "Scrolling" to "Streaming".

All other options apply only to newly added text assets.
For more information about the text settings see the Text Inspector Editor [307] .

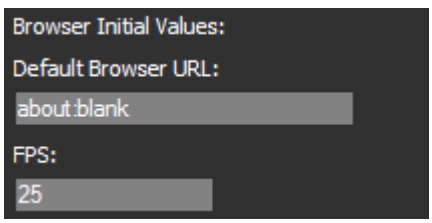
Playlist Initial Values



These options apply to newly added playlists [239].

For more information about the settings see the Playlist Inspector [202].

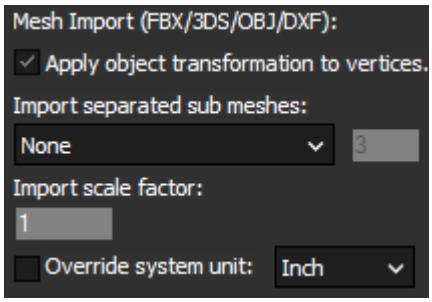
Browser Initial Values



This option applies to newly added Browser assets [279]. More settings are offered in the Browser Inspector [198]. The frame rate can be adjusted under Resources (Global) [155].

- Default Browser URL
Enter a URL that you wish to assign to a Browser asset as soon as it is created in the Project tab, e.g. https://www.christiepandorasbox.com

Mesh Import Initial Values



The following options are offered in an additional dialog as soon as you import an object. The dialog opens with the values set up in the Configuration tab.

- **Apply object transformation to vertices**
This is of interest if your object was saved with sub meshes or levels and one sub mesh exists multiple times. In case you de-activate "Apply object transformation to vertices" it can happen these copied sub meshes are imported with a different position, i.e. their own 0,0,0 origin (pivot point) has moved to the coordinate system's origin, the global 0,0,0. point.

- **Import separated sub meshes**
This is of interest if your object was saved with sub meshes or levels.
None: The imported object consists of only one mesh. All sub meshes are merged into it.
All: The original hierarchy is not changed. All sub meshes will be available as separate meshes. In the Project tab, double-click onto the main object and a tree structure will open with all available sub folders and sub meshes. Please note, that it is possible to use either a single sub mesh or an entire folder entry above it; the folder merges all consisting sub meshes when assigned to a Layer.
Specifies Levels: The original hierarchy can be flattened. Enter how many levels you like to use. If the original file was saved with more "deeper" levels, they will be merged.

- **Import scale factor**
Enter a scale factor for newly added meshes. For example: 0.5=half size; 2.0=double size

- **Override system unit**
If the "Scale factor" does not return the expected result, you may enable the option to override the system units the file was saved with. If you see this scaling issue with 3DS files, in most cases it helps to set "Inch".

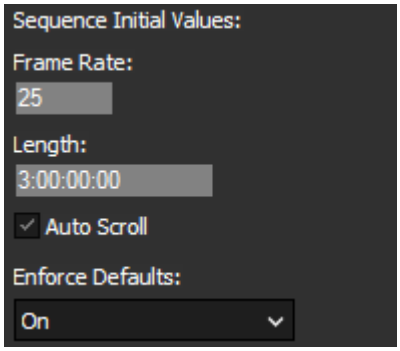
6.3.4.6.2 Sequence

The section "Sequence" in the User category [139] from the Configuration tab enables you to set up (initial) properties of timeline specific elements like the sequence itself, container, cues and others.

Line Height

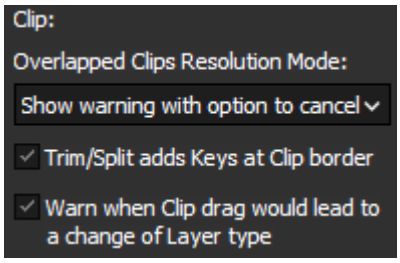
Choose a height (Standard (=small), Medium or Large) for the lines in the Sequence tab [292] in which the Layer or Parameter information is shown. This setting can also be found in the Sequence Button Bar [293].

Sequence Initial Values



These properties are applied as soon as a new sequence is added to the project. For more information about these and other sequence settings see the Sequence Inspector [204].

Clip

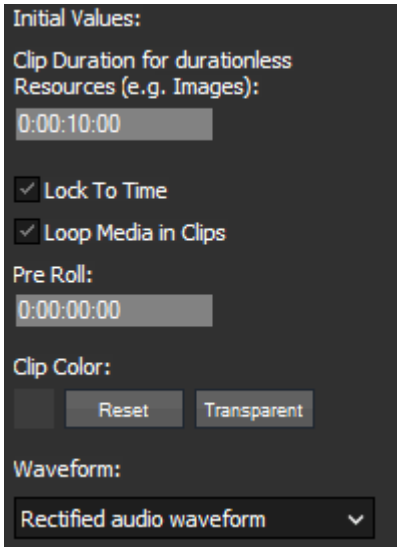


- Overlapped Clips Resolution Mode
When a container is being copied (or dragged) to a time where another container already exists there is a conflict. With the options from the drop-down menu you can decide how this conflict should be resolved.
- Show warning with option to cancel: A pop-up warns about the container overlap. You can choose to cut the existing container or to cancel the action.
- Cancel operation: The action that would lead to the overlap will simply not be executed. This is the same result as choosing "Cancel" in the above mentioned pop-up.
- The existing container will be cut at the beginning and/ or end of the pasted container. This is the same result as choosing "Ok" in the above mentioned pop-up.

- Trim / Split adds Keys at Clip border
When you trim a container or split it, per default a new key is added at the new container end. Image, that you have a container with a value fade from 0 to 255. If you split the container in the middle, a key with value 125 is added to new clip borders. Uncheck this option if these keys should not be generated automatically.

- Warn when clip drag would lead to a change of layer type
Since version 5.5 containers can be dragged [299] from one layer to another. When a container is dragged to a different layer type (that does not share the same parameters as the original layer) its borders turn red and a dialog pops up that warns you about the parameter key loss and needs to be confirmed. If you like to deactivate this additional dialog, untick this check box.

Clip Initial Values

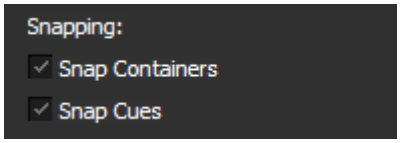


- Clip Duration for duration-less Resources
Images, meshes and other resources do not have a duration like videos. However, if you drag them onto a device in the timeline, they create a container that needs to have a duration. Per default, this container is 10 seconds long. If you like to change this for all new containers, enter a new time. For 5 seconds you can enter "5:00" or simply "500".

- Lock to Time
- Loop Media in Clips
- Pre Roll
- Clip Color
Change the properties that should be applied as soon as a new clip / container is added to the timeline. For more information about these and other clip settings see the Clip Inspector [206] and Clip Summary Inspector [206].

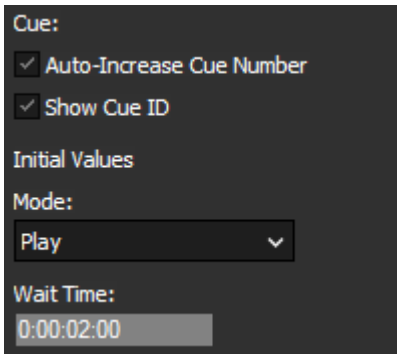
- Waveform
Audio Tracks [661] of Pandoras Box include a new parameter "Waveform" that displays audio peaks. In case you like to align other content (on Video Layers etc.) as accurate as possible it is recommended to zoom in closely. Use the [+] key on the main keyboard or the Zoom-in icon  in the Sequence Button Bar to do so. You can choose between two waveforms, a standard one with positive and negative peaks or a a rectified waveform with only positive peaks as depicted. You can find this setting in the Configuration tab > User > Sequence [142] > Clip settings

Snapping



- Snap Containers, Snap Cues
If you drag one or several containers or key frames in the Sequence [292] tab, you will notice that they snap to cues and to the borders of other containers. Key frames can also snap to other key frames within the same container. Zooming in and out could help finding the required snap target faster. To disable snapping temporarily, you can hold the [Alt + Shift] keys whilst dragging. Uncheck the options "Snap Containers" or "Snap Cues" if you do not like to snap to these elements permanently.

Cue



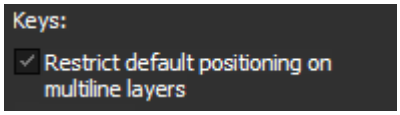
- **Auto-Increase Cue Number**
When checked, the Cue Number will automatically increase when the next cue is created. When unchecked, all cues will be created with ID 0 and can be edited later on.

- **Show Cue ID**
Per default the ID from a cue is shown in front of its name. If you like to see the name only, untick this check box.

- Initial Values
-- **Mode**
Choose whether a newly created key should be a Play-key or another mode.

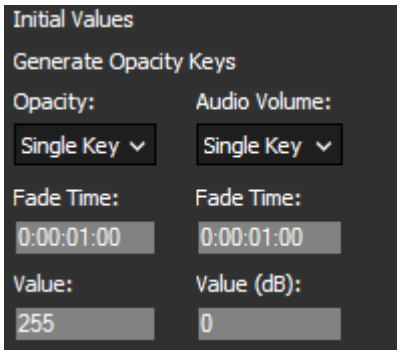
-- **Wait Time**
If you generate a Wait-key it has a default Wait time of 2 seconds. If you like to change this for all new Wait-keys, enter a new time. For 5 seconds you can enter "5:00" or simply "500".

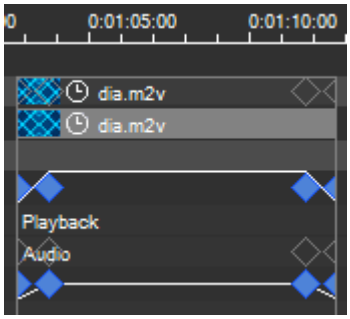
Key



This option is of interest when the line height (of a Parameter in the Sequence) is changed to "Medium" or "Large" as described at the top of this page 142. Right-clicking into a Parameter line creates a key with a certain value. As long as the "**Restrict**" option is ticked, this value is the default value, e.g. 255 for "Opacity" or 1 for "X Scale". In standard lines, all values are displayed at the same height but medium or large lines have more space and depict keys with a high value at their top border whilst lower values are placed further down. If you disable the "Restrict" option the initial key value is no longer the default value but accords to the Y-position of your click within the line borders.

Key Initial Values





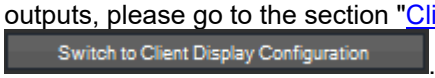
- **Generate Opacity Keys (in new Clips)**
This takes place when a container is created:
- when saving active values
- dragging media files from the Project tab directly in the Sequence
- right-clicking in a parameter track to create a key

Choose with the drop-down menus whether you want to have automatically created Opacity and / or Volume keys in new containers. You can choose to have
- a single key for a constant value
- two keys for a fade in (the maximum value for the fade is either 100% or the active value
- two keys for a fade out
- four keys for a fade in and out
- no key at all

The Fade Time can be entered in the format H:MM:SS:FF or SFF, e.g. "100" equals 1 second

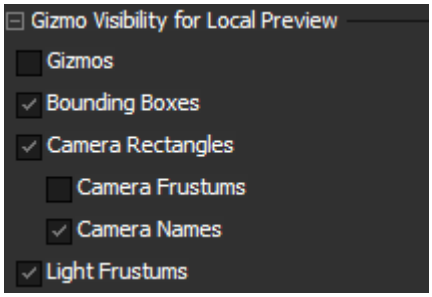
6.3.4.6.3 Preview Display

The section "Preview Display" in the User category [139] from the Configuration tab allows to influence the appearance in the Preview tab [243]. You can for example change the rendering of the Ground Grid, Gizmo and untextured objects. You can also open these settings through the right-click menu of the Preview tab > Preview Display Configuration.

Note that all settings here are for the local Preview only. To change the rendering appearance from all Client outputs, please go to the section "Client Display [158]" or click the according button 

.

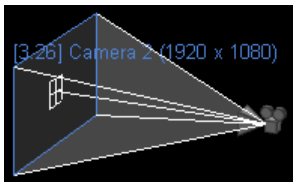
Gizmo Visibility for Local Preview



Choose to display or hide the following objects.

"**Gizmo**" describes the icon with three axes in the colors red, green and blue. Many objects in Pandoras Box have a Gizmo: Layers, Cameras, Mesh points, etc. but they all work in the same way described in the chapter Gizmos and Wireframes in the Preview [249]. Note that this check box is linked to the "Gizmo Visibility" button  in the Preview tab.

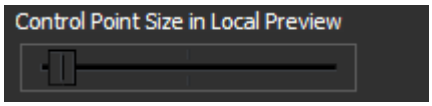
"**Bounding Box**" describes the blue box rendered around an object indicating its size. As long as you do not assign a three-dimensional object but work with flat 2D images or videos, the bounding box looks like a blue rectangle.



The three camera objects help visualizing cameras and their opening angle and Look-at point. "**Camera Rectangle**" refers to the rectangle (highlighted in blue) or border that shows what the camera "sees". The "**Camera Frustum**" includes the camera icon itself plus the pyramid connecting it with the rectangle. It consists of the Camera Wireframes (here in white), the four lines connecting the camera with each corner of the Camera Rectangle and the line to the Look-at point. Per default, the pyramid sides are shaded. The Frustum's appearance can be changed in the Camera Inspector tab [218]. Note that this check box is linked to the "Camera Visibility" button [249]  in the Preview tab. Lastly, the "**Camera Name**" (here in blue) simply means the display of the Camera device number, its device name from the Device tree tab [173] and resolution.

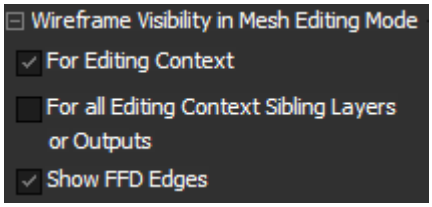
Likewise, a Light Device [666] has its own icon and Frustum which can be toggled with the "**Light Frustums**" check box. Its style can be changed in the Light Inspector [214].

Control Point Size in Local Preview

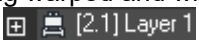


The fader influences the size of an FFD control point  , e.g when working in the Mesh Editing Mode [257].

Wireframe Visibility in Mesh Editing Mode



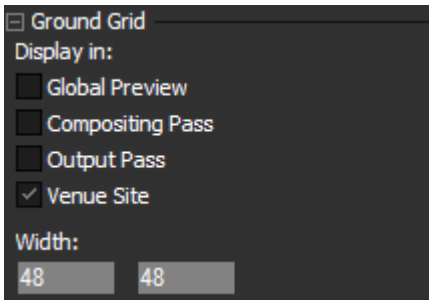
These settings refer to the rendering of Mesh and FFD grids when the Preview is toggled to the Mesh Editing Mode [257].

The check box "**For Editing Context**" toggles the visibility of the Mesh grid (not the points itself!) for the Layer that is chosen in the "Editing Context" drop-down list, i.e. the Layer that is currently being warped and which is depicted with the editing icon in the Device Tree: 

The next check box "**For all Editing Context Sibling Layers or Outputs**" toggles the Mesh grid on all other Layers and Outputs. These means that you can toggle the Mesh from the Editing Context Layer and all other ones separately. This is helpful if you like to align a Mesh to another one, e.g. within a softedge area where two Meshes overlap each other. The Mesh appearance, e.g. its color, can be changed in the Object Inspector [200].

"**Show FFD Edges**" renders the lines between FFD points.

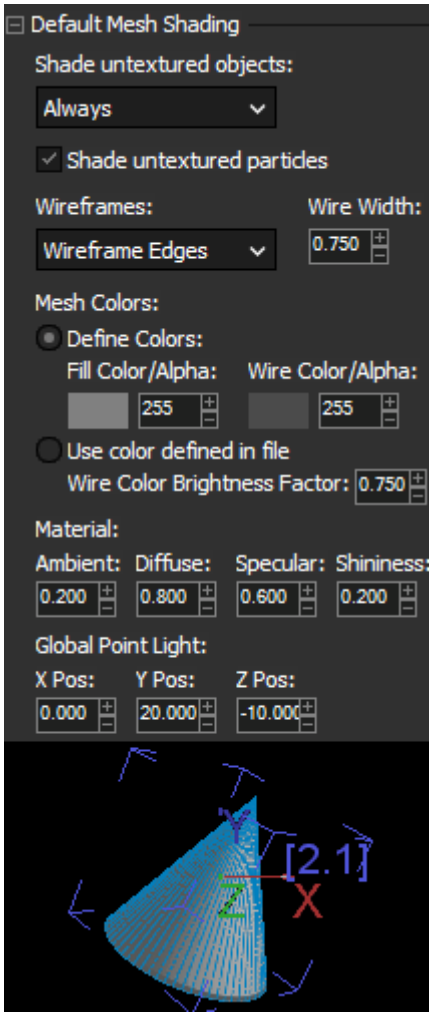
Ground Grid



- **Display in...**
With these check boxes you may choose were you like to see the Ground Grid. The "**Global Preview**" refers to the view when the "Select Preview" drop-down list is set to "All cameras" as described in the chapter Select Preview: Camera or Output View [245].
The Composition and Output Pass are described in the chapter Video Processing Pipeline [320].
Venue Sites are explained in the chapter Venue Sites [694].

- **Width / Height**
With the number fields you can influence the size of the Ground Grid.

Mesh Shading and Color



Please note that the Inspector for an object [200] allows to overwrite the below settings with individual ones.
The left bottom example shows an untextured object with gray material color and blue wireframe edges.

- **Shade untextured object**
With an activated option, even objects without an assigned texture are visible as they can be shaded according to the below settings e.g. colored and with visible wireframes.

- **Shade untextured particles**
If you assign an object but no texture (i.e. media file) to a Particle System [185], it can still be shaded according to the Particle Emitter Color [188] if this check box is ticked. If you disable this option, particles are only visible with an assigned texture.

- **Wireframes**
Choose whether an untextured object should be rendered without Wireframes, with Wireframe Edges or Wireframe Triangles.

- **Width**
Define the thickness of above mentioned Wireframes.

- **Mesh Colors**
Adjust the color of untextured objects.
--**Define Colors**
Click on the color fields to define a color that is used for the Fill Color and Wireframes. Use the number fields to adjust the transparency.
--**Use in file defined color**
If the object was originally saved with an assigned fill and wireframe color, Pandoras Box can use them also. Use the number field to adjust the wireframe's brightness.

- **Material**
Adjust the Ambient, Diffuse, Specular reflection levels and the shininess that influence how an untextured object reflects the global light, e.g. to appear rather glossy than dull. Further information... [527]

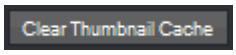
- **Global Point Light**
You may adjust the X-, Y- and Z-position of a global light that is reflected from untextured objects according to the "Material" settings.

6.3.4.6.4 Cache

The section "Cache" in the User category [139] from the Configuration tab allows to clear the internal cache in Pandoras Box.

Thumbnails are used in various places in Pandoras Box:
- the Thumbnails tab
- the Inspector when a media file in the Project is selected
- as a small thumbnail in a container in the Sequence tab
- in the Preview if "Use Thumbnails for Preview" (Configuration tab > Local Preview [150])

For every file in the Project tab, Pandoras Box generates an according thumbnail and saves it in the project folder\local_cache_thumbs.

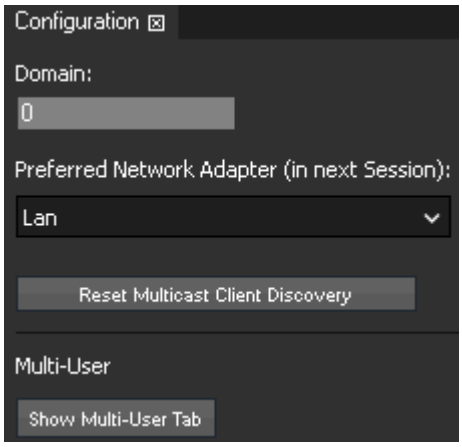


In case that you encounter any issues with loading images or thumbnails, you may press "Clear Thumbnail Cache" which deletes all files in the "thumbs" folder , and the next time you start the project, the thumbnail and image files will be rebuilt.

6.3.4.6.5 Network

The section "Network" in the Local category [139] from the Configuration tab enables you to set up connection properties applying to the Master-Client-connection. In addition Multi-User settings are available. Note that it belongs to the Local category [139] which means that the following settings apply only locally on your system.

General Connection Settings



- Domain
Per default, the Domain channel number is 0. If you have problems connecting to another PB system, make sure that the Domain is set to the same channel on all Masters and Clients [316] in the network. If the Master is on a different Domain than the Client, they cannot connect! It is recommended to only change the Domain if necessary, e.g. if there are two Masters in the same network that should only connect to a certain group of Clients.
Valid channel numbers are from 0-65535.

- Preferred Network Adapter (In Next Session)
The drop-down menu contains all network cards installed on the system. Select one and restart the application. Now the entire Pandoras Box communication of Master and Clients will be transferred by this defined adapter and its IP address. This applies for example to show data generated by Containers and their Keys in the timeline. With choosing a different network adapter for other protocols like Art-Net, you can separate different communication levels from each other.
These protocols can be assigned separately: Pandoras Box show data,DMX / Art-Net Mode [148], CITP Thumbnail Exchange [148] and Streaming ACN [149].
Here, "Art-Net" refers to the data generated when a layer has been patched or if DMX devices are included in the timeline. It does not refer to possible Art-Net data generated by using matrix patches on an output layer [682] (when working with LED walls and the Matrix Patcher [2077]).
If an explicitly named network adapter is not found, the "any" adapter will be used. "Any" is to be decided by the operating system Windows and its internal adapter order.

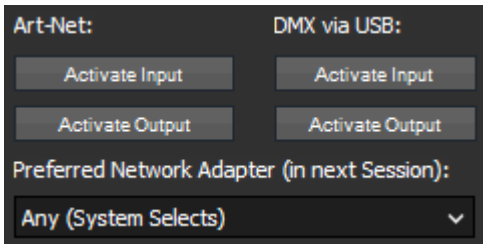
Multi-User Connection

- Show Multi-User Tab
This will open the Multi-User tab which is not activated per default.

6.3.4.6.6 Remote Control Protocols

The section "Remote Control Protocols" in the Local category [139] from the Configuration tab enables you to set up connections between your Pandoras Box Master system and other devices. You may receive and send Art-Net and DMX, exchange thumbnails via CITP or run MANet, Midi Show Control and Streaming ACN.

Art-Net and DMX



Art-Net:
- Press "**Activate Input**" to toggle Art-Net receive mode [706] for remote control of devices or sequences according to the settings in the Patch tab [228].
- Press "**Activate Output**" to toggle Art-Net output mode [721] for sending data from DMX devices added to the sequence.

Since version 5.3 the IP address does not necessarily need to be in the 2.x.x.x range to be able to receive Art-Net. Sending Art-Net is still tied to this specification. If you are working witch matrix patches [2077], you may address a different IP range there, but currently this is not possible for DMX fixtures.

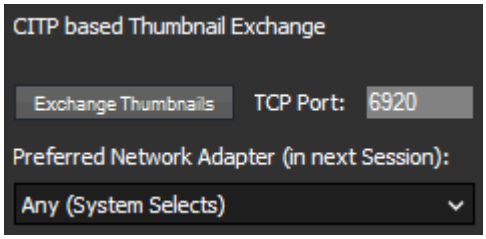
DMX via USB:
- Press "**Activate Input**" if you want to toggle DMX receive mode [706] via a local DMX Link In USB interface [1999]. This is for remote control of devices or sequences, see also the Patch tab [228].
- Press "**Activate Output**" if you want to toggle DMX send mode [721] via a local DMX LinkOut USB interface. This is for sending data from DMX devices added to the sequence.

Please note: While Art-Net supports multiple DMX universes simultaneously, a DMX Link USB interface offers only 512 DMX channels that are send on Subnet/Universe 0/0.

- **Preferred Network Adapter (In Next Session)**
The drop-down menu contains all network cards installed on the system. Select one and restart the application. Now the entire Pandoras Box communication regarding the DMX / Art-Net protocol will be transferred by this defined adapter and its IP address. This applies to the DMX / Art-Net data generated when a layer has been patched or if DMX devices are included in the timeline. It does not refer to possible Art-Net data generated by using matrix patches on an output layer [682] (when working with LED walls and the Matrix Patcher [2077]).

With choosing a different network adapter for other protocols like for the Pandoras Box show data, you can separate different communication levels from each other.
These protocols can be assigned separately: Pandoras Box show data [147], DMX / Art-Net Mode, CITP Thumbnail Exchange [148] and Streaming ACN [149].
If an explicitly named network adapter is not found, the "any" adapter will be used. "Any" is to be decided by the operating system Windows and its internal adapter order.

CITP based Thumbnail Exchange



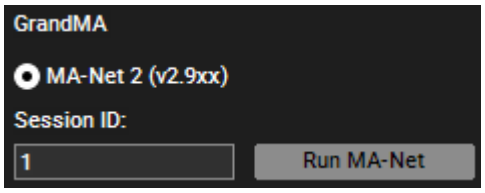
If you want to exchange Thumbnails with a Chamsys MagicQ console or with the Widget Designer:
- Enter a **TCP Port**
- Press "**Exchange Thumbnails**"

Please note: The ChamSys console / Widget Designer have to use the same TCP Port to get a connection.

Pressing "Exchange Thumbnails" again will stop the CITP connection.

- **Preferred Network Adapter (In Next Session)**
The drop-down menu contains all network cards installed on the system. Select one and restart the application. Now the entire Pandoras Box communication regarding the CITP based thumbnail exchange will be transferred by this defined adapter and its IP address. With choosing a different network adapter for other protocols like for the Pandoras Box show data, you can separate different communication levels from each other.
These protocols can be assigned separately: Pandoras Box show data [147], DMX / Art-Net Mode [148], CITP Thumbnail Exchange and Streaming ACN [149].
If an explicitly named network adapter is not found, the "any" adapter will be used. "Any" is to be decided by the operating system Windows and its internal adapter order.

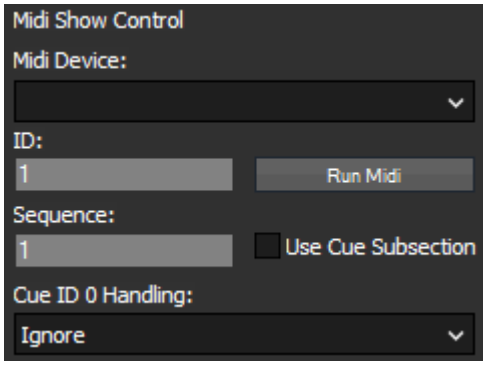
GrandMA



To interface with GrandMA lighting consoles, please enable the MA-Net:
- Enter the MA-Net **Session ID**.
- Press "**Run MA-Net**".

Since Pandoras Box 6.4.0 MA-Net version 1 and 2 are no longer supported. MA-Net Series 2 supports up to 256 DMX Universes. They can be patched in the Patch tab [228]. For more info please read the topic about DMX Input [706].

Midi Show Control



To control a Sequence via Midi Show Control Protocol:

- **Midi Device**
Choose an installed Midi Device from the drop down list

- **ID**
Enter the ID of MSC Message Mapping (= Device ID - you find it in front of the Client name in the Device Tree [173])

- **Sequence**
Enter the ID of the Sequence to be controlled via MSC

- **Run Midi**
Press Run Midi to launch MSC and MIDI Input

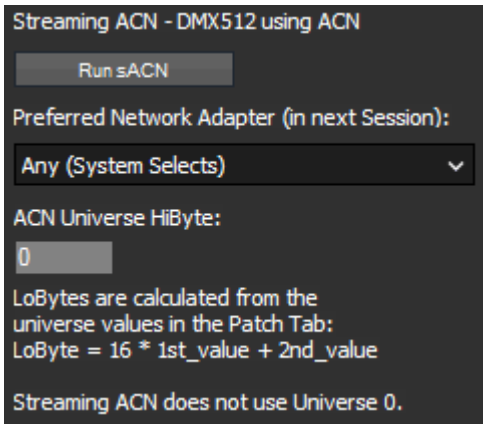
- **Use Cue Subsection**
As decimal places for cue numbers are not supported in Pandoras Box, you can use the check box option "Use Cue Subsection". Doing this will multiply the incoming cue command x1000 (for example: GO to Cue 1.020 will be interpreted as GO to Cue 1020). With this option three decimal places are supported.

- **Cue ID 0 Handling**
Choose between Regular Cue, Ignore, Stop and Stop & Reset all for Cue ID 0 Handling.

The following message types are supported:
Midi: Channel Voice Messages (all Note-On messages are processed)
MSC: System Exclusive messages (SysEx) according to the MSC specification

Please see the following links for detailed information: Midi [716] / Midi Show Control [716].

Streaming ACN

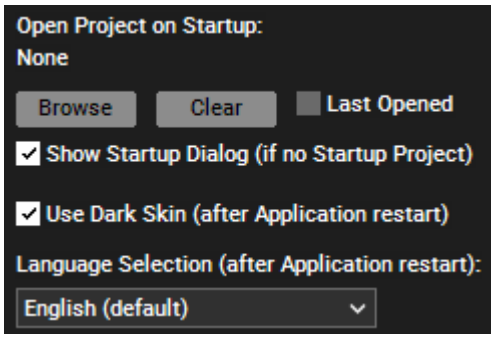


If you want to control Pandoras Box Master via Streaming ACN (sACN), please prepare the patch [228] as usual. For more info please read the topic about DMX Input [706].
To run sACN press **"Run sACN"**.
Please note that Streaming ACN does not use Universe 0, so do not patch below Universe 1.

- **Preferred Network Adapter (In Next Session)**
The drop-down menu contains all network cards installed on the system. Select one and restart the application. Now the entire Pandoras Box communication regarding Streaming ACN will be transferred by this defined adapter and its IP address. With choosing a different network adapter for other protocols like for the Pandoras Box show data, you can separate different communication levels from each other.
These protocols can be assigned separately: Pandoras Box show data [147], DMX / Art-Net Mode [148], CITP Thumbnail Exchange [148] and Streaming ACN.
If an explicitly named network adapter is not found, the "any" adapter will be used. "Any" is to be decided by the operating system Windows and its internal adapter order.

6.3.4.6.7 Startup

The section "Startup" in the Local category [139] from the Configuration tab enables you to set up the look and properties when a project is loaded.



- Open Project on Startup:
Here you can tell the system to load the last opened project or a specific project file:
- Press "**Browse**" to choose a specific project file
- Press "**Clear**" to reset the load option
- Tick the check box to always open the last opened project file

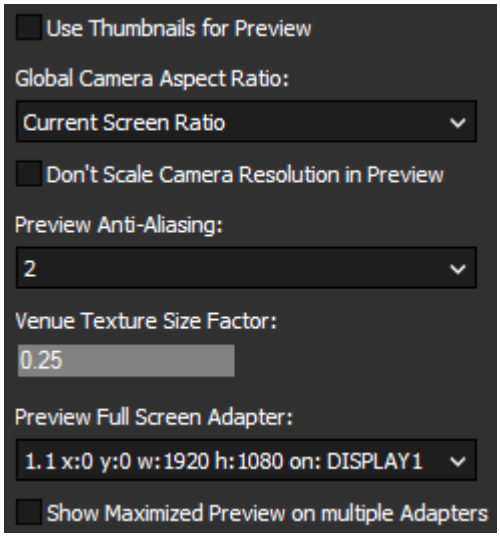
- Show Startup Dialog
This de-/activates the Startup dialog [118] that pops up as soon as Pandoras Box is loaded. If you have chosen to load a certain project on startup (via the above option), the Startup dialog is never shown.

- Use Dark Skin
If the check box is activated, the user interface is rendered in the so called Dark Skin. If the check box is not ticked, the interface looks light gray. This option takes effect when Pandoras Box loads, please restart your project (saving an empty project is not mandatory) if you like to see its effect right away.

- Language Selection
Choose a language for the user interface of the Pandoras Box master software. This option takes effect when Pandoras Box loads, please restart your project (saving an empty project is not mandatory) if you like to see its effect right away.

6.3.4.6.8 Local Preview

The section "Local Preview " in the Local category [139] from the Configuration tab enables you to set up properties of the local Preview [243].



- Use Thumbnails for Preview
Using this option will show thumbnails instead a full preview of the file. This saves graphics card power on the Master system.

- Global Camera Aspect Ratio
The Aspect Ratio of the Global Camera can be set to one of the options in the pull down list: Current Screen Ratio, 5:4, 4:3, 3:2, 16:10, 5:3, 16:9 or 17:9.
This setting refers only to the size of your Preview area in the tab. If you like to change the aspect ratio from your output, please refer to the Camera Inspector [218] or Output Inspector [220]; or change the resolution in your graphics card driver [1948].

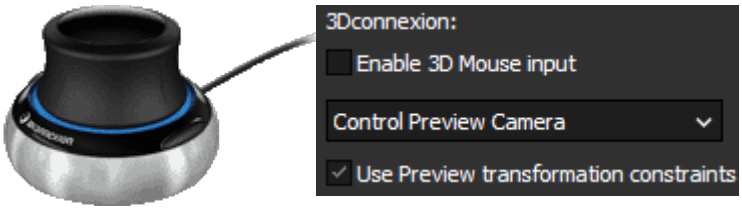
- Don't Scale Camera Resolution in Preview
This is of interest when working with Matrix Patches [2077] on a Master with Output Layers. In case you are not rendering fullscreen but in the Preview tab, the Art-Net data would be based on a scaled image. If you need to see correct Art-Net pixels even if the preview is toggled into the user interface, tick the check box. The checked option saves graphics card memory on the Master system. This is especially helpful when previewing many Clients e.g. working with a Venue Site.

- Preview Anti-Aliasing
The Anti-Alias level for the preview can be set to the value 0, 2 or 4, 4 being the maximum. For memory and performance enhancement the default value is set to 2.

- Venue Texture Size Factor
The main purpose of a Venue Site [694] is to render content from other Clients, which means that their Camera and Output textures are routed to the Venue Site. Therefore, all those clients must be toggled into the Preview. For larger projects you might want to visualize many outputs which requires the PB Master to render many textures. To save performance, the routed textures are scaled down. Per default the "Venue Texture Size Factor" is set to 0.25 . Increase it for higher quality and decrease it for more performance. You can also change individual textures in the Camera Inspector [218] or Output Inspector [220].

- **Preview Full Screen Adapter**
Activate the option **"Show Maximized Preview on multiple Adapters"** if multiple displays are connected to the master system and you like see the maximized Preview on all of them. If you select a Client's Camera or Output in the Preview and then maximize it, your first display will show the first Camera or Output and the second display the second one, etc. You can choose the first Adapter with the drop-down list.
Please note that the maximized Preview [Ctrl + Shift + F] is not the same as the Fullscreen mode [Ctrl + F]. As described in the Preview chapter [243], the maximized Preview still allows to navigate and edit as all tools, modes and view options are available within the Button Bar (shortcut [T] to show and hide it). The Fullscreen mode is not editable and only available if the Master offers an Output Layer. Settings regarding the Fullscreen mode can be found in the section "Render Engine [162]" in the Configuration.

6.3.4.6.9 3D Hardware



Any Pandoras Box Master versions support the product SpaceNavigator from 3Dconnexion. In short, this is a mouse device that allows to navigate in three axes (position and rotation). Make sure to install its driver.

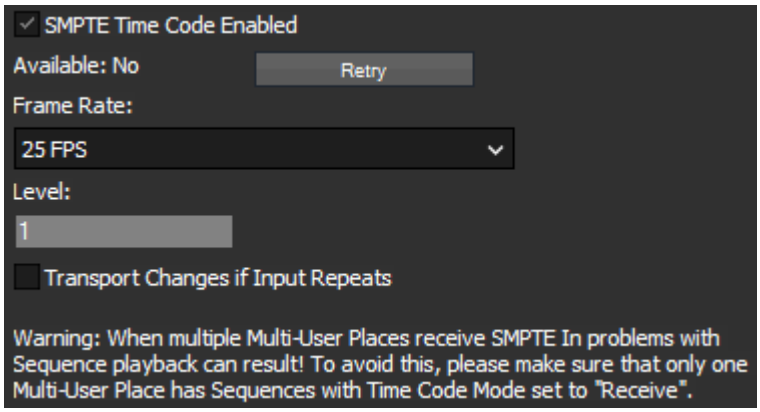
The check box **"Enable 3D Mouse Input"** en- and disables the communication with the mouse device.

The below drop-down list offers two options. With **"Control Preview Camera"** you can control the camera selected in the Preview tab [243] and with "Control selected Layers" you control this or those layer(s) that are currently selected in the Device Tree [173]. Note that the mouse input alters the parameters relatively, not absolutely.

The check box **"Use Preview transformation constraints"** refers to the constraints along the X-,Y- and / or Z-axis set up in the Preview tab. This is explained in more detail in the chapter Layer Mode and Camera Transformation Mode [251].

6.3.4.6.10 SMPTE Time Code

The section "SMPTE Time Code" in the Local category [139] from the Configuration tab allows to set up timecode options for an attached SMPTE Link interface [2000].



If you want to use a SMPTE Link interface for either SMPTE send or receive mode, please connect the SMPTE Link USB device and make sure the check box to enable the SMPTE Time Code is ticked. The availability will then switch to "YES". If not, Pandoras Box cannot find the SMPTE Link. Try to reconnect it to the hardware, restart Pandoras Box or re-install the driver found in the Download Center our web site. To set a timeline to "Send" or "Receive" please see the Sequence Inspector [204].

- **Retry**
Use the "Retry" button if you have lost communication or the SMPTE Link was removed from the USB port.

- **Frame Rate**
Setup the frame rate that you want to use for either send or receive mode of the SMPTE Link interface.

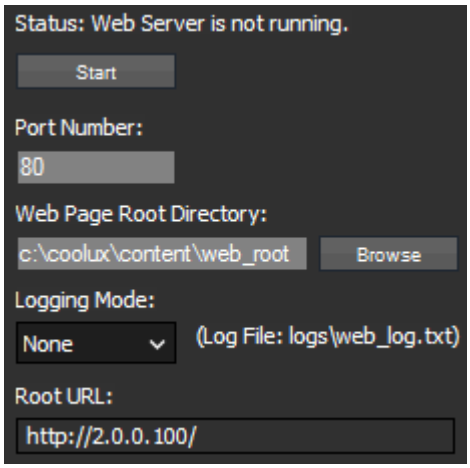
- **Level**
The level setting is important for send mode only. You may enter values between 0 and 1 to amplify the SMPTE output signal level.

- **Transport Changes if Input Repeats**
Some SMPTE sending devices continue sending the last frame instead of stopping the SMPTE transmission.
If this option is not enabled, and the SMPTE devices sends 3x an identical frame, the timeline will continue running until it receives a different frame.
If this option is enabled, the timeline will execute the Stop-Action set up for this sequence.

As written in the tab itself, please note the following when working in a Multi-User [234] environment. When multiple Multi-User Places receive SMPTE Time Code, problems with Sequence playback can result. To avoid this, please make sure that only one Multi-User Place has Sequences with the Time Code Mode set to "Receive" (in the Sequence Inspector) [204].

6.3.4.6.11 Web Server

The section "Web Browser" in the Local category [139] from the Configuration tab allows to set up the integrated web server in Pandoras Box. See here a list with all ports [781] used by Pandoras Box and Widget Designer.



The principle of Pandoras Box' Web Server

Version 5.5 comes with the newly developed Pandoras Box Web Server. It offers remote control of a Pandoras Box project that can be implemented into a web site. All PB Automation commands, known from the SDK [730], are now executable in JavaScript language [734], and thus can be embedded in the HTML code by your web developer.

There is a demo web site that shows what can be done and how the PBAutomation commands are implemented in the web site's code. If you like to view it, open a Pandoras Box Master system and activate the web server. Then go to your browser and enter your IP address and the port number, e.g. `http://2.0.0.100:6214` (alternatively you may copy-paste the URL from the Configuration tab) to - this calls the web server and shows the delivered web site. This site contains a link to the demo site which is installed per default in the web root folder under `C:\Christie\content` or `C:\coolux\content`. If you like to write your own site please make sure to place it in this web page root directory.

Please note that the internal Pandoras Box Web Server can be used without access to the internet, it is meant to be used in (secure) local networks. For bigger installations it might be of interest that external web servers can be used in combination with the PB web server. For example, an Apache HTTP Server can be used or any other web server that supports the scripting language php. This server is then used to deliver the web sites to the browser and forward commands to or receive information from the PB web server. This setup has advantages for larger installations and for those that need access to the internet (as the PB server can be invisible), or for database applications.

Settings in the Configuration tab

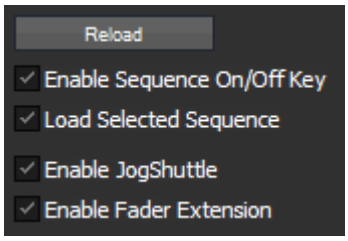
Click the **Start** button to activate the web server which acts then as an interface. You may send commands to the web server, it "translates" them and forwards them to Pandoras Box; in return Pandoras Box sends requested information through the Web Server to you.

Per default the **port** number used by the web server is 80, it can be changed if needed. As well the **directory** and **URL** for the web page root can be altered. There is a log file that logs when a command was send or received, an error or another action happened. The log entries can be filtered; choose your **logging mode** (None, Limited or Full) in the drop-down list.

6.3.4.6.12 Controller Setup

The section "SMPTE Time Code" in the Local category [139] from the Configuration tab allows to assign the Sequence faders and Cue buttons for the Christie Jog/Shuttle [1991] and Fader Extension [1993]. Without assigning them here, the controller boards will not work.

General



- Reload
Press this button if the connected controller board is not shown in the tab. This will reload the connection.

- Enable Sequence On/Off Key
This option is valid for the former Jog/Shuttle Controllers [1994]. The Sequence On/Off Key is enabled by default. This button will toggle the opacity of the selected sequence between 0% and 100%. This option may be disabled so that the sequence cannot be turned off by fault.

- Load Selected Sequence
When this option is enabled, the sequence selected by the sequence faders will be active and loaded into the Sequence Tab [292].

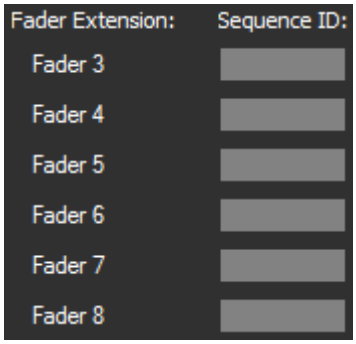
- Enable Jog/Shuttle / Fader Extension
Both check boxes en- or disable the communication with the connected controller boards.

Jog/Shuttle

- Sequence ID
The Jog/Shuttle Board allows controlling two sequences. To assign a sequence in Pandoras Box to a Sequence fader and Select button on the Jog/Shuttle board, please enter the sequence ID into the text field next to Fader 1 or Fader 2.

- Cue IDs
For each sequence that is linked to the faders, 16 specific cues [292] may be called via the Cue buttons. To assign them:
- Select the Sequence whose cues you want to link to the Cue buttons by clicking on Fader 1 or Fader 2 in the Jog/Shuttle section.
- The title of the Cue IDs changes to the Fader number and the Sequence ID linked to this fader.
- Enter a Cue ID for a Cue button.

Fader Extension

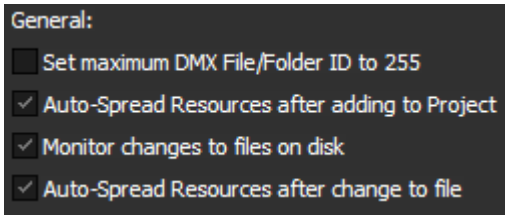


If there is a Fader Extension Board [1993] attached to the Pandoras Box Master as well, six additional faders and Sequence IDs are shown. Assigning sequences to faders and cues to buttons works in the same way as explained above.

6.3.4.6.13 Resources (Global)

The section Global "Resources" in the Global category [139] from the Configuration tab enables you to set up the general settings for resources to improve your workflow. Note that Global Resource settings influence also remote instances. In a Multi-User environment they are shared with other Places. User Resource settings [140] influence only the local instance.

General



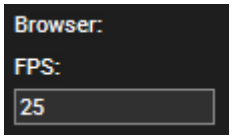
- **Set maximum DMX File / Folder ID to 255**
When checked, the File ID and Folder ID [191] can reach a maximum of 255. The next item would start with 1 (instead of 256) and a consecutive Folder ID. This option is especially useful when working with a lighting desk and / or exchanging thumbnails via CITP [148].

- **Auto Spread Resources after adding to Project**
All media files added to the project will automatically be spread to all available Clients (by default). To determine the time of spreading manually, uncheck this option.

- **Monitor changes to files on disk**
This applies when a file that is part of your Pandoras Box project is saved by another program under the same name, i.e has changed. In versions before 5.5 it was necessary to reload the file manually or to reload the entire PB project to load the new state of the file. Now, you can decide to monitor the hard disk and reload a file automatically as soon as it has changed. If you like to spread it too, please activate the next check box as well " Auto Spread Resources after change to file".
Please note that this only applies to the system from where the file was originally added to the project. If you spread a file and change it on the remote node, you will not see the updated file as the remote disk is not the source disk and was not monitored.

- **Auto-Spread Resources after change to file**
If a file change is monitored (see check box "Monitor changes to files on disk") it will be spread to available nodes too.

Browser



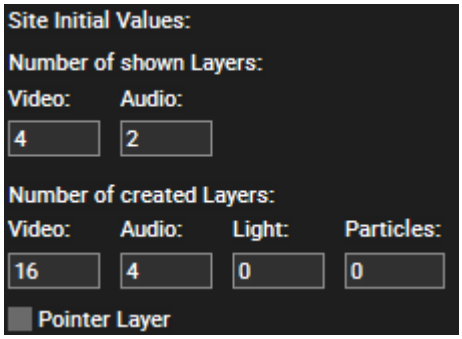
This option applies to Browser assets [279]. More settings are offered in the Browser Inspector [198]. The default URL can be adjusted under Resources (User) [140]

- **FPS**
Enter the frame rate a newly created Browser asset should be rendered with.

6.3.4.6.14 Devices / Parameters

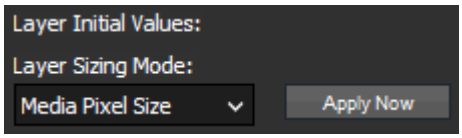
The section "Devices / Parameters" in the Global category [139] from the Configuration tab enables you to set up the general appearance and behavior of devices and parameters in your project.

Site Initial Values



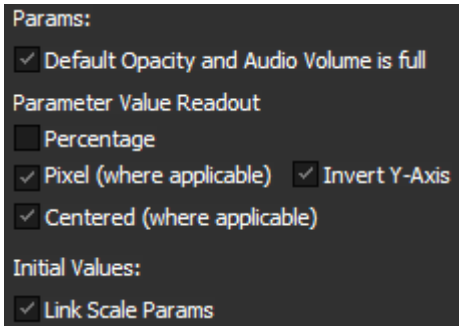
- Number of shown Layers
- Number of created Layers
- Show Pointer Layer
Enter the amount of layers [318] that should appear when a new site is added to the Devices tab. Choose whether it shows the Pointer Layer or not.
All other created, but not shown layers are hidden to ensure a good overview. To unhide or add additional layers to an existing site, right-click on the site. (see "Site context menu" [178].)

Layer Initial Values



- Layer Sizing Mode
Choose the Layer Sizing Mode [211] that should be applied for newly added layers, e.g. whether they should be stretched fullscreen or keep their original resolution.
Click "Apply now" if already existing layers should adopt that mode.

Params



- Default Opacity and Audio Volume is full
When checked, the default value for Opacity is 255 and for Volume it is 0dB, i.e. both are at 100%. This means that as soon as a media file is assigned to a layer, it will be visible or hear-able. When a container is created, and no key for Opacity / Volume is generated, the default value takes place.
In Versions before 5.5 the default value has been 0. If you like to return to that behavior you may uncheck this option.
Please note that this option results in a different show appearance when switched during or after show programming, thus it is recommended to test the show or check this option only once before starting to program.

- Parameter Value Readout
-- Percentage
Use the check box to enable percentage readout for all values of all parameters in the project, see figure b) and c). This setting can be changed at any time during runtime.

-- Pixel (where applicable)
This option is available, if the Unit Translation Mode [160] is set to "Use Fixed Relationship".
Some parameters (like Position and most Camera parameters) are then based on pixel values. This makes it possible to enter an exact pixel number, e.g. move right by 50 pixels. With a disabled check box, those parameters are based on generic units as explained here: Configuration > Unit Management [160].
In Pandoras Box, the positive X-axis points to the right direction and the Y-axis to the top. If you like to have the Y-axis pointing down, make sure, **"Invert Y-Axis"** is enabled. Both options can also be found in the section "Unit Management".

-- Centered (where applicable): Some of the parameter values appear in centered position by default (default value is here set to 0), irrespective if the readout is decimal or percentage, e.g. the following parameters: X/Y/Z Rotation Speed of a Layer or Keystone faders of an Output. If you disable this function, 0 will be the minimum and the 8 bit /16 bit decimal value (depending on the properties of each parameter) will be the maximum value. With percentage value readout the value range goes from 0 to 100%. Please note that the value parameter or the layer appearance is not changed, it is just a way of displaying the value.

☐ Percentage
☑ Centered

☑ Percentage
☑ Centered

☑ Percentage
☐ Centered

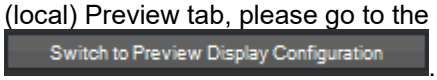
☐ Percentage
☐ Centered

- Initial Values (for Parameter)
Check the option "Link Scale Params" if all newly added layers should have a locked aspect ratio. If you now change the X Scale, the Y and Z scale follow.
For existing layers, you can link parameters by using the chain icon underneath the parameter fields, see Changing Parameter Values (Device Control) 171 and Changing Parameter Values (Device Tree) 177.

6.3.4.6.15 Client Display

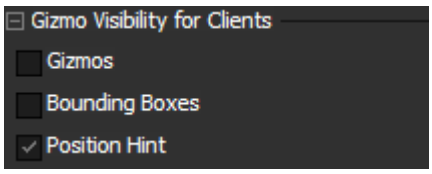
The section "Client Display" in the Global category [139] from the Configuration tab allows to influence what is rendered in the outputs of your Client (e.g. a Server). You can for example change the rendering of the Ground Grid, Gizmo and untextured objects.

Note that all settings here are for the (remote) Client Display only. To change the rendering appearance in the (local) Preview tab, please go to the section "Preview Display [145]" or click the according button



.

Gizmo Visibility for Clients



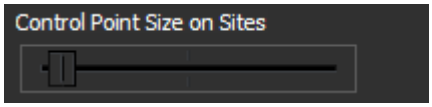
Choose to display or hide the following objects.

"**Gizmo**" describes the icon with three axes in the colors red, green and blue. In the Output only Layers have a Gizmo.

"**Bounding Box**" describes the blue box rendered around an object indicating its size. As long as you do not assign a three-dimensional object but work with flat 2D images or videos, the bounding box looks like a blue rectangle.

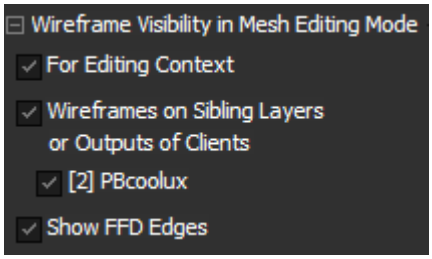
The "**Position Hint**" is visible when the Preview is toggled to the Mesh Editing Mode. As you move the mouse cursor in the local Preview tab, but look at the Mesh points in the Client Display the Position Hint helps to identify where the cursor is as it renders a circle around the according position within the Client Display.

Control Point Size on Sites



The fader influences the size of an FFD control point  , e.g when working in the Mesh Editing Mode [257].

Wireframe Visibility in Mesh Editing Mode



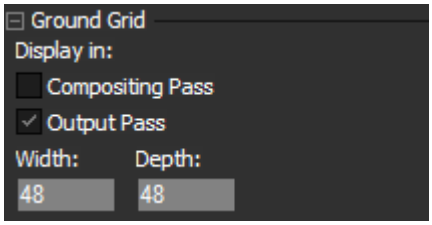
These settings refer to the rendering of Mesh and FFD grids when the Preview is toggled to the Mesh Editing Mode [257].

The check box "**For Editing Context**" toggles the visibility of the Mesh grid (not the points itself!) for the Layer that is chosen in the "Editing Context" drop-down list, i.e. the Layer that is currently being warped and which is depicted with the editing icon in the Device Tree: 

The next check box "**Wireframes on Sibling Layers or Outputs of Clients**" toggles the Mesh grid on all other Layers and Outputs. These means that you can toggle the Mesh from the Editing Context Layer and all other ones separately. You can toggle each Site by its own which is helpful if you like to align a Mesh to another one, e.g. within a softedge area where two Meshes overlap each other. The Mesh appearance, e.g. its color, can be changed in the Object Inspector [200].

"**Show FFD Edges**" renders the lines between FFD points.

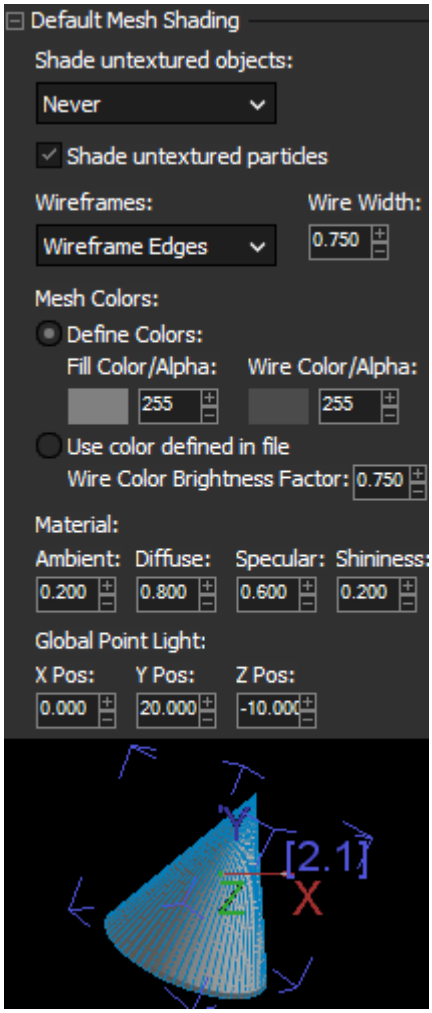
Ground Grid



- Display in...
Choose whether you like to see the Ground Grid in the Compositing and / or Output Pass which are described in the chapter Video Processing Pipeline [320].

- Width / Height
With the number fields you can influence the size of the Ground Grid.

Mesh Shading and Color



Please note that the Inspector for an object [200] allows to overwrite the below settings with individual ones.
The left bottom example shows an untextured object with gray material color and blue wireframe edges.

- **Shade untextured object**
With an activated option, even objects without an assigned texture are visible as they can be shaded according to the below settings e.g. colored and with visible wireframes. The default for the Output is set to "Never".

- **Shade untextured particles**
If you assign an object but no texture (i.e. media file) to a Particle System [185], it can still be shaded according to the Particle Emitter Color [188] if this check box is ticked. If you disable this option, particles are only visible with an assigned texture.

- **Wireframes**
Choose whether an untextured object should be rendered without Wireframes, with Wireframe Edges or Wireframe Triangles.

- **Width**
Define the thickness of above mentioned Wireframes.

- **Mesh Colors**
Adjust the color of untextured objects.
--**Define Colors**
Click on the color fields to define a color that is used for the Fill Color and Wireframes. Use the number fields to adjust the transparency.
--**Use in file defined color**
If the object was originally saved with an assigned fill and wireframe color, Pandoras Box can use them also. Use the number field to adjust the wireframe's brightness.

- **Material**
Adjust the Ambient, Diffuse, Specular reflection levels and the shininess that influence how an untextured object reflects the global light, e.g. to appear rather glossy than dull. Further information... [527]

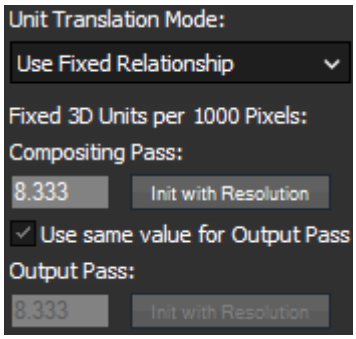
- **Global Point Light**
You may adjust the X-, Y- and Z-position of a global light that is reflected from untextured objects according to the "Material" settings.

6.3.4.6.16 Unit Management

The section "Unit Management" in the Global category [139] from the Configuration tab enables you to set up on which units Pandoras Box 3D space is based on. Since version 6, the pixel-oriented workflow is the new default and supersedes working with generic units. Of course, you can still choose between both workflows. When you create a new project, the Startup Dialog [118] offers the check box "Pixel-oriented workflow" which does the same as using the Configuration settings described below.

Pandoras Box allows you to work in a 3D space based on so called 3D units or generic units (GU).
The result you see on your display is a rendering of the 3D space done in two steps, so called render passes. First a 3D compositing and then a 3D output space is created. The chapter "Video Processing Pipeline" [320] describes the Composition and Output Pass in more detail.
The final render output is based on pixel units. Depending on the resolution set up in the Display driver [1948], the pixel width and height changes. Now, there are different ways how to translate a 3D unit to a pixel and Pandoras Box offers two options in the drop-down list "Unit Translation Mode".

Unit Translation Mode

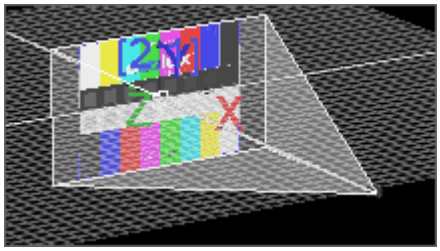


When positioning and sizing elements based on pixel values it is helpful to ensure that the entire system globally uses only one fixed relationship for translating between 3D units and pixels. This is for example of interest when displays with different resolution form one "pixel space", one large screen. Or, when you like to work with pixel values in general e.g. to be able to shift the layer exactly 512px. This form of unit translation is the default, the drop-down list "Unit Translation Mode" is set to **"Use Fixed Relationship"**. In this mode, a distance in pixels always results in the same distance in 3D units independent of the render pass resolution. If you need to change the translation mode in an existing project or the mapping, please follow the next steps. After selecting the fixed relationship, enter how many 3D units should be mapped on 1000 pixel. Per default the Output Pass uses the same translation factor and should be changed only if needed. If your Clients are already connected to the Master system, you may click the "Init with Resolution" button. This opens a dialog that lists all output resolutions from the Clients. Choose one resolution and Pandoras calculates the translation factor automatically. As a result, the cameras of each site will adopt to new default values. Whilst the Z-position stays at -25units, the FOV (field of view) changes to a value according to your translation factor. To ensure consistent translation of units, leaving the cameras in the default state is strongly recommended!

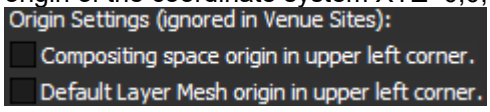
When choosing the option **"Map 16 3D Units to each Resolution"**, Pandoras Box sizes the 3D spaces so that 16 3D units exactly match the pixel width of each render pass. This is achieved by applying specific parameters to the camera; the XYZ-position equals 0,0,-25 and the opening angle, the field of view, is 35.489°.
To give you an example, no matter whether your display / render pass resolution is 1024px or 1920px, moving a layer with 8 units has the same result in both displays. If the layer's center was in the middle of the screen, it will now be at the monitor's edge.
In other words, different render passes may map same 3D unit distances to different pixel distances. In the example, 16 units were mapped to the 1024px as well as the 1920px output, hence 8 units is half the width of both monitors.
16units / 1024px = 0.15625 units / px = 15.625 units / 1000 px
16units / 1920px = 0.83333 units / px = 8.333 units / 1000 px

Origin Settings

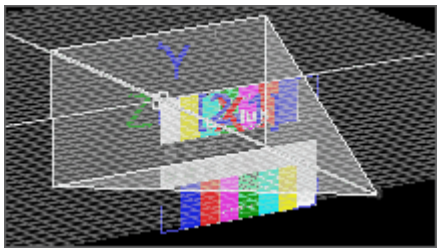
"Origin Settings" are further options that become available when working with a fixed relationship. Layers and other devices (Camera, Output) can be positioned differently in regards to the origin of the coordinate system.



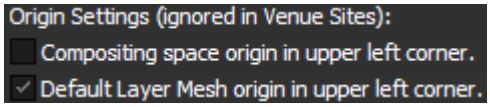
In the left image you see that the layer's and the camera's center match the origin of the coordinate system XYZ=0,0,0.

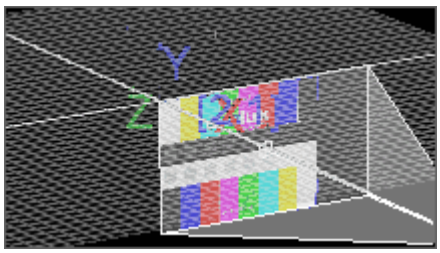


This is the default in both Unit Translation Modes.

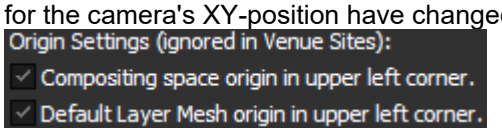


In this example the "Compositing space" is left in the center (check box not ticked) but the "Default Layer Mesh" is activated so that the 0,0,0 origin is in the layer's upper left corner.

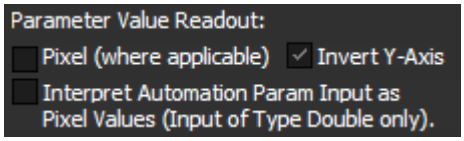




At last, in this example the upper left corner of the "Compositing Space" and the "Default Layer Mesh" match the 0,0,0 origin. Note that the default values for the camera's XY-position have changed.



Parameter Value Readout

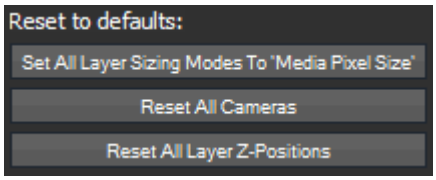


-- **Pixel (where applicable)**
This option is available, if the Unit Translation Mode [160] is set to "Use Fixed Relationship" and activated per default. As some parameters (like Position and most Camera parameters) are then based on pixel values, you can also choose to see and enter exact pixel numbers, e.g. move right by 50 pixels. With a disabled check box, those parameters are based on generic units.

In Pandoras Box, the positive X-axis points to the right direction and the Y-axis to the top. If you like to have the Y-axis pointing down, make sure, **"Invert Y-Axis"** is enabled. Both options can also be found in the section "Devices / Parameters [156]" where you can also choose to see percentage values.

The last option "**Interpret Automation Param Input as Pixel Values**" is of interest when you remote control Pandoras Box via the offered SDK [730] or via Widget Designer [786]. When you send values with the type Double to "pixel" parameters in PB, they can be interpreted directly as pixel values when the check box is activated, meaning that they are not modified (again). When the option is deactivated, PB assumes, the input values are generic units and translates them to pixels according to the translation factor. For example, sending "4.5" results in 540px taking the default translation factor 8.333 GU per 1000px.

Reset to defaults



At last, the buttons under "Reset to defaults" ensure that your workflow is based on correct settings. When changing from one translation mode to another a pop-up dialog informs you that some settings should be adjusted and offer a button do do this automatically. However you can still apply the recommended setup manually using these buttons.

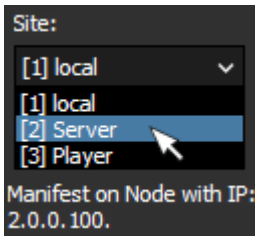
"Set All Layer Sizing Modes To 'Media Pixel Size' " changes the Sizing Mode from a Layer. Please see further information in the Layer Inspector [211].
"Reset All Cameras" resets all (active) parameters from the Camera Layer. As explained above, the translation modes set different default parameters for the cameras. In case you have already changed some parameters, please click this button (and adopt your programming).
"Reset All Layer Z-positions" resets all (active) Z-position parameters from all Video Layers. If you work with "Fixed Relationship" and the pixel-oriented workflow the pixel accuracy is only achieved when leaving the layers on Z-position 0.

All these options are already prepared for you in case you open a new project and click the check-box "Pixel-Oriented Workflow" in the Startup Dialog.

If you change to a generic unit-based workflow, there is only one button changing the Sizing Mode from a Layer (explained here: Layer Inspector) [211]: **"Set All Layer Sizing Modes To 'Fit Larger Side' ".**

6.3.4.6.17 Render Engine

The section "Render Engine " in the Sites category [139] from the Configuration tab enables you to set up properties of the local and remote render outputs.

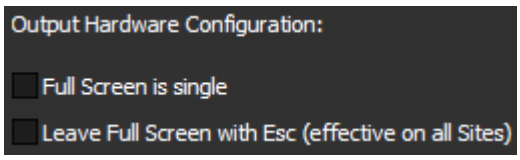


The following settings are Site sensitive, please select the according Pandoras Box system with the "**Site**" drop-down menu.
If the node is found in the network the below information will say "Manifest on Node with IP:..." otherwise it says "Node not available in Network".

These settings are also accessible with the button "Open Engine Configuration"in the Inspector for the site [210].

If your local Site offers Output Devices in the Device Tree, they can be used in Fullscreen mode [Ctrl + F] or for the maximized Preview [Ctrl + Shift + F]. Without Outputs, you cannot render in Fullscreen. As described in the Preview chapter [243], you can navigate in the maximized Preview and edit it. The local hardware setup for the Preview can be found in the section Local Preview [150] whilst this Render Engine section sets up the rendering in Fullscreen Mode.

Output hardware configuration

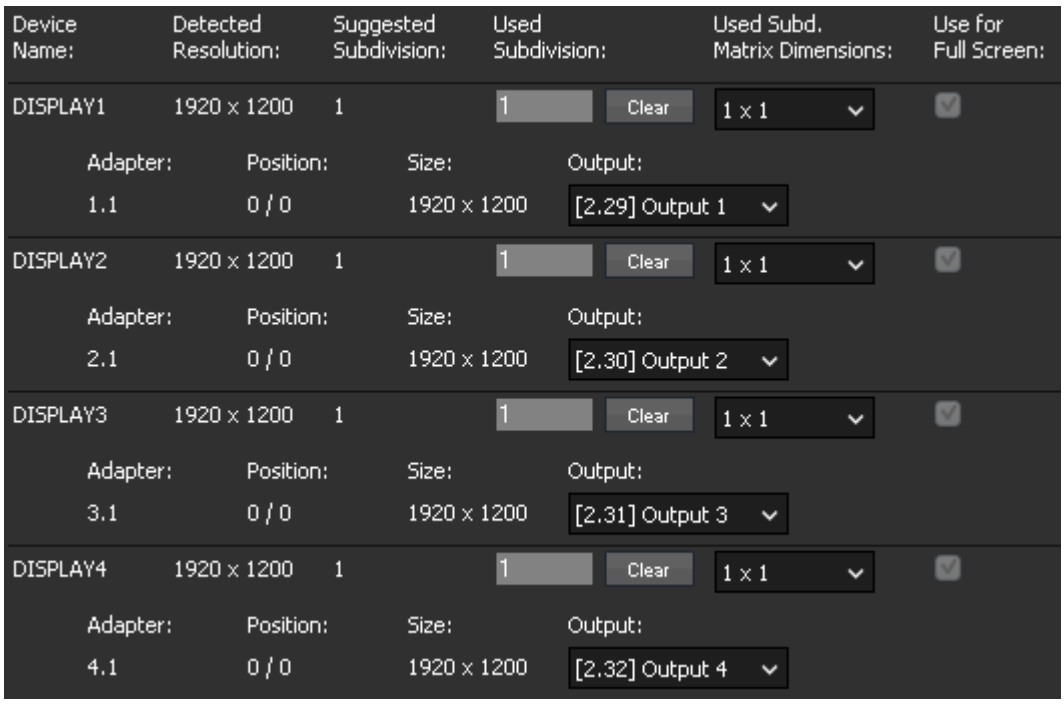


Per default, the shortcut [Ctrl + F] toggles the Preview of your Client (e.g. a Server) into fullscreen. If you like to use the [Esc] key to toggle back to the embedded Preview, tick the option "**Leave Full Screen with Esc (effective on all Sites)**".

There are two configurations for the fullscreen mode, which depend on the settings in the graphics card [1948]. The chapter Multiple displays [1952] explains how to set up your outputs using the NVIDIA Control Panel and explains the difference between using separate outputs and using one Mosaic display as one virtual display spanned across multiple displays. The chapter also includes tips in case you encounter difficulties with setting up the Mosaic mode [1957].
Alternative to the NVIDIA Control Panel, you can click the "Display Setup" button in the PB Menu [2097] and use our "Display Configuration" dialog to setup your Client displays.
In any case, it is recommended to setup the displays before starting the PB Client software so that the Master software does not display wrong information.

If the driver reports separate displays, all outputs can be configured individually regarding the ability to go into full screen. In case one screen should not toggle full screen, check the "Full Screen is single" option and the grayed out check boxes in the column "**Use for Full Screen**" become available to be activated or deactivated. This is of interest when you like to see the user interface of your Master eventhough you are in full screen mode.
In addition, you can route the physical output adapter (meaning the graphics card output) to an Output Layer. "DISPLAY1" means the primary display, "DISPLAY2" the secondary, and so on. This setting is also available in the Output Inspector [220].

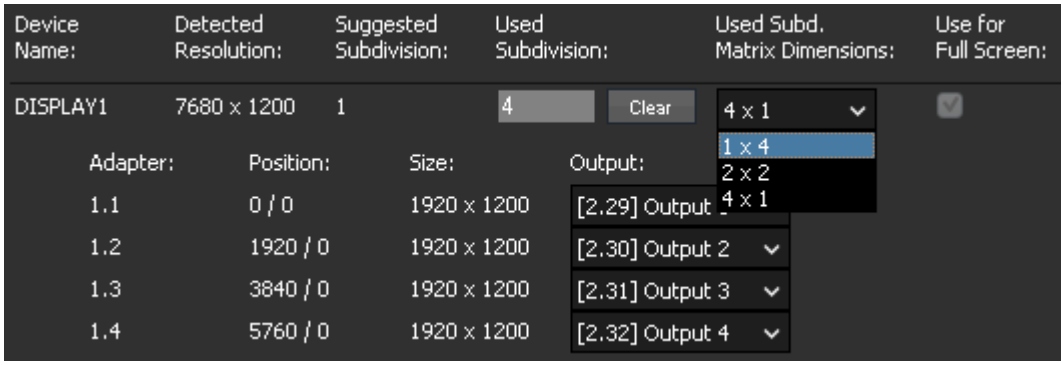
This image shows the hardware configuration from a Server with four outputs. Before, its four displays are setup as four separate displays. Pandoras Box offers automatically to link four separate Output layers to them:



Normally, the settings for Subdivisions do not need to be altered when using separate displays. "**Used Subdivisions**" allows you to divide the output into more areas. After entering "2" for example you could assign another Output layer to render in this area. Hence you would see the results of two output layers in one display whilst the display whereto the Output layer was assigned to before stays black. The drop-down "**Used Subd. Matrix Dimensions**" offers settings to align subdivisions vertically and / or horizontally (see next image). The "**Clear**" button resets the subdivisions to the default setting.

If the driver reports to be in Mosaic mode you have two possibilities. First, you can leave the default option, meaning that Pandoras Box subdivides the available output. In case you have four displays devices, the value for "**Used Subdivision**" is "4" whilst the first one starts at XY-position 0,0 and the other ones with an offset of one subdivision width or height. As a next step you can change the routing of a physical output adapter (meaning the graphics card output) to an Output Layer. This setting is also available in the Output Inspector [220].

This image shows the hardware configuration from a Server with four outputs which were setup as one Mosaic display whose resolution is reported from the graphics card to Pandoras Box. As Pandoras Box recognized that this is four times 1920x1200 it offers automatically to subdivide the large displays and link four separate Output layers to the subdivisions:



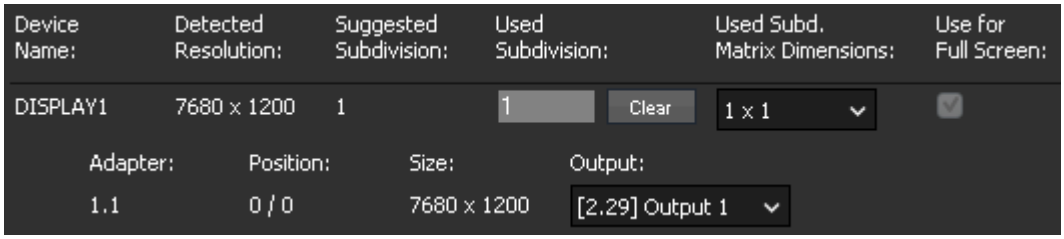
Of course, you can edit the subdivision factor "**Used Subdivision**" manually, too. The drop-down "**Used Subd. Matrix Dimensions**" offers settings to align subdivisions vertically and / or horizontally. The "**Clear**" button resets the subdivisions to the default setting.

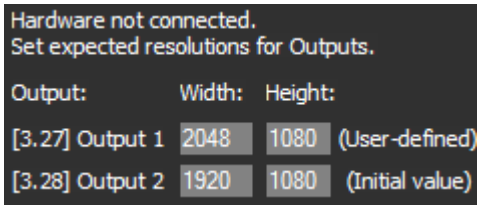
Please note that you cannot take one subdivision out of full screen!

The second possibility for the Mosaic mode is to change the subdivision count to "1". In that case, Pandoras Box handles the entire output as one software adapter. This means that you have one Camera and one Output covering the reported width / height. But as the driver splits the image, you still have an image on all graphics card

outputs.
This can be an advantage as you work with less parameters and options. The second Camera and Output have no effect and can be hidden [178] or removed from the timeline. On the other hand this also means that you loose the flexibility of having more parameters. If you like set up one display differently e.g. less opacity, softedge on another side, etc. you need both Output Layers, hence 2 or more subdivisions.

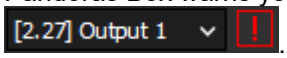
This image shows the hardware configuration from the same Server with four outputs joined in Mosaic mode. But this time the subdivision factor is set to "1" which links all four displays to a single Pandoras Box Output layer:



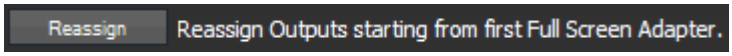


If the Client is not available in the network, e.g. because
- it is not turned on
- it is assigned with no or a wrong IP address
- the PB Client software is not started
- the PB Client software is with a wrong version

no hardware configuration can be read from the driver or the operating system. If you like to pre-program your show, you can enter the resolution you will later setup in the Client's graphics card and the Camera and Output will adopt to it. Note that this is now saved as a User-defined value. The resolution also influences some "Layer Sizing Modes" [211]. If this resolution changes in the real setup later, Pandoras Box warns you by displaying a red exclamation mark icon in this Configuration tab



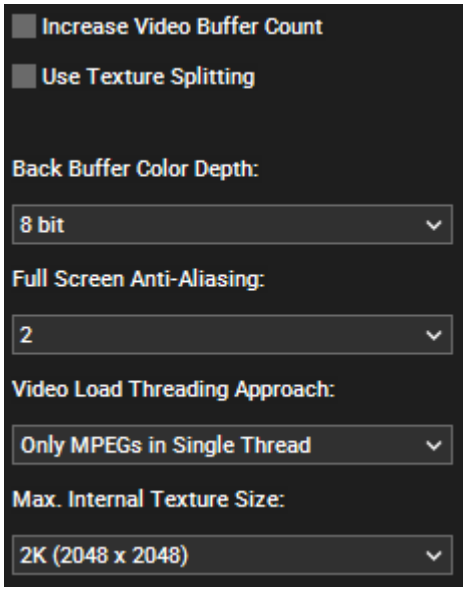
. Click on it to see a pop-up showing the user-defined resolution. Check whether this effects your programming and adjust it if needed.



The "Reassign" button routes the physical output adapter (meaning the graphics card outputs) in the same order as found displays or subdivision are listed. This is of interest if you have changed the routing and want to go back to the default setting.

Rendering Performance Settings

Again, these settings depend on the Site chosen in the drop-down list at the top.



- Increase Video Buffer Count
When using FluidFrame [191] with high-framerate content (50p, 60p files or live inputs), please check this option to ensure smooth playback. To increase the playback performance, more system and GPU RAM are used. Hence, the option should not be checked if not needed.

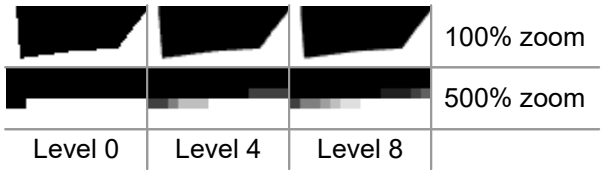
- Use Texture Splitting
"Texture Splitting" can increase the so called show performance when you are working...:
- on our Player R4 and Server R4 hardware
- with uncompressed formats
- with high resolutions above HD

As the name suggest, this option splits textures internally which has the effect that the system handles smaller (but more) textures instead of one very large one. As a result, the loading process of large textures gets smoother and thus increases the "show performance", which we state on our Performance Sheets found in the Download-Center. The show performance defines the maximum number of videos which can be started and stopped without interfering other videos.

- Back Buffer Color Depth
A color depth of 8bit per channel results in 16.7 million colors, whilst a higher color depth of 10bit per channel allows for a total of 1 billion colors. Use this option only, if you are using 10bit content (HDR etc.) and if your

outputs (displays, projectors etc.) are capable to display 10bit as well! All other Clients should be set to 8bit to safe performance.
The increased image quality is seen in higher contrast and cleaner color transitions (no artifacts or color banding) especially when using uncompressed formats.
Please make sure that the graphics card supports the higher color depth as well. For Nvidia cards, the setting can be found in the Nvidia Control Panel > Change Resolution > Output Color Depth drop-down list.

- **Full Screen Anti-Aliasing**
This refers to a anti-aliasing method used for smoothing object edges in the last rendering pass. Per default this is the Output Pass, thus the anti-aliasing works on the output mesh. In case you have set the Output's parameter "State" to Bypass [682], the anti-aliasing applies to the Compositing Pass meaning objects on layers. The higher the anti-aliasing value, the more memory and performance is drawn but the smoother object edges are rendered (as seen in the below example). The maximum anti-aliasing level depends on the graphics card. For performance enhancement the default value is set to 2.



- Video Load Threading Approach
This is of interest when you have programmed multiple containers with video content all starting at the same time. Under circumstances (number of containers, resolution of videos, etc.) this can lead to different render issues as all videos start to be decoded at the same time. With the default option "**Only MPEGs in Single Thread**" MPEG videos start one after another enhancing the robustness of the render engine. The option "**Single thread**" does the same with all videos, not only MPEG ones. A drawback is that delay is added. If you need to sync many videos, choose "**Multiple threads**" from the drop-down list and make sure that your programming does not affect the robustness of the render engine.

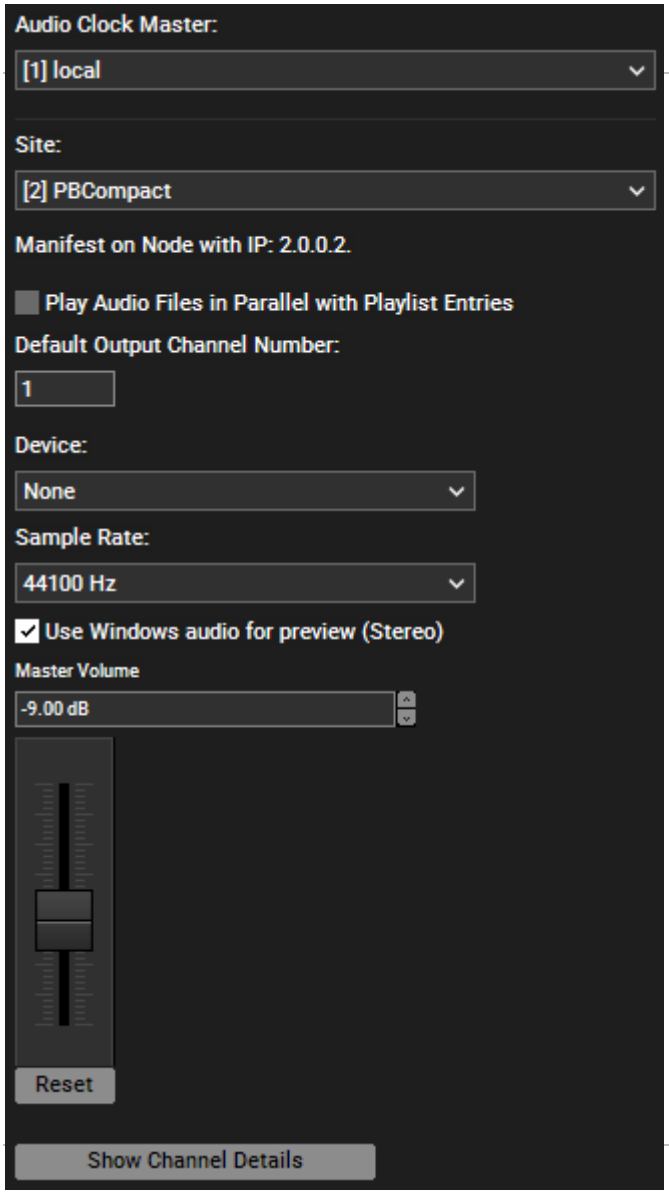
- **Max. Internal Texture Size**
During the rendering process content is buffered onto a so called internal texture. The internal texture is only called when the content is used with
- deinterlacing
- FluidFrame (= frameblending)
- particular effect combination, e.g with Blur. The description of the effects in question mention whether they depend on the internal texture.

Depending on the resolution of the content you are using with those optional features, you might need to select another entry from the drop-down list. Please note that EVERY content file is then buffered with this size. As this consumes plenty of graphics card memory it is recommended to choose the entry next in size and not simply the largest available.

6.3.4.6.18 ASIO-Audio

The ASIO Audio section in the Sites category [139] from the Configuration tab allows choosing an audio clock master and offers ASIO audio settings for all Sites (local and remote).

ASIO sound cards are used by dedicated Audio tracks [661] in the PB Master software. Audio tracks allow playing back ASIO WAV files (mono and stereo PCM Wave files in 16, 24 or 32bit) and audio inputs from optional input boards while being synchronized to a master clock.
If you are interested in sound cards available for Pandoras Box, please check the chapter "Input and Output Cards" [1947].



Audio Clock Master / Audio and Video Synchronization

Open the drop-down list and choose which system should be the "**Audio Clock Master**". The Main Master or any Client with an ASIO device can be chosen. Sub Masters (see Multi-User [234]) cannot be clock masters, but only slaves. The drop-down entries marked with "Slave only" cannot be chosen.

We highly recommend to bind the Audio Clock Master only to a system with a dedicated ASIO device including a proper clock! This explicitly excludes simple ASIO drivers.

Note that there can be only ONE master clock in the system. The ASIO clock will be the system's master clock and Timecode master. That means that all Video Layers within the entire PB network will be synced to it. Further, ASIO and SMPTE cannot be used simultaneously. If no "Audio Clock Master" is chosen, i.e. the drop-down is set to "None", the Main Master is automatically the clock master even though this clock is not an ASIO clock.

Technical information: It is possible to output ASIO on multiple systems but it is not possible to synchronize the audio on more than one system while playing back. This means, if you output ASIO on another system (than the one that generates the master clock), those Audio Tracks cannot be synched while playing back. They will be triggered in sync but as soon they are running, they are in a free-run mode. This sounds more alarming than it is, because audio files are still clocked to the device's clock which is extremely precise if you have chosen a dedicated ASIO device with a proper clock.

Site and Device Settings

Choose with the "**Site**" drop-down menu a local or remote Pandoras Box system for which you like to see the ASIO settings. Once chosen, the information on which IP address the site is manifested is displayed and whether the "**Clock State**" is "Master" or "Slave".
Open the below "**Device**" drop-down and assign an available ASIO device. Since PB version 8.0, the list also offers Dante devices.

Dante
Pandoras Box Licenses are natively supporting 32 input channels and 32 output channels without the need of any hardware, driver configuration or licensing.

If you like to use Dante to receive or transmit digital audio over IP, choose "PB Dante" from the "Devices" list and click the "Configure" button. In the dialog, choose which "Network Adapter" should be used for Dante and check the number of input and output channels. Lastly, check the "Sample Rate" as explained below.

To output Dante, assign an audio file [92] or live source to an Audio Track [661] and check the channel routing in the Track Inspector [217].
To input Dante streams, go to the Assets tab [131] and open the folder "Live Inputs > PB Dante" and choose a Dante stereo input from the PB system with the Dante device and drag it into the Project tab [278]. Per default, you will see 32 inputs (16 stereo channels). If you have entered less channels, only the according inputs will be active. If you have chosen more channels, you might need to refresh the list to see more inputs.
Last, assign an input to an Audio Track [661] from the PB system with the Dante device.

As demanded by Dante, you need a Dante Master in the network used for transmitting Dante. The Dante Master unlocks the Dante functionality per se, generates a clock and syncs the signals. The Dante Master can be any Dante interface or Dante audio card for example. The "Dante Controller" software can be used to assign the Master functionality, route signals, match sample rates for sources and outputs, etc.

For using more than 32+32 channels, we recommend to separate the network traffic from the show network and use a dedicated Dante card. Keep in mind, that each Dante output generates network traffic even though it might not play a file. The Dante card offered by Christie [1986] allows to use 256 channels in total. As said above, it can be a Dante Master.
Another advantage of a Dante audio card is redundancy. As it provides two distinct network connections and features Audinate's patented Glitch Free redundancy, audio continues uninterrupted in the event of a network connectivity issue with the first connection. The card is explained in the chapter "Dante Audio Card" [1986] in more detail.

Sample Rate & Volume

Choose a sample rate from the "**Sample Rate**" drop-down list (e.g. 41 kHz, 44.1, 48 kHz, 96 kHz as supported by the sound card). All sound inputs and sound files within the project should match this sample rate.
Please note that the Audio Tracks will play the assigned files with the sample rate that is preset in the Audio Configuration only, there is no sample rate conversion. Each sample will be played back 1:1.
There are test files with various sample rates and volumes in the Stock Assets folder.

The fader sets the **Master Volume** of your ASIO sound card. Value Range: -96 dB (muted) up to 6 dB (max.). Default value is -9 dB.

Playlist Options

With a selected ASIO device, you can choose the option "**Play Audio Files in Parallel with Playlist Entries**". This automatically plays the "according" audio file with the video in a Playlist [239]. The audio file does not need to be part of the project, but it has to be named in the same way as the video file and needs to be located in the same folder.
If you like to play the files on another ASIO channel than 1, change the "**Default Output Channel Number**" to a channel listed under "Show Channel Details". Note that always two channels are used as the audio output is a stereo signal. Hence, if the default channel is set to 1, channel 2 is automatically used also.

Windows Audio

This feature is of interest if the computer has no dedicated ASIO sound card or driver installed but offers a Windows audio device. It is checked per default as long as no other audio card is chosen. In that case you can route all audio signal (originating from Video Layers and Audio Tracks) to the audio device that is setup as the standard audio device in the Sound card settings of Windows. Please also check the sample rate (e.g. 48 or 96kHz) and bit depth (16bit) there, mostly available in the advanced properties tab. That way, you can "preview" all audio with the Windows audio device e.g. for pre-programming purposes.
When using "Windows Audio" all ASIO channels are mixed down to two channels, all odd ones to the left channel and the even ones to the right channel.
Obviously, you cannot use Windows Audio and ASIO Audio at the same time. As soon as an ASIO device is chosen within the "Device" drop-down, the Windows option is deactivated.
Please also note that a normal sound card does not synchronize audio information in the same way as an ASIO interface and therefore cannot be the Audio Clock Master. For audio playback during a show, an ASIO interface is recommended.

Multi-Channel ASIO Output

The ASIO protocol allows multichannel audio output, depending on the sound card up to 64 audio channels are supported. In order to assign a track to a specific audio channel of your ASIO sound card, please refer to the Track Inspector [217].

Show Channel Details:
This list reads the sound cards settings and gives you an overview how the channels are routed in the sound card.

6.3.4.7 Cues

The Cues tab in Pandoras Box gives you an overview of all cues used in the sequence [292] shown in the Sequence tab and allows fast changes. In addition, you can also export the information about cues as a CSV file.
The Cues tab is part of the View [287] called "Show". Generally, to open it, please click on "Tabs" in the Toolbar and choose "Cues".

Cues: [1] unnamed | Patch | Assets | Device Types | Aeon FX | FireFly Particles | Playlist

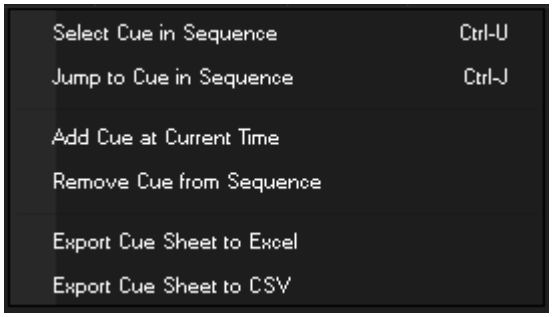
ID	Name	Note	Type	Time	Countdown	Jump/Wait	Target Cue
1	Start	show start	Play	00:00:02:23	--:--:--:--	--:--:--:--	-
3	Cue 2		Wait	00:00:06:08	00:00:01:03	00:00:02:00	-
2	Cue 3	wait for actor XY	Pause	00:00:07:09	--:--:--:--	--:--:--:--	-
4	Cue 4		Jump	00:00:15:14	00:00:03:08	00:00:00:00	-
5	Cue 5		Jump	00:00:18:17	--:--:--:--	--:--:--:--	6
6	End		Stop	00:00:23:09	--:--:--:--	--:--:--:--	-

When the Sequence is paused at a Cue (manually or via a Pause Cue), the current Cue is highlighted in the Cues tab and the next Cue is listed with a Countdown time.

The Cues tab lists all cues from the sequence (currently shown in the Sequence tab) in the order of the cue time shown in the "Time" row. Per default, all rows are activated:
Cue ID, Name, [Internal] Note, Type, Time, Countdown, Jump/Wait [Time], Target Cue
With a right-click in the title line, you can uncheck the rows you do not like to be displayed in the table. The width of a row can be adjusted by dragging the cell border with the mouse.

To change a cell, (double-)click it and start typing (a new ID, Name etc.). The "Type" cells require a double-click to open a drop-down list. You can also multi-select cells and change them at once. Click on a cell and hold either [Ctrl] or [Shift] pressed whilst clicking on another one. The selected cells are highlighted in blue. Now again, double-click to open the drop-down or start typing directly, i.e. with or without the double-click.

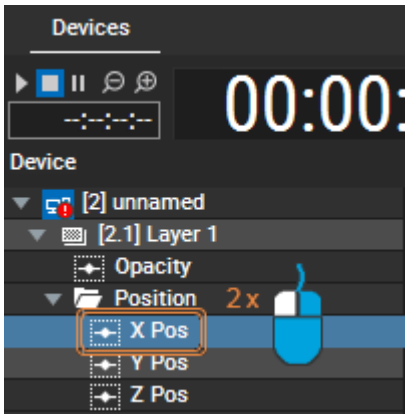
The information in the "Note" cells is only displayed in the Cues tab. All other data can also be seen and edited in the Cue Inspector [208] when a cue is selected in the sequence. Of course, changes made in any tab are synchronized with all other tabs in the user interface. The "Countdown" and "Jump/Wait" times are additionally displayed above the Device Tree. The "Countdown" can also be seen in the Time Monitor tab [309].



When right-clicking on a cue, you can either choose to "Select Cue in Sequence" to display it in the Inspector, or use the shortcut [Ctrl + U] to do so. You can also choose to "Jump" with the Nowpointer to its position in the Sequence (or use the shortcut [Ctrl + J]).
Additional options are to "Add Cue at Current Time" which adds a new cue at the current Nowpointer position or to delete the selected cue entirely.
Lastly you may export the entire cue sheet as an excel or CSV file.

6.3.4.8 Curve Editor

This topic explains how to work in the Curve Editor. It allows to fine-tune the behavior of a parameter over time. The Curve Editor opens automatically when a parameter is loaded into it, as shown below. To open the Curve Editor manually, click on "Tabs" in the Toolbar and choose "Curve Editor".

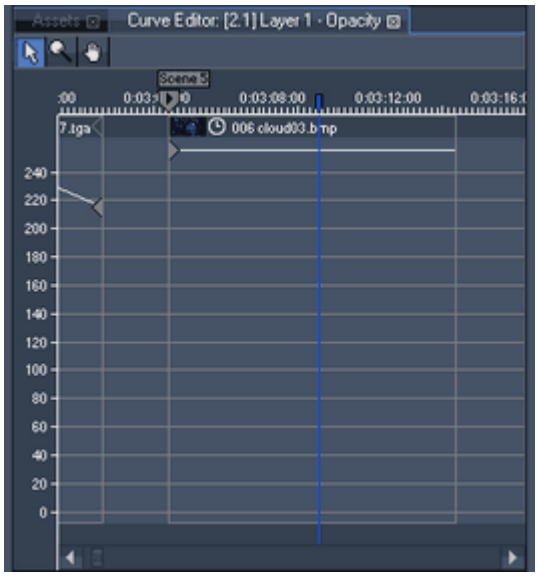


For changing the curve of a parameter, make a right-click on this parameter in the device tree and choose "Load into Curve Editor". You can also double-click the parameter name in the device tree.

Please note that you need a sequence in order to edit curves.
In the example, the opacity of Device [2.1] is loaded into the curve editor.

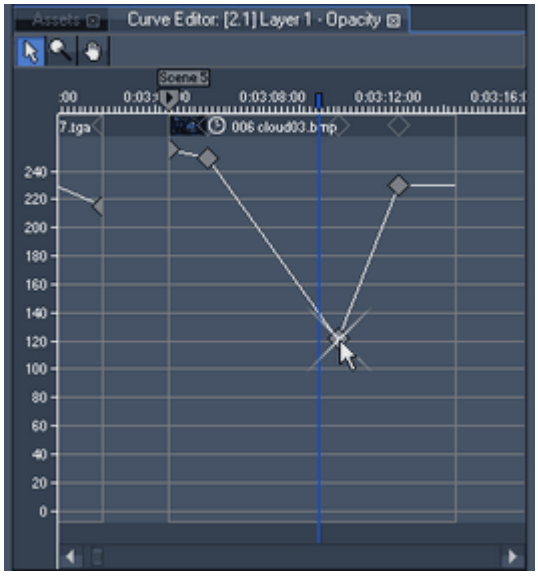
Underneath the tabs you see the timeline and the current position of the now pointer. The first time you open the Curve Editor it opens with time 0:00:00:00, press play (or hit the space bar) to jump to the time the now pointer is located in the sequence tab. From now on as soon as you position the now pointer differently in the sequence tab, the frame in the Curve Editor will follow automatically and vice versa.

As well container and keys are synchronized.



Below the timeline there is the clip container including a thumbnail and the name of the clip. The greyish borders on the left and right mark the start and end time of the clip container. The greyish borders on the top and bottom mark the value range of the parameter (depending on 8 or 16 bit values).
In this example the opacity key at the left border shows a constant value going until the end of the clip.

To navigate in the Curve Editor there are three modes:
- select and edit clips, key frames and their handles
- zoom the y-axis (the parameter value range) that is depicted on the left diagram border. Pressing [Alt] will zoom out. Alternatively you may use the shortcuts [Shift and + or -] on your main keyboard.
To zoom the x-axis (the time) use [+ / -] , just as you do it in the sequence.
- move the content depicted in the Tab in both axes (without moving the now pointer). Alternatively hold down the M key on your keyboard.



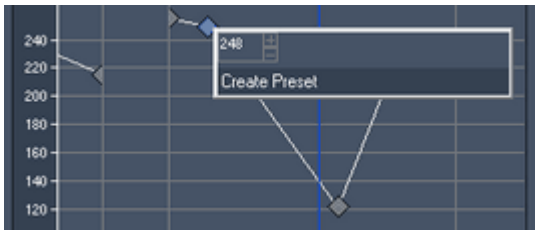
Inserting and Removing Keys

To insert more keys for the used parameter, right-click in the Curve Editor and a new key will be created.
To remove a key, select it and press [Delete] on the keyboard.

Editing Keys

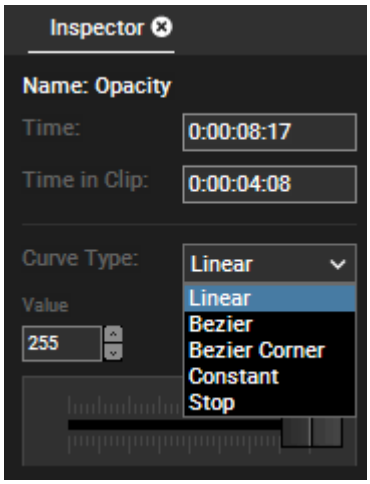
To move a key, please drag and drop it to the new position.
Moving it left and right will move it within time.
By moving it up and down, the value of the key increases or decreases.

With the shortcut [Ctrl + left or right arrow] you snap the now pointer exactly to key frame. Now, when dragging the key frame up and down it will stick to the same time.



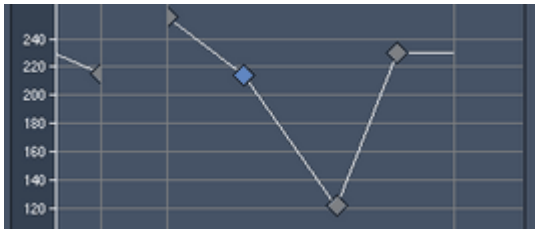
If you want to move it to a certain value, do a right-click on the key and insert the value or click inside the value field and move the mouse-cursor up and down.

Alternatively you may use the Device Control Tab or the Inspector Tab. But only the Inspector gives the possibility to multi-select item and change them synchronously.

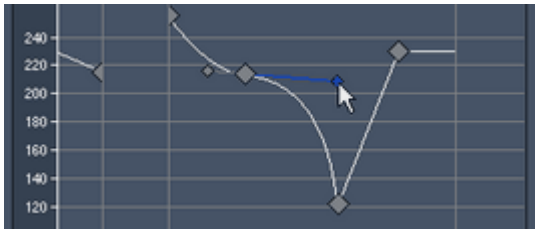


Curve Types

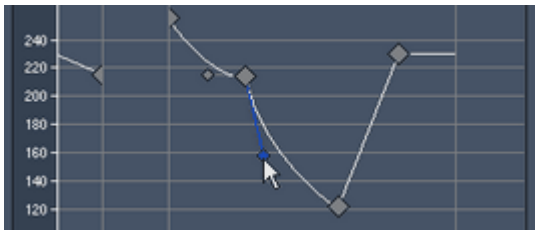
To change the curve type of a key, select the key and choose a Curve Type from the drop down menu in the Inspector Tab.
You can choose between Linear (Standard), Bezier, Bezier Corner, Constant and Stop.



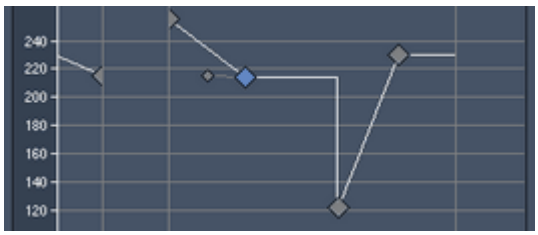
- Linear
  A linear line is drawn between the previous and next key.



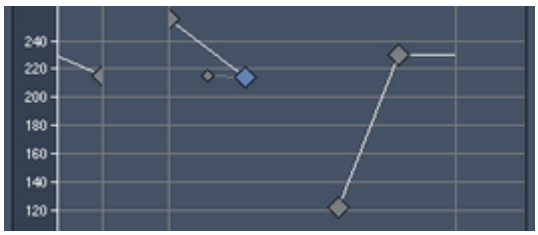
- Bezier
  The key gets two handles and you can intuitively smoothen the curve by moving the handles and changing their length. The handles always will stay in a line.



- Bezier Corner
  Compared to Bezier the handles can be modified independently from each other.



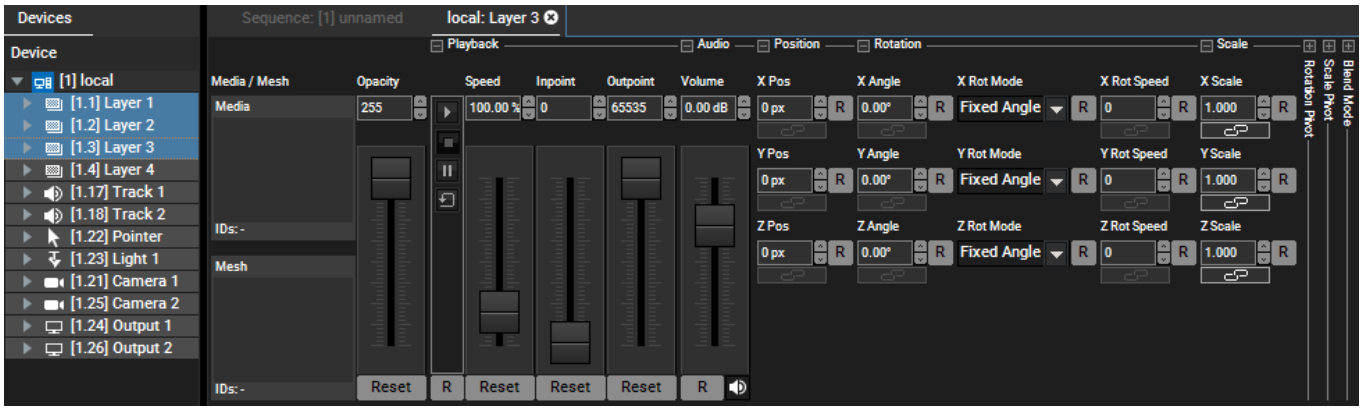
- Constant
  The value of the key will be continued until a new value is given.



- Stop
  The connection to the next key will be removed, the key's value will be held until a new value is given.

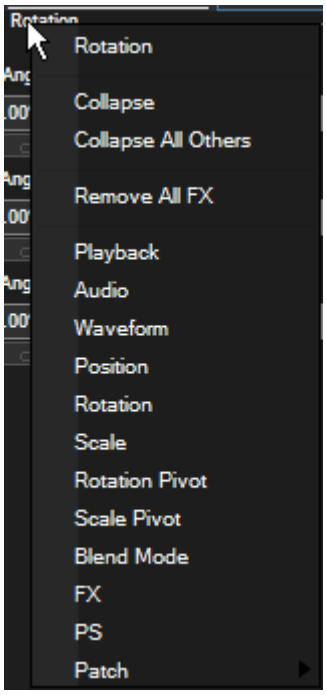
6.3.4.9 Device Control

The Device Control tab shows all available device parameters for the Device that is selected in the Device Tree [173] . If several devices are selected, the one with the gray dotted line is loaded.



Please see the chapter Devices and Layer Parameters [318] for detailed information about each parameter and available layer types. This chapter covers more general information:
- Changing Parameter Values [171]
- Expand & Collapse Parameter Sections [172]

Context Menu



If you right-click in the height of the parameter section name, the context menu opens.

- Collapse / Expand the current parameter section
- Collapse all other parameter section except this one
- Remove All FX [323] from the device
- Expand any parameter section by name.

6.3.4.9.1 Changing Parameter Values (Device Control)

This chapter describes how to change or link parameter values in the Device Control tab. [171]

The parameter values are shown in decimal readout and centered by default. To display them in percentage readout and/or non-centered, please have a look in the Configuration tab > Devices / Parameters [156].

Changing Parameter Values

You can change a parameter in the Device Control tab or in the Device Tree [173], as described here [177]. All changes made to the parameter values will turn the parameter active and red (until it is stored in the Sequence).

If there is a fader for the parameter, you have the following possibilities to change the parameters value:

- Use the fader handle for coarse control

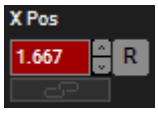
- Click the fader once and use the arrow keys [Up] or [Down] => +1 / -1 steps; hold [Ctrl] too => +10 / -10 steps.

- Click in the numeric value field above the fader and drag it up / down for sharp tuning.

- Click the arrow buttons next to the numeric value field for finest control.

- Enter the new value manually into the numeric value field (double-click first) and press [Enter] to apply the change.

- Press the "Reset" button to set the parameter back to its default value. The status "active" will be removed from the parameter.



If there is only a numeric value field for the parameter, you have the following possibilities to change the parameters value:

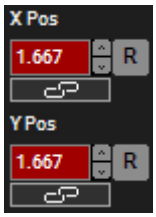
- Click in the numeric value field and drag it up / down for sharp tuning.

- Click the arrow buttons next to the numeric value field for finest control.

- Enter the new value manually into the numeric value field (double-click first) and press [Enter] to apply the change.

- Press the "R" (for reset) button to set the parameter back to its default value. The status "active" will be removed from the parameter.

Linking Parameters



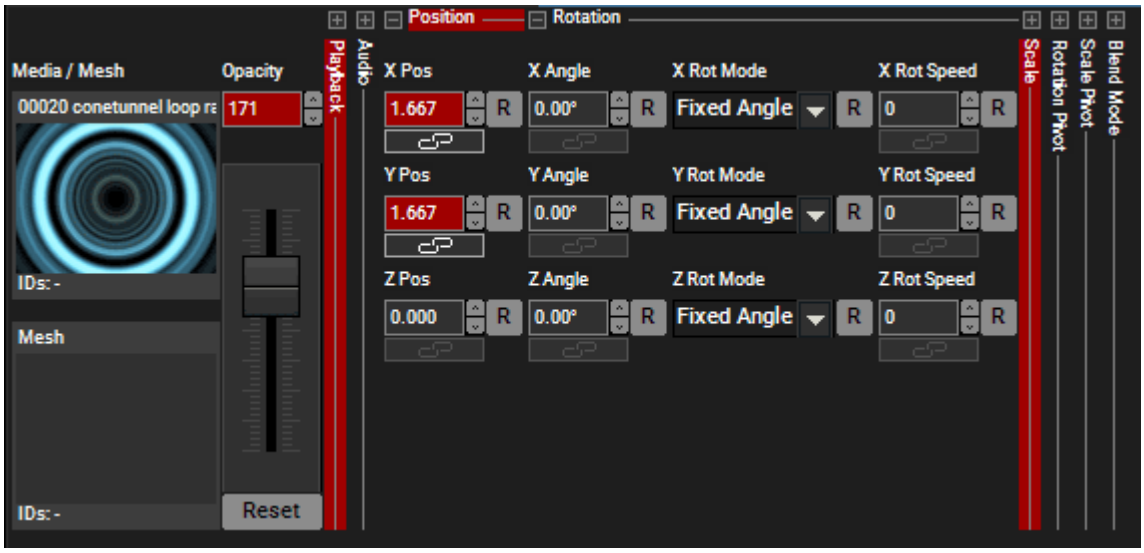
To link several parameters click on the link icons below the faders or numeric fields in the Device Control tab or in the Device Tree. If one of the parameters is changed now, all parameters linked will accept the values in relation to it. Only parameters from the same type (like rotation or scaling) can be linked! Per default, the Scale parameters are already linked.

6.3.4.9.2 Expand & Collapse Parameter Sections

This chapter explains how to expand or collapse parameter sections in the Device Control tab [171].

Expand & Collapse Parameter Sections

To have a better overview in the Device Control Tab, parameter sections may be shown in collapsed or expanded mode. Below you see, that only the sections for the Position and Rotation parameters are expanded.



To collapse / expand a parameter section:
- Click the "-" or "+" icon in front of a parameter section name
- Double-click the parameter section name or its following line
- Double-click in the area below the parameters
- Use the right-click menu [171]

6.3.4.10 Device Tree

The Devices tab (or Device Tree) in Pandoras Box is the main place to browse and edit all available Sites and Devices (i.e. Layers) in the project.

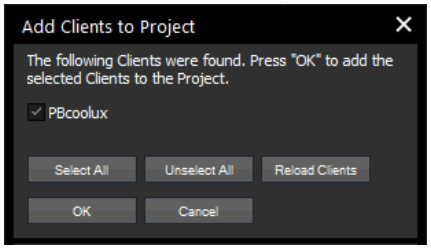
The Device Tree...
- is always linked to the Sequence Tab in order to browse and edit keys stored to the sequence (both tabs cannot be closed!)
- allows a Site to be included in / excluded from the Preview (see below)
- allows you to sort and (un-)hide Sites, Layers, parameters and effects, see next chapter [175]
- offers distinct columns for different information: parameter names, values and various (clickable) icons, see next chapter [177]

In case you are rather interested in the devices itself, e.g what is the meaning of a parameter in a Video Layer, how to add an effect to a layer or how to use a DMX device (DMX fixture) please refer to the topic about "Devices and Layer Parameters [318]".

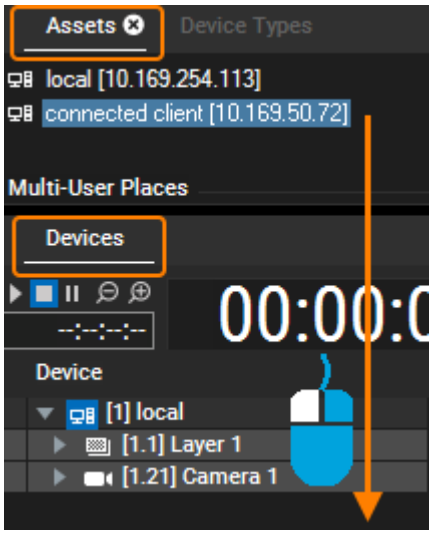
If you are working with a single Pandoras Box system (i.e. in stand-alone mode), the "local" Site in the Device Tree lists all layers that are part of your system. You can add as many Output Layers as there are licenses available on your dongle(s).
If you are working with more systems (i.e. a Master-Client-setup) you need to add these Clients to the Device Tree, which automatically includes all necessary layers. To do this, there are two workflows.

Adding Connected Clients to the Device Tree

In case you like to add Clients which are already connected, please make sure that they are:
- started (in the same Version [125] as the Master system) and showing the Pandoras Box Client window [316] and
- set to the correct IP address and Pandoras Box domain [147].

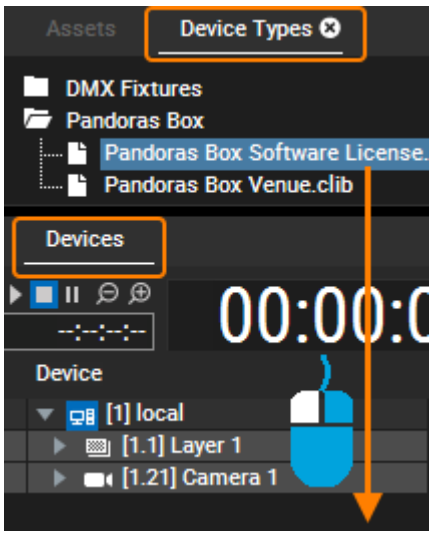


Since version 6.0.3, PB offers to add Clients to the Project automatically when starting a new project on your Master software. All existing Clients are visible in the dialog "Add Clients to Project". The list can be reloaded with the "Reload Clients" button. You can select or unselect individual Clients or all of them by pressing the according button. Press "Ok" to add all selected Clients to the Device Tree. The next chapter [175] describes how to reorder the Sites.



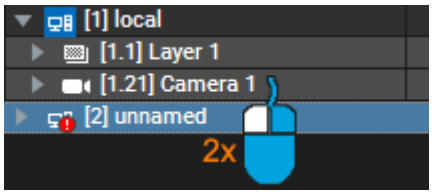
To manually add Clients, please go to the Assets tab [131] which lists all connected PB systems. Drag them from there to the Devices tab. You can also multi-select them by holding the [Ctrl] key.

Adding Clients to the Device Tree for Pre-Programming

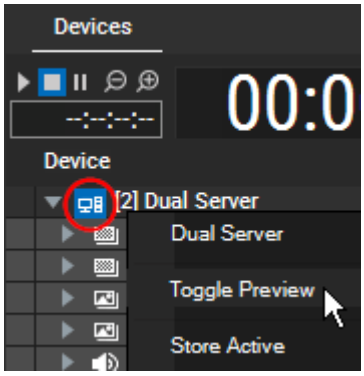


If your Clients are not connected yet, e.g. if you like to pre-program a show, you may drag an according system from the Device Types tab [183] into the Device Tree. Simply open the "Pandoras Box" folder and choose the entry "Software license". A pop-up asks you, how many Outputs should be added to that system.

Later on, when the real systems are connected, you can simply select the pre-programmed Client and enter the IP address in the Device Inspector [210]. A Client can manifest on that system if the number of Output Layers is less or the same as the number of available licenses on the dongle(s). If there are more Output Layers, PB offers to delete exceeding Output Layers automatically. You can of course also remove Output Layers manually in case you like to save or copy your programming (i.e. Containers on Output Layers) first.



After adding a system to the Devices tab, it is shown with collapsed layers. To unfold/fold a site double-click on it.
The right-click menu offers a rename command [F2] in case you like to work with more descriptive names.



Per default, a Site is not included in the Preview [243] when added to the project. Note that its icon is not highlighted. To add a Site to the Preview rendering, right-click it to open the context menu and choose "Toggle Preview". Now, the icon is highlighted in blue.

See the following sites for more information:
Site, Layer and Parameter Structure [175]
Changing Parameter Values (Device Tree) [177]
Context Menus - Device Tree [178]
Device Selection [182]

6.3.4.10.1 Site, Layer and Parameter Structure

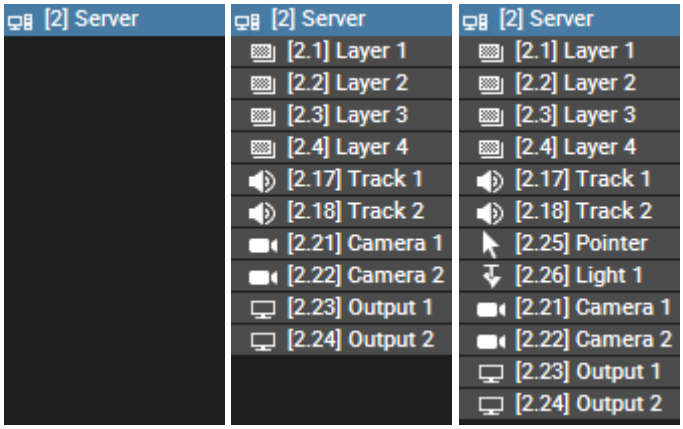
This topic explains how to add, hide and show Sites, Devices (i.e. Layers) and Parameters in the Device Tree tab and how to change the render order of Layers and Sites. For more information about the Device Tree, please see the introductory chapter [173].

Device Structure in the Device Tree

When adding a Site (from the Assets tab [131] or Device Types tab [183]) to the Device Tree, it is shown with collapsed layers (see left image).

Double-click a Site to unfold the default device structure as seen in the middle image. The default view includes for example the first four Video and two Audio Tracks as well as all available Cameras and Outputs. The number of layers shown per default, can be changed in the Configuration tab > Devices / Parameters [156].

You may (un-)hide layers or add new ones. The right image shows the Server with an added Pointer and Light Layer.



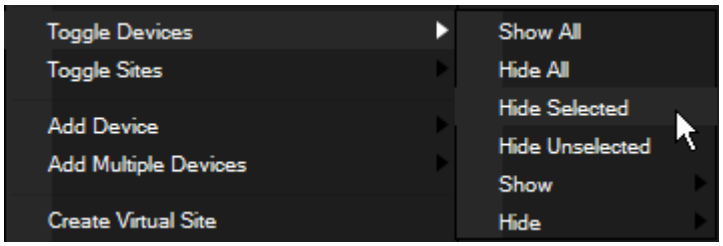
The commands can be found when right-clicking on the site itself. "Hide" lets the layer disappear temporarily; "Toggle Devices" allows to unhide again and "Add Devices" lets you choose from a layer list that can be added to the Device Tree. All other commands are explained in the next topic "Context Menus [178]".

The next topic shows what you can do when unfolding a Device to see its parameters.

Hide and Show Sites and Devices

For a better overview and faster navigation, you can hide Sites and Devices in the Device Tree. Simply select the Site or Layer, right-click it and choose "Hide" from the context menu. To select several Layers, select the first one and hold the [Ctrl] or [Shift] key while clicking on other devices to select them too.

The "Toggle Sites" and "Toggle Devices" entry from the Device's or Site's context menu offers many options to hide or show Sites and Devices again.



By the way, if you store a View [310], the Site and Device visibility in the Device Tree is also stored.

Hide and Show Parameters

The Device Tree can also be filtered on the parameter level. To do so, right-click on any Device and choose "Toggle Parameter" to hide or show individual items or all items at once. In addition, the list shows you the keyboard shortcuts to apply the filtering to any device selection.

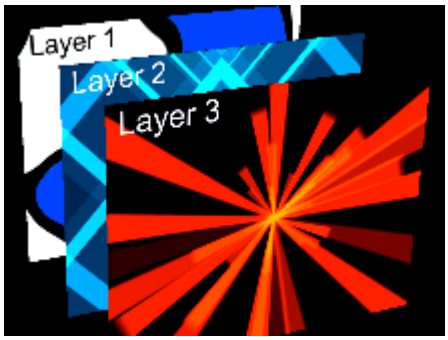
When working in Device Tree, the following shortcuts apply to current selected Devices:

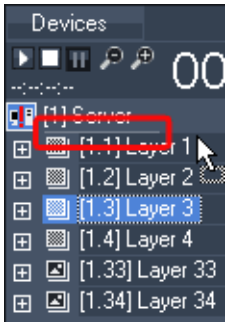
A	Show all Parameters
H	Hide all Parameters
U	Show only Parameters used in Sequence/Show all Parameters
M	Show/Hide Media Parameter
O	Show/Hide Mesh (Object) Parameter
I	Show/Hide Opacity Parameter
V	Expanse/Collapse Playback Parameters
X	Expanse/Collapse Audio Parameters
P	Expanse/Collapse Position Parameters
R	Expanse/Collapse Rotation Parameters
S	Expanse/Collapse Scale Parameters
T	Expanse/Collapse Rotation Pivot Parameters
D	Expanse/Collapse Scale Pivot Parameters
B	Expanse/Collapse Blend Mode Parameter
F	Expanse/Collapse FX Parameters
G	Expanse/Collapse Particle System Parameters

Changing the Render Order of Layers and Sites

All Video Layers are arranged close to each other in Z order:
Layer 1 is the bottommost layer, all other layers enqueue above this layer 1.

The device tree allows you to change this layer order (without having any Z Position changes applied to any layer). Simply unfold the site and drag and drop any layer to another position.

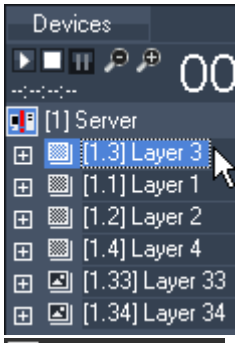


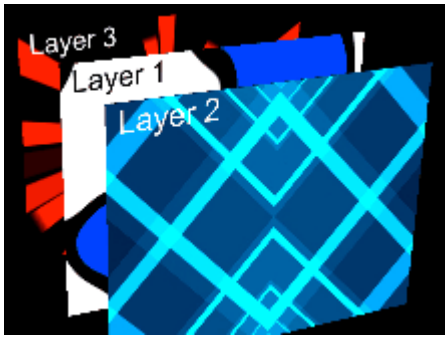


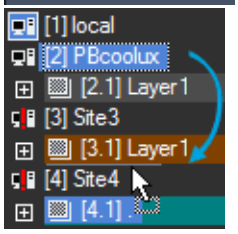
The two images above show the default layer order.

Then, layer 3 is selected and dragged above layer 1. Note that the mouse cursor changes and that the new position of the selected layer is indicated by a slender white line (enhanced with a red rectangle in the left image).

After dropping layer 3 it now sits above layer 1 in the device tree The result is that layer 3 will be positioned behind layer 1 now in Z order and the preview looks like in the image below.

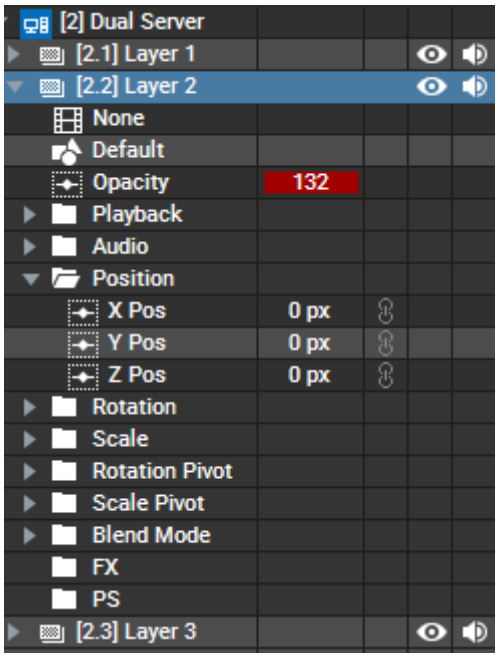






Since version 6.0.3 it is also possible to re-structure entire sites in the Device Tree via drag and drop. Simply drag the Site to its new place as depicted in the left image. Again, note the horizontal line and the cursor that indicates a valid position. As with Layers, reordering Sites also influences the render order. If you add two Sites to the Preview and they have the same X / Y Camera Offsets, you will see the Sites output of the bottom Site as it overlays the other one. Note that you do not need to change the Camera Offsets as the Site Inspector offers a Preview offset [210].

6.3.4.10.2 Changing Parameter Values (Device Tree)

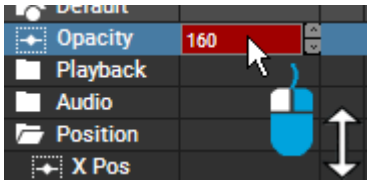


This topic explains how to change a parameter value in the Device Tree and how to use the Device and parameters icons. For more information about the Device Tree, please see the introductory chapter [173].

First, double-click a Device to unfold its parameters, or single-click the arrow icon. Many parameters (like the position parameters) are grouped but can be unfolded in the same way. The default view includes all parameters. As explained in the previous chapter you can show and hide parameters.

As you can see in the image, there are distinct columns for different information: parameter names, values and various icons are placed separated.

Changing a Parameter Value



The device tree allows direct changes to any parameter value. Double-click the value you like to edit, you will see a numeric box. Now you can...
1) left-click the value and drag it
2) click the small up / down icons
2) double-click in the numeric box and enter a new value using the keyboard. Hit [Enter] to apply the change.

If not already, the value is now an active value. The active state is shown in the value column and on the left side of the layer or parameter name.

Opening the Parameter in the Curve Editor

Double-clicking a parameter name (first column) opens the Curve Editor [169] and loads the parameter there.

Clickable Icons

As you can see, there are further columns next to the columns for the parameter name and parameter value. They display icons that can be clicked for direct access.

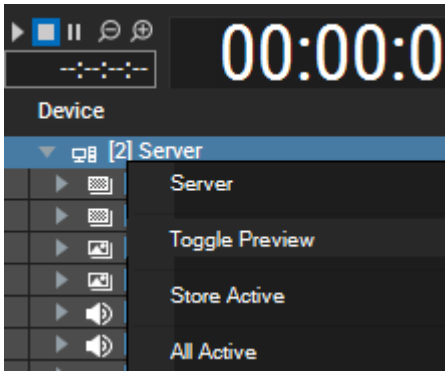
The third column displays "Link" icons where applicable. Open the "Position" parameters for example. Now you can link the "X Pos" and "Y Pos" parameters which means that any change to one value is also applied to the linked value. You can also link icons in the Device Control tab as explained in the chapter Device Control Tab [171].

The forth and fifth column display icons to mute the (video) rendering or audio. Alternatively, you can also click the "Mute rendering" or "Mute audio" option in the Inspector [211] after selecting the Layer.

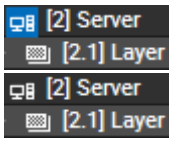
6.3.4.10.3 Context Menus - Device Tree

This topic explains the context menus from a Site, a Device [179] and a Parameter [181]. For more information about the Device Tree, please see the introductory chapter [173].

The Context Menu of a Site



[Site Name]
This simply shows the site's name.



[Toggle Preview]
Toggles the site to be included in / excluded from the preview [243]. When the site is included in the preview the site icon is highlighted in blue as seen in the top image.
If it is not in the preview the icon not highlighted.

[Store Active]
Stores all active parameters of the site's devices as containers to the sequence.

[All Active]
Activates all parameters of all devices of this site (every parameter will turn red).

[All Active (Partially Active Devices)]
Activates all parameters of all layers of this site that already have active parameters.

[Clear All Active]
De-activates the active status off all parameters of all devices of this site. The values will be kept and not set back to default.

[Reset All]
Resets all parameters for all devices of this site to their default values and removes their active status.

[Reset All Active from Remote]
This is of interest when working in a Multi-User environment [234]; it resets all parameters for all devices of this site to their default values and removes their active status that was received from a remote Multi-User Master.

[Take Over Activity]
This is of interest when working in a Multi-User environment [234]; it changes the status from "remote active value"

which is indicated by a beige / yellow color to "local active value" which is indicated by a red color. For more information please read the "Multi-User" chapter [234].

[Rename]
Enables you to rename the site. You can do this by pressing [F2] as well.

[Remove]
Removes the site irrevocably from the device tree. You will have to confirm this action in a pop-up window.

[Hide]
This command will hide the site in the device list. To show it again, right-click on another site and choose the "Toggle Sides" command as explained below.

[Toggle Devices]
Choose if you want to show or hide all devices, or show / hide single devices of this site in the device tree. A time-saving way to select and hide more than one layer is using the right-click menu of a device.

[Toggle Sites]
Choose which sites you want to show or hide in the device tree.

[Add Device]
Adds one new Device (layer) to your site. A pop-up lets you choose the possible type of layers which depends on your PB product.

[Add Multiple Devices]
Adds several new Devices (layers) to your site. A pop-up lets you choose the possible type of layer which depends on your PB product.

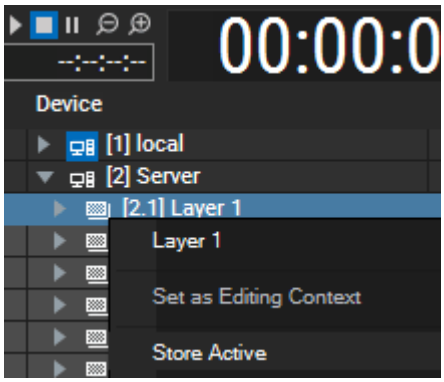
[Create Virtual Site]
This command will create a Virtual Site out of all selected sites. Please see the chapter "Virtual Site" [311] for detailed information.

[Clone Site]
This creates a copy of the Site including the entire layer structure (layer names and IDs, render order and all Inspector settings) as well as the programming. You simply need to enter a new IP address. This is especially useful when programming backup scenarios or a show where many Sites equal each other but the Virtual site feature is not possible.

[Enter or Leave Fullscreen]
This command toggles the site into fullscreen and out of fullscreen.

[Patch]
This sub menu offers patch commands.
"Reveal in Patch tab" opens the Patch tab and loads the selected Site.
All other commands [Set Patch Start Values, Copy, Paste, Export Patch State as CSV] are also available in the context menu from the Patch tab which is described in the topic "Site Patch [232]".

The Context Menu of a Device or Layer



[Device Name]
Shows the device's / layer's name (in the example above: Layer 1)

[Set as Editing Context]
This is of interest when working in the Preview tab with the Mesh Editing Mode [257] or Canvas Editing Mode [256] and loads the selected layer as the "Editing context". Please see the linked chapters for more information.

[Store Active]
Stores all active parameters of the device as containers to the sequence.

[All Active]
This command will activate all parameters of this device (every parameter will turn red).

[Clear All Active]
De-activates the active status off all parameters of this device. The values will be kept and not set back to default.

[Reset All]
Resets all parameters of this layer to their default values and removes their active status.

[Reset All Active from Remote]
This is of interest when working in a Multi-User environment [234]; it resets all parameters for this device to their default values and removes their active status that was received from a remote Multi-User Master.

[Take Over Activity]
This is of interest when working in a Multi-User environment [234]; it changes the status from "remote active value" which is indicated by a beige / yellow color to "local active value" which is indicated by a red color. For more information please read the "Multi-User" chapter [234].

[Rename]
This command will enable you to rename the device. You can also do this by pressing [F2].

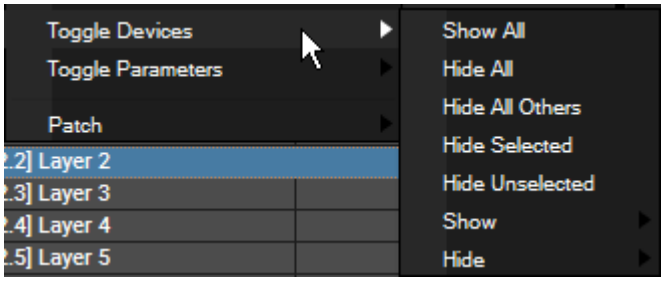
[Remove]
Removes the layer from the site. Please note that the layer will not only be hidden as with the command "Hide".

[Copy FX Structure]
This command copies the FX structure of this device in order to provide it for another layer. This refers to the FX types and their order.

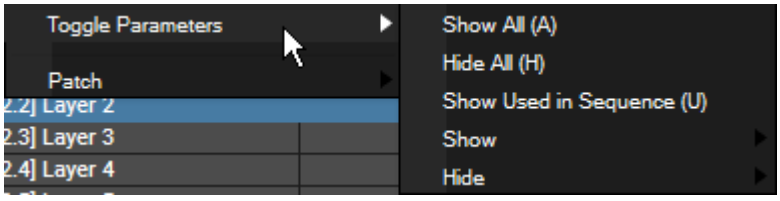
[Paste FX Structure]
Copy a FX Structure from another layer first. This command then pastes the FX structure to this layer. If this layer already contains FX, it will add the copied FX structure after the existing FX.

[Remove All FX]
This command removes all FX of this device. Please note: Once an effect is removed from the layer, it won't be available any more for all of the layer's clip containers which used this effect.

[Hide]
This command will hide the device in the device tree.



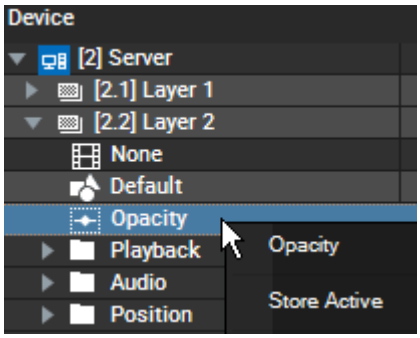
[Toggle Devices]
The pop-up allows to show and hide layers. This can be done with single or all layers, or depend on your current layer selection.
Selected layers are highlighted blue. To multi-select layers, press [Ctrl] or [Shift].



[Toggle Parameters]
The pop-up allows to show and hide parameters. This can be done with single or all parameters, or depend on your currently used once (stored keys in the sequence).
In addition, the list shows you the keyboard shortcuts [314] to apply the filtering to any device selection.

[Patch]
This sub menu offers patch commands.
"Reveal in Patch tab" opens the Patch tab, loads the according Site and selects the Device or Layer.
All other commands [Create Template, Set Patch Start Values, Copy, Paste, Export Patch State as CSV] are also available in the context menu from the Patch tab which is described in the topic "Site Patch [232]".

The Context Menu of a Parameter



[Parameter Name]
Shows the parameter's (in the example above: Opacity).

[Store Active]
Stores all active parameters of the device as containers to the sequence.

[Active]
Activates the parameter (parameter will turn red).

[Clear Active]
De-activates the active status off this parameters. The value will be kept and not set back to default.

[Reset]
Resets the parameters to its default value and removes its active status.

[Take Over Activity]
This is of interest when working in a Multi-User environment [234]; it changes the status from "remote active value" which is indicated by a beige / yellow color to "local active value" which is indicated by a red color. For more information please read the "Multi-User" chapter [234].

[Hide]
Hides the parameter in the device tree. To show it again, right-click on the layer and choose the "Toggle Parameter" command as explained above. The previous topic "Site, Layer and Parameter Structure [175]" shows as well the keyboard shortcuts to toggle parameters.

[Toggle Key Mode]
Allows you to detach the parameters events from the existing clip containers. This applies to all sequences.
Please note: This command cannot be undone. For more details see the topic "Programming on the Timeline [299]".

[Load into Curve Editor]
To edit the parameter curve do a right-click and choose "Load into Curve Editor" or double-click on the Curve Editor icon in front of the parameters (you will find the Curve Editor Tab under Tabs - Curve Editor). Please see the Curve Editor [169] description for detailed information.

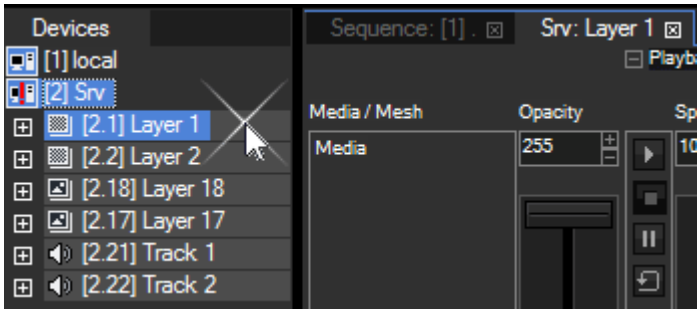
[Patch]
This sub menu offers patch commands.
"Reveal in Patch tab" opens the Patch tab, loads the according Site and expands the Device or Layer to select the Parameter.
All other commands [Create Template, Copy, Paste, Export Patch State as CSV] are also available in the context menu from the Patch tab which is described in the topic "Site Patch [232]".

6.3.4.10.4 Device Selection

This topic explains ways how to select Sites and Devices in the Device Tree. If you like to save selection-groups, please refer to the topic "Groups [281]". For more information about the Device Tree, please see the introductory chapter [173].

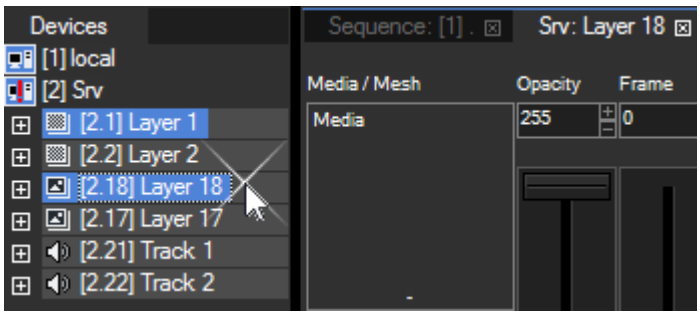
Selection of a Site or Device

Select a Site (i.e. a Server) or a Device (i.e. a Layer) by left-clicking on it. The layer icon will turn blue and the selected device will be loaded into the Device Control tab [171]. In addition, the Device properties are visible in the Inspector [211].



Multi-Selection

To select several devices, select the first one and hold down the [Ctrl] or the [Shift] key while clicking on another Device to select it too.
When several devices are selected, the last one selected will be loaded into the Device Control tab [171]. Note that this Layer is outlined with a dashed line. You can simply click on another Layer to load it into the Device Control tab, but keep the [Ctrl] or [Shift] key selected to maintain the selection. This is especially of interest when working with the Align feature [124].



Once you have done your selection, all parameter changes (adding new media/mesh, moving a fader) on one of these selected layers will be assigned to all selected layers. E.g.: Changing the opacity of layer 1 will result in opacity changes in all selected layers.

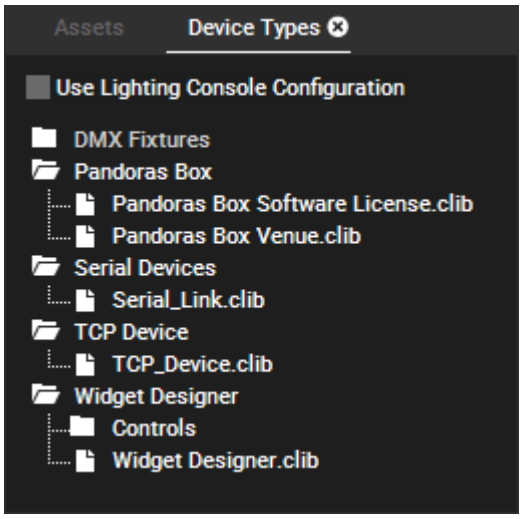
Press [Esc] to deselect, or select another item.

Selection Group

Pandoras Box offers the possibility to save which Devices are currently selected. Simply right-click on the "Groups" folder in the Project tab [278] and say "Create Group" to save the current selection as a selection group that can be recalled any time. For more information please refer to the topic "Groups [281]".

6.3.4.11 Device Types

The Device Types tab shows you all devices which you can use for programming. Simply, select a device and drag it into the Device Tree [173] list.



By the way, if you are starting a new project on your Master and the Pandoras Box Client systems are already connected, a pop-up will ask you whether you like to add the Clients automatically to the Device Tree. Alternatively, you can add Clients manually from the Assets tab as they appear there as soon as they are available in the network.

In case you like to program with Pandoras Box Clients that are not connected yet or with one of the other devices from the tab, select the according device and drag it into the Device Tree. If the device shows up with a red "!" it just means that it is not connected yet. As soon as it is available in the network (and in some cases also started the according software) select it in the Device Tree and enter the IP address in the Inspector tab [210].

Please find detailed information about each device and its parameters, in the topic "Devices and Layer Parameters [318]".

More information:
- Layout and commands from the Device Tree tab [173]
- Programming and storing containers and navigating through the Sequence tab [292]

Multi-License

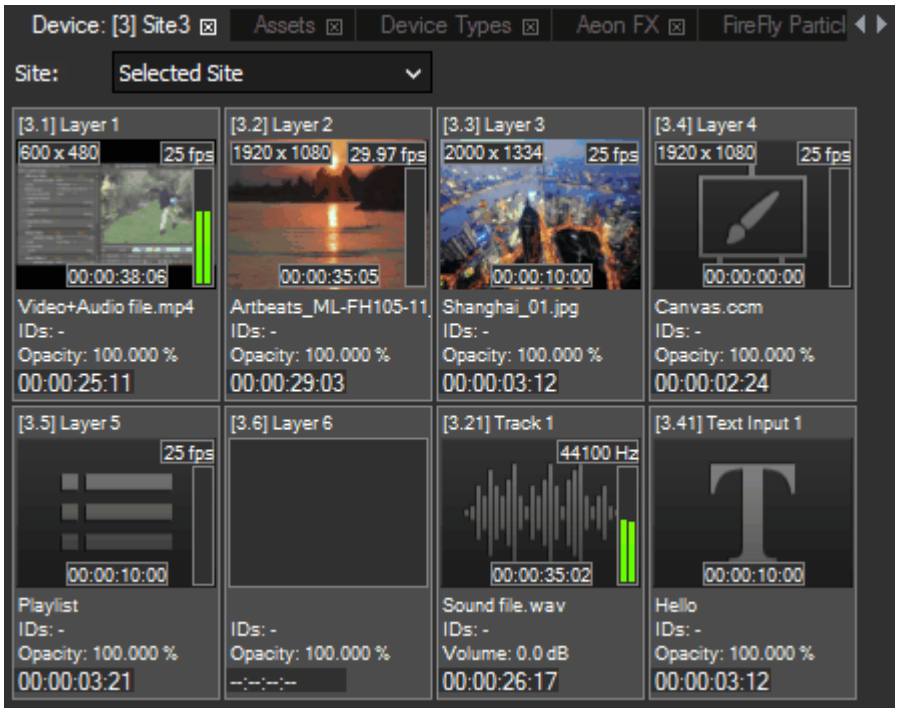
If you are adding the "Pandoras Box Software License" a pop-up will ask how many Cameras and Outputs should be applied. For each Output Layer you will need one license. If your Site contains four Output Layers, it can only manifest on a PB system with four licenses. If it holds only three, a dialog offers to remove one Output Layer. Multiple licenses can be stacked by combining dongles or they can be applied to a single dongle.

Using Lighting Console Configuration

If you activate the "Use Lighting Console Configuration" check box, all Pandoras Box devices that you then add to the Device Tree will be patched already. In the Patch tab [228] you will see that the Layers (or rather parameters) are patched according to our DMX tables [708] which include the most relevant parameters. This DMX library can be loaded on most lighting desks to control Pandoras Box.
In addition some "Multi-FX" [444] effects will be added to the Layers which allow to choose various effects. This has the advantage that all Layers have the same Patch channels instead of changing channel numbers due to individual effects on various Layers.

By the way, you may start Pandoras Box in a dedicated Lighting Console Mode. This is an option in the start-up dialog [118] or when opening a new project (File menu [119]). In this mode, the check-box is already activated.

6.3.4.12 Device Viewer



The Device Viewer tab in Pandoras Box shows an overview of the content that is playing on the Site that is either selected in the Device Tree [173] or selected in the drop-down menu.

The Device Viewer tab is part of the View [287] called "Show". In case you closed it, please click on "Tabs" in the Toolbar and choose "Device Viewer" or use the shortcut [Ctrl + I]. In difference to most other tabs, the Device Viewer can be opened multiple times.

The Audiometer depicts the Audio level of a resource containing audio, that might be an audio file playing on an Audio Track [661] or a video file containing audio information playing on a Video Layer [647]. Too see all information, please check the following:

- a Site must be previewed and selected,
- the (audio) files must be spread to the Master in order to play them there in case you add them from a remote location to the Project,
- the Master needs to have a sound card, but not necessarily an ASIO one.

In addition, only devices displayed in the device tree are shown in the device tab (no hidden devices).

Depending on the media type [90], additional media information is displayed with the thumbnail:
- in the top left corner you will see the media size
- in the top right corner there is the frame rate or sample rate
- on the right side the Audiometer displays the audio level if the resource contains any (Configuration tab > ASIO Audio [166])
- on the bottom of the thumbnail the clip container length is displayed
- below the thumbnail you see the media's name, its DMX file and folder ID and the current opacity or volume value
- the last entry at the bottom shows the countdown until the container's end, in case the content is saved in the timeline .

Selecting a layer field here in the Device Viewer will also select the Layer in the Device Tree and show the according information in the Inspector [190].

Since version Pandoras Box 6.1 the Device Viewer tab has a new drop-down menu to select a Site and hence, the tab can now be opened multiple times (via the Menu "Tabs"). This feature is very useful when you need an overview of several Sites.

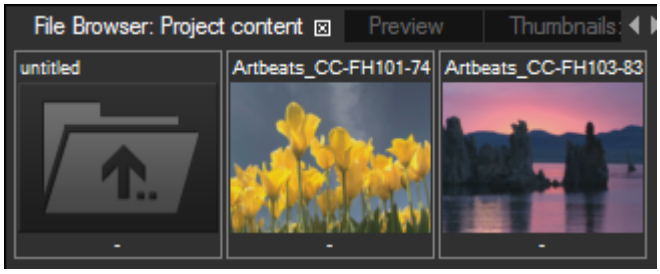
As a reminder, you can right-click a pane and choose to split it horizontally or vertically and then drag the second Device Viewer to the new location. This is explained in the chapter "Layout [313]".

6.3.4.13 Encoder Extension

The built-in Pandoras Box Encoder Extension allows to transcode most AVI -and Quicktime- as well as WMV files to Mpeg1 or 2 up to a 4k resolution. Please see the chapter "Encoder Extension" [103] for more information...

6.3.4.14 File Browser

The File Browser is available in Pandoras Box since version 6.4.0. Right-click on a folder in the Project [278] tab or in the Assets [131] tab, and you will see the option to open this folder in a new File Browser.



It opens in a new tab and shows the content from the respective folder as thumbnails. You can open as many File Browsers as you need. If you are interested in changing the layout of the user interface, please see the chapter "Layout [313]". It describes how to add or break-out panes and how to save and load views.

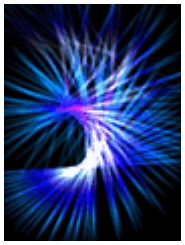
The Thumbnails [310] tab is still available but as always, you can open it only once and the displayed content changes dynamically whenever you select another folder. The File Browser on the other hand is more static as it keeps displaying the same content.

Files from an Assets tab folder that are displayed in the Thumbnails tab or File Browser can be dragged into the Project tab. After adding files to the project you can assign them to a Layer as described below.
Files from a Project tab folder that are displayed in the Thumbnails tab or File Browser can be dragged into the Sequence tab. You may also double-click a thumbnail in order to assign the file to the currently selected Layer.
Please be aware that all thumbnail images are stored locally. If the media is not present on all Clients you may have to spread the media to all Clients in order to assign it to the selected devices. This is described in the Assets chapter [131] in more detail.

6.3.4.15 FireFly Particles

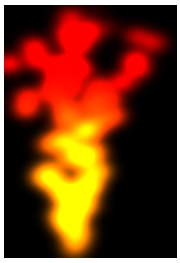
The FireFly™ particle system allows creating elaborate real-time particle effects in a true interactive 3D compositing space. There are 3D and 2D particle systems available on Pandoras Box.

Examples for Particles

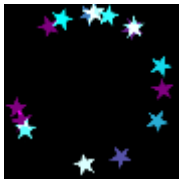


The Stock Assets in Pandoras Box offer various textures you can try and play around with, e.g. Textures > Gradients, Noise or Particles.

For the depicted example, a gradient image was stretched and rotates slowly around itself. Simple parameter choices allow creative content creation and adding of visual effects on the fly, not only for interactive solutions.



For this example a blurry white dot was chosen for "Media" and the "Point" emitter type emits particles with an opening angle of 45°. The particles are orange and change later to red, they only appear yellow in the beginning as they overlap more and the colors add up. All parameters are explained below.



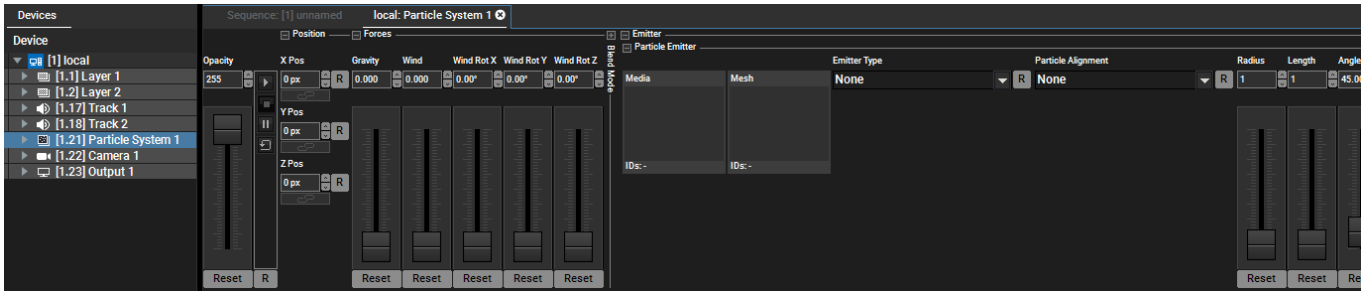
Here, star particles are emitted from a "Circle" emitter type which is rotated towards the Camera. Each star rotates around itself during its life time and changes from pink to white to blue. The Speed parameter is 0.

With the "Blend Mode" parameter you can setup, how particles blend with the background.

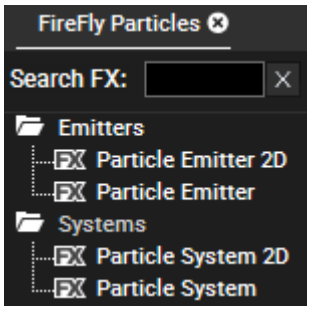
Since version 8, Particles are rendered based on the blend mode of the Layer. In prior versions, they were mixed with the background as if their blend mode was "Add". With the ability to choose a distinct blend mode, you can now also use particles darker than the background or create other effects.

Particle System Layer

The easiest way using particles, is to add a "Particle System Layer". Simply right-click a Site and choose "Add Devices". This layer is optimized and offers only parameters mostly needed for particles.



Old Workflow: Adding Particles to a Video Layer



The older workflow is still supported but requests more steps and has no advantages. In order to use the Firefly Particles you have to add a Particle System to a Video Layer first. As a second step you may then drag one or several emitters onto the specific system.

First, open the Particles tab: Open the Tabs menu and click on "FireFly Particles". Open the "Systems" folder and drag a 2D or 3D Particle System onto a Video layer in the Device Tree. Alternatively, select the Layer to see its parameters in the Devices Controls tab and drag the Particle System on the empty space right next to the last parameter.

With the particle system you define the environment, setting up the opacity of the system, Gravity and Wind. To see how these settings affect the emitters, you have to add at least one emitter.

Next, open the "Emitters" folder and drag a 2D or 3D Particle Emitter onto the Particle System (not the layer), either in the Device Controls tab or in the Device Tree on the entry "Particle System" under "PS".

The particle system section is now enhanced by the emitter parameters.

Controlling Particles with Widget Designer or via the SDK

Of course, you can also control the parameters of Particles using the SDK [730] or Widget Designer [786], e.g. through interactive devices like the AirScan [1988] or sensors like Phidgets [1383] and more.

In general, the exact parameter name is needed for remote controlling. The name is case-sensitive and all spaces count. The parameter list [1514] shows all parameter names of a Particle Layer and their value ranges. So if you like to control the "Opacity" or "Gravity" you simply type this name as it is. For all parameters in the Emitter group, e.g. "Radius", the parameter name starts with "Particle Emitter" and the pipe "|" character, e.g. Particle Emitter|Radius

If you are using the old workflow with adding a Particle System and an Emitter to a Video Layer, the names are in most cases identical, but with an additional "Particle System|" in front of it, e.g. "Particle System| PS Opacity" or "Particle System| Gravity" or "Particle System|Particle Emitter|Radius".

Note that all position values are displayed in Pandoras Box in pixels but the input via the SDK has a different range per default. The chapter Configuration tab > Unit Management [160] explains more about this and the check box "Interpret Automation Param Input as Pixel Values".

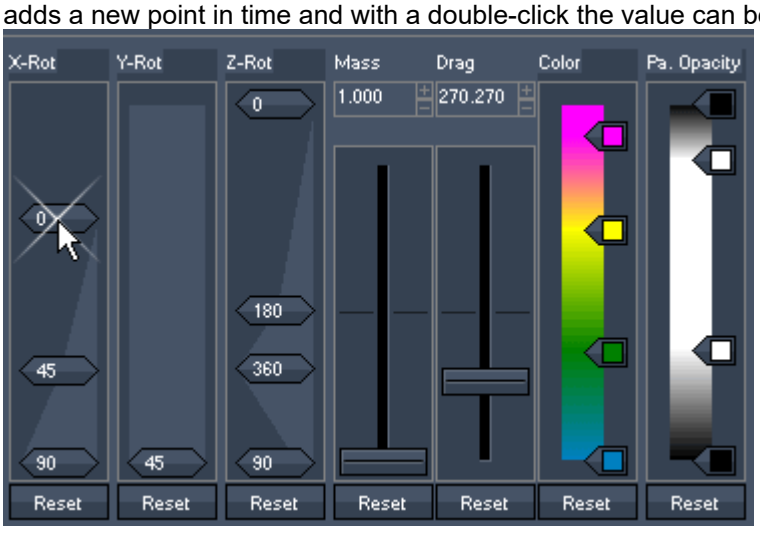
As you see below, there are some "life time" parameter like "Speed" or "Color"; those ones can only be remote controlled via the SDK not Widget Designer.

Parameter Overview

System / Forces Parameter	Description
PS Opacity	Opacity of the whole Particle System
Gravity	Amount of Gravity affecting the emitters (see Mass and Drag parameters)
Wind and Wind Rotation	Amount and Rotation of Wind affecting the emitters (see Mass and Drag parameters)
Emitter Parameter	**Description**
Media and Mesh	Media and (3D) object file used for emitting particles
Emitter Type	Choose how the emitting particles appear: None, Point, Line, Area, Circle, Sphere, Cylinder
Particle Alignment	Alignment of the emitting particles: None - particles (just like other layers) are in XY plane and look along Z-axis Billboarding - particles always face Camera Flight Direction - particles look in the direction they are flying

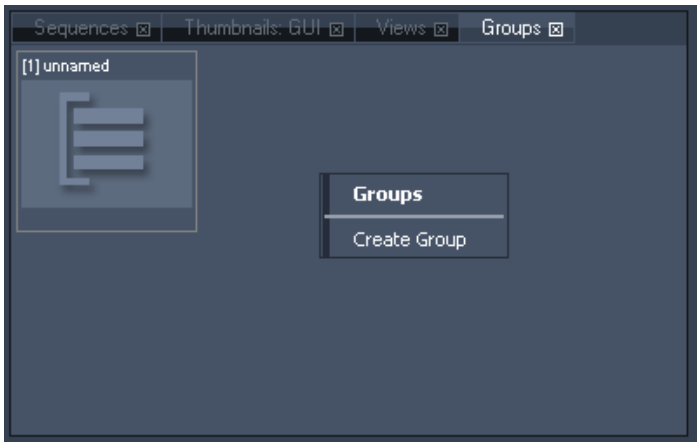
Radius	First size parameter for all "Emitter Types" except "Point" (in units, a PB fullscreen is always 16 units wide)
Length	Second size parameter for all "Emitter Types" except "Point" and "Line"
Angle	Opening angle of particle source
Range	Range of applied "Speed" 0 - constant Speed for all particles >0 Speed value varies from particle to particle
X / Y / Z Pos	Position of "Emitter Type"
X / Y / Z Angle	Rotation of "Emitter Type"
Rot Pivot X / Y / Z Pos	Position of rotation pivot 654
Count/Sec	Amount of particles emitted per second
Time to Live (Sec)	Amount of seconds a particle will live until it disappears (= lifetime)
Speed *	Speed of particles (lifetime depending)
X / Y / Z Scale *	Size of particles (lifetime depending)
X / Y / Z Rot *	Rotation of particles (lifetime depending)
Mass	Mass of particles; the flight behavior of an objects is defined by its momentum (mass*velocity or speed); the momentum decreases the influence of gravity and wind
Drag	Drag of particles; as drag is a force which decreases the velocity / speed of an object it resists the particle's motion, thus it increases the influence of gravity and wind
Color *	Color added to particles (lifetime depending)
Particle Opacity *	Opacity of particles (lifetime depending)

* : These parameters allow defining a curve over the particles lifetime. A right-click into the fader's background adds a new point in time and with a double-click the value can be defined.



6.3.4.16 Groups

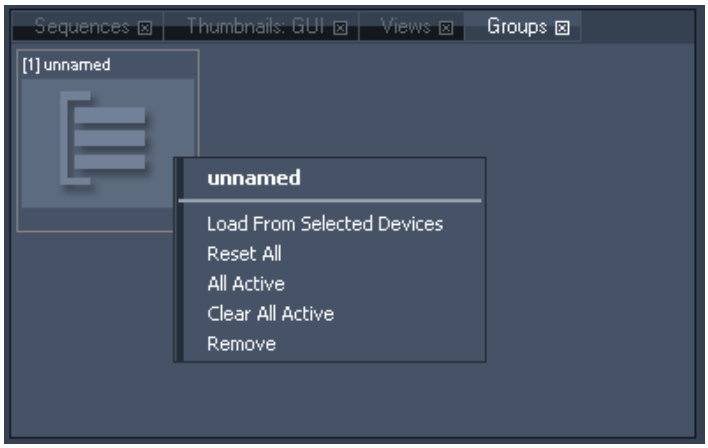
The Groups tab lets you create, edit and select groups in a browser view. The same functions are accessible in the Project tab > Groups [281].



To create a new group, select multiple devices in the device tree.

Then right-click in the Groups tab to create a new group based on the current selection.

To select a group, only press [Esc] to clear the current selection and click on the group item in the group browser to select all group members in the device tree.



If you want to edit an existing group, right-click on the desired group and choose one of the following actions:

- Load From Selected Devices
This command will overwrite the actual group selection with the current selection in the device tree.

- Reset All
This command will reset all parameters for all devices stored within this group.

- All Active
This command will activate all parameters of all devices stored within this group.

- Clear All Active
This command will de-activate all parameters of all devices stored within this group.

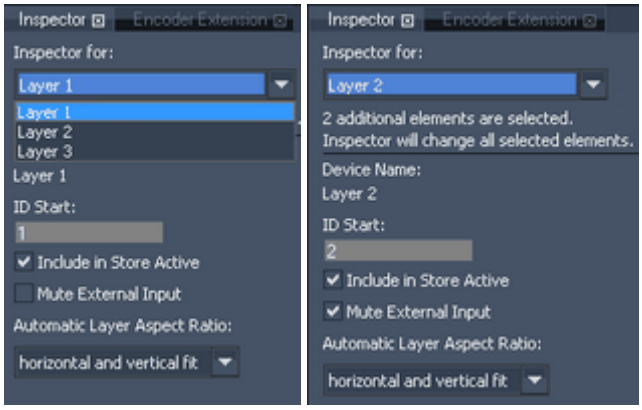
- Remove
This command will delete the group from the project.

6.3.4.17 Inspector

The Inspector is a dynamic, context sensitive property window that will constantly display the properties of the selected items in the user interface.

The Inspector tab is part of the default View [287]. In case you closed it, please click on "Tabs" in the Toolbar and choose "Inspector" or use the shortcut [Ctrl + M]. Even if it was already open and simply overlaid by another tab, it is now the visible tab within the pane [313].

Based on the selected item, the inspector offers different control and property settings.
Since version 5.1 it is possible to multi-select items and modify them all at once. This applies to all possible items such as key frames and containers, cues, files and layers.
If you select items that have different settings at the time you select them the changes done apply to all in the same way. That is, a setting that is possible to be a toggle function (e.g. a check box) would not toggle the other way if it was already used in one of the selected items. Please note, there is no grayed out symbol if items with different settings are within the selection. The options shown in the inspector always refer to the item last selected or to the one afterwards chosen via the drop-down menu the inspector offers. Underneath the drop-down you see the information how many more items you have currently selected.
The only exception of the rule are explained in the file inspector [191].
Generally changes made for IDs apply to the first item selected. The IDs for the other items are set incrementally



The different Inspectors are described in the following order:

Inspectors to be called from Project Tab:
- File Inspector [191]
- Folder Inspector [197]
- Browser Inspector [198]
- Canvas Inspector [199]
- Image Sequence Inspector [199]
- Object Inspector [200]
- Sub Mesh Inspector [201]
- Playlist Inspector [202]
- Text Inspector [203]
- Sequence Inspector [204]
- Tab Button Inspector [205]

Inspectors to be called from Timeline Tab:
- Clip Summary Inspector [206]
- Clip Inspector [206]
- Cue Inspector [208]
- Parameter Keyframe Inspector [209]

Inspectors to be called from Device Tree:
- Device Inspector [210]
- Video Layer Inspector [211]
- Pointer Inspector [213]
- Light Layer Inspector [211]
- Track Inspector [217]
- Camera Inspector [218]
- Output Inspector [220]

6.3.4.17.1 File Inspector

The File Inspector opens when a media file is selected in the Project tab [278]. The left depicted Inspector shows all properties of a video file. A still image has less options. The File Inspector is also included in the Folder Inspector [197]. If you do any changes in the Folder Inspector, they apply to all files as if they were multi-selected.
The Inspectors for other resources (Browser [198], Canvas [199], Image Sequence [199], Object [200], Playlist [202], Text [203]) are described separately.
Since version 5.7 it is possible to set initial values in the Configuration tab > Resources (User) [140], e.g. if you always want an activated FluidFrame option.

The thumbnail shows a frame of the content with additional information about the length, resolution, frame rate, color depth and (embedded) audio.

Time Offset [Milliseconds]
This option is available for audio content. Enter a time in milliseconds to offset this content if played on the Sequence. Audio delay settings are available per Resource, Layer or on Clip Level. That enables you to set up different delays for different output signal chains.

Folder and File ID
Set numeric virtual file and folder IDs for DMX [706], PB Automation [730] and Widget Designer [786] remote control and thumbnail exchange [148]. To easily address several files in ascending order, move over to the Folder Inspector [197] and enter the Folder ID and the start File ID.
Press the button "Clear" to remove the virtual file and folder ID.

Anisotropic Filtering
Anisotropic Filtering does pixel smoothing for real-time video and image scaling, default is on.

Ignore Thumbnail for Preview
Even if using the "Thumbnail for Preview"-Option (see Configuration >Local Preview [150]) this file can nevertheless be rendered and played back in full quality instead of showing its thumbnail, (e.g.. if you need to know where exactly the video playback currently is). Check the Ignore Thumbnail for Preview option to make this exception for this file.

Video Alpha channel
If movie type supports embedded alpha, this option lets you activate the alpha channel. The alpha channel inside images is always activated.

FluidFrame
Choose this option for slow motion or smooth motion frame-adaptive frameblending and framerate conversion. With FluidFrame™, a smooth cross-conversion that can take any input and output framerate, supporting both interlaced and progressive sources and clips is possible now. This way for example 59.94 video input signals are displayed without any frame drop on 60hz output displays. FluidFrame™ can therefore also handle 50hz to 60hz real-time conversions.
In addition to this, it allows smooth playback of highspeed frame rates such as 50 or 60p content and higher. FluidFrame™ Technology can be applied to any live video input or content playback scenario ranging from straight forward Live Video processing to the synchronization of multiple HD SDI streams for Live Stereo 3D applications.
Please note:
When using FluidFrame with 50p, 60p files or live inputs, please check the option "Increase Video Buffer Count [164] " to ensure smooth playback. In all other cases this mode should NOT be used because it requires more system and GPU RAM.

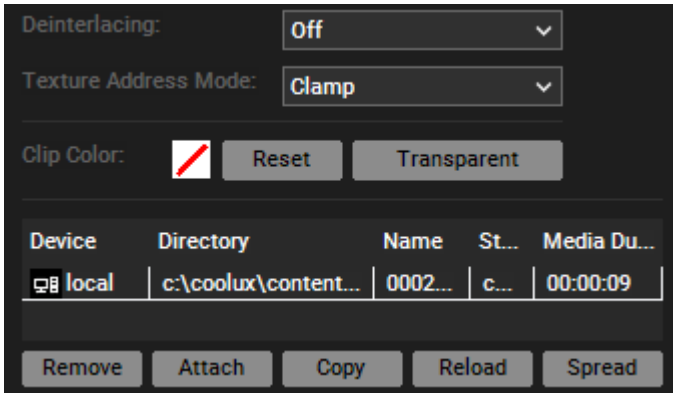
Optimize Mpeg Color Space
MPEG has a reduced colorspace, this option stretches the colorspace (black will become a real black).
Do Colorspace Conversion in Decoder

The MPEG format saves colors in the so called YUV colorspace, whilst Pandoras Box requires the RGB colorspace. Per default the YUV>RGB conversion is done by the graphics card. The advantage is that performance is saved. The disadvantage is that the card's conversion is different to the one used when actually creating an MPEG file. This is for example the case when taking use of any encoding features in Pandoras Box, i.e. Video Export or the Encoder Extension. But also when encoding an MPEG with other programs and tools.
In other words, the Video Export converts RGB>YUV, when using this exported file in Pandoras Box, the graphics card converts YUV>RGB. As the two ways of converting colors do not match, there are different RGB values. To convert in the same way, tick the check box. Note that this will consume more performance and is optimized for the Pandoras Box encoding and decoding features. There still could be a color difference when encoding MPEGs with third party tools.
This option can be used in addition to stretching the colorspace, as described above.

Underscan (Title Safe)
This option is mostly designated for Live Input Signals to crop the image to the Title Safe Area.

Optimize Looping
Check this option if a looped file does not play back smoothly when jumping from the last to the first frame.

Mute Sound:
If a video files contains audio as well, you may check this option if you want to completely mute the sound part of the video file without using the Volume fader.



Deinterlacing:
This is a real-time (zero-delay) de-interlacer for live-input sources based on top or bottom field first. The option fieldblending adds a half-frame delay.

Texture Address Mode
This setting determines how the texture is going to be addressed on objects that use texture coordinates beyond the range [0-1].
- Wrap: the texture will be repeated (wrapped) to infinity
- Clamp (by default): the texture's last pixel will be stretched out to infinity
- Mirror: the texture will be flipped back-and-forth to infinity.

Clip Color
In case this media file the Inspector refers to is saved as a container in the Sequence, the Clip Color is the color the container is filled with. Per default it is gray. In case you set it to "Transparent" it will be of the same color the Layer is set to (see Layer Color [211]).
By clicking into the color field a dialog opens where you can choose any other color for the container. Now, any time you save the file to the Sequence, the container will be filled with the chosen color.
Example: Select a folder in the Project (automatically, the Inspector now refers to all files inside) and set up a purple Clip Color. Select a different folder and choose a different color for it. If you now save files from both folders to the timeline, you can see instantly which folder they belong to.
Note that the (global) Clip Color set up in the File Inspector can be overwritten with the Clip (summary) Inspector [206]. Simply select the containers that should be different and select a new Clip Color.

Spread Selected Resources
In case, you selected multiple files, this option becomes available and spreads the selected content.

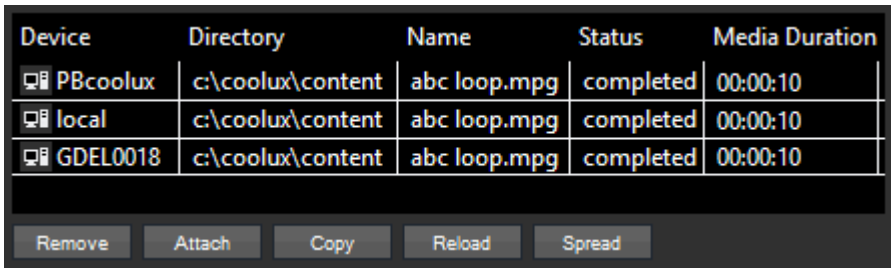
Clean up Selected Inconsistent
In case, you selected multiple files, this option becomes available and removes all inconsistent files. Please see the File Location Table [193] for a detailed explanation about inconsistencies [194].

File location table
Please see the topic File Location Table [193] for a detailed explanation.
Please note, that the changes made in the File location table do NOT apply in a multi-inspector. If you have more than one file selected these changes only apply to one file at time.

6.3.4.17.1.1 File Location Table

The file location table can be found at the bottom from various Inspectors, e.g. the File Inspector [191] or the one from another resources in the Project tab.

The purpose of the table is file management. It gives an overview of the location and status of a file and provides buttons to copy it or attach another file for example.



Background Knowledge: How Pandoras Box Handles Files

To understand the table, it is mandatory to know how Pandoras Box accesses and spreads files.

As a first step, a file needs to be imported to the Project. This is usually done by selecting a file in the Assets tab [131] and dragging it into the Project tab [278]. The project simply saves a link to the file location on the hard drive. In case you work in a network with other remote Clients and Multi-User Places and they are part of your project, imported content is automatically spread to them too, which creates a hard copy of the file on the respective system(s). In Pandoras Box, however, only the link for each system is saved and listed in the file location table. For more information about importing files, please see the chapter Project [278].
A system can only access a file, if it is available on its own hard drive under the saved path. It accesses a file as soon as this file is either played as a container [296] at the current time in the timeline or programmed as an active value [127] in the Master software. The Master (and Sub Masters) will show the file in their Preview whilst Clients render it in the (full screen) window.

If you like to see files only locally, and spread them later for example, you can deactivate the automatic Spread function in the tab Configuration > Resources (Global) [155]. However, if you do so, make sure that all files are available on the local system. This might require a manual spread in case you import content from a remote system. This is especially important, if you want to transfer show files to other systems later on for backup scenarios or other purposes or if you like to bundle [120] the project. For a long time the recommended workflow in Pandoras Box was to import files exclusively from the local hard disc system of the Master system. You do not need to follow this recommendation if you understand manual spreading and the file location table.

Background Knowledge: File Paths and Live Inputs

As said above, "spreading" content creates a hard copy of the file on the respective system(s). Whilst the file name itself stays the same on all system, the file path could be modified. This happens automatically and normally you do not have to worry about it. However, sometimes it is necessary to understand why different file paths are listed in the file location table. And in case you need to access the file on the remote system directly, e.g. in order to delete it, you obviously need to know, where to look for it.

No matter whether you import files from a Master or Client system, the source location will be used as a reference for remote file management. In most cases, the source and target location match, but there are exceptions. This table shows the general rules:

Source Path	Target Path
C:\Christie\content or C:\coolux\content	C:\Christie\content or C:\coolux\content
C:\different folder	C:\different folder On systems where C:\Christie\content or C:\coolux\content is a partition: ...\content_Pandoras_Box_Data\remote_content\different folder
X:\folder on external drive	C:\folder on external drive On systems where C:\Christie\content or C:\coolux\content is a partition: ...\content_Pandoras_Box_Data\remote_content\folder on external drive

Source Path	Target Path
C:\project path after saving\assets listed as "[assets]" in file location table, see Project Assets [133]	C:\Christie\content\projects\project path after saving\assets C:\coolux\content\projects\project path after saving\assets

As introduced many years ago, Pandoras Box prefers and defaults into `C:\coolux\content`. Christie hardware always mounts content drives (RAIDs) into this folder to act as a regular folder and all content data is placed into the right location. This way the OS-drive is not overloaded with content or cache files. Since Pandoras Box 6.4.0 (Q4 2019) we mount content drives into `C:\Christie\content`. However, we care for compatibility and continue to use the coolux folder if it exists already. Only when installing Pandoras Box 6.4.0 (or higher) on a brand-new system, the Christie folder will be created. Existing hardware can be used as it is and also a mixture of hardware configurations is no issue at all.

Live Inputs, i.e. video or audio input cards or devices or StreamiX feeds [724], are also listed with a directory, however this "code" is unique and different for each system and device. The chapter Input and Output Cards [1947] explains all available cards for Pandoras Box hardware. If you like to use a graphical or audio input, please drag it to the Project tab in the same way as you would do it with "normal" content. Bare in mind, that each Client will list its own live inputs and that live inputs are not shared by Pandoras Box over the network. In other words, if only your Client but not the local system offer a video card, you will not be able to preview the source on the local system, but if you assign it to a layer on the Client, it will display it. If your local system does have a video card connected to the same video signal, you can attach it to the Client's live input. The attaching feature is explained further down.

File Location Table

As depicted above, the table has the following columns. Each row lists a system (Clients or Multi-User Place) the file was spread to.

Devices
Computer name of the system loading the file

Directory + Name
File path and the file name from the file that is loaded on the system

Status
Informs about status of the file access. In case there is a problem with the file, e.g. it is inconsistent, the file and all folders in the Project tab that contain this file are marked with a red exclamation mark.

aborted:	PB tried to copy the file to the specified path but failed. Check the hard drive and try to reload the file or to remove and spread it again, respectively to import it to the project again. Another reason could be, that the spread was canceled manually, e.g. in the Task Manager [305], or because the network connection was lost.
completed:	Everything is ok, PB copied the file to the specified path and the system can access and play it.
inconsistent:	The system cannot read the file under the specified path, or it does not exist at all. The fastest solution could be to reload the file, or to remove and spread it again, respectively to import it to the project again. Attaching the consistent file is also possible. This is explained further down under "Attaching files" in the forth example. If that did not work, check whether the file exists on the system or why it was not possible to copy it. If the file does exist, check whether the system can open the file with an external software. If this works, the following reasons could cause an inconsistency in PB: - file resolution too high => check whether the license [62] supports the file resolution - unsupported file format => in case you are using external codecs, check that they are installed on all systems
0-100%	That is the progress of the file transfer as the file is being spread and is also listed in the Task Manager [305].

Media Duration
Playback duration of the file

Buttons in the File Location Table

Most buttons below the table refer to the selected entry, simply click on the system and it is highlighted in blue.

Remove
This button removes the selected entry from the table. Bare in mind, that Pandoras Box never deletes files on the hard drive. Hence, "removing" only removes the link to the file from the project. If a system is not listed in the table, it cannot play back the file in case it is used, i.e. a Master cannot show the file in its Preview and a Client cannot show it in its (full screen) render window.

Attach
This button generates a new entry for a system. In other words you can say that a system should load another file, e.g one in a lower resolution. This powerful feature is explained below in more detail.
A dialog opens where you can select the new system and according file path. Note that the file type must match, i.e. only an image can be attached to another image but not to a Live Input for example. A system should only be listed once in the table; remove the unwanted entry if necessary.

Copy
Click this button if you like to spread the file to a different location than the one chosen by Pandoras Box (see above table with source and target location). A dialog opens where you can select the remote system and your file path. The file is then copied there and the new location is automatically attached.

Spread
This spreads the file to all PB systems that are part of the project and available in the network if they do not have the file yet or if they have a different version. Note that you can use the "Spare from Spread" function which simply does not spread a file to a system. You can either say (in the Site Inspector [210]), that a specific Client should never receive any files. Or you can exclude a system from receiving files from a specific folder created in the Project tab, which is set up in the Folder Inspector [197].

Reload
This button simply reloads the selected file again. In some cases the spread or save process was not finished when PB tried to access the file for the first time. A reload then helps and marks the file as consistent. In other cases, the file was overwritten with new information and needs to be reloaded manually.

Configure (in case of Video Live Input)
The Configure button is only available for video live inputs.
If your Master is equipped with a video input board, the Configure dialog shows settings depending on the board. If you are working in a Master-Client setup and your Client is equipped with a video input board [1947] sold by Christie or another Deltacast video input board you can configure the board through the Master. Depending on the type of board, the dialog shows other settings. Currently, DisplayPort and HDMI inputs can toggle the chroma subsampling mode whilst SDI inputs have no settings. If your Client is equipped with another DirectShow board, please configure it through the Client interface [317].

Please note, that the changes made in the File location table do NOT apply in a multi-inspector. If you have more than one file selected these changes only apply to one file at time.

Attaching Files

The attach function is a very powerful tool in Pandoras Box. In a nutshell, it achieves that different files are loaded in different systems even though the Master is programmed with only one file. This is helpful for various programming tasks as seen in the following examples.

Example 1: Live Inputs
A Master and a Client are both equipped with an HDMI card for example and the signal that runs to the Client is split and is also connected to the Master. After dragging the Client's Live Input from the Assets tab to the Project tab, you can select it to see the file location table in the File Inspector. Live Inputs cannot be spread, so there is only one entry for the Client. If you click the "Attach" button you can select the local system and the according Live Input. Now, there are two entries in the table. If you now play the Live Input, you will see the Master's signal in the Preview and the Client's signal in the Client's render window. Of course, you can also do this with more systems but bear in mind that you can only attach a Live Input to another Live Input.

Example 2: Different file version, e.g. lower resolution
A Master runs on a smaller laptop and a Server client on current Server hardware. The content for the show consists of videos in 4K resolution. Whilst the Server can play these files, the Master cannot, maybe because it is

not equipped with proper hardware (graphics card, hard drive or CPU). If you now program the content with an activated Preview, the Master will not be able to perform well as it reaches its hardware limitations. For that case, Pandoras Box offers the feature to show only thumbnails in the Preview [150] instead of the full content. This thumbnail however is in a very small resolution and does not play. If you have the videos in a smaller resolution, you can copy them onto the Master's hard drive and attach them. Remember to remove the entry that points to the 4K content.
If your content is not video content but many image sequences, bare in mind, that it might not be enough to lower the resolution as image sequences play only well on SSD drives!

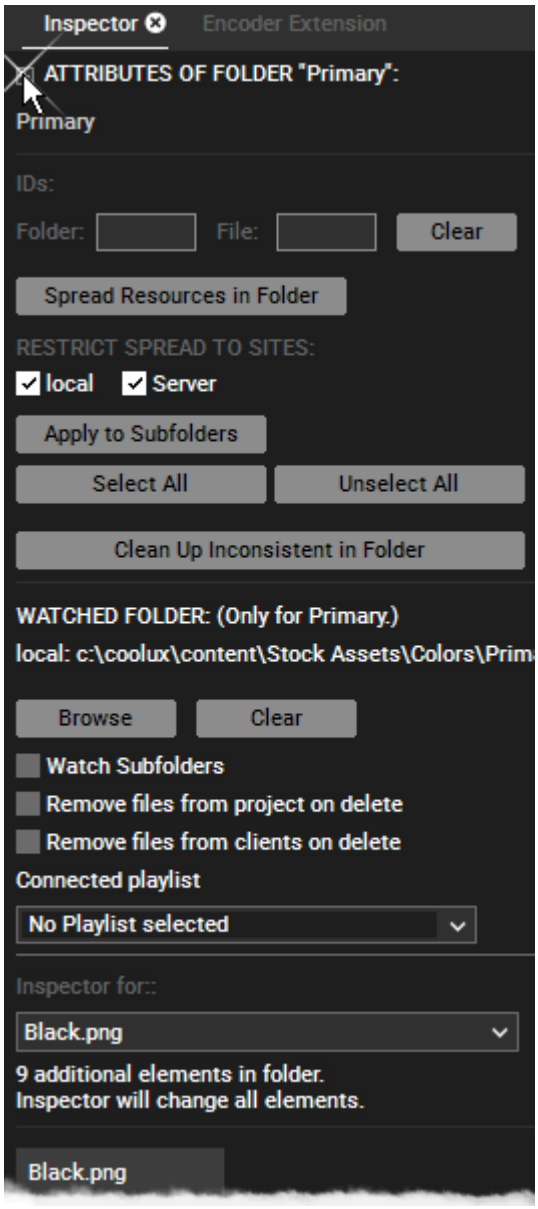
Example 3: Multi-Softedge with many Clients
A Master is connected to many Clients, e.g. Compact Players with a single output. All Compact Players project on a large screen, meaning that each Compact Player is responsible for a small cutout or slice of an image with a huge resolution. As the total resolution can not be played back on one Compact Players, either due to the license or due to the playback performance, each Compact Player has individual content. Now, you could program with all Compact Players being individual Sites in the Device Tree. However, the longer or more complex the show is, or the more changes are done, it is just a question of time until the operator makes and error and forgets to make an opacity fade for one Client or to move content. It is much easier, faster and secure to combine all Compact Players to a Virtual Site and program fades and containers only once. This normally has the disadvantage that the content is the same on all Clients, as you program only one container with one media file. But this disadvantage can be compensated if you attach the according content "slice" per each system.

You could even program a Playlist with attached files!

Example 4: Inconsistent media files
A Master is used for programming a show. Many files are imported to the project and used in containers on the timeline. Now, some files need to be moved to another directory on the hard disk or renamed. This is done with the Windows Explorer, for example. In Pandoras Box, those media files can not be played back anymore. Note that images are cached internally when the project is loaded, hence they can be still seen. When you select one of these moved media files in the project and look at the File Location Table in the File Inspector, you will see that they show up as "completed" but as soon as the "Reload" button is clicked, the file's status will turn to "inconsistent". If you moved many files, it is much faster to save, close and re-open the project again as all media files in the Project tab are checked upon Project load. Now, all inconsistent media files are marked with a red exclamation mark in the Project tab and their status in the File Location Table is "inconsistent". You could of course import all "new" files using the Assets tab but then, you need to drag them into the Sequence too, meaning that all containers need to be checked. In a complex show this is very sensitive to errors and attaching files is a better workflow.
Select an inconsistent media file to load its File Inspector. Click the "Attach" button and locate the "new" file. This adds a second entry to the File Location Table which states "completed". Now, select the entry that says "inconsistent" and click the "Remove" button. As a result, Pandoras Box now loads the new file wherever the old one was used in the Sequence.

Attaching files using the Widget Designer
The above examples show that attaching files, can be easier, can save time and can be less error-prone. However, attaching files with many Clients and file versions, can also be quite time consuming as it is much manual work. For larger shows or for scenarios were you expect that the content will change many times, it is definitely a good idea to think about using the Widget Designer for attaching and removing entries from the file location table. With smart pre-programming including the use of Folder and File IDs in Pandoras Box and Variables in Widget Designer, attaching files can be done in a millisecond. For more information, please see the Widget Designer chapters Scripting Language [1511] to learn more about the general scripting workflow, Variables [1900], Custom Script Button [822] and the command list with the commands to attach [1583] and detach [1585].

6.3.4.17.2 Folder Inspector



The Folder Inspector opens when a folder is selected in the Project tab [278]. It includes the properties of the folder itself which can be viewed when clicking the + icon as seen in the left image. In addition it supplies the File Inspector of all files that are part of the selected folder. If you do any changes in this File Inspector, they apply to all files as if they were multi-selected.

Folder and File ID
Set numeric virtual file and folder IDs for DMX [706], PB Automation [730] and Widget Designer [786] remote control and thumbnail exchange [148]. All files in this folder will get the same folder ID, the file ID will be counted up based on the given file ID. If needed you can set the maximum number to 255 in the Configuration tab > Resources (Global) [155]. In that case the folder ID will increase as well.

Clear
Press "Clear" to remove the virtual file and folder ID.

Spread Resources in Folder
This spreads the entire content of the folder to all sites according to the below options "Restrict Spread to Sites".

Restrict Spread to Sites
Per default, the check boxes for the local Master and all connected Clients are ticked, meaning that content is spread to them. In case you don't like to spread content to a Client, deactivate its check box. Note that this does not delete the content if is was already spread.
"Apply to Subfolders" copies the check box status to all subfolders within the selected folder. To (un-)select all check boxes click the according button.
Hint: If you right-click on a folder in the Project tab, you may also find the option "Spread to...".

Clean up Inconsistent in Folder
Removes all inconsistent files from the folder. Please see the File Location Table [193] for a detailed explanation about inconsistencies [194].

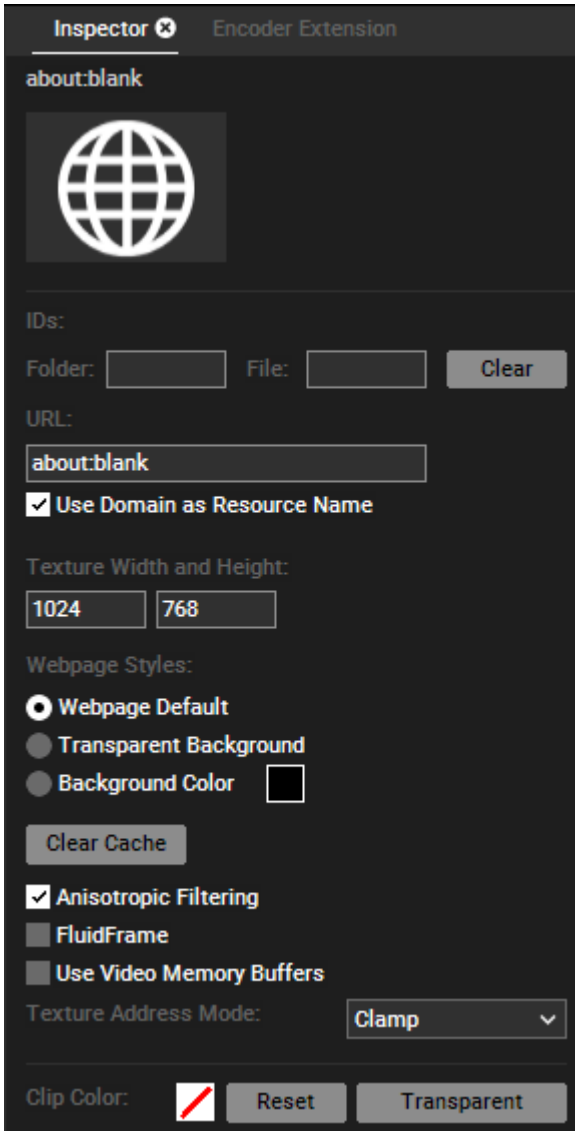
Watched Folder
First of all, make sure the option "Monitor changes to files on disk" in the Configuration tab > Resources (Global) [155] is activated (which it is by default).
Click the button "Browse" to choose a folder from the hard disk that should be watched. If you like to watch subfolders as well tick the according check box. This will automatically add all subfolders as Watched Folders.
All files that are part of the watched disk's folder will be added to this Pandoras Box project folder. All files that are afterwards added to the watched folder will automatically appear in the PB folder. If you are missing files, check whether the format can be read from Pandoras Box [90]. Files that are changed will be reloaded by Pandoras Box. Files (and sub folders) that are deleted in the watched folder will remain in the PB folder unless "Remove files from project / clients on delete" is checked.
Click the button "Clear" to delete the Watch function.

Connected Playlist
Since Pandoras Box version 6.4 you can directly link a Playlist [239] to a watched Windows folder. This way Pandoras Box automatically adds and removes files to its Playlist. Simply open the drop-down list and choose one of the Playlists that are part of your project already. If you don't have a Playlist yet, right-click in the project tab and choose "Add Playlist". More information can be found in the chapters Playlist [239] and Playlist Inspector [202].

Inspector for
Please see the File Inspector [191] for information about these options. If you do any changes in this File Inspector, they apply to all files from the folder as if they were multi-selected!

File Management
Please see the chapter File Location Table [193] for information about these options.

6.3.4.17.3 Browser Inspector



The Browser Inspector opens when a browser asset is selected in the Project tab. Please refer to the chapter "Context Menu - Project" [279] if you like to know how to create and work with a browser asset. If your displayed web page includes links you would like to click, please read the chapter about Layer Picking [252].
If you like to set up initial values for new Browsers, please refer to the Configuration tab > Resources (Global) [155].

Folder and File ID
Set numeric virtual file and folder IDs for DMX [706], PB Automation [730] and Widget Designer [786] remote control and thumbnail exchange [148].

Clear
Press "Clear" to remove the virtual file and folder ID.

URL
Enter a web site's address that the browser should open, e.g. https://www.christiepandorasbox.com/

If you like to open a web site saved on a hard disk, the syntax for the URL is: `file://host/path`.
Example for local hard disk: `file:///C:/Christie/web site/index.html`
Example for remote hard disk: `file://2.0.0.100/Christie/web site/index.html`
Note that rendering from a remote path will add delay. The space character in the folder name "web site" can also be represented with "%20". The remote folder needs to be shared via Windows so that other computers can access it.

Use Domain as Resource Name
If this option is ticked, the browser asset in the Project tab refers to the URL. In this example it is called coolux.de

Texture Width and Height
Set the size of the texture the browser renders on.

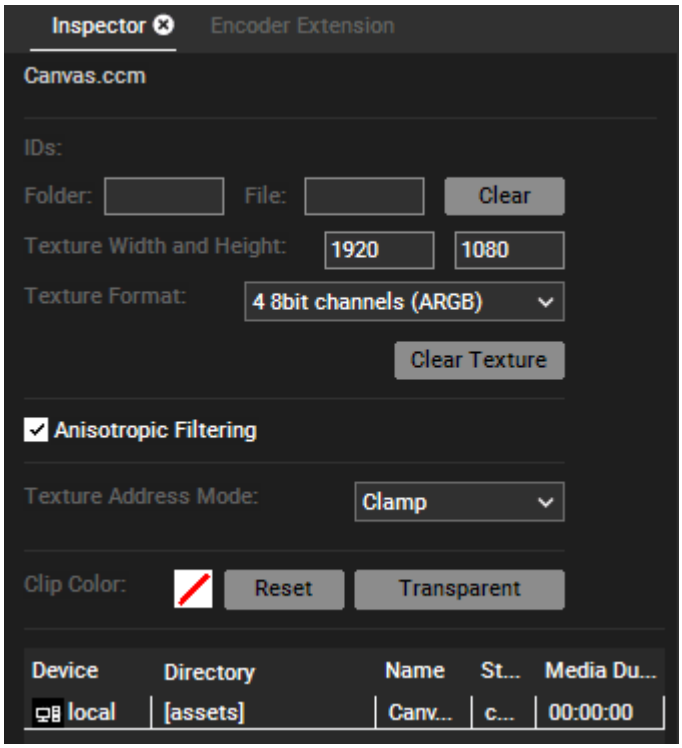
Webpage Styles
Decide how the web page's background should be displayed. You may choose to display it like a normal browser would do, to replace it with transparency or a custom color.

Clear Cache
This clears the browser cache which stores temporary information from the web site including styles and images.

Please see the File Inspector [191] for the options "Anisotropic Filtering", "FluidFrame", "Use Video Memory Buffers", "Texture Address Mode" and "Clip Color".

Additional, advanced options including font anti-aliasing for web content and FPS settings can be found in the file "tweak_config.txt" in the installation path, e.g. `C:\Program Files\Christie\Pandoras Box 8.0.0\data\config`.

6.3.4.17.4 Canvas Inspector



The Canvas Inspector opens when a Canvas is selected in the Project tab [278].

Folder and File ID
Set numeric virtual file and folder IDs for DMX [706], PB Automation [730] and Widget Designer [786] remote control and thumbnail exchange [148].

Clear
Press "Clear" to remove the virtual file and folder ID.

Texture Width and Height
Set up the size of the Canvas. This should match the size set up in the application used for drawing onto the Canvas.

Texture Format
If you like to draw on the Canvas within the Preview [256], please leave the default setting "4x8bit channels ARGB" which allows a data range of 8bit (0-255=256 values) for the alpha, red, green and blue channel for each pixel of the Canvas.
The other two options are only of interest when using the Canvas for technical, non-visual applications with the PixelWarp parameter on an Output device [686] and in combination with the automation interface which is explained in detail in the SDK chapter [730].

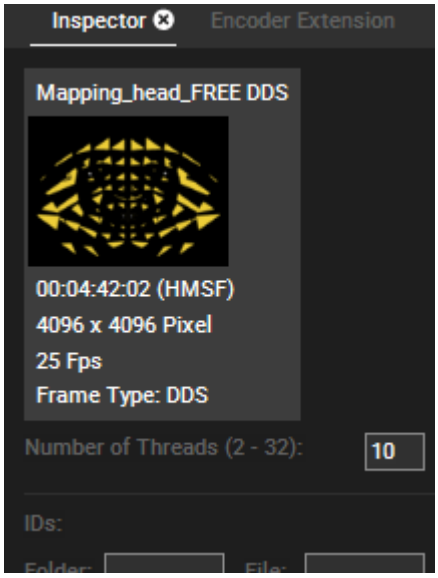
Clear Texture
This deletes all input actions from the Canvas so you can re-start with an empty one. In case you created a Canvas from an image, this command will also erase the image pixels, turning it into a black texture.

Anisotropic Filtering and Texture Address Mode
Please see the File Inspector [191].

Clip Color and File location table
Please see the File Inspector [191] and chapter File Location Table [193].

Note: To save performance a Canvas should only be spread to those Sites which display it because they also receive all Canvas updates (drawing etc.). In earlier versions there was a dedicated option in the Inspector to achieve this. Since the "Spare from Spread" option in the Device (Site) Inspector [210] and "Restrict Spread to Sites" option in the Folder Inspector [197] now includes Canvas resources too, this extra option was removed.

6.3.4.17.5 Image Sequence Inspector



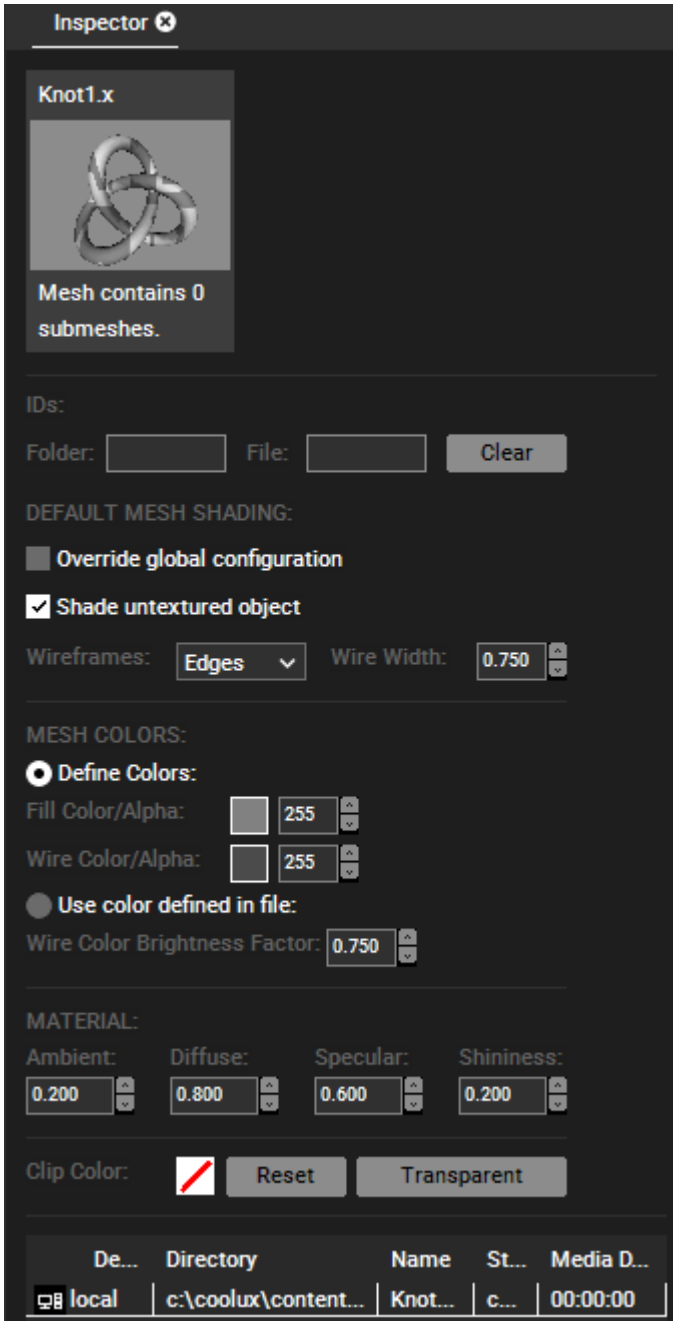
The Image Sequence Inspector opens when an image sequence [95] is selected in the Project tab [278].
Since version 5.7 it is possible to set initial values in the Configuration tab > Resources (User) [140], e.g. if you always want an activated FluidFrame option.

Number of Threads:
This alters how many threads are called by the Pandoras Box application from the operating system in order to run the sequence. The number of threads must be smaller than the total number of images in the sequence. Please read the chapter Image Sequence Formats [95] for more information.

Frame Ordering
Choose whether the image sequence should be played
- from the first to the last frame
- from the last to the first frame
- from the first to the last frame and again back the first
- from the last to the first frame and back to the last

Please see the File Inspector [191] for all other options like "Anisotropic Filtering" etc. and the topic File Location Table [193].

6.3.4.17.6 Object Inspector



The Object Inspector opens when an object (also an Editable Mesh) is selected in the Project tab [278].
Since version 5.7 it is possible to set initial values in the Configuration tab > Resources (User) [140].

Folder and File ID
Set numeric virtual file and folder IDs for DMX [706], PB Automation [730] and Widget Designer [786] remote control and thumbnail exchange [148].

Clear
Press "Clear" to remove the virtual file and folder ID.

Override global configuration
In the Configuration tab > Preview Display [145] you can set up how meshes without an assigned texture are rendered per default. Check this override option to "activate" all further options in order to render the selected mesh different from the default settings.

Shade untextured object
With an activated option, even objects without an assigned texture are visible as they can be shaded according to the below settings e.g. colored and with visible wireframes.

Wireframes
Choose whether an untextured object should be rendered without Wireframes, with Wireframe Edges or Wireframe Triangles.

Wire Width
Define the thickness of above mentioned Wireframes.

Mesh Colors
Adjust the color of untextured objects.

Define Colors
Click on the color fields to define a color that is used for the Fill Color and Wireframes. Use the number fields to adjust the transparency.

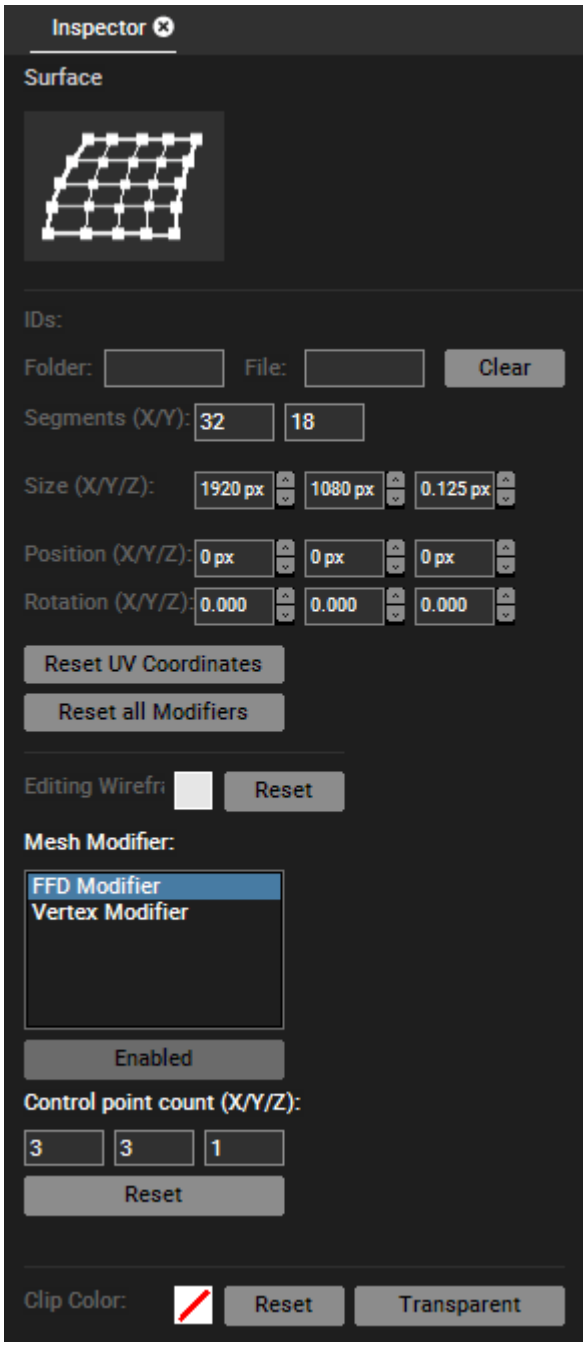
Use in file defined color
If the object was originally saved with an assigned fill and wireframe color, Pandoras Box can use them also. Use the number field to adjust the wireframe's brightness.

Material
Adjust the Ambient, Diffuse, Specular reflection levels and the shininess that influence how an untextured object reflects the global light, e.g. to appear rather glossy than dull. Further information... [527]

All settings from "Shade untextured object" to "Material" can also be set for the local Preview in the Configuration tab > Preview Display [145] > Default Mesh Shading.

Clip Color and File location table
Please see the chapter File Inspector [191] and chapter File Location Table [193].

6.3.4.17.7 Sub Mesh Inspector



The Sub Mesh Inspector opens when a Sub Mesh or Surface from an Editable Mesh is selected in the Project tab [278].

Folder and File ID
Set numeric virtual file and folder IDs for DMX [706], PB Automation [730] and Widget Designer [786] remote control and thumbnail exchange [148].

Segments
Here you can change the horizontal and vertical Segment count of a Mesh which is known as Mesh points in the Warper [2129]. The advantage of many Segments is that you can warp with more detail whilst the disadvantage is that this takes more time. If you project on a surface that is only partly complicated whilst other parts are more plane and would allow fewer Segments, the Planar or Perspective UV Mapping Mode [269] might offer an alternative solution as it allows to assign a UV map from one source to multiple meshes.

Note that you should adjust the Segment count **before** warping as any change will erase

Size, Position and Rotation
All these settings influence the mesh before it is even assigned to a Layer. Enter pixel values for "Size" and "Position" and degrees values for "Rotation". Note that working in the Sub Mesh Mode within the Mesh Editing Mode [257] in the Preview changes these values as well!

Reset UV Coordinates
This resets the UV Coordinates and applies the default UV map. You can change the UV map with the Planar or Perspective UV Mapping Mode [269] from the Preview.

Reset all Modifiers
This resets the FFD and Vertex Modifier in case you have moved FFD or Vertex points using the according modifier in the Mesh Editing Mode [257] from the Preview.

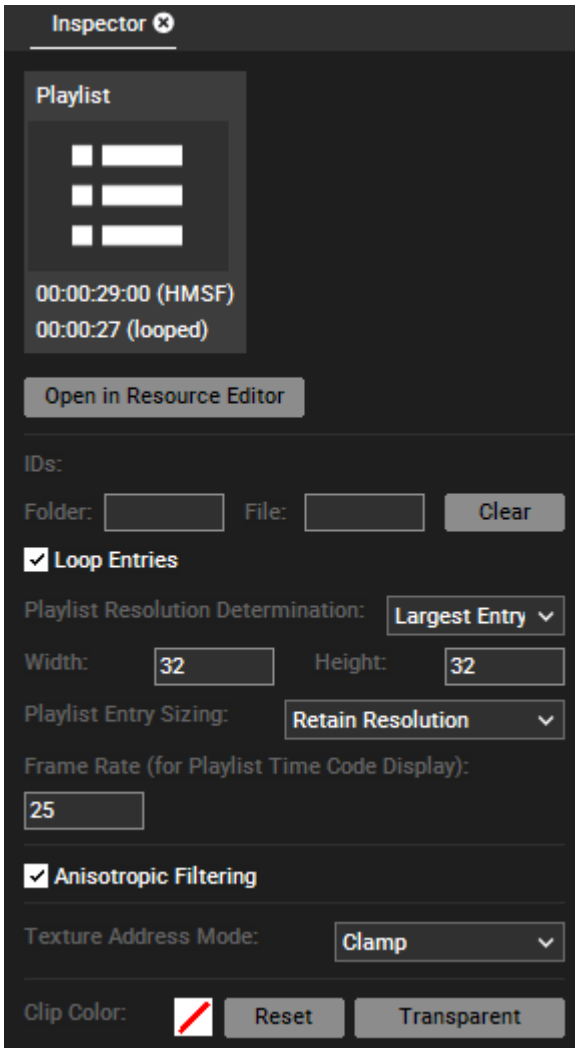
Editing Wireframe Color
If you click on the colored box, a dialog opens were you can choose another color for the mesh which is visible in the Mesh Editing Mode when the FFD or Vertex Modifier is selected. This is especially useful if you need to overlap several meshes. Coloring one red, and the other green for example will result in yellow lines if both meshes overlap (partly).
The Reset button assigns a light gray color again.

Mesh Modifier
You can select either the FFD or the Vertex Modifier and click the below buttons.
The "Enabled" button toggles the selected modifier off and on again.
The "Reset" button resets all points of the selected modifier to the initial default position.

If you select the FFD Modifier, you can change the "Control point count" below the list. The according count for the Mesh "Segments" can be found at the top of the Inspector.

Clip Color
Please see the File Inspector. [191]

6.3.4.17.8 Playlist Inspector



The Playlist Inspector opens when a Playlist is selected in the Project tab [278].
Please note that you can set up initial values for new Playlists, in the Configuration tab > Resources (User) [140].

First of all the Inspector depicts the duration. The first time refers to the duration if the playlist is played back once (here: 29 sec) and the other if the playlist loops (here: 27 sec). In this example the cross-fade time from the last to first clip is set to 2 seconds.

Click the "Show in Resource Editor" to display the playlist and its content in the Playlist tab [239].

Folder and File ID
See the File Inspector [191].

Loop Entries
This option lets the Playlist play endlessly. When the last entry is reached, it cross-fades to the first again.

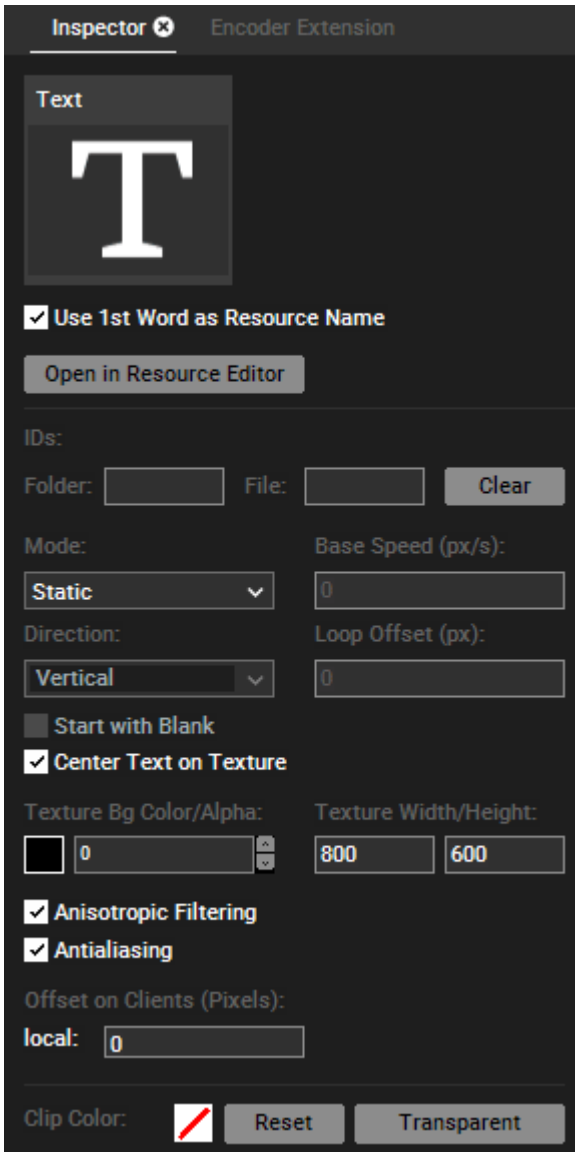
Playlist Resolution Determination
Choose the texture size each content file should be rendered on. You can for example choose the size from the "Largest Entry" or enter a custom width and height after choosing "User Defined" in the drop-down list.

Playlist Entry Sizing
Choose the aspect ratio for the texture each content file is rendered on. "Retain resolution" renders the playlist's content in 1:1 mode, having the advantage of not possibly stretching the content but the disadvantage of not filling the fullscreen and allowing different sizes.
All other option like "Fit Horizontally" or "Stretch Fullsize" possibly alter the content's size. Please see the Video Layer Inspector [211] for an illustrated explanation for the different Fit-options.

Framerate
Choose a framerate for the playback of the playlist, this is only for calculating the duration. The playlist's content files are not converted to this framerate.

Please see the File Inspector [191] for "Anisotropic Filtering", "Texture Address Mode" and "Clip Color".

6.3.4.17.9 Text Inspector



Create your Text Input by right-clicking on a folder in the Project tab [278] > Add Text Input. Select it to see its properties in the Text Inspector.

Use 1st Word as Resource Name
This names the Text input in the Project tab with the first word entered in the Text Input Editor [307].

Open in Resource Editor
In the Text Input Editor you may set up all options regarding the content of the selected Text Input, e.g. the text itself. Some of the options from the Editor are available in the Text Inspector too and apply to all (multi-) selected Text Inputs. The Editor also allows to transfer style properties to other Text Inputs. Please note, that .net framework 4.0 is needed to display this tab.

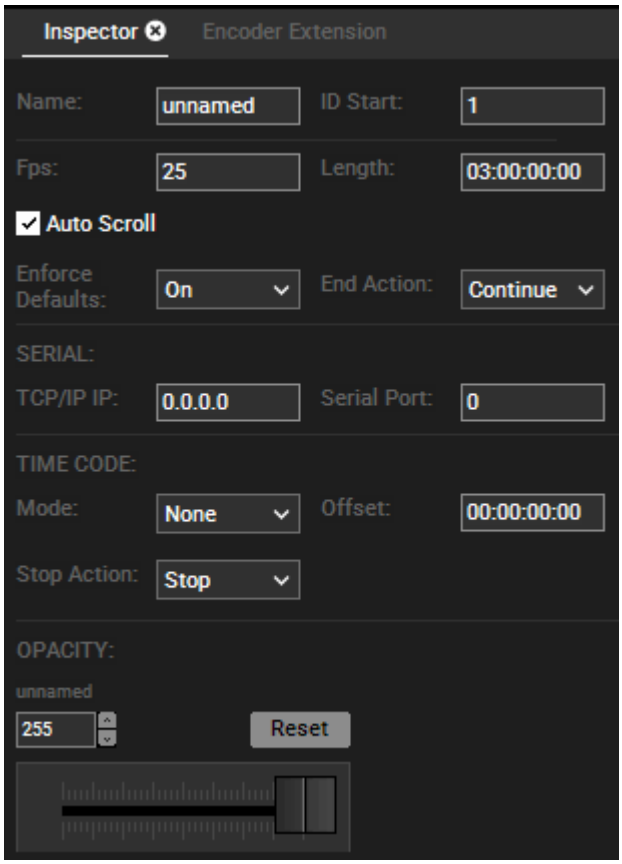
Folder and File ID
See the File Inspector [191].

Mode /.../ Texture Width/Height
Please see the Text Input Editor [307] for all options from "Mode" to "Texture Width/Height".

Offset on Clients
Enter a pixel value to offset the Text Input for the according Client. This is especially useful when rendering text that spans across multiple outputs from different Clients.

Please see the File Inspector [191] for "Anisotropic Filtering", "Antialiasing" and "Clip Color".

6.3.4.17.10 Sequence Inspector



The Sequence Inspector opens when a Sequence is selected in the Project tab [278].

Name
Enter an internal name for the Sequence that is for example shown in the Project [278], Sequence [292] or Sequence Control [303] tab.

ID Start
Enter an ID which is for example used for DMX [706], PB Automation [730] and Widget Designer [786] remote control.

Fps
This changes the displayed time resolution in the Sequence tab [292], i.e. how many frames are available per second. Please note that this does not effect the frame rate used for rendering files! Pandoras Box renders each file according to its own frame rate, the final results seen on a display device depends on the refresh rate [1949] set up in the graphics card driver.

Length
Default is three hours. You can manually set a new length e.g. 30 minutes by entering 00:30:00:00 or just 300000. Apply the new length with pushing Enter!

Auto Scroll
If Auto Scroll is on (by default), the timeline will always show the area around the nowpointer when it is set to play. With the Auto Scroll option unchecked, you will be able to scroll to any point of the timeline while the nowpointer is running.

Enforce Defaults
You can choose between three states: On, Off and Only Mesh/Media
Please see the chapter Enforcing Default Values [300] for a detailed description!

End Action
Once the sequence reaches the end of its length, you may choose a specific action:
The sequence stops and jumps to first frame, it will pause or it will continue playing.

Serial
These settings apply when a Serial Link [2044] or TCP/IP device is used to remote control the sequence. The supported commands are listed in the chapter explaining the TCP/IP input protocol [720].
[Pandoras Box] <== TCP/IP ==> [Serial Link] <== Serial ==> [Serial Device sending commands to control the PB sequence] or
[Pandoras Box] <== TCP/IP ==> [any device set up with an TCP Server that sends commands to control the PB sequence]

TCP IP / IP
This refers to the Sender's IP address.
Please note, that the used TCP/IP port from Pandoras Box is always 23 and can not be configured.

Serial Port
In case you work with a Serial Link, this refers to the serial port on the Serial Link where your serial device is attached to.
In case you work with another TCP device, set this port to "1" as this is a virtual port that is also part of the command sent to PB.

Time Code
These settings apply when a SMPTE Link [2000] is used and set up in the Configuration tab > SMPTE Time Code [152].

Mode
NONE - this sequence will not be effected by SMPTE timecode
SEND - this sequence will send its timecode via the SMPTE link device. Please note that if a sequence in your project is set to Send-Mode, all other sequences will be set into None-Mode.
RECEIVE - the sequence will be controlled via the incoming SMPTE timecode.

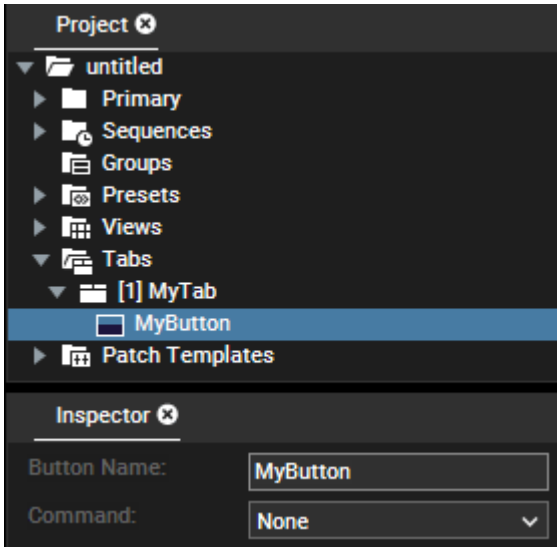
Offset
If needed enter a timecode offset here.

Stop Action
Choose which action should be executed if incoming time code stops. The timeline play mode can be set to Stop, Pause or Continue.

Opacity
This fader works as a master fader for the opacity and volume of the sequence (and all the layers that have clip containers in this sequence).
The sequence opacity works as a value multiplier. If you have the sequence fader at a 50% position (value 127) all containers or layers will be shown with an opacity that is 50% of their stored or active opacity value. If one layer has a current active opacity value of 200, it will be shown with 100. In the same way, another layer that has a stored opacity key will reduce its opacity by one half.
This also means, if a container has no opacity key or value i.e. it is only visible because the default opacity [156] was set to full, it is not influenced by the sequence fader! This way you can include or exclude layers from the sequence "group".

In the same manner the volume parameter is influenced.

6.3.4.17.11 Tab Button Inspector

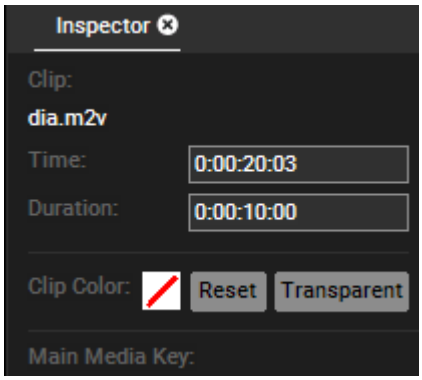


The Tab Button Inspector opens when a Button from a Tab folder [288] is selected in the Project tab [278].

Button Name
Change button name and text

Command
Choose a command from the drop-down list, e.g. "Paste to Selected Devices".

6.3.4.17.12 Clip Summary Inspector



The Clip Summary Inspector opens when a container is selected in the Sequence tab [292].

The container is highlighted with a blue outline.

Clip
This is the name of the clip. Right-click the media file in the Project tab [278] if you like to rename it.

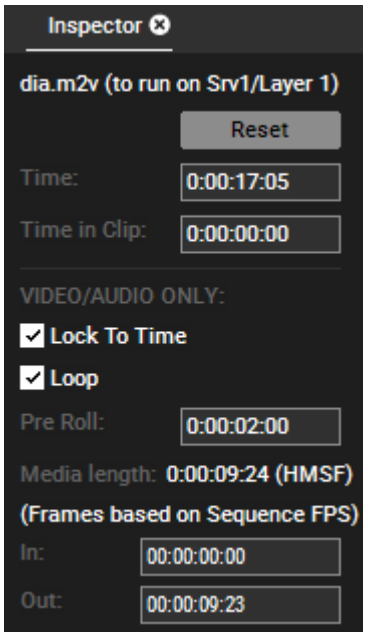
Time
To change the start timecode of the clip, enter the new timecode using the following syntax: enter e.g. 0:00:06:09 or just 609 to get the new time 0:00:06:09.
To just add e.g. two seconds to the current time, enter "++200", to subtract e.g. two seconds, enter "--200".

Duration
To enlarge or to reduce the duration of the clip, enter the new duration using e.g. 0:00:12:00 or just 1200 to get the new duration of 12 seconds.
To just add e.g. two seconds to the current duration, enter "++200", to subtract e.g. two seconds, enter "--200".

Clip Color
The Clip Color is the color the container is filled with. Per default it is gray. In case you set it to "Transparent" it will be of the same color the Layer is set to (see Layer Color [211]). By clicking into the color field a dialog opens where you can choose any other color for the container.
Note that in case you have chosen Clip Colors in File Inspectors [191], they are overwritten with individual container colors. If you like to reset the container to be colored according to the File Inspector, click "Reset".

All following settings named "Main Media Key" and "Video / Audio only" are a copy from the Clip Inspector [206]. They refer to the media assigned as the main media to the layer. Note, that some effects have their own media clip. Open the FX track and select this media to see its properties and adjust for example a Pre Roll time.

6.3.4.17.13 Clip Inspector



The Clip Inspector opens when a clip within a container is selected in the Sequence tab [292].

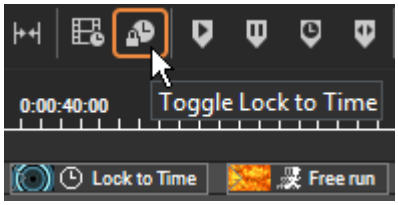
The media track is highlighted in blue. Note, that some effects have their own media clip. Open the FX track and select this media to see its properties and adjust for example a Pre Roll time.
When you select a mesh or object, you will see less options. When selecting an audio clip there is one additional setting "Time Offset Millisecs" that allows to offset the audio content.
The Clip Inspector is also part of the "Clip Summary Inspector" which opens when a container is selected in the Sequence tab. In that case the same settings are listed below the title "Main Media Key".

Reset
This removes the content from the Container and places a "None" key instead.

Time
To change the timecode of the start timecode clip, enter the new timecode using the following syntax: enter e. g. 0:00:06:09 or just 609 to get the new time 0:00:06:09.
To just add e. g. two seconds to the current time, enter "++200", to subtract e. g. two seconds, enter "--200".
Note that the container itself stays on the same time. So by shifting the clip, you just start with another frame when entering the container. When you shift the left container border to the right, i.e. you crop the container, the clip stays at the original time which has the same result as shifting the clip to the left.
Empty frames are not possible, thus you can only shift the clip to the left, meaning that you start with a later frame. The "Time in Clip" adjust according to your entry.

Time in Clip
Whilst the "Time" expressed the start of the clip regarding the absolute timecode, the Time in Clip expresses the time relative to the clip beginning. So by default, this value is 00:00:00:00 as the clip starts at the same time as the container. If you crop the clip container on its left side, this value will get negative and shows you the starting time of your clip. You can change this value back to 00:00:00:00 to let the video inside the clip start from its beginning. Note that the "In" parameter also effects which frame of the clip is seen when entering the container.

Lock to Time
There are two different modes how a container clip is played back.To switch the mode, you can either use the "Lock To Time" option in the Inspector or the toggle button in the button bar [293] displayed above the Sequence. The mode is depicted by a small icon in the container.



Per default, "Lock To Time" is enabled. The clip icon is a clock. In this mode, the clip synchronizes to the timeline. If the nowpointer is paused inside the container, the clip playback is also paused. If you have programmed video playback keys [649], the sequence playback overrides them.

If you uncheck the option, the container is in "Free-run" mode. The clip icon is a running-stick-figure. In this mode, the clip playback does not depend on the sequence playback mode. As soon as the nowpointer enters the container, its video playmode keys (play once, play loop, pause, stop) take effect.

Loop
This option is only available with the Lock to Time option.
Per default, the Loop option is enabled which means that a clip starts again from the beginning in case its container was extended manually and is now longer than the duration of the clip itself.
But imagine, you have a video on the timeline and after playing it back you like to keep the last frame as a still image on the screen. First, you increase the length of the container in the sequence, then you select the container and untick the Loop check box. Now, the last frame is hold.

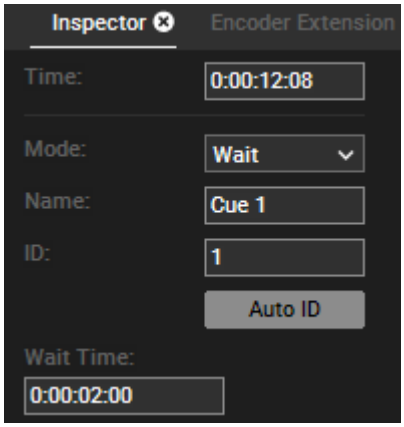
Pre Roll
Since version 6.4.3, the default value for pre rolling content is 2 seconds. If the Pre Roll time is 0, the media clip is loaded into RAM as soon as the nowpointer enters the container. The higher the resolution of the media is, or the more containers start at the same time, the longer the loading process takes. To display the media faster, Pandoras Box can start pre-loading it before the container actually starts. The Pre Roll Time defines the starting point ahead of the container. The Pre Roll Time can overlap with a preceding container.

In, Out
When the nowpointer enters the clip, it will start/stop/loop at the given In- and Outpoint. These Playback parameters [649] can be adjusted in the Device Control tab or by entering the exact frame count in this Inspector.
You can also enter a timecode value:enter e.g. 0:00:06:09 or just 609 to get the new time 0:00:06:09.
If you like to loop within the In- and Outpoint you can adjust the duration from the container easily: right-click on the container > "Set duration to Main Media In/Out"
Delete the values in the Inspector to get the original value.
Please note that in addition to the In- and Outpoint you can start with a later frame by using "Time in Clip" as explained above.

Time Offset [Milliseconds]
This option is available for audio content. Enter a time in milliseconds to offset the audio content. Audio delay settings are available per Resource, Layer or on Clip Level. That enables you to set up different delays for different output signal chains.

6.3.4.17.14 Cue Inspector

The Cue Inspector opens when a Cue is selected in the Time Bar above the Sequence tracks:



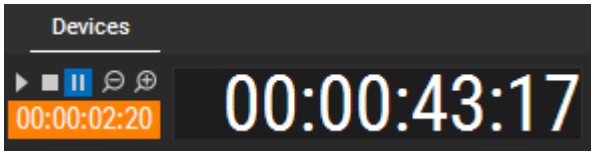
Time
To change the start timecode of the clip, enter the new timecode using the following syntax: enter e.g. 0:00:06:09 or just 609 to get the new time 0:00:06:09.
To just add e.g. two seconds to the current time, enter "++200", to subtract e.g. two seconds, enter "--200".

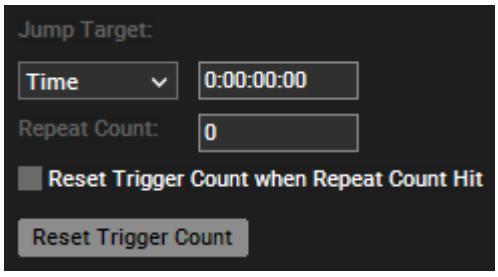
Mode
Available cue mode settings are:
- Play: the timeline play status will stay the same when reaching the Cue (Timeline is playing = it continues playing; Timeline is paused = it continues pausing)
- Pause: the timeline play status will change to "Pause" when reaching the Cue
- Stop: the timeline play status will change to "Stop" when reaching the Cue. This means that the nowpointer stops and then jumps to the timecode 0:00:00:00
- Jump: when reaching the Cue the nowpointer will jump to the Jump Target entered and the timeline keeps its previous play status
- Wait: the nowpointer will wait at the Cue for the Wait Time entered and then keeps on playing if it was in play mode before.

Name
By default the cue is named with its Cue ID as shown in the tool tip on top of a cue. A new cue name may be entered here.

ID
The numeric cue ID is increased for each new cue by default, this function may be deactivated in configuration tab [139]. To change a cue ID enter a new one here.

For Wait Cues the following option is available:
Wait Time
Enter here the time the nowpointer should wait on this cue before continuing to play. The remaining wait time is displayed and highlighted in orange in the devices tab below the play, stop and pause buttons.





For Jump Cues the following options are available:
Jump Target
Enter here the jump target timecode if cue mode is set to "Jump", using the following syntax: enter e.g. 0:00:15:09 or just 1509 to get the new time 0:00:15:09.
To just add e.g. two seconds to the current time, enter "++200", to subtract e.g. two seconds, enter "--200".

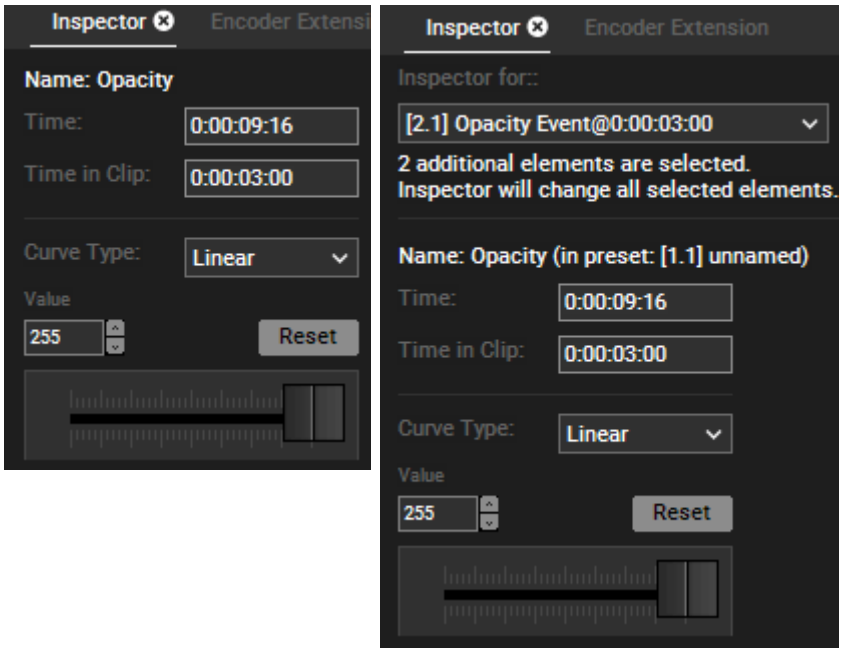
Repeat Count
Enter here the amount of iterations if cue mode is set to "Jump", 0=endless loop. If repeat count is set to a number >=0, the jump cue will be executed this amount of times and then will be ignored to infinity. The amount of remaining repeats are displayed in front of the Cue Name. When the timeline is stopped (not paused) the repeat count will always be reset to the value entered here. To reset the Trigger Count automatically after the last repeat, please check the option [Reset Trigger Count when Repeat Count Hit].

Reset Trigger Count when Repeat Count Hit
When this option is checked, the Trigger Count for the Jump Repeat will be reset automatically after the last repeat.

Reset Trigger Count
By pressing this button the Trigger Count will be reset manually to the value entered as Repeat Count.

6.3.4.17.15 Parameter Keyframe Inspector

The Parameter Keyframe Inspector allows you to set up the following:



The left image shows the standard inspector for a keyframe.
The right image shows the multi-inspector of three keyframes that belong to a preset [283].

Note the additional drop-down menu where you can choose the keyframe the below information should refer to.
The keyframe's name in the menu consists of:
[2.1] - the site and layer ID it belongs to,
Opacity Event - the parameter it belongs to, and
@0:00:00:03:00 - the time it belongs to.

Underneath the drop-down menu you see the information that two more preset keyframes are currently selected.
Under the horizontal line the standard inspector lists all customizable control and property settings.

Name
Informs you about the parameter's name the selected keyframe belongs to. In case the selected key belongs to a preset, the preset's ID and name are listed as well.

Time
To change the timecode of the key, enter the new timecode using the following syntax: enter eg. 0:00:06:09 or just 609 to get the new time 0:00:06:09.
To just add eg. two seconds to the current time, enter "++200", to subtract eg. two seconds, enter "--200".

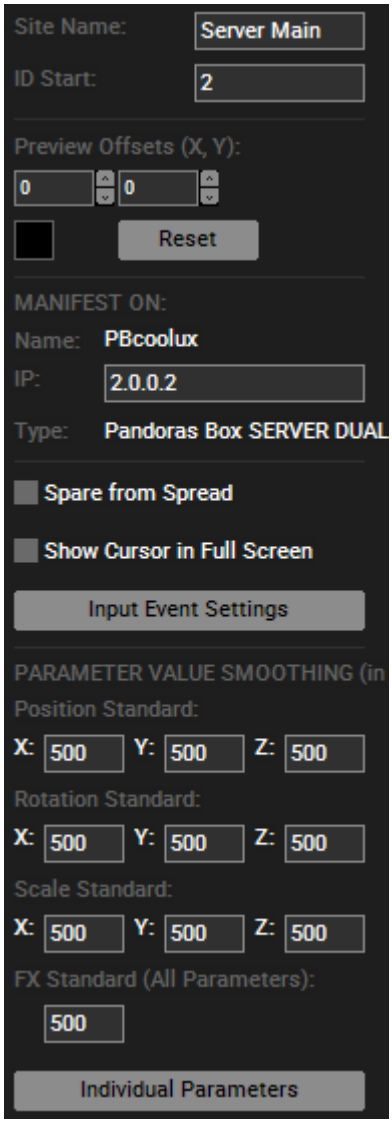
Time in Clip
To change the Time of the key inside the clip, enter the new time using the following syntax: enter eg. 0:00:06:09 or just 609 to get the new time 0:00:06:09.
To just add eg. two seconds to the current time, enter "++200", to subtract eg. two seconds, enter "--200".

Curve Type
You can choose between a Linear (Standard), Bezier, Bezier Corner, Constant and Stop curve type. To edit the curves please open the Curve Editor [169]. The Curve Editor [169] topic also illustrates and explains the different types.

Value
To change the key's value, enter the new value or move the fader.

6.3.4.17.16 Device Inspector

The Device Inspector opens when a site (node) is selected in the Device Tree tab [173].



Site Name
Change site/node name here

ID Start
Set up the numeric start ID

Preview Offset (X,Y)
Setting up the Preview [243] Offsets: apply an X and Y offset for the global cam by typing in the value or by clicking in the black box and moving the mouse cursor up and down. Press the Reset button to set the Preview Offsets back to default.

IP
Edit the device IP address this site should refer to. The "Name" and "Type" inform you about the computer network name and license type of the site.

Spare from Spread
Disables content to be spread to this device.

Show Cursor in Full Screen
Enables to see the mouse cursor in the Client´s fullscreen window. This is useful when working with Layer Picking in the Preview [252] or with Clients. Instead of this option you may as well use a Pointer Layer [665], giving you more options to influence the look of the mouse cursor.

Input Events Settings
The pop-up dialog offers settings regarding the feature "Layer Picking". In short, you can route input events like a mouse click to Pandoras Box' layers, for example to execute links in a Browser Asset [279].
The first option "Show Cursor in Full Screen" conforms with the above described option.
If you like to use Layer Picking on an output (instead of only in the preview tab and the Global Cam), the left setting "Enable device" must be activated, either for Output 1 or 2 or both.
The settings to the right routes the Layer Picking data to either a Widget Designer Device [696] and, or to the layers.
For more detailed information and a step-by-step example, please refer to the chapter about Layer Picking [252].

The Output Inspector [220] offers the same input event settings, so you can access them there too.
To see the Input Event Settings for all systems in the Device Tree combined in one dialog, right-click in the Preview tab and choose "Show Input Events" there.

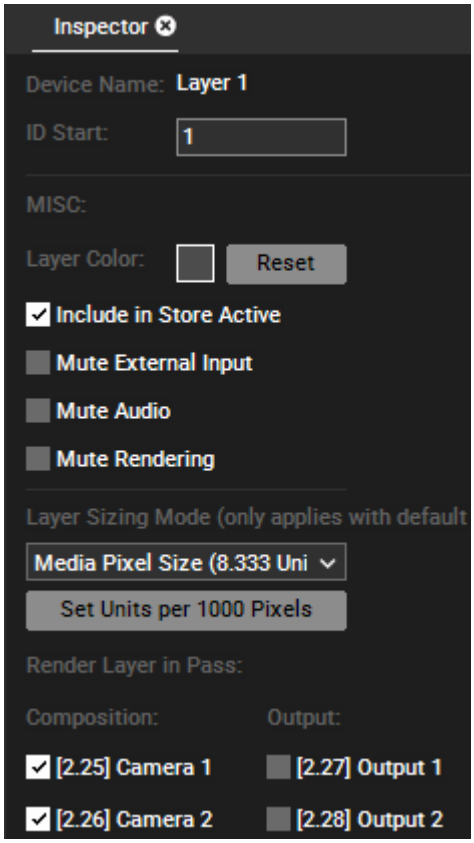
Parameter Value Smoothing
These settings only have an effect for all incoming data, like from the fader of the device control, DMX, data from the Widget Designer, Art-Net etc. The timeline behavior will not be influenced by this.
Parameter Value Smoothing defines the time in milliseconds the object will need to reach the given parameter value. In other words, active values are not applied instantly or abruptly (0ms) to the parameter but within the smoothing time in order to smooth movement.
The default value of 500ms applies to all Translation, Rotation, Scaling and Effect parameters. You can enter a minimum of 0ms and a maximum of 65535ms.

The button "Individual Parameters" opens a dialog where you can set the smoothing time for almost every parameter separately for each layer. For tracking scenarios you might need to decrease the time frame for position changes etc. and for color changes you might want to deactivate the smoothing entirely.

The button "Open Engine Configuration" gives access to hardware related engine settings especially influencing rendering performance. Please refer to the chapter Configuration tab > Render Engine [162].

6.3.4.17.17 Video Layer Inspector

The Layer Inspector allows you to set up the following:



ID Start
Numeric start ID

Layer Color
Per default each layer in the Sequence tab is displayed in gray. Clicking into the gray field opens a color dialog where you can choose a different color. Note that you can also color a container [192].

Include in Store Active
Uncheck this option to exclude the layer from all Store Active commands.

Mute External Input
Tick this check box to block this layer from any external input, like Art-Net, Input Data from Widget Designer etc.

Mute Audio
Tick this check box to mute the audio content of the Layer. Note that the Mute icon in the Device Parameters tab is automatically clicked. Also, the according icon in the Device Tree [177] changes automatically.

Mute Rendering
Tick this check box to mute the audio content and stop rendering the video content of the Layer. Note that the Mute icon in the Device Parameters tab is automatically clicked. Also, the according icon in the Device Tree [177] changes automatically. "Muted" Layers are not part of the rendering process anymore. If you simply set the opacity for a Layer to 0, it is part of the rendering but just not visible.

Layer Sizing Mode
Choose from the drop-down list how this layer should handle the aspect ratio of all media files assigned to it. See below for examples and a detailed explanation.

Render Layer in Pass
The check boxes allow to toggle the Layer's visibility in the available Composition and Output Devices. Per default, the layer is included in the composition pass as it is visible for all available Cameras - it will be part of the so called render target and then be send through the entire output pass. It will be affected by its settings e.g. warp object, keystone or softedge etc.
If you change the default behavior, e.g. uncheck the Camera option(s) and check the Output option(s), the layer is only part of the output pass. It will not be seen by the Camera as it is sent directly to the according Output. In return it will not be affected by the output settings (except perspective settings) and can be placed outside the warped / softedged area. This is especially interesting when working with multi-softedged projection(s) and blacklevel compensation [646] is needed. Find more information in the topic Video Processing Pipeline [320].

The check boxes depend on the number of Cameras / Outputs. A Server with two Outputs, for example, always has two Cameras too and you may decide whether a layer should be only rendered by one of them. A possible application is working with different outputs for a key & fill scenario.

Layer Sizing Mode

By default the Layer Aspect Ratio is set to "Media Pixel Size" meaning that each media file is displayed with its native aspect ratio and size. In other words, it is not stretched nor scaled. Note the "Media Pixel Size" is a dynamic mode that depends on the translation factor set up in the Configuration tab > Unit Management [160]. For example, if you have an output with a width resolution of 1920px (and a height resolution of 1080px) the according translation factor is 8.333. Now, the Layer Sizing Mode is based on 8.333 as well. If a 1920x1080px testpattern is assigned to a Layer, it will be displayed fullscreen. A 1024x768px texture will be displayed accordingly smaller. In case the Layers of a second Client with a 1024x768px output are set to "Media Pixel Size" as well, the same texture is fullscreen whilst the first testpattern would be cut off.

From the other modes, the first three scale the media files to fit them into the screen width or height keeping the file's aspect ratio. The others stretch the media files to fit a certain aspect ratio.

To keep it simple, the description refers to the "output size". In fact, the scaling depends on the aspect ratio of a Camera device which can be set up in the Camera Inspector [218]. Per default, the Camera's aspect is linked to the Output's aspect which again depends on the resolution of a connected display.

Important note in case you are using outputs with different resolutions. Note that all scaling modes apply to each Site individually but not individually for each Camera. For example, if you have two displays with a different resolution connected to two Servers, both displays will show a Layer with the setting "Horizontal Fit" as large as their own screen width. However, if you connect both displays to a single Server, the result will be different. This is because the scaling happens once per Site and depends on the Camera with the larger aspect ratio.

This is the original media file with a resolution of 1000px*1000px. All examples below show the image with an X-,Y-scaling of 1.0. Likewise, the scaling mode only applies if no Mesh file is assigned to the layer.

Media Pixel Size
When the layer sizing mode is set to "Media Pixel Size", the media file is displayed with its native aspect ratio and size. It is not stretched nor scaled.
The first example, shows an output resolution of 1920px * 1080px. The image size is 1000px so there is a 40px border at the top and bottom for example.
The second image shows an output resolution of 4096px * 2160px.

As this example shows, keep in mind that displays with different resolutions do not look the same because displaying a media file with its native resolution takes up more or less space per output!
See above for an explanation of the unit calculation.

Fit larger side
When the layer sizing mode is set to "Fit larger side", the media file is displayed with its native aspect ratio but it is scaled until it fits one screen side. Depending on the resolution of the image and output, this is either the screen width or height. In this example, the image is scaled to fit the height which has the same result as the setting "Vertical fit".

Horizontal fit
When the layer sizing mode is set to "Horizontal fit", the media file is displayed with its native aspect ratio but it is scaled to fit the screen width.

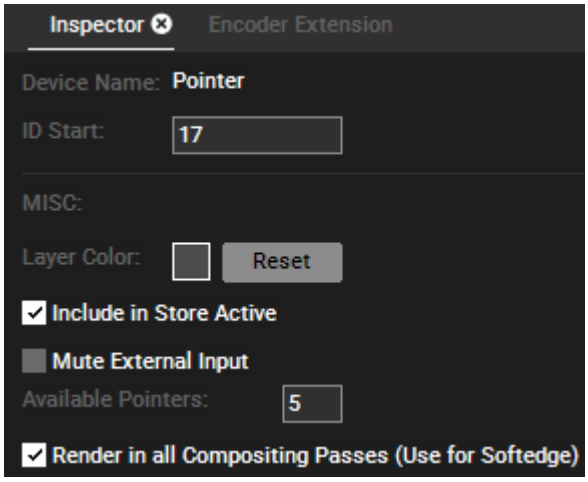
Vertical fit
When the layer sizing mode is set to "Vertical fit", the media file is displayed with its native aspect ratio but it is scaled to fit the output's height.

5:4 - 4:3 - 16:10 - 16:9 - 1.85:1
When the layer sizing mode is set to one of the fixed ratios, the media file is stretched and also scaled to fit the screen width (which is always 16 units).

1:1
When the layer sizing mode is set to one "1:1", the media file is stretched to be 1 unit wide and 1 unit high, whereas the screen width is always 16 units.
This mode is for example of interest when working with generic units and not based on pixels. Simply set the media file to 1x1unit and then assign your individual X- and Y-scaling.

6.3.4.17.18 Pointer Inspector

The Pointer Inspector opens when a Pointer Layer [665] (used for Layer Picking [252])is selected in the Device Tree tab [173]. More information about the Pointer Layer can be found in the Pointer Layer chapter [665].



ID Start
Numeric start ID

Layer Color
Per default each layer in the Sequence tab is displayed in gray. Clicking into the gray field opens a color dialog where you can choose a different color. Note that you can also color a container [192].

Include in Store Active
Uncheck this option to exclude the layer from the Store Active command.

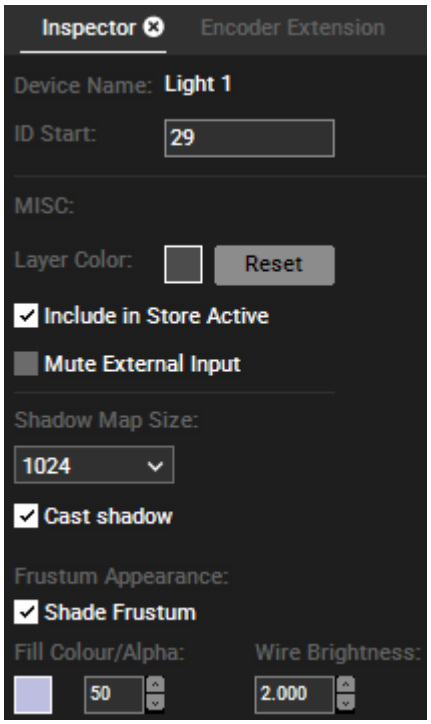
Mute External Input
Tick this check box to block this layer from any external input, like Art-Net, Input Data from Widget Designer etc.

Available Pointers
Choose how many inputs should be substituted with the Pointer Layer. If you are working with an Windows XP system only one input is possible, whereas Windows 7 and above is able to work with multi-touch events.

Render in all Compositing Passes (Use for Softedge)
Per default, the Pointer Layer is now included in the Composition pass. Hence, it is part of the so called render target and influenced by any Output settings, like warp object, keystone or softedge etc. Find more information in the topic Video Processing Pipeline [320].

6.3.4.17.19 Light Layer Inspector

The Light Layer Inspector allows you to set up the following. More information about the Light Layer can be found in the Light Layer chapter [666].



ID Start
Numeric start ID

Layer Color
Per default each layer in the Sequence tab is displayed in gray. Clicking into the gray field opens a color dialog where you can choose a different color. Note that you can also color a container [192].

Include in Store Active
Uncheck this option to exclude the layer from the Store Active command.

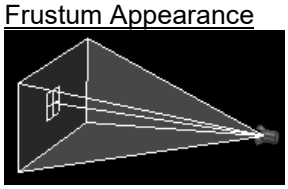
Mute External Input
Tick this check box to block this layer from any external input, like Art-Net, Input Data from Widget Designer etc.

Shadow Map Size
Choose from the drop-down list how high the resolution from the light texture should be. By default the Shadow Map Size is set to 1024*1024 px.

The example below shows the difference between 256 - 512 - 1024 - 2048 - 4096. Keep in mind that a higher resolution uses more performance.

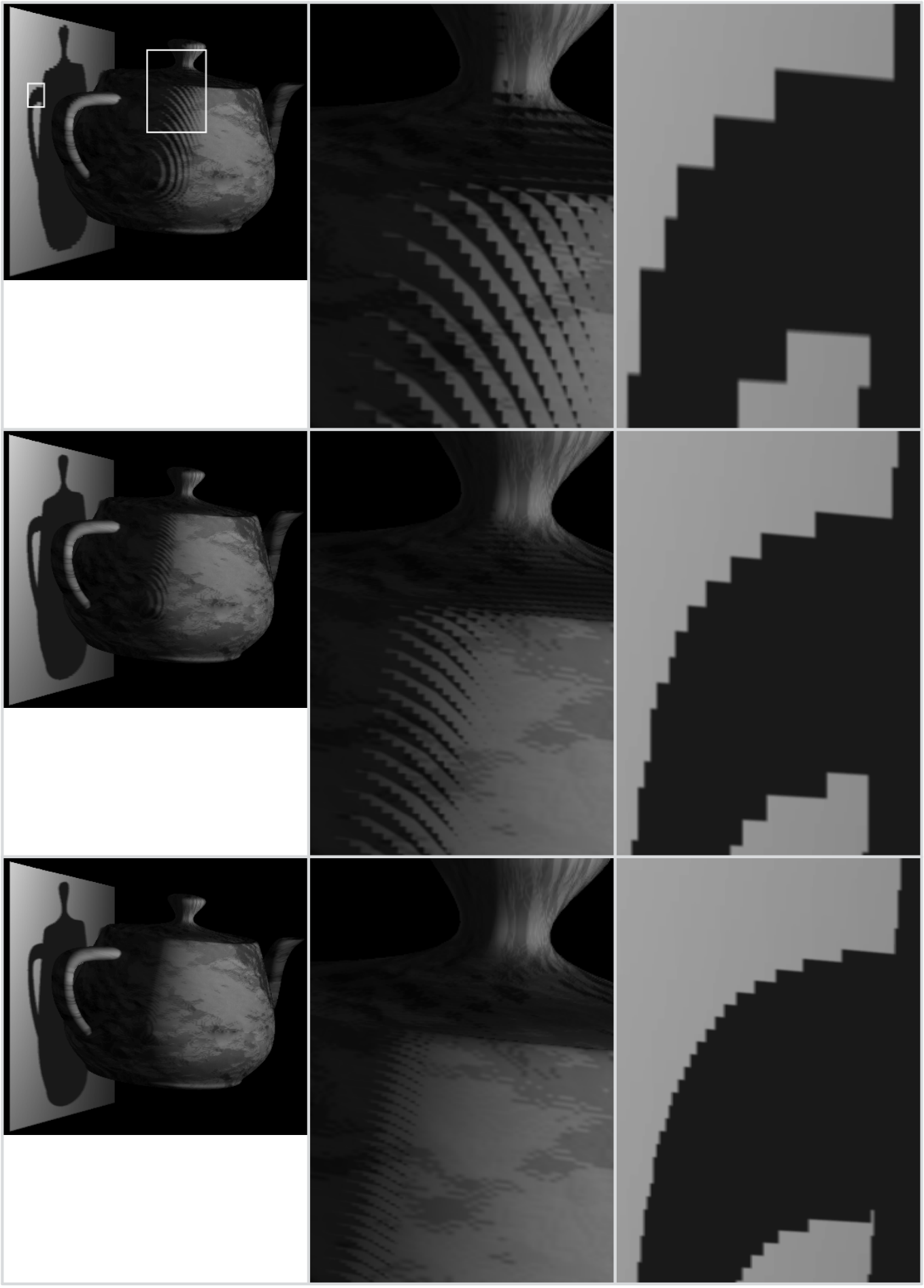
Cast shadow
Per default, a light casts also shadow. Uncheck the option if you do not like to render shadows. Note that the option for the Light effect [527], e.g. "1 Light" [532] are influenced by this setting. If you set the effect to "Cast & Receive" it can only cast a shadow when the linked Light Layer casts it too.

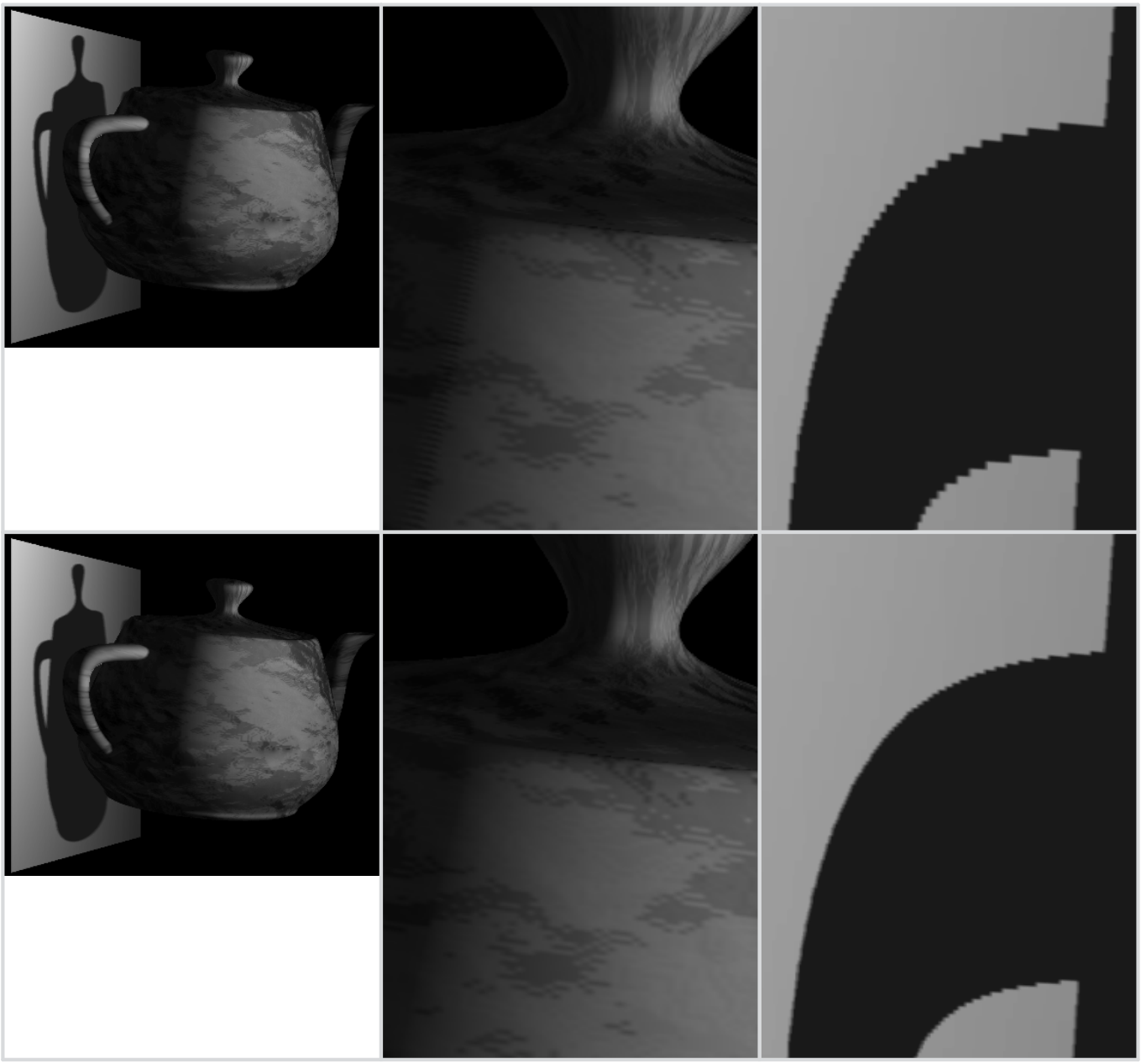
Frustum Appearance



If the Preview is switched to the Layer or Camera Transformation Mode [250], the Light symbol shows the Light Layer's position and the Frustum visualizes the opening angle and Look-at point. Per default the Frustum is shaded with a very light blue/purple color with a transparency level of 50. If you like, you can change the transparency, click into the color field to choose another color or untick the check box "Shade Frustum" if the Frustum should be fully transparent. The Light Wireframes can be altered too. The color can be set with the Frustum's color and the brightness with the number field.
Hint: You can also change the "Layer Color" (in the Device Tree [173]) explained further above to the same color to have an optimal overview, e.g. when many Lights are in use.
Note: You can hide the entire Frustum including the Light icon in the Configuration tab > Preview Display [145].

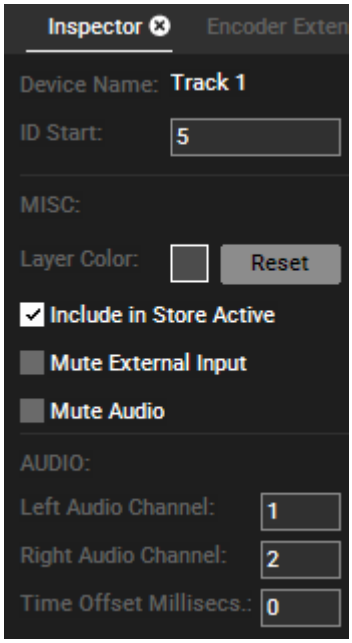
Lastly, the following images show the difference between a Shadow Map of 256 - 512 - 1024 - 2048 - 4096 (from top to bottom).





6.3.4.17.20 Track Inspector

The Audio Track [661] Inspector allows you to set up the following:



ID Start
Set up the numeric Start ID.

Layer Color
Per default each layer in the Sequence tab is displayed in gray. Clicking into the gray field opens a color dialog where you can choose a different color. Note that you can also color a container [192].

Include in Store Active
Uncheck this option to exclude the track in the Store Active command.

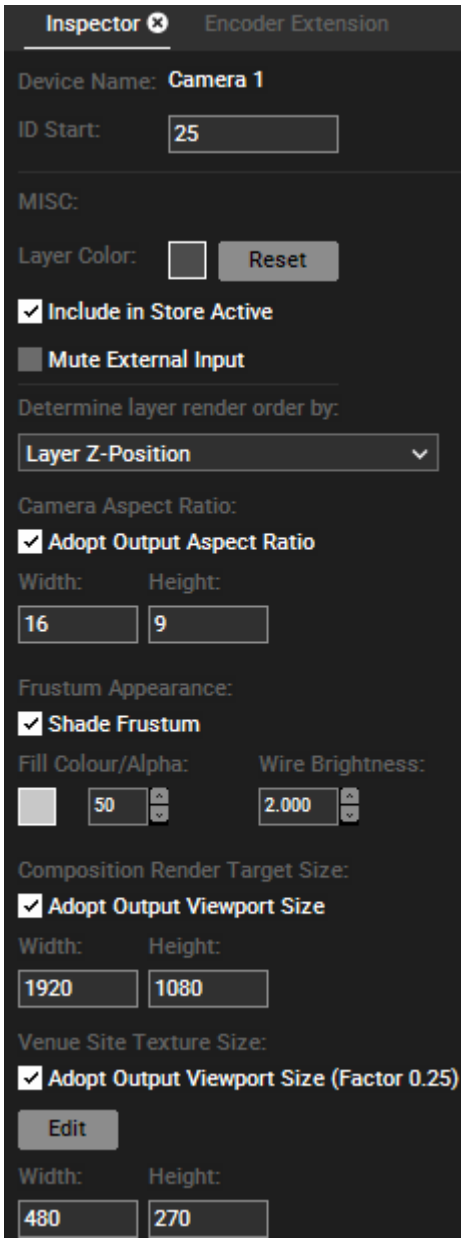
Mute External Input
Tick this check box to block the track from any external input like Art-Net, Input Data from Widget Designer etc.

Mute Audio
Tick this check box to mute this Track. Note that the Mute icon [661] in the Device Parameters tab is automatically clicked.

Left and Right Audio Channel
Choose the channels of your ASIO sound card this track should be routed to. Per default, all tracks are stereo tracks, thus they are routed to two channels of the sound card, whereas the left channel is routed to output x and the right one to output x+1. Hence Track1 is routed to output channels 1+2, Track 2 to 3+4 etc.
If you like to use one channel only for playing mono files, you can simply set the second channel to "0". Alternatively you can use PAN parameter [661]. In this case, it might be of interest that you can store a parameter outside a container [299].

Time Offset [Milliseconds]
Enter a time in milliseconds to offset all content played on this Track. Audio delay settings are available per Resource, Layer or on Clip Level. That enables you to set up different delays for different output signal chains.

6.3.4.17.21 Camera Inspector



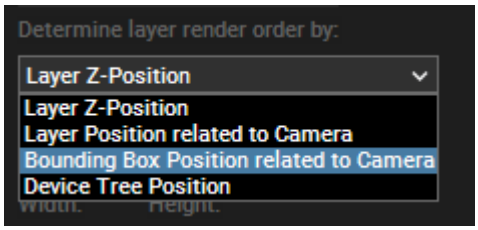
The Camera Inspector opens when a Camera Layer is selected in the Device Tree [173].

ID Start
Setup the numeric start ID of the camera device.

Layer Color
Per default each Layer in the Sequence tab is displayed in gray. Clicking into the gray field opens a color dialog where you can choose a different color. Note that you can also color a container [192]. Hint: You can also change the "Frustum Appearance" explained further down to the same color as the Layer to have an optimal overview, e.g. when many Cameras are in use.

Include in Store Active
Uncheck this option to exclude the Layer from all Store Active commands.

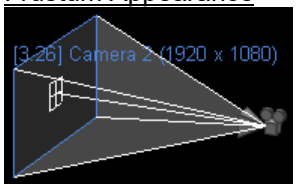
Mute External Input
Tick this check box to block this Layer from any external input, like Art-Net, Input Data from Widget Designer etc.



Determine layer render order by
Per default, the render order for Layers is defined by their Z-position. This means, that the Layer with the smallest Z-position is rendered first. Under circumstances, e.g. when you program different or moving camera positions in a Venue Site, it is necessary to change the render order. Simply choose the desired mode from the drop-down list and the current view is re-rendered.

Camera Aspect Ratio
Per default, the aspect ratio adopts to the ratio set in the Output Layer [220], respectively the one set up in the Configuration > Render Engine [162] which again accords to the (Client's) graphics card settings. If you like to detach the Camera, first untick the check box and enter then enter the new width and height. The chapter Warping Tutorial [264] shows an example when this is helpful.
Note that this simply influences the aspect ratio, the resolution itself can be changed further down under "Composition Render Target Size".

Frustum Appearance



If the Preview is switched to the Layer or Camera Transformation Mode [250], the "Camera Visibility" button  toggles the Camera icon and the Frustum to visualize the opening angle and Look-at point. Per default the Frustum is shaded with a light gray color with a transparency level of 50. If you like, you can change the transparency, click into the color field to choose another color or untick the check box "Shade Frustum" if the Frustum should be fully transparent. The Camera Wireframes can be altered too. The color can be set with the Frustum's color and the brightness with the number field.

Please see the chapter Gizmos and Wireframes in the Preview [249] for more information.
Hint: You can also change the "Layer Color" (in the Device Tree [173]) explained further above to the same color to have an optimal overview, e.g. when many Cameras are in use.

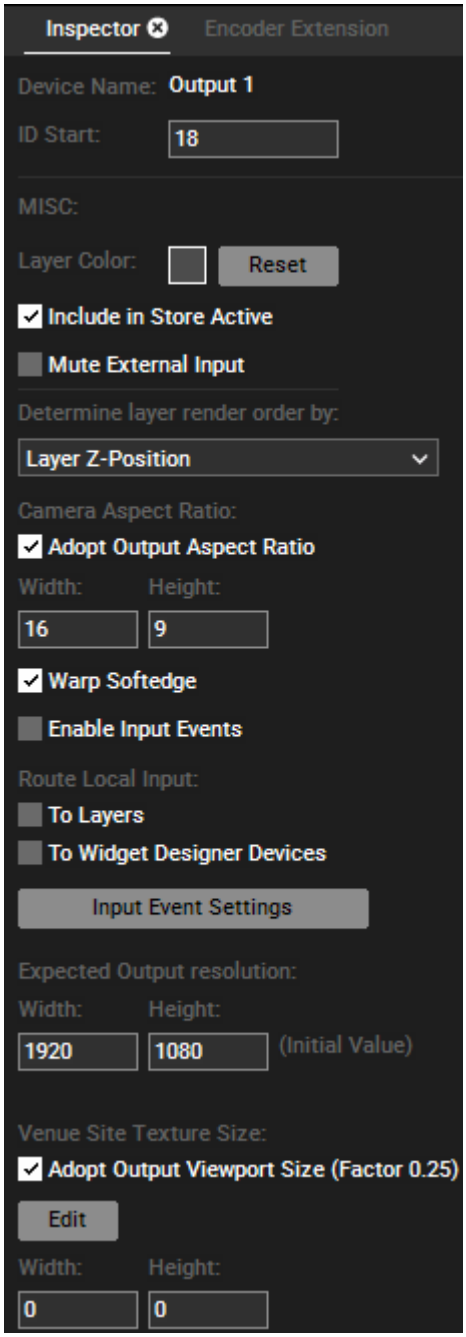
Composition Render Target Size
Just as with the aspect ratio explained above, the Render Target Size adopts per default to the settings from Output Layer but can be changed when the check box is unticked. The render target is the texture "send" from the Camera to the Output. For more information, please read the chapter Video Processing Pipeline [320].

Venue Site Texture Size

The main purpose of a Venue Site [694] is to render content from other Clients, which means that their Camera and Output textures are routed to the Venue Site. Therefore, all those clients must be toggled into the Preview. For larger projects you might want to visualize many outputs which requires the Master to render many textures. To save performance, the routed textures are scaled down. Per default the "Venue Texture Size Factor" is set to 0.25 in the Configuration tab > Local Preview [150]. Increase it for higher quality and decrease it for more performance. The "Edit" button opens the Configuration tab to change the default "Venue Texture Size Factor" that automatically multiplies the reported resolution and enters new numbers into the "Width" and "Height" field. If you untick the check box "Adopt..." in the Inspector you can overwrite the number fields.

6.3.4.17.22 Output Inspector

The Output Inspector opens when an Output Layer is selected in the Device Tree [173].

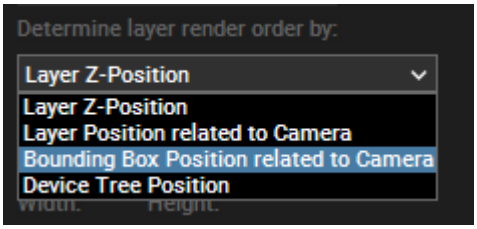


ID Start
Setup the numeric start ID of the camera device.

Layer Color
Per default each Layer in the Sequence tab is displayed in gray. Clicking into the gray field opens a color dialog where you can choose a different color. Note that you can also color a container [192].

Include in Store Active
Uncheck this option to exclude the Layer from all Store Active commands.

Mute External Input
Tick this check box to block this Layer from any external input, like Art-Net, Input Data from Widget Designer etc.



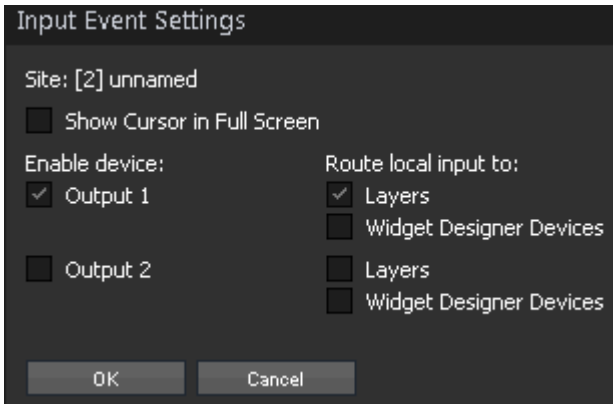
Determine layer render order by
This option is of interest when a Layer is not (only) part of the Composition Pass but (also) the Output Pass, which can be set up in the Layer Inspector [211]. See also the chapter Video Processing Pipeline [320] for more information.
The render order for Layers is defined by their Z-position. This means, that the Layer with the smallest Z-position is rendered first. Under circumstances, e.g. when you program different or moving positions for the Output, it is necessary to change the render order. Simply choose the desired mode from the drop-down list and the current view is re-rendered. Note that "camera" refers to the Output (because it has an internal camera too) and not the Camera Layer.

Camera Aspect Ratio
Per default, the aspect ratio adopts to the ratio set up with the width and height settings either further down or shown in the Configuration tab > Render Engine [162] which again accords to the (Client's) graphics card settings. If the Client is not connected yet, the fields show "expected resolutions" which you can alter.
If you like to detach this, first untick the check box and enter then enter the new width and height. The chapter Warping Tutorial [264] shows an example when this is helpful for the Camera.
Note that "camera" refers to the Output (because it has an internal camera too) and not the Camera Layer.

Warp Softedge
Per default, Softedge settings are also influenced by the Keystone [687] and Mesh [684] parameters from the Output Layer. Uncheck this option if the softedge blending should not be warped.
Note that you can access this functionality with the "Warped [688]" check box in the Output Device parameters too.

Enable Input Events
Tick the check box if you like to use the feature Layer Picking [252]. This activates the respective Output in the "Input Event Settings" dialog so that it receives mouse or touch input events. The dialog can be opened with the same-named button further down or in the Site Inspector [210].

Route Local Input
Please activate the above check box "Enable Input Events" first if you like to use the feature Layer Picking. Now you can route the input events to either a Widget Designer Device and, or to the layers. This activates the respective check boxes in the "Input Event Settings" dialog which can be opened with the same-named button further down or in the Site Inspector [210].



Input Events Settings

The pop-up dialog offers the same options regarding the feature "Layer Picking" as described above but includes other outputs too. In short, you can route input events like a mouse click to Pandoras Box' layers, for example to execute links in a Browser Asset [279].

The first option "Show Cursor in Full Screen" toggles the visibility of the mouse cursor in the Client's fullscreen window.

If you like to use Layer Picking on an output (instead of only in the preview tab and the Global Cam), the left setting "Enable device" must be activated, either for Output 1 or 2 or both.

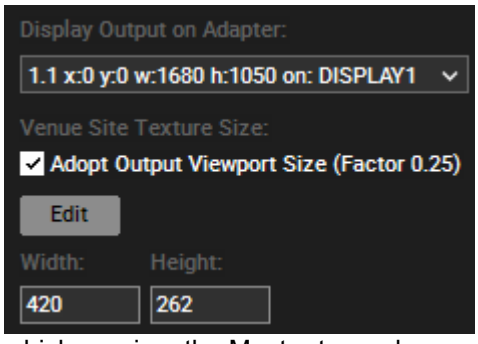
The settings to the right routes the Layer Picking data to either a Widget Designer Device [696] and, or to the layers.

For more detailed information and a step-by-step example, please refer to the chapter about Layer Picking [252].

To see the Input Event Settings for all systems in the Device Tree combined in one dialog, right-click in the Preview tab and choose "Show Input Events" there.

Expected Output resolution

This part is only shown in case your Client is not connected yet, otherwise it is replaced with the below option "Display Output on Adapter". It displays the expected resolution for the Width and Height. You can alter the initial value, for example if you know which resolution your Client will have and your show pre-programming depends on the resolution. The same options are shown in the Configuration tab > Render Engine [162].



Display Output on Adapter

This part is only shown in case your Client is connected already, otherwise it is replaced with the above option "Expected Output resolution". The drop-down list shows the output adapter in the same way as it is set up in the Configuration tab > Render Engine [162].

Venue Site Texture Size

The main purpose of a Venue Site [694] is to render content from other Clients, which means that their Camera and Output textures are routed to the Venue Site. Therefore, all those clients must be toggled into the Preview. For larger projects you might want to visualize many outputs which requires the Master to render many textures. To save performance, the routed textures are scaled down. Per default the "Venue Texture Size Factor" is set to 0.25 in the Configuration tab > Local Preview [150]. Increase it for higher quality and decrease it for more performance.

The "Edit" button opens the Configuration tab to change the default "Venue Texture Size Factor" that automatically multiplies the reported resolution and enters new numbers into the "Width" and "Height" field. If you untick the check box "Adopt..." in the Inspector you can overwrite the number fields.

6.3.4.18 Media Encryption

Pandoras Box Media Encryption is a technology that encrypts media in order to protect content from being played back by unauthorized persons or systems without permission.
Please note, that the optional Media Dongle, that adds extra security to the encryption feature, is discontinued until further notice.

Encryption

With a Pandoras Box Master you can choose to encrypt [224] media by generating two ASCII strings. The Media Encryption is based on AES 256-bit technology similar to the DCI Digital Cinema Standard, which means you will get the same powerful level of encryption that most governments use for secret documents.
Currently, MPG (including MXL, M2V, ...) and WAV files are supported. Encrypted media obtains the ending ".mpx" or ".wax".

Please note, that the 64bit version of Pandoras Box introduced with version 6.4, uses another encryption algorithm. Hence, those encrypted media files can only be decrypted with another 6.4 (or higher) PB system. In return, media files encrypted earlier can only be decrypted with a version below 6.4.

Decryption

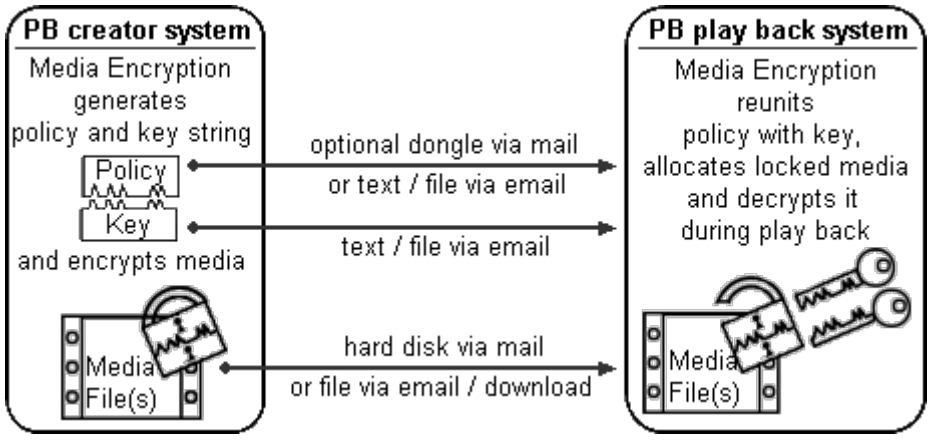
Any Pandoras Box system, either Master or Client, is able to decrypt the media with the correct ASCII strings while playing it back:

The content gets decrypted on the fly when being played back on the timeline or as an active value. There is no decryption period beforehand, thus there is no decrypted version of the content on your hard drive. Each frame needs to be decrypted (except the thumbnail). Only PB systems are able to understand the ASCII string and decrypt the media !

Of course, encrypted content cannot be exported using the Video Export [135] feature. In return it is possible to encrypt an exported video itself.

Policies, Keys and the Media Dongle

The encryption is based on two different ASCII strings - the key string and the policy string. At all times, the end customer needs both to decrypt media.
The key itself encrypts the content. The media "remembers" which key can unlock it. If you encrypt the same content with two different keys you will get two different encrypted files.
The policy holds meta information regarding the validity of a key. A key without meta information cannot be used, in return, meta information without the key is useless as well. Currently the only meta information is the time stamp. You can either decide to have no time restriction or to create a string that can only validate the key for a certain time.
As described later on [227], it is possible to modify a policy, e.g. if you need to extend the validity. That means that the customer keeps the same encrypted media file and the same key, only the policy will be updated.



In case of a time-based encryption, you can save and transfer the policy as a "digital policy" or as a "dongle policy" meaning that it is written onto a Media Dongle. Until further notice, the Media Dongle is discontinued and cannot be purchased.
Working with a Media Dongle will give you the safest encryption solution possible. The Media Dongle holds the

definite time reference itself and does not refer to the system time. The Media Dongle is more secure in other terms as well. As seen in the image above, you can transfer the strings (key string and policy string) in different ways to the customer, including the possibility to send both via email. In that case you have no chance to control whether he sends them to somebody else as well. If you choose to export the policy to a Media Dongle and send it via mail, the customer needs to have that special hardware. You have created the necessity of a physical key which cannot be copied.

Example:
It is possible to create content, encrypt it once with a key and create two different policies, both being allocated to the same key. Let´s say the unlimited policy is written onto a Media Dongle and a time-based policy is a more simple "digital policy". You will then have one encrypted file, one key string but two policy strings.
You may then send one media file, one key and one Media Dongle to the show operator. He will be only able to play the content on a PB system if the Media Dongle is inserted. The other media file, key and (time-based) digital policy goes to the customers in the event agency. They will be only able to play the content on a PB system within a given time limit. They may as well send the key and policy to another person.
Of course you can encrypt as many files as you wish with the same key. In this scenario it has the following advantage: Now, as soon as the agency or the operator want some changes to be done, you can encrypt new content with the key and send only the media files to the two parties, both being in possession of the key and being restricted to their policy version.

Please note that currently only one Media Dongle per system is read and each dongle can hold only one policy. Likewise only one digital policy can be assigned to each PB system. As a result,one dongle policy plus one digital policy can validate two keys per PB system.

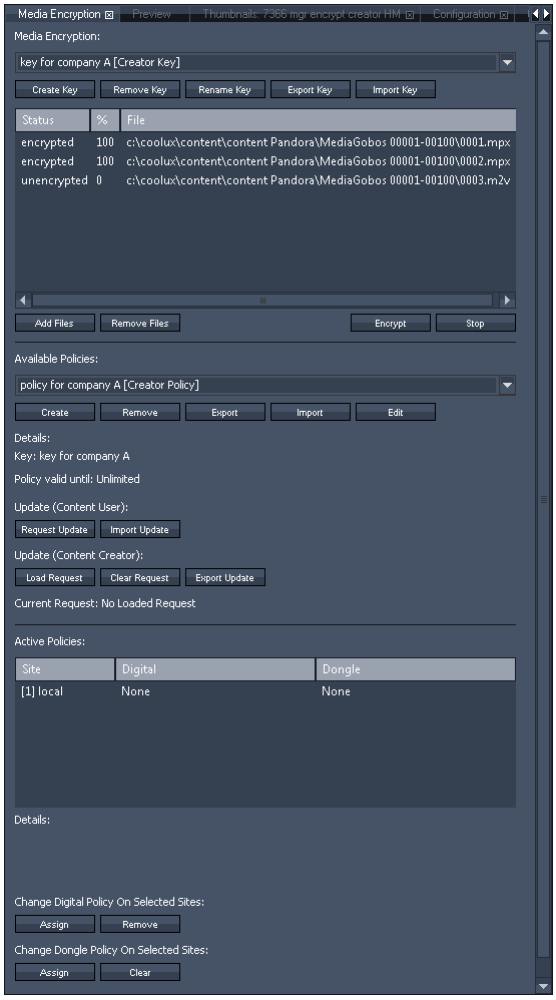
In the future it might be possible to create meta information such as a fixed MAC addresses etc. If you want to contribute your feedback to this new feature please contact the support team.

The difference between a creator key / policy and an user key / policy is described in the following chapter 224.

Media Dongle hardware specifications:
- Battery Life Warranty: 4 Years
- Real Time Clock Accuracy (@ 25°C +/- 5°C) < 12.8 minutes/year
Consequentially, we recommend to factor 13 minutes per year into your time limits.

6.3.4.18.1 Encrypting Media

To open the Media Encryption Tab, please click on 'Tabs' in the Toolbar - Media Encryption.



In order to encrypt media files please follow these steps:

Key: Click "Create Key" in the Media Encryption tab to generate a Creator Key .The appendix "[Creator Key]" is automatically added to the given name.

Media: Now choose the media files (MPEG or WAV) which are part of your project and add them to the key via the "Add Files" command. If you have chosen a file by mistake, select it and click "Remove Files".

Encryption: Select the command "Encrypt" to start the actual encryption. The "status" and "%" row inform you whether the media is unencrypted, in progress or already encrypted with the chosen key. As a rule of thumb, two minutes fullHD footage take 10sec to be encrypted.
Already encrypted files will not be encrypted again, there is no need to remove them from the list.

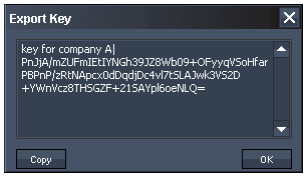
The encrypted files (mpx or wax) are saved in the same sub folder on your hard disk as the original files. Please note that another encryption with a second key will overwrite the first file when the original file was not moved to another folder.
The encrypted files are automatically added to the project and will be shown in the list as soon as you choose the used key via the drop down menu. The list helps you to keep an overview and to allocate the key if you have forgotten it.

You may choose whether you like to encrypt the files before even assigning the policy. This is because the key encrypts the media. The policy validates the key and has nothing to do with the media itself.

Policy: In order to setup the time limit, please choose "Create".
In the pop up dialog choose a Creator Key from the drop down list. This will join the policy and the key. Metaphorically speaking it imprints the necessary meta information on the key. Only then, the key is of value for the end customer.
A policy will always remember the key it was allocated to at last.

Time Limit: Choose if you want to have an unlimited policy or a time-based encryption.

How does a time-based policy work on a Media Dongles?
After having chosen a time restriction including the day and time, the time difference to your local time will be added to the time reference on the Media Dongle, and later on from the dongle to the customer´s time. This way the timeout is always calculated correctly even if the system´s time has changed.

Key Delivery: 

It doesn't matter which string you export first. For security reasons there is no text file generated automatically on your hard drive. Select the "Export Key" button to generate the key string. Click the "Copy" button and insert the string to a file of your choice or into an email directly. The key is always transferred digitally.

Policy Delivery: The policy string can be transferred as a "digital policy" or as a more secure [222] "dongle policy":
To export the policy string and send it for example via email, select the "Export" button and copy the string.
To write the policy string onto a Media Dongle, scroll further down to the list with "Active Policies" and select the site the Media Dongle is attached to. Then click the button "Assign" at the very bottom of the tab where it says "Change Dongle Policy On Selected Sites". The list will update automatically and the column "Dongle" will show your policy. In case there was already an older policy string, it will be overwritten by the new data. The button "Clear" deletes the current policy from the dongle.
Now, you need to ship the Media Dongle to the customer and send the key via mail.

At any time you may rename the key and policy, this does not alter the functionality! The button "Edit" allows you to edit a policy before exporting it, you may either choose another Creator Key or time-limit.

If you have created a key it is saved within the project and the appendix "[Creator Key]" is automatically added to its name. After exporting a Creator Key it will loose this special status and will become a regular key.
What is special about a creator key? Only the "creator" of a key can create a corresponding policy, exchange it with another policy or prolong the time limit if necessary.
As the creator of a key, you can always play the encrypted files – without a policy being required, even connected Clients will play them back. A creator key can only exist within the original project, the key is saved as a part of it. The project has to be kept/stored in case more files should be encrypted with that particular key or in case a Client demands content updates.
When exporting the key and importing it into a new project it will not be a creator key anymore. A regular key cannot be joined with another policy! The key will only be valid if the assigned policy is allocated and still active. This counts even when there are no specific limitations saved.

In general you can use any key in the system for encrypting media content, not just a creator key. Of course you are restricted to the originally assigned policy when choosing an imported key. Without the policy, no content can be played out.
Additional media files can be encrypted even later on using the same key as for the content before. As long as the customer is in possession of the key and the policy, he will be able to play out content that was encrypted at a later stage. In this case no update would be necessary.

Same distinction applies to policies. After creating a policy the appendix "[Creator Policy]" is added to its name. Only creator policies can be edited and load (creator) keys, in other words imprint the time stamp on them. After exporting a Creator Policy or writing it to a Media Dongle it will loose this special status and will become a regular policy. A regular policy cannot be edited at all, except of its name.

Please note: We strongly recommend to:
- use specified creator projects. If you want to use encrypted content yourself it is more secure to work with projects not containing the original creator key but the exported key.
- backup the creator projects to a separate place.
- keep encrypted content and the key and policy strings separate from each other.

It may be useful to create a data sheet to keep an overview which file was encrypted with which key+policy and to whom it was sent. In addition you may want to create a folder on your (external) hard drive where you copy the encrypted version to.

6.3.4.18.2 Decrypting Media

To open the Media Encryption Tab, please open the "Tabs" menu in the Toolbar and choose "Media Encryption".

In order to decrypt content you have to copy the key, the policy and the content to the project. The order in which you go through these steps doesn't matter, since the files can be played only if all elements are present:

Key: Copy the string from the email or text file. Select "Import Key" and click the "Paste" command. After clicking "OK" the key string is imported to the project and spread automatically to all connected Clients and to all Clients added to the project later on.
You may rename the key (and policy) at any time, this does not alter the functionality!

Policy: If you have received a digital policy via email or an text file, import it in the same way by selecting the "Import" button. As policies are not spread automatically, you need to assign them to each PB system. To do so:
- choose the particular policy in the policy`s drop down list
- scroll further down to the list with "Active Policies" and (multi-)select the site(s) where you want the selected policy to be active
- click the button "Assign" at the very bottom of the tab where it says "Change Digital Policy On Selected Sites".

The list will update automatically and the column "Digital" will show the according policy. If you have assigned a wrong policy simply choose another one in the drop down menu and repeat the mentioned steps.

If you have received a dongle policy via one or more Media Dongles insert them to the PB systems which should be able to decrypt media. The dongle and its containing policy are recognized automatically. The column "Dongle" in the list with "Active Policies" will show the current status. As well the policy will be added to the drop down list "Available Policies" in the PB Master (even though no dongle is attached there). The appendix [From Dongle] informs you that it is no regular digital policy.

Now, on each PB Master and PB Client the policy recognizes imported keys and validates the according one.

Please keep in mind that every system that has to decrypt media needs to have a valid key, that may require a Media Dongle per each system!
As soon as you would unplug a Media Dongle, assign a wrong digital policy, exceed the allowed time-limit or remove the key in the Master, the system cannot decrypt any more and either the preview or the fullscreen window will show a dummy picture instantly. The dummy picture tells you whether there is an issue with the key or the policy status.

Media: 

Now drag the encrypted content into the project. The content recognizes its key and can be unlocked. The files are automatically added to the particular key list. If it cannot find the correct key it will be substituted with a dummy picture.

The icons in the project tab tell you whether a file is
a) encrypted with a key / policy that is not part of the project,
b) encrypted with a known key / policy
c) not encrypted.
The status does not affect the functionality of spreading. E.g. the Master will spread encrypted content to all Clients, even though it is not equipped with a dongle-based policy. Only those Clients with the correct key and policy will actually decrypt the media.

We strongly recommend not to change keys or policies during a show! Any change in a key or policy status or Media Dongle connection will update all encrypted files.

6.3.4.18.3 Updating a Policy

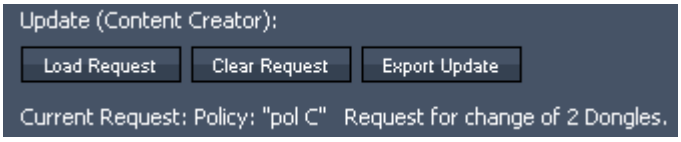
In case the key is allocated to a time-based policy it might be necessary to prolong it.

Client:

Choose the policy you wish to prolong. Click the command "Request Update".
Copy the string and send it to the content creator. There is no need to send the Media Dongle!

If you are working with dongle-based policies, please note that the string created for the request holds not only information about the policy itself but also information about attached dongles. While creating the request string, please be sure that all Clients are connected and all dongles are plugged in if they need to be updated. Otherwise a particular Media Dongle can not be updated later on without sending a new request string! This makes the entire encryption process more safe.

Creator:



Open the project with the corresponding creator key and creator policy. Copy the string you received from the end customer. Click the button "Load Request" and paste the string. The information line tells for how many Media Dongles the update will be valid and eases to keep an overview.
Choose the policy the customer wishes to extend and click the button "Edit". After entering new limitations click the command "Export Update". Copy the string and send it to the customer.
The button "Clear Request" simply discards the request information. Please note that you cannot export an update string without loading a request string. Your customer needs to generate that request string first hand.

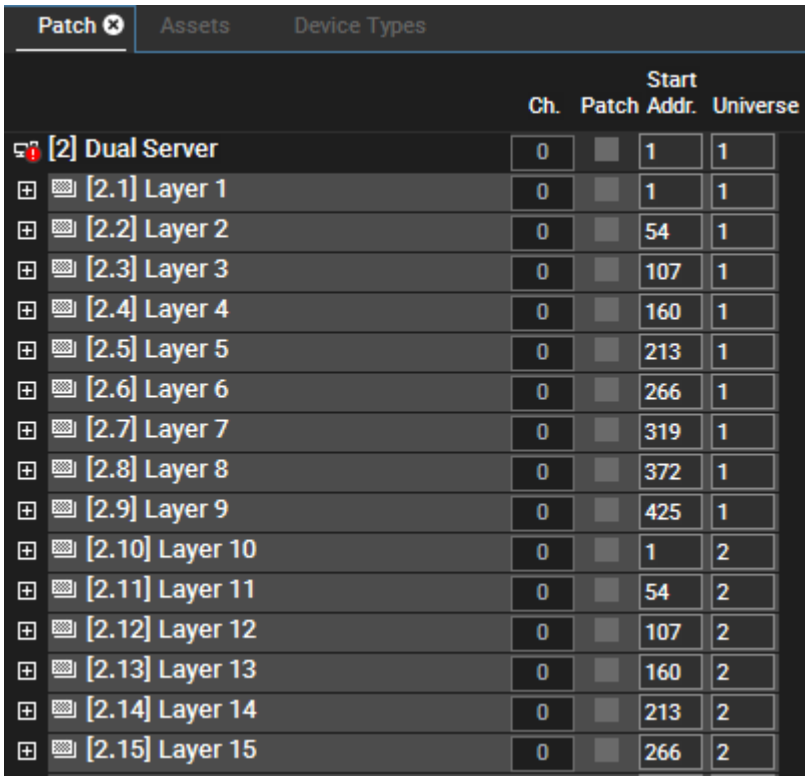
Client:

This time you click the button "Import Update" and paste the update string you received from the content creator.

The policy status from all systems in the entire network is updating automatically.
If one Client is not online whilst updating it will receive the prolonged digital policy as soon he is connected to the Master system.

In case you are working with dongle-based policies, the dongle data is automatically overwritten with the new policy. A pop up lets you know if all dongles (that were part of the original request) can be reached and updated. You may continue updating even though one Media Dongle is missing. As the update procedure can be executed repeatedly, simply import the update as soon as the dongle is available. Meanwhile the policy will not only be listed with the appendix "[From Dongle]" but also with "[Variant]" in order to inform you about different time limits within the network.

6.3.4.19 Patch



The Patch tab is not part of the default views, please open it from the Tabs menu [122] from the main tool bar. The Patch tab always shows the (one) Site that is currently selected in the Device Tree [173] or all Sequences when the Sequence folder is selected in the Project tab [278].

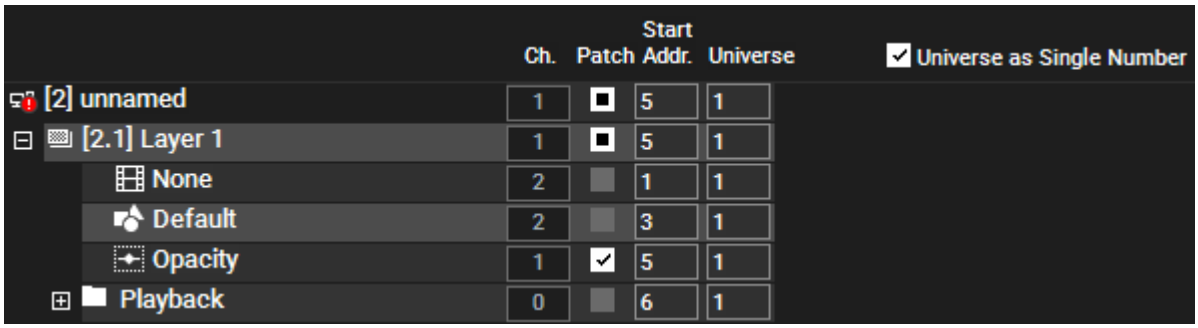
In the Patch tab you can assign DMX addresses for Sequences and for all Devices which appear in the Device Tree. In case you like to control Pandoras Box Layers from incoming Art-Net or DMX data, you can either patch Sequences, entire Sites or individual Layers. On the other hand, if you like to output Art-Net or DMX data from your Pandoras Box timeline, first add an according DMX fixture [691] to the Device Tree and patch then all parameters or only some of them. In other words, Pandoras Box Devices and Sequences are patched for DMX input whilst all DMX devices are patched for DMX output.

Please read the chapters DMX Input [706] and DMX Output [721] for more information about the entire workflow. You can activate incoming or outgoing DMX or Art-Net data in the Configuration > Remote Control Protocols. [148]

The following two chapters explain how to patch a Site [229] and a Sequence [233].

Overview

This part explains what the information means that is displayed in the Patch tab. Remember that the patch information relates to the Site that is currently selected in the Device Tree [173] (or all Sequences when the Sequence folder is selected in the Project tab) [278].



This image shows a partly patched Site. The "Patch" check box for "Opacity" is checked which means that the parameter is patched to the displayed address: DMX channel 5 on Universe 1.
The Patch check box for the Layer or Site is filled with a lighter gray color to indicate that some but not all Parameters are patched. The address for the Layer and Site show also [5,1] as this is the starting address. The other parameters are not patched as indicated by the empty check boxes. The address fields show the starting address in case they are patched.

You can change "Address" and "Universe" to suit your needs but "Channel" is an information that updates automatically and cannot be changed by you.

Universe

Per default, the universe can be entered as one number because the option "Universe as Single Number" is checked. If it is not ticked, a new field named "Subnet" appears. When using Art-Net it is common to refer to an Art-Net Subnet and Art-Net universe or ID. If you are using MA-Net, please refer to the following translation chart:

Grand MA Patch	Pandoras Box Patch	
Universe.Channel	Ch.	DMX Universe
1.1	1	0 : 0
2.1	1	0 : 1
3.1	1	0 : 2
15.1	1	0 : 14
16.1	1	0 : 15
17.1	1	1 : 0

6.3.4.19.1 Site Patch

This chapter explains how to patch a Pandoras Box Site. In detail, it shows, how to..
- check the default patch [229] when a Site was added with the option "Lighting Console Configuration"
- patch a Site manually [230]
- use the commands in the context menu [232]

Please read the chapter Patch Templates [290] if you like to learn how to use templates for custom patches.

Remember to activate DMX or Art-Net Input in the Configuration tab [139] in the section Remote Control Protocols [148] to receive incoming data. Please see the chapter "DMX Input [706]" for the general workflow regarding DMX Input.

Patch in Lighting Console Mode

As explained in more detail in the chapter "Device Types [183]", the "Lighting Console Configuration" can be chosen when opening a new project. Alternatively you can enable the check box in the Device Types [183] tab. All Pandoras Box Sites that you then add to the Device Tree will be patched already. As you see below, the Layers (or rather parameters) are patched according to our DMX tables [708] which include the most relevant parameters. This DMX library can be loaded on most lighting desks to control Pandoras Box.
In addition some "Multi-FX" [444] effects will be added to the Layers which allow to choose various effects but keep a constant parameter count to maintain a fixed channel count for the patch.

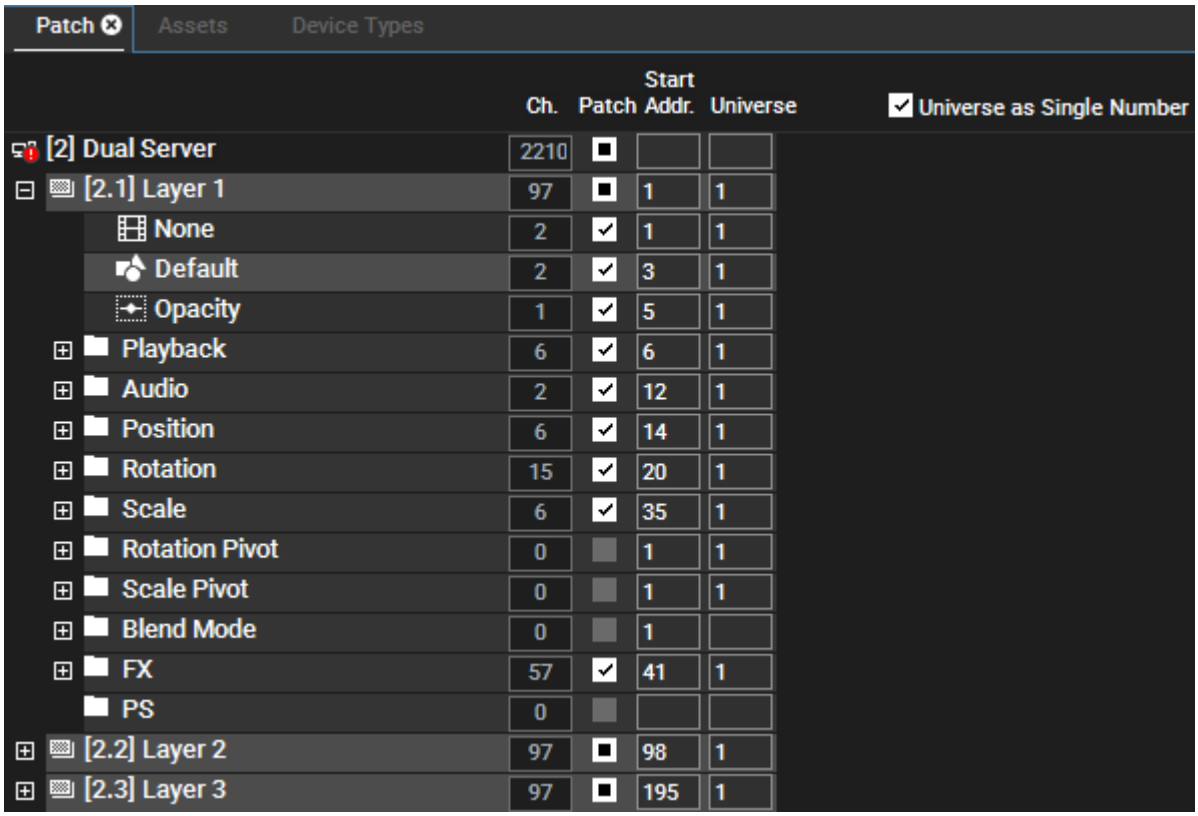
The below image shows a Server with two outputs with the default "console" patch. If DMX Data was coming in already, the parameters would be highlighted in red to show the active value state.

The first field, "Channel", shows that for the entire Server, 2210 channels are patched and for "Layer1" it is 97 channels. When you expand the Layer, you can see that the channel count appears for each parameter. Some parameters like "Opacity" need one (8 bit) channel because their value range is 0-255. Other parameters with two DMX channels either need two individual 8 bit values (like "Media" or "Mesh") or a 16 bit value because their value range is 0-65535. Please check out the DMX tables [708] for more information about the parameters.
Now you can also see which parameters are actually included in the patch, that is those with the activated "Patch" check box. Some parameters have an empty "Patch" check box, e.g. "Blend Mode", which means they are not included in the default patch. Because the Layer or Site is partly patched, their check box is partly activated as indicated by the lighter gray color.

"Start Address" and "Universe" show the starting DMX channel and universe for the Site, Layer and parameter. As explained in the last topic [229], if you like to enter a "Subnet" too, please uncheck the option "Universe as Single Number".

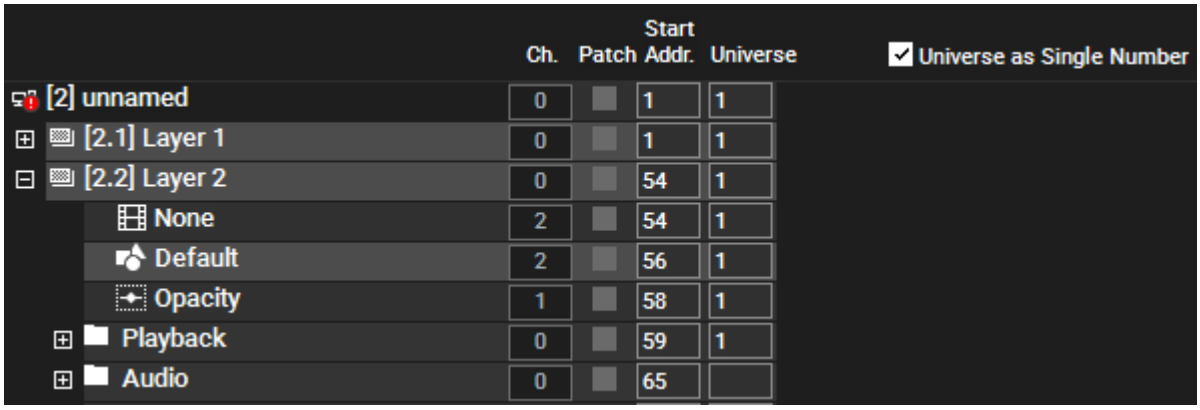
Most likely, you will not have to change anythings in here as this also requests to change the patch in the lighting console. However, if you need to, simply use the check boxes to patch / unpatch a parameter or change a channel or universe by entering another number. This is explained in more detail below under "Manual Patch" or under "Entering new Addresses [231]"

If you just like to start with a different universe, the best way is to make a right-click on the Site and choose "Patch > Set Patch Start Values". Enter your new universe as a single or two numbers [229] and press "OK".



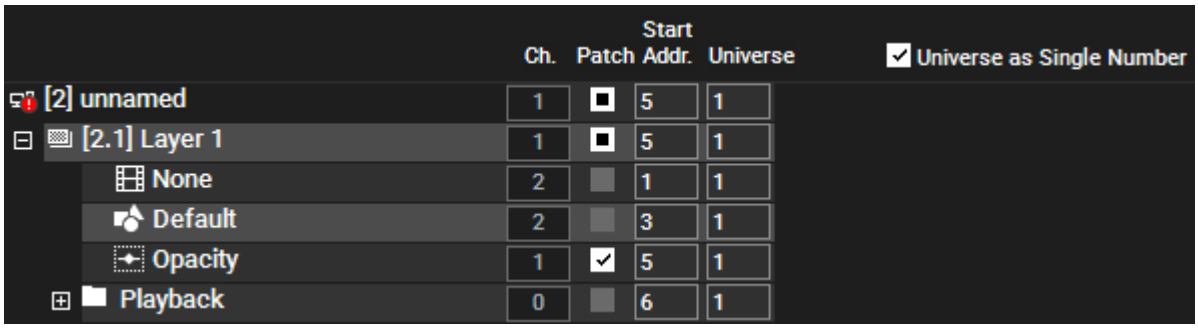
Manual Patch

If a Site was added to the Device Tree without the "Lighting Console Configuration", no parameters are patched and no effects are added automatically to the Layers.



This image shows a Site that is not patched. The "Patch" check box is not set for any single Parameter, Layer or Site.
The first field, "Channel", shows that "0" Channels are patched per Site or per Layer. "Address" and "Universe" show the starting DMX address in case anything would be patched.
When a Layer is expanded, so that the individual parameters are displayed, "Channel" shows the channel count that needs to be occupied by this parameter. In the example, "Opacity" needs one (8bit) channel because the value range is 0-255. Other parameters with two DMX channels either need two individual 8bit values (like "Media" or "Mesh") or a 16bit value because their value range is 0-65535.

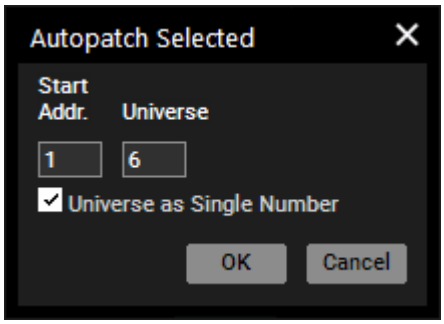
You can now activate the "Patch" check box for a Parameter, Layer or Site.



The above image shows a partly patched Site. By clicking the "Patch" check box for "Opacity", this parameter got patched to DMX channel 5 on Universe 1. If DMX Data is coming in already, the parameter would be highlighted in red to show the active value state.
The Patch check box for the Layer or Site is filled out to indicate that some but not all Parameters are patched.
The address for the Layer and Site shows also [5,1] as this is now the starting address.

Entering new Addresses

Whilst, "Channel" is an information that updates automatically, "Address" and "Universe" for any Parameter, Layer or Site can be changed from you to suit your needs. Remember that the "Patch" boxes need to be ticked first.

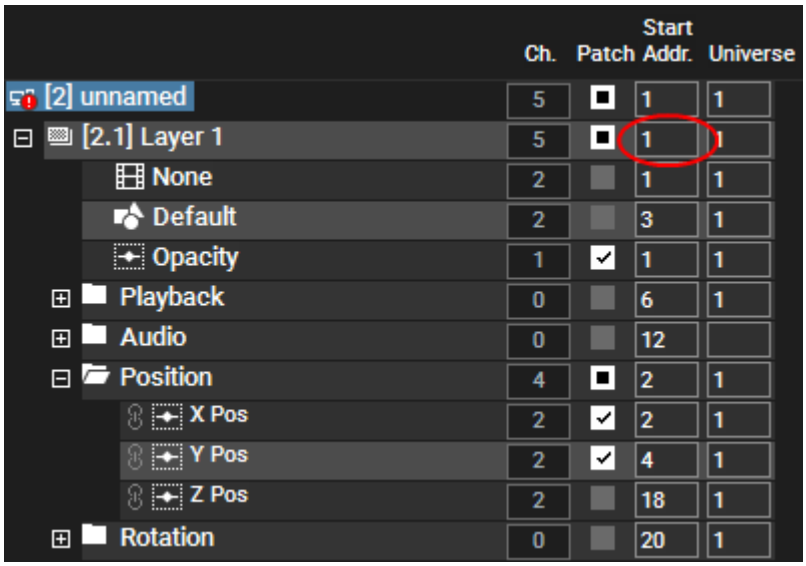


The best way to assign increasing addresses is the "Autopatch Selected" dialog. Right-click a Layer or Site and choose the command "Set Patch Start Values". Enter your new universe number and press "OK". All parameters (with an activated "Patch" check box) will be patched to an incrementing address. If a Layer does not fit into the rest of the last Universe, a new Universe is started so that the Parameters do not span across two Universes.

The Universe can be entered as a single or two numbers [229].

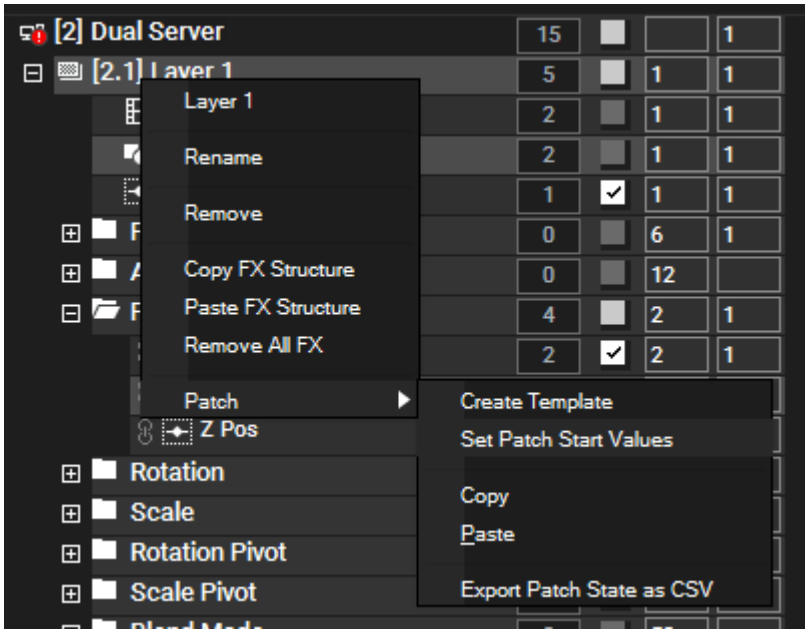
You can also enter the address in the Patch tab directly. When entering a Universe number on the level of a Layer or Site, all underlying Parameters get the same Universe.

When you enter a "Start Address" on the level of a Layer or Site, the address increases automatically for the included Parameters. If you patch more than 512 channels you will be asked, whether you like to "Wrap" or "Shift" the index. "Wrap" simply uses all available channels even if the Parameters of a Layer would span across two Universes. "Shift" always starts with a new Universe for a Layer if it does not fit into the rest of the last universe. In other words, first enter a Universe and then a starting address if you like to patch the channel and universe automatically.



This image shows that "Opacity", "X Pos" and "Y Pos" are patched. After checking the boxes, "1" was entered in the "Addr" field for Layer 2.1 to patch the Layer automatically. This has the result that "Opacity" is patched to channel 1, "X Pos" to 2 and "Y Pos" to 4.

Context Menu in the Patch Tab



In the Patch tab, you can right-click on a Parameter, a Layer or Site to see their context menus. The commands [Rename, Remove, Copy / Paste FX Structure and Remove all FX] are described in the chapter about the context menus in the Device Tree [178]. They also include the following Patch commands, so you can execute them from many places.

Under "Patch" you will find more options related to patching.

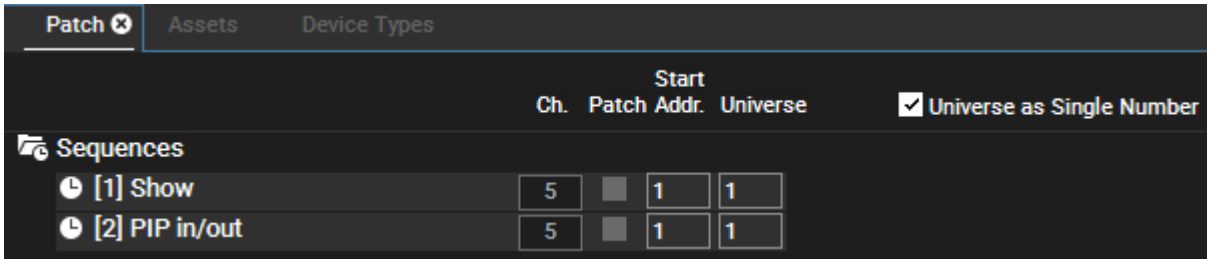
Create Template
This saves the patched parameters to a template which is added to the Project tab. See further down under "Patch Templates [290]".

Set Patch Start Values
This opens a dialog where you can enter a new channel and universe number, which can be entered as a single or two numbers [229]. All included parameters (with an activated "Patch" check box) will receive a new address that increments automatically according to the channel count. If a Layer does not fit into the rest of the last Universe, a new Universe is started so that the Parameters do not span across two Universes.

Copy / Paste
First, select a Layer and make a "Copy" to save which parameters are patched. Then, right-click another Layer and choose "Paste" to patch the same parameters. You might want to enter a new starting address for the newly patched parameters. You can do this either with the "Set Patch Start Values" command from above or by entering a new starting address in the Patch tab.

Export Patch State as CSV
This exports a CSV file that includes all necessary patch information, i.e. channel count, starting address and names for all patched Sites, Layers and Parameters.

6.3.4.19.2 Sequence Patch



When selecting the Sequence folder in the Project tab [278], the Patch tab shows all available Sequences within the project.

Simply click the "Patch" check box to patch a Sequence. If you like, you can enter another DMX Address or Universe. The Universe can be entered as a single or two numbers [229].
Remember to activate DMX or Art-Net Input in the Configuration tab [139] in the section Remote Control Protocols [148] as described in the chapter "DMX Input [706]". In contrast to the Site Patch, incoming values are not displayed as active values.

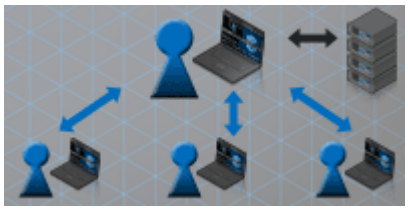
The Sequence DMX table [708] shows that every Sequence has always five channels:

5 CH	Parameter		Values	Meaning	Default
1	Opacity	8 Bit	0 1.255	Sequence not visible Opacity for all layers on sequence	255
2	Transport Control	8 Bit	0 128 255	Pause Play Stop	0
3	Cue	8 Bit	0 1.255	no Cue Cue ID	0
4	Frame	16 Bit	0 65535	Set nowpointer to frame pos	0

6.3.4.20 Multi-User

The Multi-User tab in Pandoras Box allows to setup the Multi-User feature and gives an overview on all Places (i.e. Multi-Users) online with their IP address and network name. The tab is not contained in the default view [287]. To open it, go to the Menu Bar and choose Tabs > Multi-User.

The Multi-User Feature in General



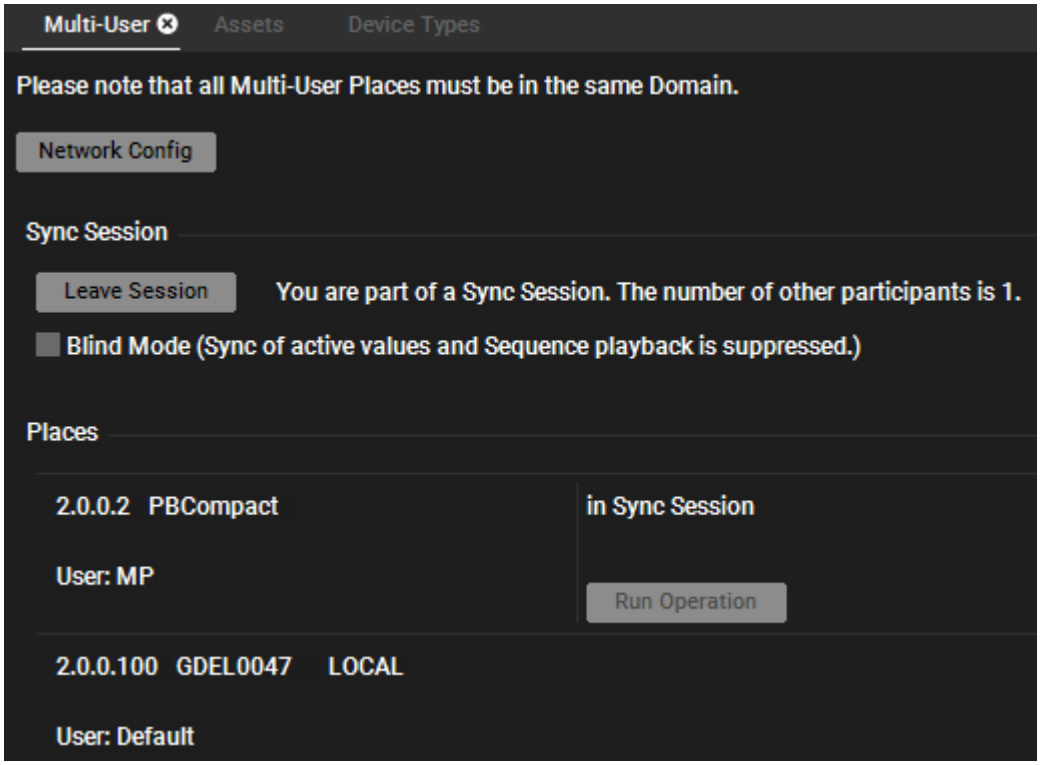
Pandoras Box version 6 introduces the Multi-User feature allowing several operators to work on the same Master file. As the ways to build shows with Pandoras Box are so versatile, Multi-User editing is a flexible feature. This means that you can choose what exactly you would like to synchronize and how often or when you like to do so.
As soon as another Master (without Output Layers!) is in the same LAN network and shares the Domain it can participate as another Multi-User. Output Layers cannot be added to participating Masters. The new tab Multi-User allows to setup the Multi-User environment and gives an overview on all Users online.

The first possibility for Multi-Users is to work in a "Sync Session" where all changes are synced continuously and seamlessly between all Users. However, you can also activate the "Blind Mode" within the Sync Session for working with an independent Nowpointer and Active Values. This allows you for example to jump to "your" time in the Sequence to store content or to view content in the Preview.
You can leave and (re-) join the Sync Session at any time. When joining a Sync Session you are asked to load the session project.

Alternative to (fully or partly) synchronized programming, you can work asynchronously in a separate project and send ("push") your changes at a certain time, e.g. checked or newly encoded content, warp and softedge settings, parts of a Sequence, etc. The "Run Operation" button in the Multi-User tab opens a dialog to define these Pull and Push Operations.

All settings that are listed in the Global category [139] in the Configuration tab apply to all Multi-Users whilst the other ones apply only locally [139]. The User category [139] now features profiles which are shared with others so they can apply one of them too. This is explained below [236] in more detail.

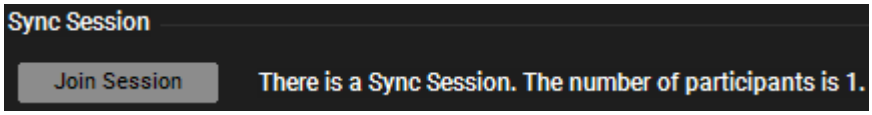
Setting up a Sync Session



This Multi-User tab shows that there are two Places (= two Users) online. They are in the same network, share the same Domain and the Master software is started. The User Profile [236] is displayed too.

Multiple Masters in the same LAN network (and Domain) participate automatically as Multi-Users. Now they have the choice to work in a Sync Session. As a possible scenario for a Sync Session, imagine, for example, that one operator is assigned to the Warping job whilst another one previews and arranges content in the Project tab and a third operator wants to setup the connection to Clients and their output resolution.

The first Master (from now on the Main Master) clicks the button "Start Session" in the tab. All Clients will automatically connect to the Main Master. His project becomes the so called Session Project.
All other Masters have then the option to join this session. Alternatively, the Startup menu offers the possibility to start a project and a sync session directly and to join one.



When other Masters join, they are asked to load the Session Project and become Sub Masters. A Sub Master can control a Client but the connection is through the Main Master. **Important note:** If a Sub Master chooses the command "Take Over All Sites in Domain" (from the Backup Menu [123] or the Assets [131] tab) he will become the Main Master!
Sub Master can leave and (re-) join the Sync Session at any time. Please note that other Users are not notified if somebody leaves the session.
The ASIO audio settings [166] in the Configuration tab allow to choose an "Audio Clock Master". Only the Main Master or any Client with a setup ASIO device can be chosen. Sub Masters cannot be clock masters, but only slaves.

The Multi-User tab shows the status next to each system, e.g. "in Sync Session". The status bar shows "MAIN" or "SUB" in the right corner next to the version.

As described above, all Masters are (in-)directly connected to the Clients. In the Device Tree, Clients are shown with the same icon. Either it is the connected one, or the one with the red exclamation mark on top in case the Client is not connected. The icons in the Main Master are colored white as usual. The icons in all Sub Master are colored in beige to emphasize that the connection is through the Main Master.

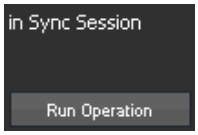
In short: the Main Master is connected to the Clients, hosts the Sync Session and can be the clock master [166]. As described below, he cannot enter the Blind Mode.

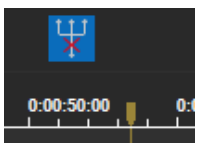
Within a Sync Session all changes are synced between all Users, including the:
- Device structure (for the Device Tree and Sequence),
- Sequences (with all Containers and Cues),
- Resources (meaning folders and content in the Project tab),
- Active Values and Sequence Playback (if not in Blind Mode).

Tip: In a Multi-User setup you might not want to spread all content to all Masters. First, deactivate the Auto-Spread option in the Configuration tab > Resources (Global) [155]. The right-click menu in the Project tab then offers options to spread content or folders to specific sources. In addition, the Folder Inspector [197] offers spread options.

Troubleshoot: In case you encounter problems like not seeing another User, please check your network. Make sure that the ports used by Pandoras Box [781] are not blocked (firewall, anti-virus applications, configurable switches etc.). If you are using configurable switches and would like to use IGMP Snooping, make sure that ALL switches in your network are configured accordingly. Otherwise, IGMP Snooping can be deactivated as well.

Blind Preview and Blind Programming

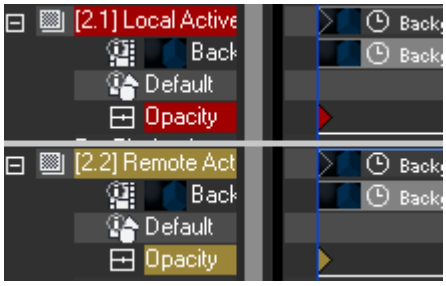




As explained above, all project data is synced in a Sync Session. If you like to detach the Active Values and Playback status (including the time for the Nowpointer), you can activate the option "Blind Mode". Note that this option is only available for Sub Masters (those that join an existing session). You can either click the according button in the Multi-User tab or the last icon in the Sequence Button Bar [293] above the timeline.
Now you can use the partly synchronized Master as a Blind Preview system or a Blind Programming system. It is for example possible to change the timeline to the next cue and preview or verify the content. Or you can edit the scene by activating parameters and storing them to the timeline without interfering with the current output of the show. However, you need to make sure to not work at the time from the Main Master's Nowpointer! Note that content spreading also uses performance on the Main Master.

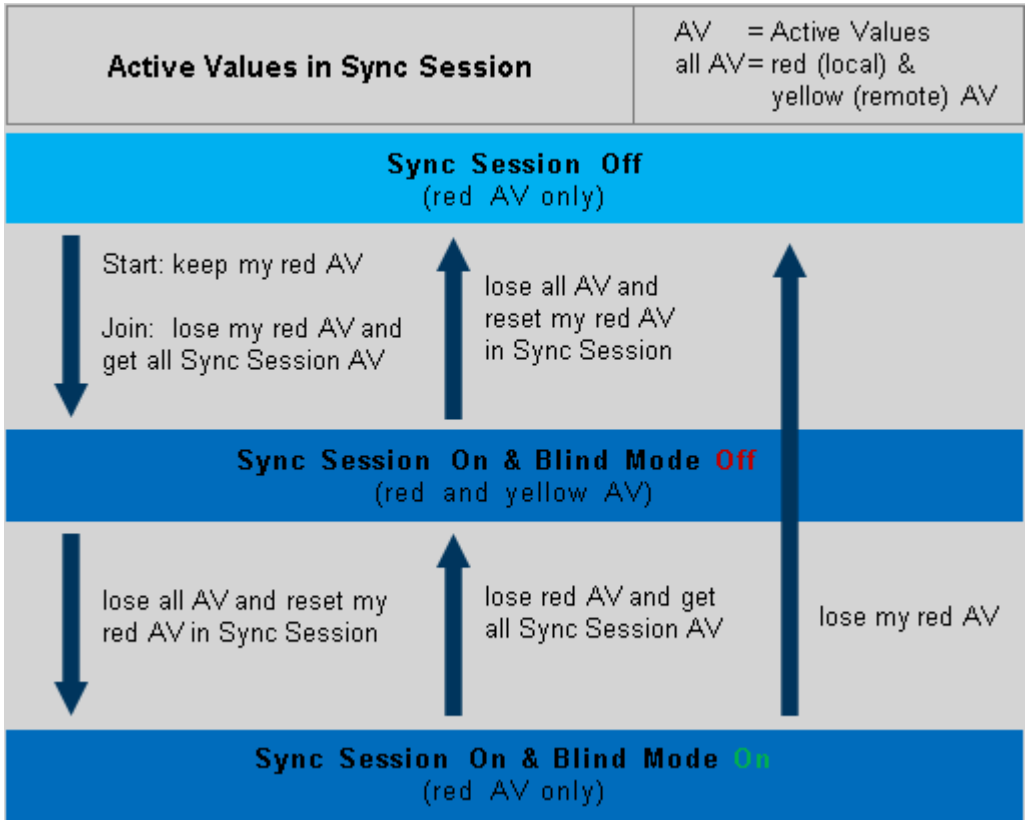
When the Blind Mode is activated, the Multi-User tab of other systems will display the status: "In Sync Session" and "Is in Blind Mode".

Local and Remote Active Values



With the possibility to synchronize active values, the necessity arose to distinguish local active values from remote active values. If active values are synchronized, their effect in the Preview is visible for all Multi-Users. However, the programming behavior is different and the GUI display reveals their origin:
Local active values are highlighted in red whilst a remote active value is colored in beige / yellow.

All "Store Active" commands will only store red local active values!
You can right-click on a Parameter / Layer / Client and choose to either "Reset" (respectively "Reset All Active from Remote") or to "Take over Activity" in order to adopt them to local active values.
Note that the tab "Active Values [127]" displays local and remote active values without distinction.
Remote active values can only exist in a Sync Session without an activated Blind Mode. Hence, they dis-/appear when you toggle the Sync Session state or the Blind Mode on or off. In addition red local active values may change. Please see the following diagram.



Saving and Loading Synchronized Projects

The "Save" and "Save as" option save the session project on all synced systems. The chapter "File menu [119]" difference between both options, and the third saving option "Save as Copy" are

When starting with all Users on another day, the Main Master loads the project and starts the session again. All others can then join it as usual. This option is also included in the Start-up dialog. Note that the saved project data does not include information about other Users.

User Profiles

Since Pandoras Box V8 you can save specific settings from the Configuration tab in so called User Profiles. Per default, User settings apply to the local instance only. However, the profiles are part of the project and also synced with all Multi-Users. Note that if no other profile is created and selected, all Users rely on the "Default" profile. This

has the result that changes done locally are synced and applied to other instances too. But any User has the choice to create or select an individual profile. All profiles except the "Default" one can be renamed or deleted, which again is synced. Keep in mind, that profiles must have a unique name and cannot be deleted if another Multi-User uses it. Read more about User Profiles in the chapter "Configuration 139".

Asynchronous Programming

So far, continuous synching was explained. For some events however it might be more convenient to not sync every small step but to work asynchronously, at least for a certain time until a more or less final state is achieved and should be shared.
Before you begin, it is recommended to decide for a hierarchy as it makes sense that there is a leading Master to which all others send their selected project data.

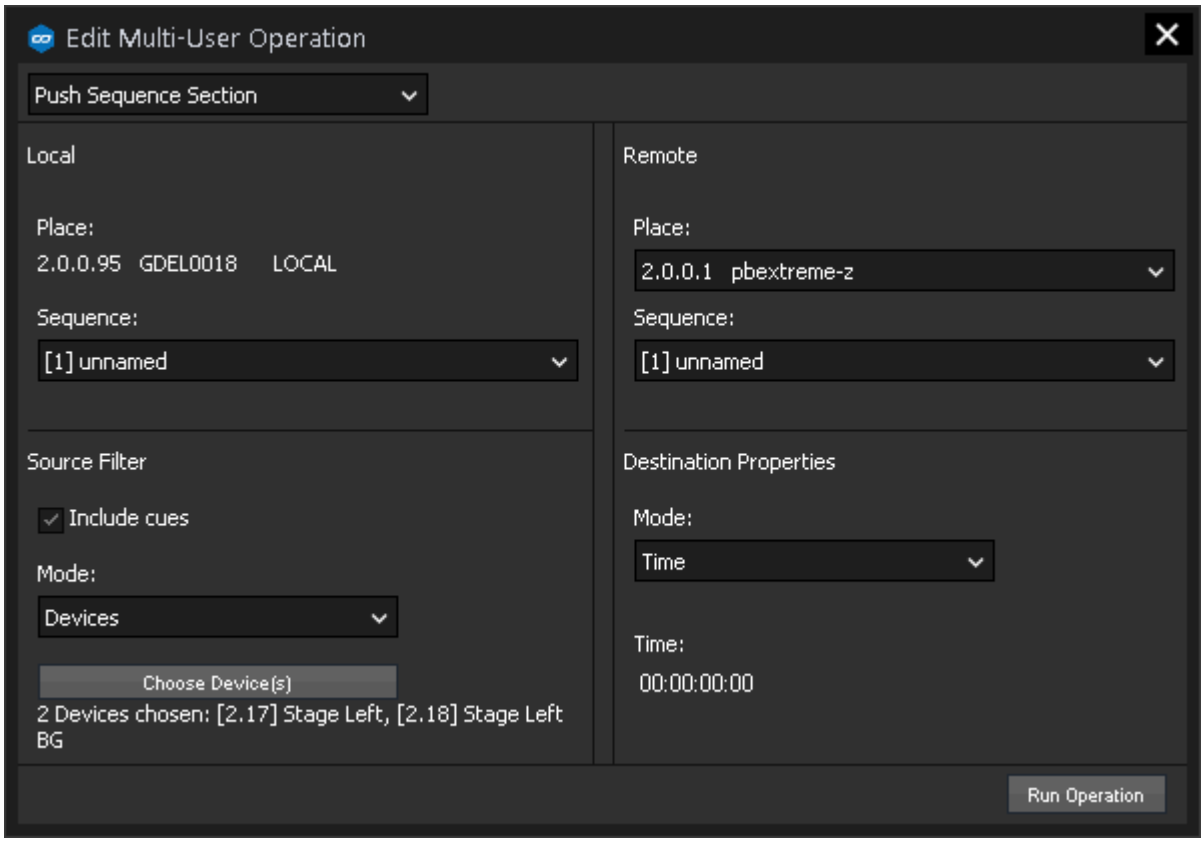
First, decide whether you want to work in a separate project or whether you like to start with the same project that includes for example already some Clients with named Layers and some Resources in the Project tab. For the later case, there are two possibilities. First, you can open a Sync Session as described above and leave it again after having synced the project once. Alternatively, one Master loads the project. All other Users open the Multi-User tab and click on the "Run Operation" button depicted next to the network address and name of this Master. This opens the "Edit Multi-User Operation" dialog where the drop-down menu offers the possibility to "Pull Project" which copies the project to the local system.

Now every operator can work in his own project and click the "Run Operation" button whenever he needs to.

Run Operation

As described above, the "Run Operation" button is for operators who program asynchronously. If the operator likes to distribute (parts of) his work, he clicks the "Run Operation" button depicted next to the network address and name of another Master. This opens the "Edit Multi-User Operation" dialog where the drop-down menu offers the following possibilities:

Operation	Description
Pull Project	Load the currently opened project from another system.
Push Project	Send your currently opened project to a another system. The chosen system loads the project without asking to save its own current project.
Pull Sequence	Choose a remote system and a Sequence ID. The other Sequence is added to your Sequence folder in the Project tab, meaning that it might get a new ID.
Push Sequence	Choose a Sequence ID and remote system to send it to. The other system adds your Sequence to its Sequence folder in the Project tab, meaning that it might get a new ID.
Push Sequence Section	Send a part of your Sequence to another system. You can define the source Sequence ID, and whether Cues are included or not. You can send the entire Sequence ("No Filter") or a part which can be defined with a starting and ending Cue or with Devices. In the Device dialog you can multi-select using the [Shift] and [Ctrl] key. Regarding the receiving system you can choose the remote Sequence ID and whether your data should be added at a certain time or Cue.
Transfer Resources	Choose a remote system to view its content tree next to your own content tree from the Project tab. You can now drag and drop files and folders from left to right and vice versa.

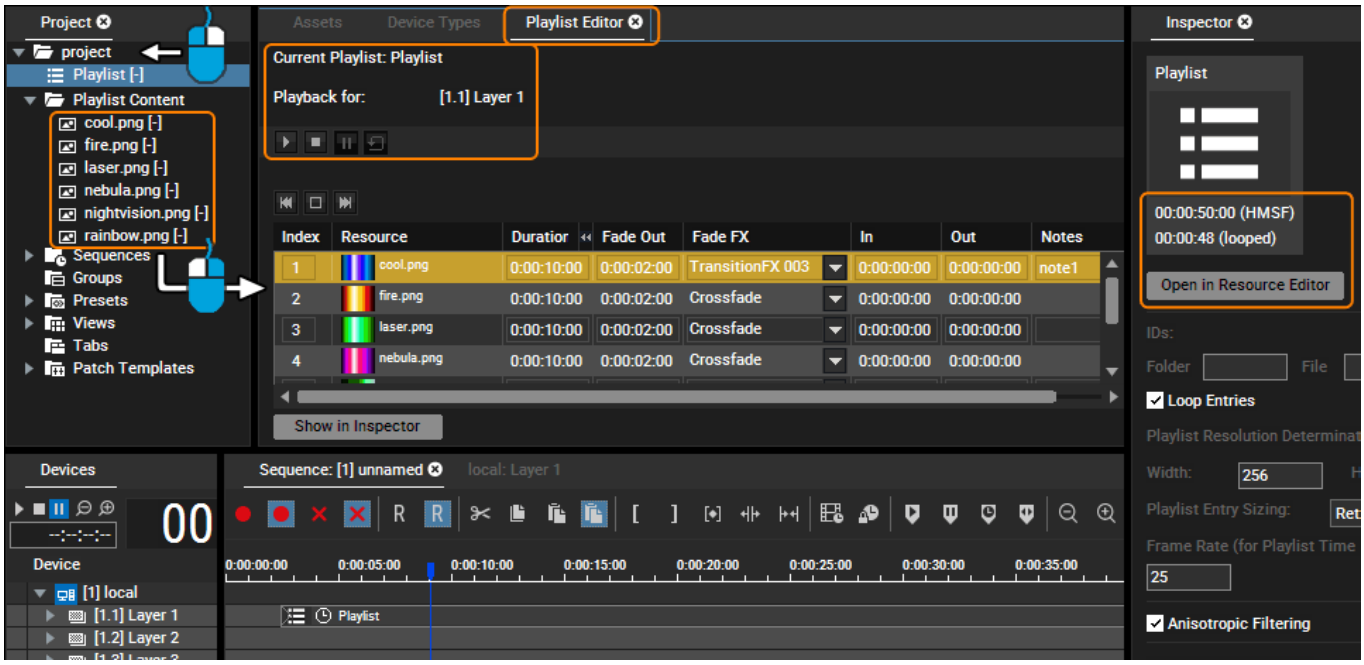


TCP and Art-Net Output

In MU scenarios, DMX devices [691] as well as the TCP device or Widget Devices [696] will behave similar to regular Site. This means they will "manifest" onto the Main system and will not send out any data from Sub Master. Therefore the Art-Net output on Sub Masters can be activated, but no value will be sent. In the event of a "takeover" the new Main Master will establish all TCP connections and start to send Art-Net if specified in the configuration. Doubled Art-Net output is prevented the same way as TCP or WD commands are not sent twice.

6.3.4.21 Playlist

A Playlist offers an easy and fast possibility to play a certain number of media files in a row without the need of programming separate containers on the sequence.



Adding, Deleting and Reordering Content in the Playlist

To create a Playlist, simply right-click any folder in the Project tab [279] and choose "Add Playlist". Select it and click the button "Open in Resource Editor" in the Inspector to load the Playlist in the tab "Playlist Editor". Alternatively, you can also switch to the View named "Content".

Now you may drag and drop the following files from the Project tab into the Playlist tab:
- images
- image sequences
- videos
- audio files

Each newly dragged resource will simply be added to the end of the list (even when the Playlist is already playing back). If you like to put the files in another order, (multi-)select an entry and drag it between to others or assign a different "Index" number. Deleting files is possible with the Delete key on your keyboard or by choosing the respective command in the right-click menu.

Duration and other Settings in the Playlist

The duration of images is automatically set to 10 seconds; all other media keeps its real duration. You may change the duration, fade time and transition effect - again, multi-selection applies the chosen setting for all selected items. Further to the right you find "In" and "Out" times that take effect on videos or audio files only and define the starting and ending frame. Notes for internal use can be added too.

The overall duration of all files minus the fade times result in the length of the Playlist. The fade times need to be subtracted as fading two files is obviously shorter then playing them back one after another. The duration is displayed in the Playlist Inspector [202]. It offers general settings like Folder and File ID, resolution or aspect mode.

Assigning a Playlist to a Layer

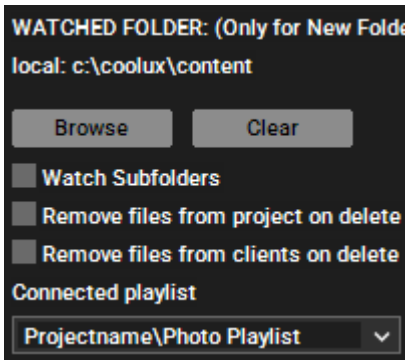
As a last step, the Playlist needs to be assigned to a Video Layer. You may assign it as an active value or program it on the timeline as one single container. Please note, that you may assign the same Playlist to several Video Layers. Then the playback status can be different per layer, e.g. one layer can be in pause mode showing the file with index 5, the other in play mode between index 10 and 11. The Playlist tab offers a drop-down menu containing all layers the Playlist has been assigned to. Depending on this drop-down menu, the Playlist changes and displays the layer's playback mode and the currently active index file with a green highlight. In the image above the chosen Playlist on Layer 1 is currently fading between two images.

Please note that not only during the cross-fade the Playlist consumes the memory for the current and the following content file, thus it needs the same performance as assigning the assets separately to two layers!

Playing ASIO Audio with Videos

If you setup an ASIO device in the configuration tab > ASIO audio [166], you see there also the option "Play Audio Files in Parallel with Playlist Entries". This automatically plays the "according" audio file with the video in a Playlist [239]. The audio file does not need to be part of the project, but it has to be named in the same way as the video file and needs to be located in the same folder. Please note that the duration of both files does not need to match.

Linking a Playlist to a Watched Folder



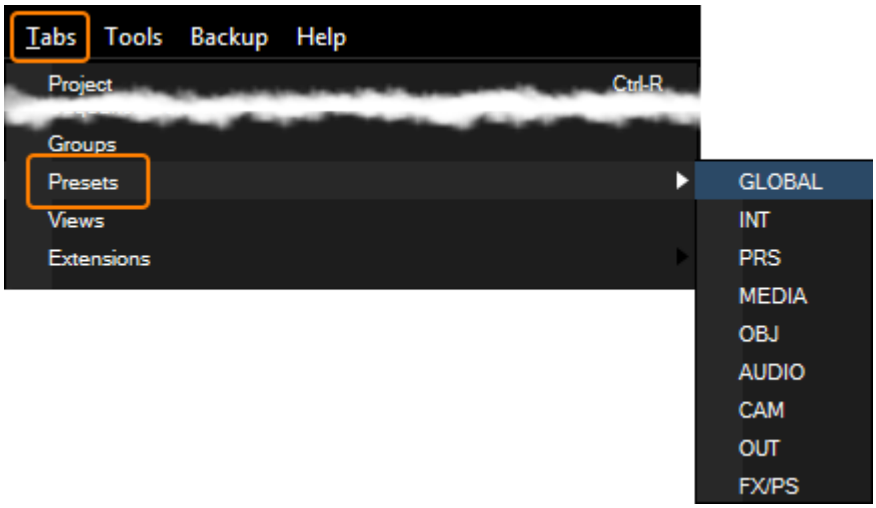
Since Pandoras Box version 6.4 we features the functionality to directly link a Playlist in Pandoras Box [239] to a watched Windows folder. This way Pandoras Box automatically adds and removes files to its Playlist. It can therefore run in standalone mode and does not need to rely on another application like Widget Designer [786] performing this task. This makes it easier and faster especially for fixed installs or when dropping photographs into Pandoras Box on the fly. You can find the new options in the (Watch) Folder Inspector [197].

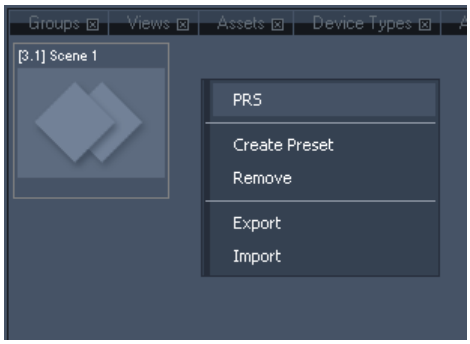
Step-by-step instruction:
- In the project tab, right-click a folder and choose "Add Playlist" and afterwards "Add Folder".
- Select the folder and look at the bottommost options named "Watched Folder" in the Inspector. (Later on, you might need to open the folder attributes by clicking the topmost + icon.)
- Click "Browse" and select the Windows folder you like to watch. Now, additional options appear for the Watch Folder.
- Open the drop-down list and select the Playlist you like to link to the Watch Folder.
- Optionally, select the Playlist and in the Inspector, click the button "Open in Resource Editor". All files from the Windows folder are part of the Playlist. If you are missing files, check whether the format can be read from Pandoras Box [90].

6.3.4.22 Presets

The Presets tab lets you create, edit, paste and apply presets in a browser view. To open a preset tab, navigate to Tabs - Presets and choose one of the preset banks to be shown in the tab. You may as well make a right-click on a preset type in the project tab and choose "Open Browser".

Please note that the preset feature itself is described in the Preset 283 chapter.





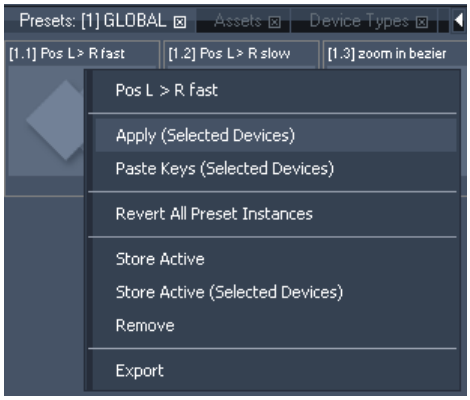
The context menu of the Presets tab provides the following options for creating and organizing presets:

[Create Preset]
Creates a snapshot preset by looking at the current active values and copying them as a preset key (at zero time) according to the current preset bank.

[Remove]
Deletes the entire preset bank and all including presets. A new preset bank can be created in the Project Tab

[Export]
Saves all including presets as an XML file.

[Import]
Adds the previously exported presets to the preset bank.



The context menu of the preset itself provides the same options for creating and organizing presets as in the project tab (except the rename function):

[Apply (Selected Devices)]
Applies a preset, starting at the first selected layer and activates those layer parameters that are stored inside the preset. The values are nor inserted into the timeline.

[Paste Keys (Selected Devices)] or make a single left-click on the preset
Creates a new instance from the preset by pasting its keys the into the currently open sequence starting at the first selected layer.

[Revert All Preset Instances]
Discard the changes made in one or more instance and make them all comply with to the master preset again.

[Store Active]
Stores all active parameter values to the preset that are allowed to be stored in this preset type.

[Store Active (Selected Devices)]
Stores all active values of the selected devices to the preset.

[Remove]
Removes the preset.

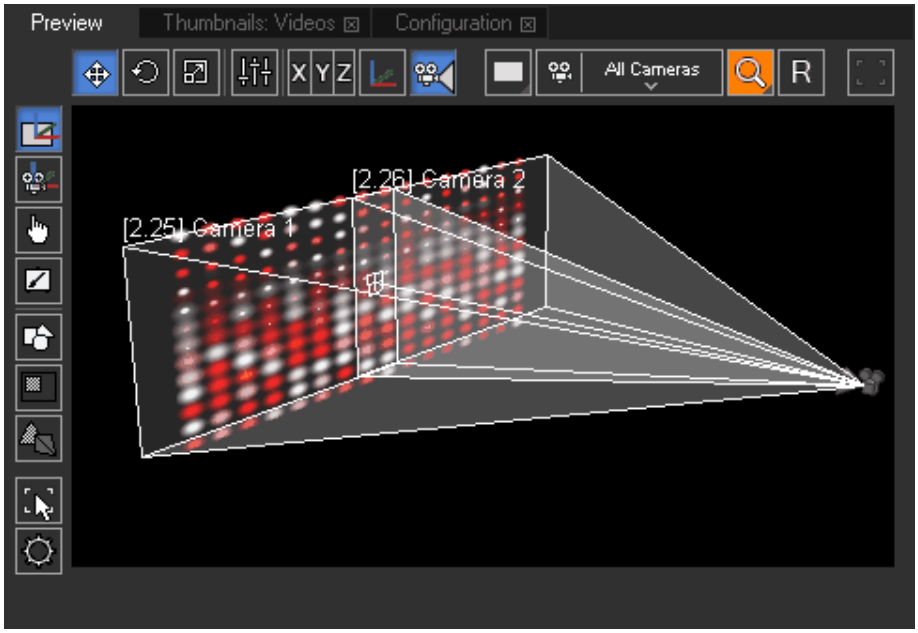
[Export]
Saves all including presets as an XML file.

6.3.4.23 Preview

The Preview tab in Pandoras Box renders your programming. Further, you may edit Layers, Cameras, Meshes etc. directly in the Preview.

With version 6 we introduced new powerful features and enhanced the structure of the Preview by adding buttons and drop-down lists around the main window. They call....
Modes: These buttons are fixed, each mode is covered in the linked chapters.
Tools: These buttons vary because they depend on the chosen mode.
Views: These options are fixed and influence what the Preview displays, e.g. multiple view ports with Camera and Output views [245].

Modes	Tools	View



Layer [250]

Camera Transformation [250]

Picking [252]

Canvas Editing [256]

Mesh Editing [257]

Planar UV Mapping [269]

Perspective UV Mapping [269]

Per default, the Preview tab is part of the main user interface. However, you can also choose to work in a maximized Preview. The according bottom left button or the shortcut [Ctrl + Shift + F] toggle your Preview tab to a fullscreen window. All tools, modes and view options are available within the Button Bar (shortcut [T] to show and hide it). In addition you can navigate [248] and use the mouse cursor as usual. Each Master offers this feature.

If the Master has Output devices listed in the Device Tree, it offers an additional function as their Preview is also their Render Output. As this is not the editable Preview anymore, there is no Button Bar etc. and you cannot switch views. Use the according top right button or the shortcut [Ctrl + F] to toggle to fullscreen.
In case you work with multiple displays, you can choose your display adapter within the Configuration tab > Render Engine [162].

In case you work with multiple displays, you can choose your display adapter within the Configuration tab > Local Preview [150] (also accessible through the settings button). More Preview settings can be found under Configuration > Preview Display [145].

Note that the Preview window is protected with a watermark. It only appears in the Preview, never in the Output!

If working with Art-Net LED walls, you can toggle the patch on and off with the "Select Matrix Mask" command from the Previews context menu [275].

Important information about the Preview

First, the Preview of the Master renders the content from all Clients in full quality. To render efficiently, you can decide which Client(s) are included in the Preview. The next chapter [244] explains how to toggle a Client's Preview on and off. This is of interest in case you work with many Clients and / or content with high resolution, frame rate,etc..

Second, in case you work in a Master-Client setup, content needs to be copied to all hard drives. If the Master is missing the content, its Preview cannot render it. If the Client is missing the content, its output cannot render it even though the Preview shows it. The Preview does not provide a real-time feedback of the Client's outputs!

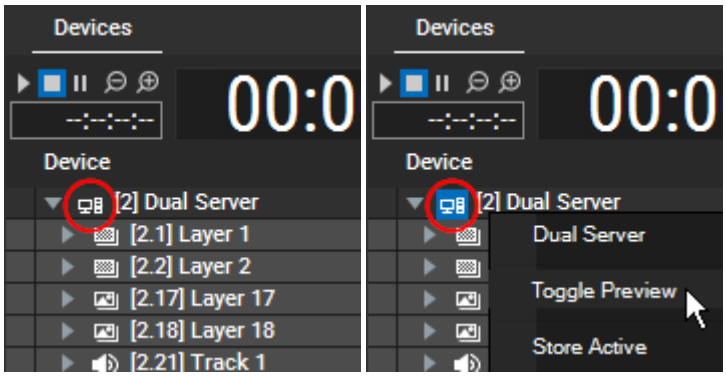
Third, the Preview allows you to pre-program a show without having Clients connected (yet).

The next chapters:
- let you know how to include Sites into the Preview [244]
- explain the different Camera or Output views [245]
- describe how to zoom, pan and rotate [248] the Preview
- explain what wireframes and pivot handles [249] are for
- inform how you can use the and Preview as an interactive interface with pivot handles [250] (Layer Mode and Camera Mode) and Layer Picking [252] (Picking Mode)
- cover the advanced Preview Modes: Canvas Editing Mode [256] for drawing and masking, Mesh Editing Mode [257] for warping and last, Planar and Perspective UV Mapping Mode [269] for texturing
- describe the context menu from the Preview

6.3.4.23.1 Including Sites into the Preview

This chapter explains how to toggle Sites to the Preview. For other topics regarding the Preview tab please see the introductory chapter [243].

Per default, a Site is not included in the Preview when added to the project. The left image shows the Device tab [173] with a Site excluded from the Preview; note that its icon is not highlighted.



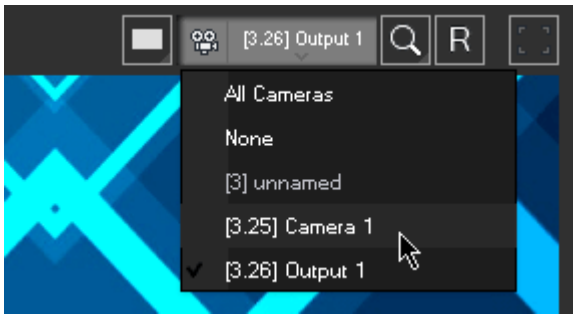
To add a Site to the Preview rendering, right-click it to open the context menu and choose "Toggle Preview". Now, the icon is highlighted in blue as seen in the right image.
Please note: In order to see the Client(s) in the Preview, the Master needs to render their files. It does not scale the content in any way! That means that the full content needs to be encoded and displayed. The more Clients you add to the Preview tab, the more content files are playing, the higher the content's resolution or frame rate is, the more performance is required on the Master PC. Please remember to spread the content to the Master and the Client likewise.

In case you are using the Audio Tracks [661] from the local Site, keep the Preview activated as video and audio output is bound to the "Preview" functionality.

The next chapter explains the preview views [245]. For example you can load a Site's Camera or Output into the Preview or you can preview all Sites at once with the "All Cameras" view.

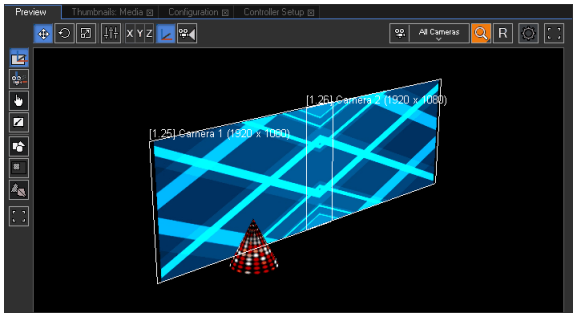
6.3.4.23.2 Select Preview: Camera or Output View

This chapter explains the views for the Preview. For other topics regarding the Preview tab please see the introductory chapter [243].



You can load various views into the Preview tab, there are three ways to do so:
- open the upper right drop-down menu "Select Preview"
- right-click in the Preview and select "Select Preview"
- right-click on a Camera or Output layer in the Device tree and select "Load in Preview"

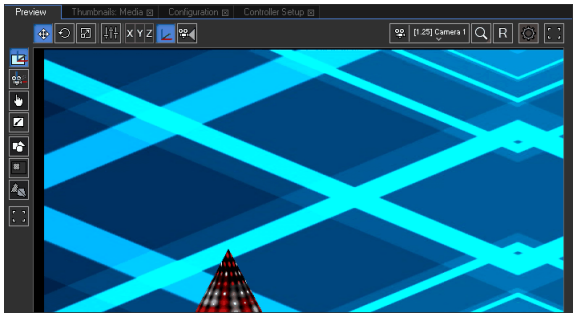
The "All Cameras" view is always available but the Camera or Output view(s) depend on the Camera or Output Layers from your Sites / Clients.



The left image is an example for the "All Cameras" view. It allows previewing all included Sites at once. Furthermore the whole 3D space is shown instead of only the specific area seen by the Site's Camera device.
The next chapter explains how to navigate in the Preview tab [248].

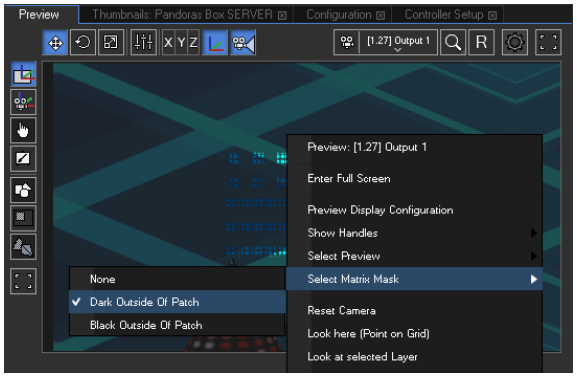
Per default, Camera targets are shown as white rectangles with a label hovering above it naming the Camera Layer and resolution. Resolution and aspect ratio can be changed in the Inspector of the Camera [218], whilst Camera position etc. can be changed in the Camera Device [675].

In addition to the rectangle you can toggle more wireframes and handles. This is explained in one of the next chapters: Preview handles and wireframes [249].



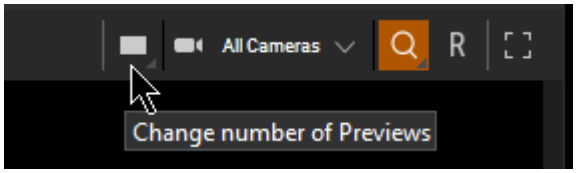
The Camera view shows what a Camera from a Site "sees", that is the so called render target or Composition pass. The render target is forwarded to the Output Pass where Output parameters like Softedge or Warp can change it. Hence the Output view shows the final result, the so called back buffer, that is also output through a Client's graphics card output. The chapter "Video Processing pipeline" [320] explain the render passes in more detail.
The left image shows the left Camera from the scene depicted above.

By the way, renaming Cameras and Output Layers in the Device Tree also changes their names in the drop-down menu.

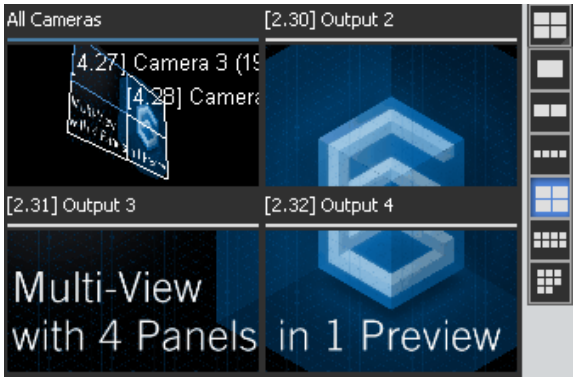


When working e.g. with LED walls that are controlled via Art-Net you can choose to preview the Art-Net output instead of the video (e.g. DVI) output. Whilst being in the Output (!) view right-click and select a matrix mask. To use the matrix feature you first need to create a patch in the Matrix Patcher [2077] and export it to your Pandoras Box project. Drag the new format ".pbx file" on any Output Layer [682]. Now, you may choose to preview it. Right-click in the Preview > Select Matrix Mask > Dark or Black outside of Patch. **Dark Outside of Patch** dims the area not covered by the patch whilst the **Black** option does not display it at all. Choose **None** to see the entire area without the highlighted patch.

Multi-View



Since version 6 the Preview can be toggled to a Multi-View window. Per default you see one view port as usual, but now you can also choose to work with a split view with 2, 4 or 8 view ports in various arrays.

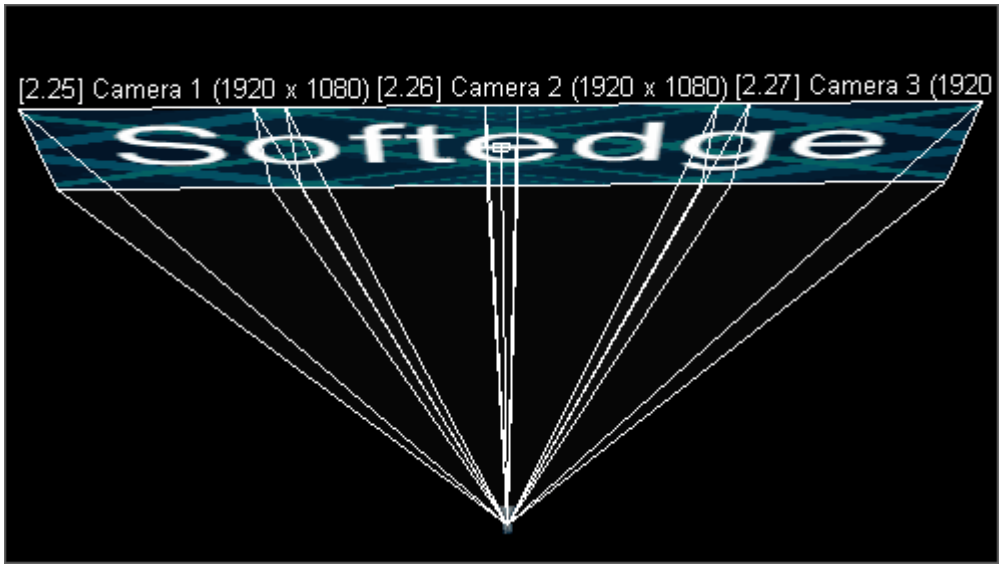


If you have loaded the "All Cameras" view before toggling to more view ports, the newly created ones will show "None"; in case you have loaded a Cameras (or Output) view, the newly created windows will automatically show other Camera (or Output) views if they are available. Of course you can change each view port individually using the drop-down menu. Please note that for the time being it is not allowed to view a Camera and the corresponding Output at the same time. Another restriction in this version is that one view can be loaded into one view port only. In other words, the "All Cameras" view or "Camera 1" can not be depicted twice.

Independent from the number of view ports, it is possible to maximize the Preview to full screen. This is especially of interest in case you warp several Outputs in the Master using the Editable Mesh feature.

Previewing different Camera offsets

In the example below a softedge projection with 4 outputs is shown (the 4 Cameras have different X-offsets).



When changing the Preview to show the single cameras, each one would show its area according to the wireframes:

Camera 1

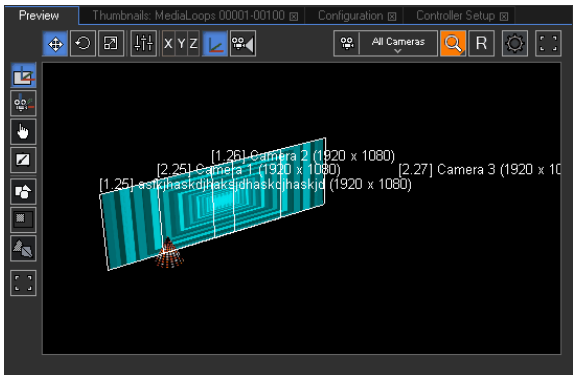
Camera 2

Camera 3

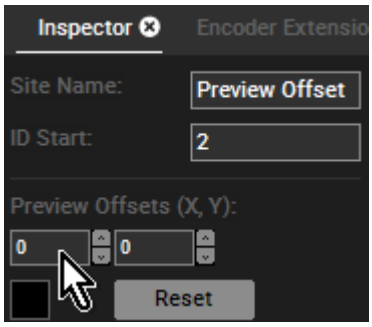
Camera 4

Previewing several Clients with equal Camera offsets

Having Sites that run separately from each other (with no X / Y Camera Offsets), the Sites outputs will overlay in the Preview by default. The output at top is from the Site that was added to the project at last. In the Inspector of Site 1 and Site 2 the Preview Offset may be changed. This way both Sites can be shown at the same time.



When changing to the All Cameras view, only Site 2 is visible, because it was added to the project at last, it overlays the Preview image of Site 1.
In the Device Tree tab select Site 1 or 2 and turn over to the Inspector...



...and modify the Preview Offset of a Site by

(1) entering X and Y values manually in the text fields or
(2) click in the little black box and move the mouse while holding the left mouse button clicked.

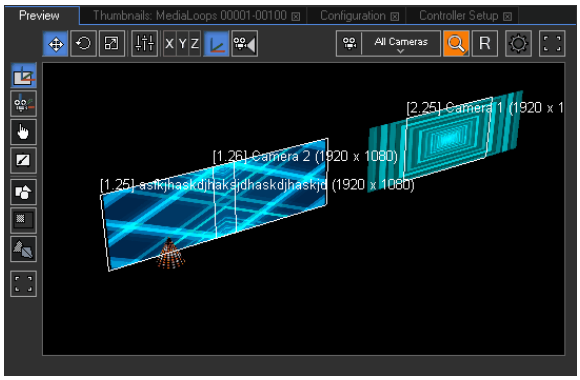


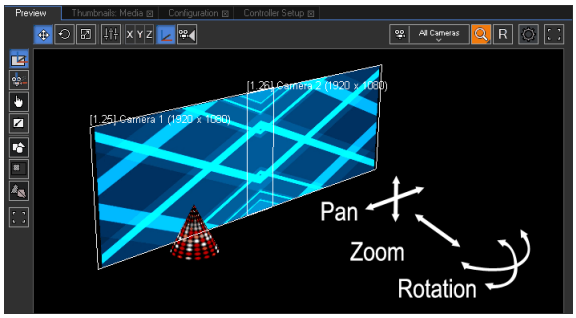
Image left: having different Preview Offset values applied to the Sites allows you to see both at the same time.

The next chapter 248 explains how to rotate, pan and zoom within the Preview tab.

6.3.4.23.3 Navigating in the Preview

This chapter explains the navigation in the Preview. For other topics regarding the Preview tab please see the introductory chapter [243].

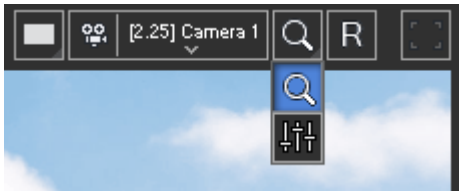
Within the Preview you can look at your composition from any point: you can zoom, rotate and pan. You can do so in the All Cameras view as well as in a Camera or Output view. Views are explained in the previous chapter [245].



For **zooming** use the mouse scroll wheel.

For **panning** keep the middle mouse button (scroll wheel) pressed whilst moving the mouse. Panning changes the Viewpoint and Target at the same time

For **rotating** keep the middle mouse button (scroll wheel) and the [Alt] key pressed whilst moving the mouse. Rotating the view changes the Viewpoint only.



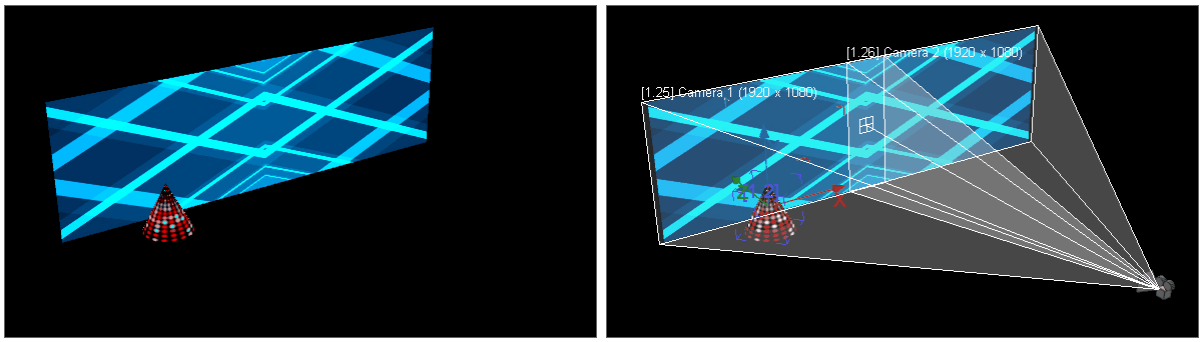
Per default, the **Camera Interaction Mode** is set to "Zoom". As soon as you change the view, e.g. rotate it, the Zoom button is highlighted in orange. The R-button applies the default view and resets your changes. The same happens when you use the shortcut [Ctrl + 0] or choose the command "Reset Camera" from the right-click menu.

The other Interaction Mode, "Parameters", is of interest when you are in a Camera or Output view. Any view navigation is now represented in active Viewpoint and Target parameters, hence can be stored to the timeline. In this mode, only the R-button resets parameter changes.

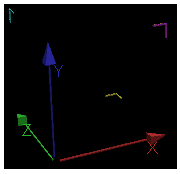
The next chapter [249] explains what Gizmos and Camera Frustums are ...

6.3.4.23.4 Gizmos and Wireframes in the Preview

This chapter explains the various handles that can be rendered in the Preview to help you in your programming. For other topics regarding the Preview tab please see the introductory chapter [243].



The left image shows the Preview with only the layers visible.
For the right image all handles are turned on:



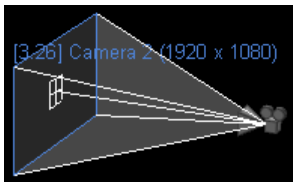
- Gizmo

In the example, the Gizmo is depicted on the Layer with the red cone. It is the icon with three axes in the colors red, green and blue. Many objects in Pandoras Box have a Gizmo: Layers, Cameras, Mesh points, etc. but they all work in the same way.

Enter the Layer Mode and click the "Gizmo Visibility" button  . Now, activate the Move Mode and change the Layer's position by dragging one of the axes. Note that the axis highlights as soon as the mouse cursor touches it. When the mouse touches one of the corners (e.g. the yellow one) both associated axes highlight (red and green).

A Layer can be moved, rotated and scaled and each transformation has its own Gizmo. The center of each Gizmo marks the Layer's Pivot Point for the selected transformation. For more information see the next chapters about the Layer and Camera Transformation Mode [250], the Rotate [652] and Scale Mode [653] and the Pivot [654].

As said above, not only Layers have Gizmos. Depending on the mode you are in, the "Gizmo Visibility" button toggles Gizmos from other objects.



- (Camera) Frustum and Wireframes

The "Camera Visibility" button  toggles the Camera icon and the Frustum to visualize the opening angle and Look-at point. The Frustum includes the Camera Wireframes (here in white) and Area (in blue) plus the name (also blue). The Frustum's appearance can be changed in the Camera Inspector tab [218].
A Camera can be moved with the Gizmo in the same way as described above. The blue highlighted Camera area can also be dragged.

The "Camera Visibility" button can be found in the Layer and in the Camera Transformation Mode. In case the Camera Frustum Visibility is toggled off, only the blue rectangle and name is shown. To switch these items off, please go to the Configuration tab > Preview Display [145].

Following the Camera's visualization, a Light Device [666] has its own icon and Frustum. The style can be changed in the Light Inspector [214]. The visualization is not influenced with the "Camera Visibility" button but can be toggled in the Configuration tab > Preview Display [145].

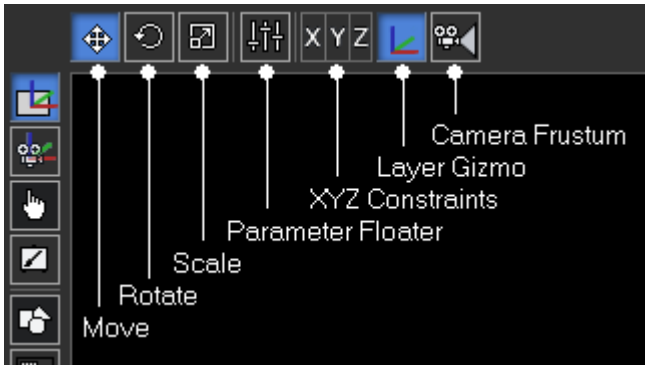
The Configuration tab > Preview Display [145] offers more settings influencing the appearance in the Preview for example Ground Grid and shading settings. All settings are also available for the Client Display under Configuration tab > Client Display. [158]

The next chapter explains the Layer and Camera Transformation Mode [250].

6.3.4.23.5 Layer Mode and Camera Transformation Mode

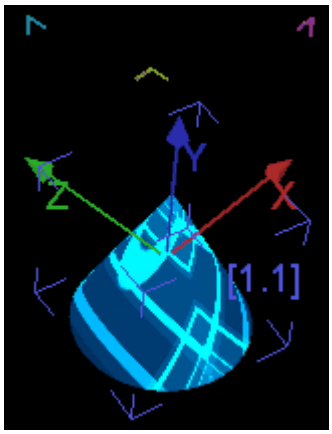
This chapter explains how to interact directly with Layers and Cameras in the Preview. For other topics regarding the Preview tab please see the introductory chapter [243]. The chapter "Navigating in the Preview" [248] explains how to pan, rotate and zoom the Preview. All keyboard shortcuts are listed on the Shortcut [314] page.

The Layer Mode allows to select, move, rotate and scale Layers directly within the Preview interface. For further Layer interaction please see the next chapter about "Layer Picking" [252].
The Camera Transformation Mode allows to select and move Cameras directly within the Preview interface. Please go to the "All Cameras" view [245] first.



In both modes you can:
- move the selected Layer or Camera; Layers can also be rotated and scaled
- open the dialog "Parameter Floater" showing the parameters from the selected Layer or Camera
- set a constraint, e.g. an activated (blue highlighted) X constraint transforms along the X-axis only
- toggle the Layer and Camera Gizmo [249]

Layer Mode



First of all, please go to the "All Cameras" view [245] and then activate the Layer Gizmo  , the button should be highlighted in blue.

Select a Layer by clicking on it in the Preview or the Device Tree. Now, the handles for the X, Y and Z-axis of the selected Layer are shown in the Preview, as seen in the left image. The so called bounding box and the Device ID of the selected Layer are shown as well in dark blue (in this example: device ID 1.1).

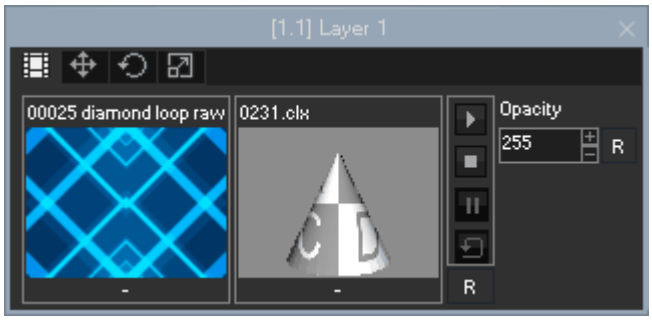
Now, choose whether you would like to move, rotate or scale the Layer (shortcut [1], [2],[3]) by clicking on the according button above the Preview. Click anywhere on the Layer and drag it. Note that the transformation is applied in the XY-plane. When holding the [Shift] key, the transformation will be applied along one axis only. For example if you move the mouse from left to right the X-axis is taken but as soon as the up-down movement is greater, the Y-axis is taken.

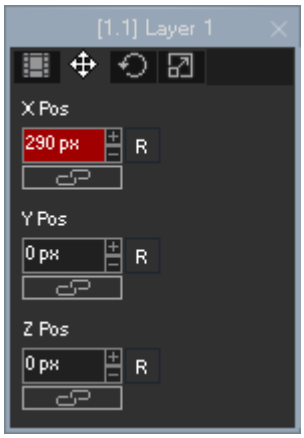
If you now position the mouse pointer on one handle of the Gizmo, it is highlighted in orange. Dragging the mouse results in a transformation along one axis. You can also work in another plane by highlighting one of the corners shown in magenta, cyan and yellow.

As an alternative to working in distinct axes, you can apply constraints. With an activated X Constraint (shortcut [X]) the Y and Z-axis are shown in gray and any transformation happens along the X-axis only.

For the Rotation [652] and Scale Mode [653] you can also change the Pivot point. Drag the Layer whilst holding the [Alt] key. The topics about the Rotation Pivot [654] and Scale Pivot [655] show how a moved Pivot affects the transformation.

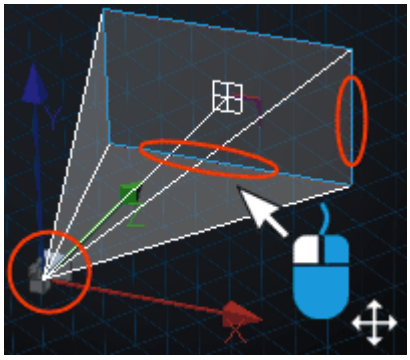
To multi-select several Layers, press [Ctrl].





The Parameter Floater (shortcut [Ctrl + P]) shows the most important parameters from the selected Layer. Of course they are synchronized with the Device Control tab [171]. You can choose to see the parameters for Media Control, Position, Scaling or Rotation.

Camera Transformation Mode



First of all, please go to the "All Cameras" view [245] and activate the Move mode.
If you now click on the Camera rectangle it highlights blue and you can drag it around. This changes the X and Y Offset parameters.
Note that constraints work in the same way as for Layers. When holding the [Shift] key, the transformation will be applied along one axis only. For example if you move the mouse from left to right the X-axis is taken but if the left-right movement is smaller than the up-down movement, the Y-axis is taken. Alternatively, you can apply constraints with the X,Y buttons.

Cameras of a Server have more parameters available: the Viewpoint [677] and Target [678] XYZ position. First, highlight the "Camera Visibility" button  and the Camera Gizmo  . Now you can either drag the Camera icon around which influences the Camera Viewpoint and Target simultaneously, or you can drag the Viewpoint and Target separately. Hold the [Alt] key down to move the Camera / Viewpoint only and the [Alt] and [Shift] key for the Target. The interaction with the Gizmo works in the same way as for Layers.

The Parameter Floater (shortcut [Ctrl + P]) shows the parameters of interest for the selected Camera. Of course they are synchronized with the Device Control tab [171]. You can see the parameters for Lens Shift and for Serves also the Viewpoint and Target.

The next chapter explains the Picking Mode [252].

6.3.4.23.6 Picking Mode

This chapter explains the (Layer) Picking Mode of the Preview which allows to use content as an interface, i.e. interact directly with it. To give an example, Layer Picking can be used to execute hyperlinks in rendered web pages, so called Browser Assets [279]. For other topics regarding the Preview tab please see the introductory chapter [243]. If you are interested how you can use the Preview's interaction data to draw on a Canvas asset, please see the next chapter "Canvas Editing Mode" [256].

Layer Picking "captures" the position from the mouse pointer and if the mouse is moved in your Master's Preview tab or the Client's fullscreen window the feature recognizes what layer is touched and where exactly on the layer the mouse is located. A layer can now be used as a true interface as mouse clicks can be passed through.
The term "mouse pointer" actually does not only refer to a mouse device but to any device that is recognized by the operating system as a pointing device and can send these input events. This includes for example an AirScan [1277] or a touch screen if they are set up to do "clicks". If you are working with an Windows XP system only one input event can be sent, whereas Windows 7 (and above) is able to work with multi-touch events.

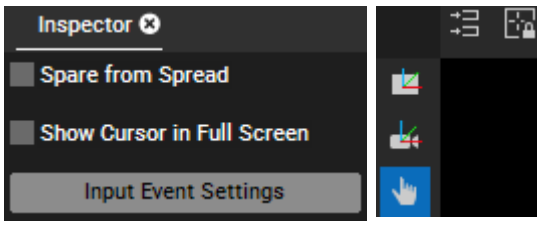
The input events can be sent from a local pointing device. Let's say for example you have a Client that renders layers in full screen that should be clicked with an AirScan. Connect the AirScan directly to the Client to receive local input events. Another possibility is to connect the AirScan to another computer which sends the input events to the Client through a so called "Mouse Injection". A Mouse Injection node can be found in the Widget Designer.

If you like to be able to click in the rendered Preview tab on your Master, simply toggle the Preview to the  "Picking Mode".

First example: Layer Picking with Web Page

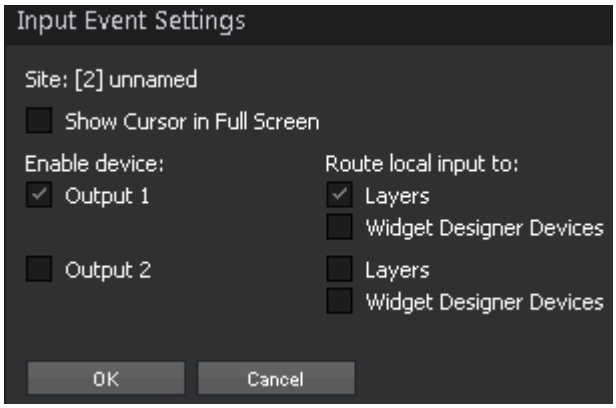
Please follow these steps to use Layer Picking with a input device connected to a Client that is toggled to fullscreen and shows a web page for example.

In your Pandoras Box project, create a Browser Asset [279]. Right-click in the Project, choose Add Browser and assign an URL in its Inspector. Then assign the Browser to a Layer from the Client. Of course you can work with a Client connected or not connected yet.
(Please do not change the default render pass, i.e. leave it assigned to the composition pass [320].)



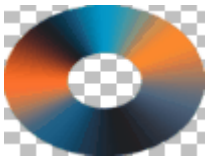
To activate click events, open the dialog "Input Event Settings".
a) Select the Client in the Device Tree tab [173] and in its Inspector [210] click the according button .

b) Toggle the Preview to the Picking Mode, and open the  "Show Input Event Settings". This opens a dialog that shows the settings for all systems in the Device Tree.

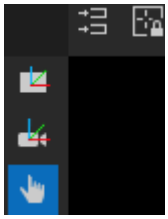


Activate the **Output** in which you like to click and the option **Layers**. This passes clicks directly through to the layers and executes underlying links for example. The Widget Designer option is covered in the next example.
Now, the mouse capturing works
- in the fullscreen window from the Client and
- in the Master's Preview if toggled to the Picking Mode in the Output or All Cameras view [245].

The check box "Show Cursor in Fullscreen" (in the Device Inspector or in the dialog) lets you display the mouse cursor on-top of the Client's fullscreen.
In the Master's Preview you will see a special click cursor.



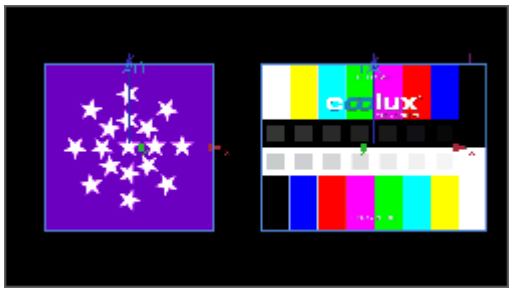
Alternatively you may use "Pointer Layers [665]"; they offer more creative options like displaying a picture or even a Particle System instead of a simple cursor. Depending on the layer's opacity, this Pointer Layer is automatically shown:
a) in the Preview tab if you are in the Picking Mode and toggled to the according Output View
b) in the Client's small render window
c) in the Client's fullscreen render window
Optional: Create a Pointer Layer, if you like. Right-click the Client in the Device Tree and choose "Add Device" > "Pointer".



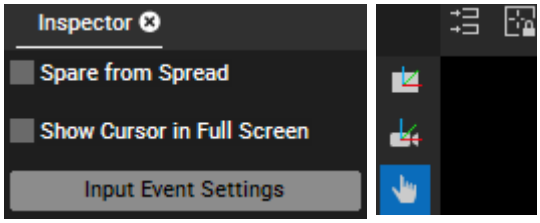
Without a Pointer Layer you will see the click cursor in the Preview. With a Pointer Layer you will see that one instead. If you wish to see the click cursor additionally, activate the  "Always Show Cursor" button.

Second example: Send Layer Picking data to Widget Designer

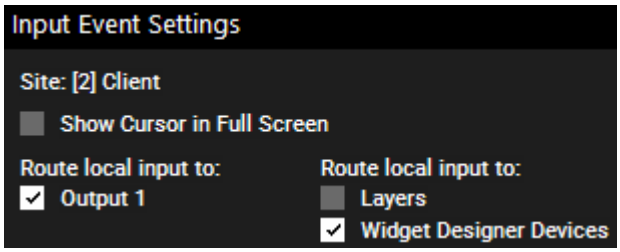
In addition to the Preview or Fullscreen interaction, you can also route the data to Widget Designer Device, to execute actions there too. It is also possible to know where the mouse is located e.g. on-top of Layer 1 and work with this data.



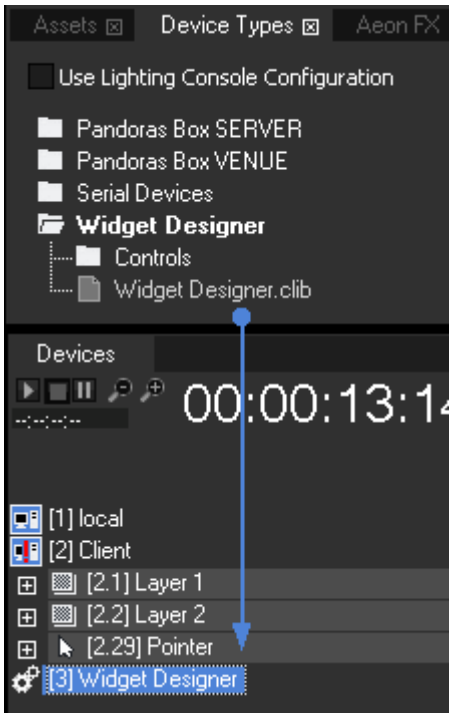
Start Widget Designer (4.0 or above) on the same or another PC.
In Pandoras Box, create a new project and program layer 1 and 2 next to each other so that your Preview looks something like the example to the left. Please do not change the default render pass, i.e. leave it assigned to the composition pass [320].



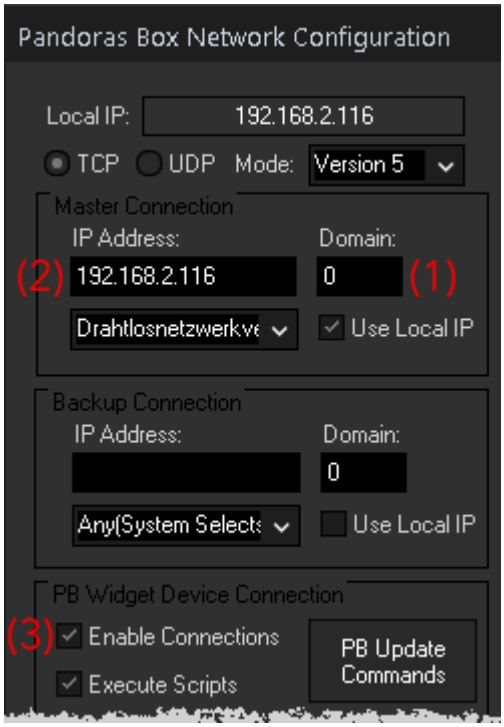
To activate click events, open the dialog "Input Event Settings".
a) Select the Client in the Device Tree tab [173] and in its Inspector [210] click the according button .
b) Toggle the Preview to the Picking Mode, and open the  "Show Input Event Settings". This opens a dialog that shows the settings for all systems in the Device Tree.



Activate the **Output** in which you like to click and the option **Widget Designer Devices** . This passes the cursor position through a Widget Designer device in the timeline to the Widget Designer application. By the way, the option **Layers** that we used before can be check or not.
The check box "Show Cursor in Fullscreen" (in the Device Inspector or in the dialog) lets you display the mouse cursor on-top of the Client's fullscreen.
In the Master's Preview you will see a special click cursor. Alternatively you can use a Pointer Layer as described above.

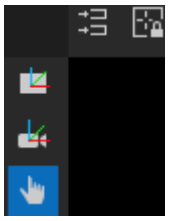


To add a Widget Designer Device [696], go to the "Device Types" tab > "Widget Designer" > and drag the "Widget Designer.clib" into the Device Tree.
Currently, the device cannot connect and is depicted with a red mark. Select the Widget Designer device in the Device tab to see its Inspector. Enter the **IP address** from the Widget Designer and the icon will not display the red mark anymore. If it still does, check the IP address and domain number from PB (in the Configuration tab) and from WD as described in the next step.



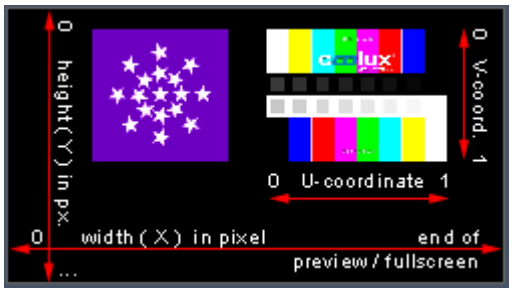
In Widget Designer, open the Connection menu and choose "PB Configuration [1256]". Then make sure that the Domain (1) and IP address (2) under "Master Connection" matches with the PC where the Pandoras Box Master is running on. In Pandoras Box, the IP address is shown in the Assets tab, the Domain in the Configuration tab.
Have a look in the section "PB Widget Device Connection" and enable the check box "Enable Connections" (3). You can close the dialog, but remember the button "Input Tester", it opens a dialog that displays incoming information.

Press [Alt + N] and create the following node: Input > Pandoras Box > Layer Interaction [1058] (the input nodes "Layer Mouse Input [1059]" and "Layer Touch Input [1060]" can be used in later projects as well). Make a right-click on the node and choose "Item Properties". In this dialog again, make sure that the IP address matches with Pandoras Box. Leave the dialog open.



So far we have set up the data routing, now we can already use Layer Picking within a fullscreen window of a Client. If you like to use Layer Picking in the Preview tab, enter the Picking Mode by clicking on the according button on the left site from the Preview and switch the view to the Output [245] you have activated in the Input Event Settings dialog.

Please note, that the data is only transferred when the mouse is actually moved.



Now we can move the mouse cursor across the Preview tab or the Client's fullscreen display and see the according data in the Item Properties dialog from the input node in Widget Designer. Alternatively you can open the "PB Network Configuration [1256]" dialog again and click the "Input Tester" button.

The node reports the Site ID and Device ID as soon as the mouse touches a layer. Of course its opacity value must be greater than zero.

X and Y relate to the width and height from the Preview tab or fullscreen window and are measured in pixels.

U and V relate to the layer's texture coordinates. They run from 0 to 1, whilst (0,0) marks the upper left and (1,1) the bottom right corner of a rectangular texture. Of course, having an object e.g. a cone assigned to a layer will result in other coordinates. The UV-coordinates [274] are saved within the object file.

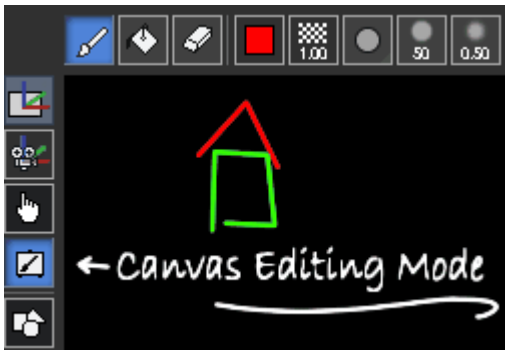
The mouse buttons (value 0 or 1) and the wheel (value depending on mouse) report whether the according button was pressed or the wheel was used.

The left image shows the reported data in Widget Designer: the mouse touches layer 2 from Server 2 and hovers above its UV-coordinate (0.1,0.5) whilst the XY-coordinate in the Preview tab is (316,160) pixel.

If you like to use the data to click on Widget Designer's Custom Script Buttons [822], please see its chapter. You can also use further filter nodes, e.g. the "Is in Bounding Box node" to check whether the pointer is in a certain area.

The next chapter explains the Canvas Editing Mode [256].

6.3.4.23.7 Canvas Editing Mode



This chapter explains the Canvas Editing Mode of the Preview which allows you to use content as a drawing background which allows for example to draw masks directly in Pandoras Box. For other topics regarding the Preview tab please see the introductory chapter [243].

If you like to draw in your Preview you first have to create a Canvas Asset. To do so, right-click in the Project tab [278] and choose "Add Canvas", save the new Canvas and assign it to a Layer. In case you like to use the Canvas as a mask, make sure to assign it to a Layer overlaying all other Layers, e.g.:
- use the last Layer in the render order before the Camera
- toggle the Layer into the Output render pass [211] only

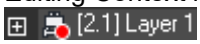
Alternatively you can right-click on any image file already part of your project and choose "Create Canvas from Image". This is of special interest in case you fine-tuned a mask with third-party software and want to re-import the Canvas. To export it initially, right-click the Canvas and choose "Export".

In case you like to change the Canvas size, please go to the Canvas Inspector [199].

To draw on the Canvas, go to the Canvas Editing Mode and make sure that the Editing Context (highlighted above in orange) is set to the Layer your Canvas is assigned to. Now you can pick the Brush or Flood tool and draw on the Layer. There are various other tools that let you choose the color or brush softness for example. With the Erase tool you can turn pixels transparent again.
To draw a straight line click once with the brush tip and hold the [Shift] key when clicking the second time.

The Drawing Canvas can also be used for a more creative application, e.g. to draw onto a facade or other 3D object. As for most interactive applications, the Widget Designer is of great help. Please see the chapter about the "Layer UV Draw to Canvas" [1245] node.

Note that in the Device Tree, the Layer that is picked as the Editing Context is marked with the Editing Context icon.



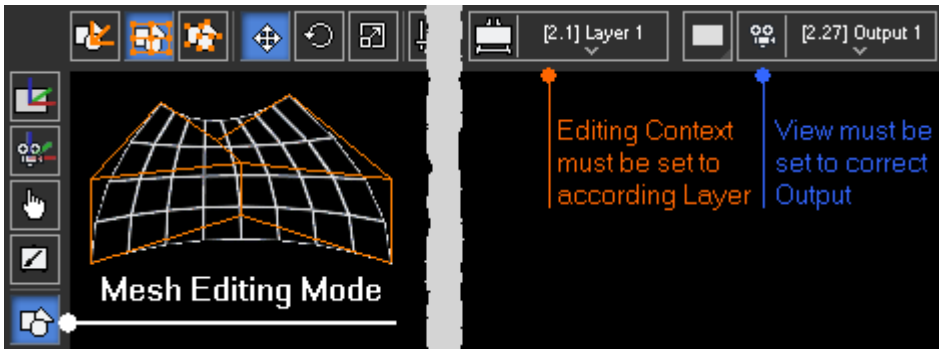
In case you delete the Canvas from the Layer whilst the Editing Context is still referring to it, the Layer will be marked with a small red dot.

To export the mask as a PNG file, right-click on the Canvas and choose "Export".

The next chapter explains the Mesh Editing Mode [257].

6.3.4.23.8 Mesh Editing Mode

This chapter explains the Mesh Editing Mode of the Preview which allows you to warp directly in Pandoras Box. For other topics regarding the Preview tab please see the introductory chapter [243]. The next chapters cover theory and practice of warping. First, general warping questions [259] are explained before a tutorial [264] shows step-by-step how to warp in PB in a Master-Client setup but also in stand-alone mode. The topic about the Multi-User tab [234] explains the possibilities to work with several operators on one project, e.g. if one or more operators warp the screens whilst other ones take care of the timeline and content.



How to Start with the Mesh Editing Mode

The fastest way to warp an Output is to select it in the Device Tree [173] and press the "Edit Warp Mesh" button in the Device Control tab [171]. This will automatically...:

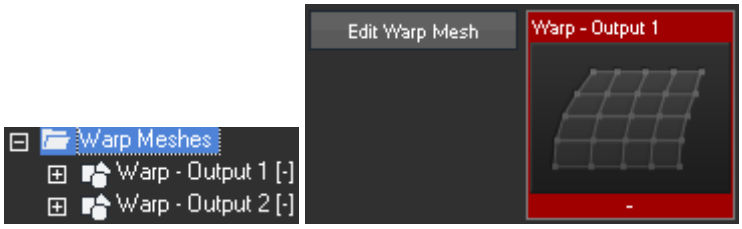
1- create an Editable Mesh with the size of your output resolution and a 3x3 FFD [259] grid, name it "Warp - Output no." under the project folder "Warp Meshes" and save it in the project path.

2- assign the Editable Mesh to the according Output.

3- toggle the Preview to the Mesh Editing mode.

4- set the correct Editing Context and View which is then scaled out slightly in order to make the selection of FFD points easier.

If you click the "Edit Warp Mesh" button a second time, only step 3 and 4 are executed.



To do this manually...

1- right-click in the Project tab [278] and choose "Add Editable Mesh" and save it. Right-click the Mesh and choose "Rename" in case you like to change the name. Check the size in the Sub Mesh Inspector [201]. Note that the newly created Editable Mesh folder lists one Sub Mesh called "Surface" which is a planar 2D Mesh in fullscreen size with a 3x3 FFD [259] grid.

2- assign the Editable Mesh or the contained Sub Mesh called "Surface" to your Output.

3- toggle the Preview to the Mesh Editing mode.

4- set the Editing Context (in the above image its highlighted in orange) and the View (highlighted in blue) to the according Output and scale it out slightly by scrolling with the middle mouse button.

Of course you can also assign the Editable Mesh to another Device e.g. a Video Layer.

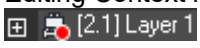
A Mesh can include one or multiple Sub Meshes. If you assign the Main Mesh to a Layer, and there are multiple Sub Meshes all of them are assigned, i.e. the Layer texture is shared on all Sub Meshes. Of course, you can also assign only one Sub Mesh. To do so, you can open the Main Mesh folder and pick the Sub Mesh you like. If you

like to assign all Sub Meshes to individual Layers, right-click the Main Mesh and choose the command "Assign Sub Meshes to Devices Consecutively" after selecting the "starting" Layer.

If you like to edit an imported object, simply right-click it in the Project tab and choose "Create Editable Mesh from Mesh".

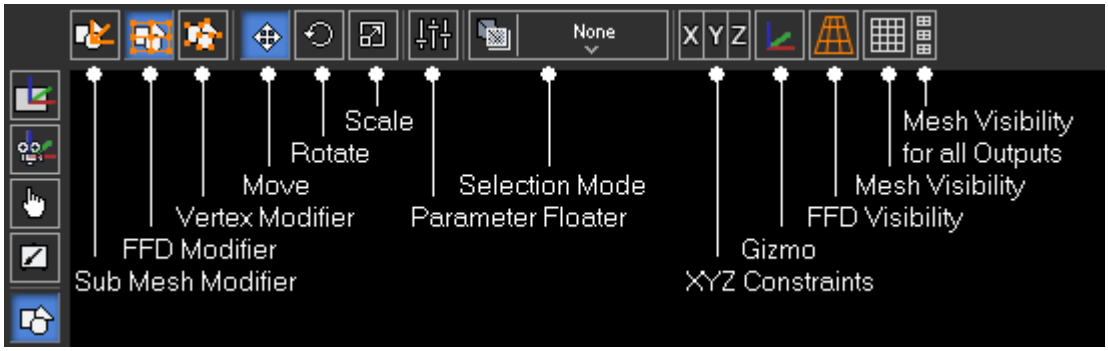
The Object Inspector [200] lets you change general settings like the wireframe color whilst the Sub Mesh Inspector [201] includes options like the Segment count (known as Mesh points in the Warper), the Control Point count (aka FFD), Mesh size and segments, etc. If you are not familiar with the warping process, please also read the next chapter covering general warping questions.

Note that in the Device Tree, the Layer that is picked as the Editing Context is marked with the Editing Context icon.



In case you delete the Mesh from the Layer whilst the Editing Context is still referring to it, the Layer will be marked with a small red dot.

Tools for the Mesh Editing Mode

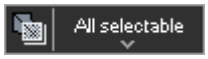


As soon as the Preview is set to the Mesh Editing Mode, the Mesh Segments (per default light gray) and FFD Control Points (per default orange) become visible. Above the Preview, there are many tools available.


Choose whether you like to select the entire Sub Mesh, FFD points, or Mesh / Vertex points. The shortcuts are the letters M, F and V.


Choose whether you like to move, rotate or scale the selection. The shortcuts are the numbers 1, 2, 3 on the main keyboard.
The workflows how to transform are explained below the table.


The Parameter Floater [Ctrl + P] shows the exact position in pixels. If you have selected more than one point, the "In Layer" position refers to the center of your selection. The "In Layer" position is the absolute position in regards to the Layer bounds whilst the "Local Change" is a relative input, i.e. it is added or subtracted from the absolute position and then reset to 0.



If you work with an Editable Mesh that contains several Sub Meshes (Surfaces) you have the choice to select either one of the Sub Meshes or only the one you choose from the drop-down list.

If you highlight one / several Constraints, a transformation is only executed along the chosen axis / axes.


These buttons toggle the visibility of the Gizmo, the FFD grid and the Mesh Wireframe in the local Preview. The last button toggles the Mesh Wireframes for all Outputs. For more options, go to the Configuration tab > Preview Display [145] or Client Display.


This button can be found on the left site at the bottom and toggles your Preview tab to a fullscreen window. All tools are available within the Button Bar, that you can toggle with the shortcut [T]. Hence, the maximized Preview gives you the best overview when warping.

To select one object or point, simple click on it or draw a selection box around it. To make a multi-selection you can either draw a greater selection box or, in case you select FFD or Vertex points, use the shortcut [Ctrl + an arrow key] after having selected the first point. The shortcut [Alt + an arrow key] moves the selection in terms of selecting the adjacent points.

You can transform the selection using the mouse, the keyboard or the dialog "Parameter Floater". With your mouse you can click on one of the selected points and then drag it horizontally or vertically. When holding the [Shift] key, the transformation will be applied along one axis only. For example if you move the mouse from left to right the X-axis is taken but as soon as the up-down movement is greater, the Y-axis is taken.

You can also work with the Gizmo, if it is visible. As soon as you position the mouse pointer on one handle of the Gizmo, it is highlighted in orange. Dragging the mouse results in a transformation along one axis. You can also work in another plane by highlighting one of the corners shown in magenta, cyan and yellow.

As an alternative to working in distinct axes, you can apply constraints. With an activated X Constraint (shortcut [X]) the Y and Z-axis are shown in gray and any transformation happens along the X-axis only.

Step-by-Step Description how to Deform a Mesh

Depending on your preference you can work either using the mouse or the keyboard, or both. At any time you can maximize the Preview tab in the Master with this button  and toggle the Button Bar with [T].

	Mouse	Keyboard
Choose the Edit Mode	 Choose whether you like to select: Sub Mesh FFD points, or Mesh / Vertex points.	Alternatively, use the shortcuts: - [M] - [F] or - [V]
Select points	Select a single point by simply clicking on it. Or, drag a selection box around one or several points.	First, select all points with [Ctrl + A]. [Up/Down/Left/Right] key selects a single neighbor point. [Ctrl + arrow] adds the neighbor point(s) to the selection. [Alt + arrow] moves the current selection.
Choose a Transformation Mode	 Choose how you like to transform the selection: Move Rotate Scale.	Alternatively, use the shortcuts: - [1] - [2] - [3]
Toggle the Axis Locks (if necessary)	 Apply a Constraint if you want to transform along one axis only. Another possibility is to use the Gizmo.	Alternatively, use the shortcuts [X], [Y] and / or [Z]. You can also hold the [Ctrl] key whilst dragging your selection.
Edit the selection	Now, simply click on one of the selected objects / points and drag it with the left mouse button up/down or left/right. Alternatively, you can open the Parameter Floater  and drag the according number field up or down.	Move the selection with the shortcut [Shift + Up/Down/Left/Right arrow]. Alternatively, you can open the Parameter Floater with [Ctrl + P] and enter a number to an according number field.

The next chapter covers general warping questions [259].

6.3.4.23.8.1 Theory: Warping Questions

This chapter covers warping questions related to the Mesh Editing Mode [257] in the Pandoras Box Preview and the warping process in general. Whilst the next chapter explains step-by-step the practice, this chapter is all about the theory.
For other topics regarding the Preview tab please see the introductory chapter [243].

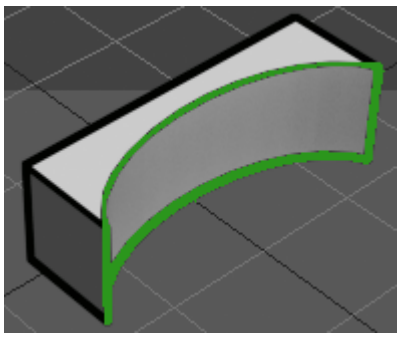
Projector and Lens Setup

Before the warping process can start, you need to make sure, that the projectors are setup correctly. This includes the physical projector position as well as settings like lens shift, zoom and focus. Hint: Some operators prefer to

reset the projector to the factory settings before they start working as this makes sure that no hidden setting is overseen.
Position-wise, the projected images obviously need to cover the screen entirely. If you do the blending in Pandoras Box, the overlap (at the smallest distance!) should be a minimum of 256px due to the internally used 8bit gradient. If you divide 256px with the projector resolution and multiply this with 100%, you get a more handy percentage number, e.g. roughly 13% for HD resolution and 25% for XGA. Many projectors have internal grid test pattern that can help finding a nice overlap; simply count the horizontal grids and if you know, that the overlap is 25% it's obvious that at least a quarter of all grids must overlap each other. Further down it is explained how to calculate and measure the overlap in meters. A smaller overlap is not recommended as it becomes harder to blend both projectors seamlessly. More overlap eases the softedge process but the larger the projected image is zoomed and the more pixels are "outside" the screen, the less resolution you have "on" the screen. Thus, the goal is to find a good position where you do not loose pixels but cover the screen and at least the minimum overlap area.
In regards to quality and final resolution, it is always better to spend time with positioning the projected image using the available hardware tools in difference to correcting a bad position with too much warping and masking.
For blending both projectors, it is important that color and brightness are equal. As you can not increase the brightness from a darker projector, the reference is the worst projector.
Last, check whether the resolution is set correctly in the projector and the Client's graphics card. Please keep also in mind, that the PB Client software should be closed and restarted when graphics card settings [1948] are changed. Ideally, you should check that the Client's output settings are reported correctly to the Master. Go to the Configuration tab > Render Engine [162] and select the Client in the top drop-down list.

If one of the mentioned points has to be changed later, you have to redo the warping! Especially for fixed installations, EDID Managers guarantee, that the resolution and frame rate do not change due to a unforeseen shut down, power loss etc.

Screen Measurements and Markers

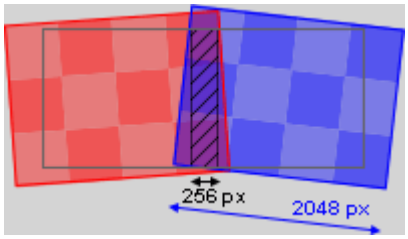


The example from the tutorial refers to a projection on a flat wall. However, all surfaces - no matter whether they are bended or include corners - work in the same way. Just keep in mind that your measurements refer to the unwrapped width and height of the surface. In other words, you always follow the geometry of the surface as shown with the green line in the image.

For measuring the screen, the overlap or projection area, or the center of it, it might be needed to place markers on the screen. Even though it is not absolutely mandatory to do this, you will notice very fast that markers help a lot during the warping process. The time that you invest in measuring and marking important points will pay off later as the warping itself will be easier, faster, less sensitive to errors and most likely result in a better quality. Especially the risk of warping wrongly due to a bad perspective position is limited. Without markers you can only rely on your optical perception and depending on your position, the projection and surface it can be very hard to see whether a line is really straight, whether a grid is uniform and whether distances are even. How you place markers or how many you need can not be answered in general as this depends on the projection, the environment and your personal preferences and skills. Markers can be drawn (more or less permanently) with a pencils or pen or any paint which even includes ultraviolet paint that only becomes visible with a UV lamp. Markers can also mean tape strips that are attached temporarily (and can be removed nicely). Lastly, you can use a laser distance meter for temporary markers. There is a wide choice of different laser solutions; many operators prefer using cross line lasers with tripods. For screens that are far away or very large it is useful to work with (video) cameras.

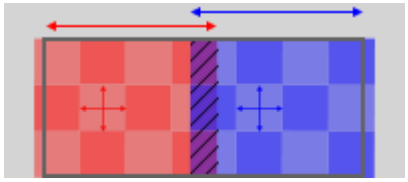
Overlap and Aspect Ratio

The softedge area refers to the smallest area where two projectors overlap. For the depicted example, this would be the hatched area. If you do the blending in Pandoras Box, the overlap should be a minimum of 256px due to the internally used 8bit gradient. If you divide 256px with the projector resolution and multiply this with 100%, you get a more handy percentage number, e.g. roughly 13% for HD resolution and 25% for XGA. However, sometimes it is needed to calculate (or check) the overlap very exactly or maybe you like to place markers and you need a number in meters that you can measure.



In the left example, the horizontal resolution is 2048px. If you measure the projected width, divide it by 2048px and multiply this with 256px, you get the distance for the minimum overlap. For example: 2048px = 3m and 256px = ?m; ?m = 3m / 2048px * 256px = 0.375m
If you cannot see or measure the entire width, you can of course substitute the numbers, e.g. with half of the resolution and half of the width. Keep in mind, that the results gets less accurate the smaller your reference distance is.

Now, let's look at the possibilities regarding the aspect ratio for projections without preplanned content that matches the screen. For this example, let's assume it is not possible to align the projectors better so the above image shows the starting point for the warping process.
Looking at the red projection first, the right side needs to be aligned at the right side of the (hatched) overlap. The top and bottom are now defined by the screen area. The left side needs to be shifted in such a way that the squares in the test pattern stay even. The test pattern shows a 4x3 grid because our projectors also have a 4x3 aspect. For the blue projection the same criteria apply, laterally reversed of course.



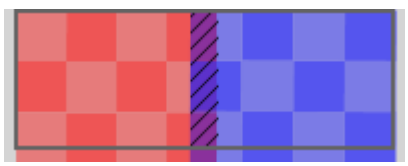
This image depicts the ideal (intermediate) result. Note that a square in the red projection has the same height as width and also corresponds to each blue square. If we applied a circle image or Text Input to a Layer and moved it around, it would be perfectly even everywhere on the screen.

However, with this result we obviously overshoot the screen. We cannot move the left and right side in, because this distorted the final aspect ratio. One solution would be to blank both areas. You can either apply a black image to a Layer and position it accordingly, or draw a mask that covers both sides, or apply a "Crop" effect [412] to the Output Layer. Another solution is to use the possibility to change the aspect ratio of the Camera in the Camera Inspector [199]. Simply measure the "on-screen" width depicted with the red and blue arrow and enter this with the on-screen height in the Inspector. This is also possible when the content was prepared with a certain aspect ratio. In that case you just need to make sure, that the projector covers the screen according to the plan and use this aspect ratio in the Inspector.
The units do not matter as the ratio is always the same, meaning that 4:3 has the same result as 1024:768 or 2048:1536. However, the entered numbers must be whole numbers, hence it makes sense to measure in centimeters. For the left example the measurements result in an on-screen width of 233cm and height of 187cm hence we enter "233" and "187" into the Inspector. The area that is now seen by the camera is exactly the area that needs to cover the screen. If you start with this workflow, it makes much sense to use a test pattern in the aspect of 233:187 and the total width of the output resolution. This is because Pandoras Box always refers to the horizontal width, so any new aspect will always cover the horizontal width and the height adopts. In our example, the output resolution is 2048px. Hence the ideal test pattern should be 2048x1644px according to this calculation: 2048 / 233 * 187. In the tutorial [264] we do not use custom test pattern but keep working with the default ones which requires one more adoption.

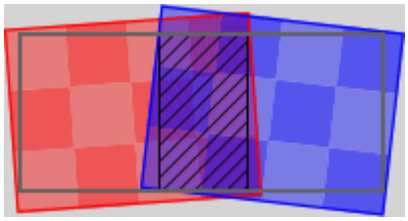
If your test pattern or content was pre-defined and has a different resolution, e.g. 2000px, you can either scale the Layer with the factor 2048/2000=1.024 or reduce the Z-distance from the camera to the Layer. The default distance is -3000px, so the new distance would be -3000px / 1.024 = -2930px. Of course, you can also adjust the numbers by eye, without the need to calculate them exactly.

If you like to practice this to get a better understanding, you can do this also with a small projection, for example with one projector on a piece of paper. Draw an area on the paper and try to find the best and fastest way to project an image there using the above explained workflow with the Camera Inspector. Advanced users can also adopt the pixel conversion factor under Configuration tab > Unit Management [160].

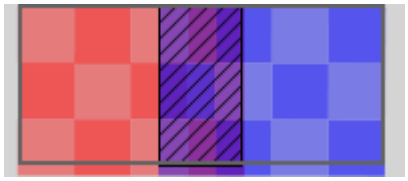
In case the screen is wider than above, and the images do not fill out the screen they must be warped differently. Align the outer edges with the screen edge but again, keep an eye on the aspect ratio so that each square stays even.



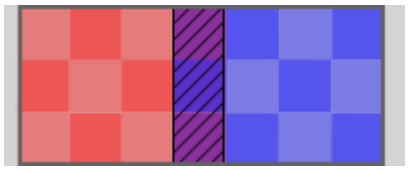
The result will look like this: we overshoot the screen vertically. Again, you can either blank this area out, or measure the on-screen width and height to change the aspect mode in the Inspector which is the recommended way.



For this example, the projector's overlap is larger which makes two solutions possible.



This would be the result if the FFDs are adjusted starting at the overlap edges and covering the screen. Just as in the last example, we overshoot vertically and can either blank the bottom area or better, adjust the Camera's aspect ratio.



This alternative would also be possible. Here the FFDs are adjusted referring to all outer screen edges. The disadvantage is that the overlap is reduced but if it is large enough anyways, it is a feasible, acceptable and fast warping solution.

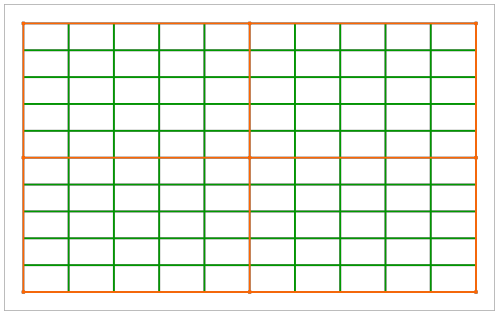
Test Pattern

Needless to say, it makes sense to use images as test pattern that cover the entire projection area and match the resolution of the content used in your show. In the best case, you have at least one background image that suits the needs of a test pattern. This means that it features circles and / or grid lines to make sure, the projectors overlap perfectly and form an image in the correct aspect ratio. Different colors and gradients help during the softedge process. If no images are available, you can also make use of the Stock Assets content found in the Assets tab. The sub folder "Testpattern" includes different test images in all kinds of resolutions. At best, you get familiar with an image editing software and workflows to create test pattern that suit your needs. Then you are flexible and prepared for the worst case as you can prepare or edit test images that fasten your warping work significantly and increase the quality of it.
It is also worth, to get familiar with the Mesh count setting. Each Mesh consists of horizontal and vertical lines anyhow so you can also orientate the warping by these lines. The following two paragraphs explain this in more detail.

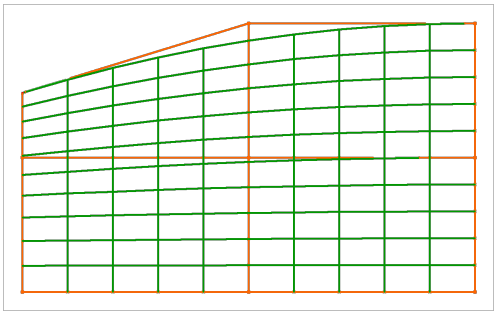
Mesh versus FFD

This paragraph describes the difference between a Mesh point (Vertex) and an FFD point, thus it is covering a fundamental function of the warping process.
The pictures below show a 2D plane with a green-colored 10x10 Mesh and orange-colored 3x3 FFD.

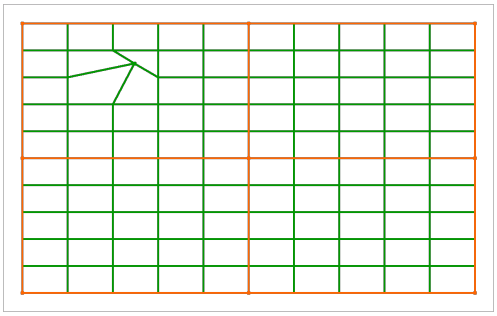
See here the differences between moving a FFD control point and moving a Mesh point. Please note that the FFD is only a helping tool to set up the Mesh, you will not see the FFD in the final object.



This shows the grid without any editing. The green lines represent the Mesh, the orange lines the FFD.



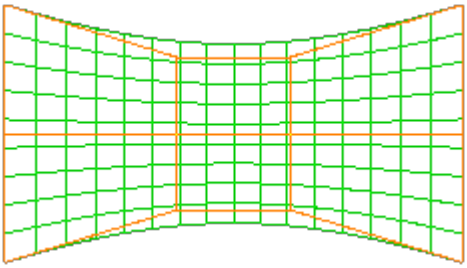
The top left FFD control point is moved further down. The whole Mesh is affected by this change: the horizontal lines are bend together on the top left side, the Meshes outline gets curved.
This effect can be of advantage or disadvantage. In the beginning of the warping process it can simplify and accelerate the workflow as it is not necessary to move each individual Mesh point. The further the warping process develops the more it is necessary to apply changes to particular pixels only. At this point the FFD is not sufficient any more as it affects large areas of the grid.



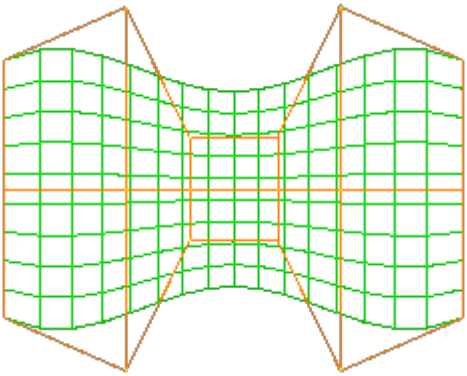
Now only one Mesh point is moved. Only the segment lines between the moved point and the four neighbor Mesh points are affected by this.
The more you are experienced with warping the better you will be able to answer the question how many Mesh points a grid should have. This differs from application to application. If too little points are chosen it won't be possible to apply the detail changes that are necessary. This is especially crucial when setting up Meshes for a softedge projection as the pixels must overlap each other exactly in the overlapping area.
If too many points are chosen, the warping process is lengthened unnecessarily as all points must be adjusted.

Setting up FFD Control Points and Mesh Segments

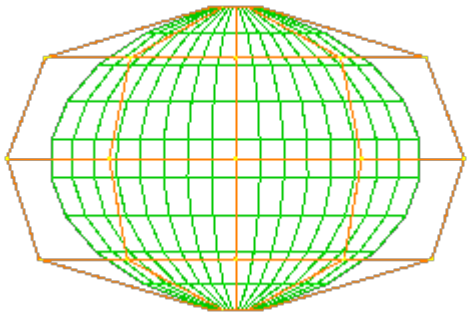
Before you start moving the control points you have to decide with how many FFD control points and Mesh segments you want to work. The FFD and Mesh count is set up in the Sub Mesh Inspector [201]. The perfect amount depends on your screen surface and outline.
The more warping projects you have done, the faster you will be able to tell the best FFD and Mesh count. If you are not sure in the beginning, simply make a guess and start warping. You will see quite fast, that you have picked too many or too less FFDs. Too many FFD points are not that bad, you simply spend more time in the FFD-phase but you could save time during the Mesh-phase - at least if not way too many FFDs were picked. If you have picked too less FFD points you will notice that you will have to move Mesh points at a very early stage of warping. This will definitely be more time-consuming than starting all over with a new Mesh. In addition, it is easier to obtain a good quality Mesh with FFDs instead of offsetting single Mesh points too far. If you like, you can save the current project and have a quick look whether a new Mesh with more FFDs does give you better results and eases your work. This can be found out in a few seconds but save minutes or even hours.



Example 1, a 4x3 FFD
A simply bend screen will go well with only 3 vertical FFD control points. Horizontally there needs to be done more warping, thus 4 FFDs work better. The more smooth the outline has to be, the more horizontal Mesh segments you should take. In this case there are 20 Mesh segments.



Example 2, a 6x3 FFD
Curved screens that are more complex will be easier to handle if you increase the amount of FFD control points. In this case there are 6 horizontal control points and still 3 vertical ones.
The curved outline of the Mesh is the result by only moving the FFD control points.



Example 3, a 5x5 FFD
Spherical screens are bend in all directions. They require to increase the amount of vertical FFD points as well. In the depicted example a Mesh with an 5x5 FFD is shown. If the projector looks straight on the equator, the Mesh deformation will be quite homogenous and the FFD should look similar to the example.

The amount of Mesh segments can be altered as long as working only with the FFD control points. As soon as a Mesh point is moved, the amount should not be changed any more as this will reset the Mesh and your changes will be lost.
In general, the amount of Mesh segments depends on how exact the warp needs to be. Firstly this is a question of how complex or detailed the screen is, including the outline, as shown in the above example with the simply bend screen. If the screen is quite flat itself but has a very detailed outline, it could be a faster solution to create a mask [256] instead of increasing the Mesh count.

Secondly, it is important whether you are projecting with a single projector only, or if several projectors overlap each other. Within the softedge area the pixels from both projectors must overlay perfectly. This requires a higher Mesh count. As a rule of thumb, at least 7 Mesh segments should lie within the overlap area.

Many operators prefer to change the Mesh color for different outputs using the Inspector for the according Sub Mesh [201] and its setting "Editing Wireframe Color". If you choose for example a red Mesh for one output and a green one for the other, it will become much easier to warp the overlap as both lines add up to a yellow color. However, please note that sometimes you can not bring the lines together. In those scenario your goal is to get the mesh lines parallel and with an equal distance. An example would be if you need to get the largest possible overlap and the amount of segments in the blend area cannot be a (reasonable) whole number. E.g., if the output resolution is 1280px and the overlap needs to be 400px and not any smaller!

By the way, it could be helpful to work with Mesh segments that have the same height as width because it is easier for the warping operator to see whether squares (in difference to rectangles) are equal. It also makes it easier to measure the screen and place markers. If your projector has an aspect ratio of 16:9, you could set up a Mesh count of 16 by 9 during the FFD-phase and increase it to 32 by 18 or even 48 by 27 before starting the Mesh-phase. As explained above under "Aspect Ratio [260]", it is also possible to use the aspect ratio of the Camera or on-screen area, respectively.
With the rougher Mesh it is easier to see and get equal squares whilst moving the FFD points. Hence you have a very good basis when starting to work with the Mesh where the higher Mesh count might be needed because of details in the structure or an overlap with another projector. Doubling or tripling the Mesh count has the advantage that your markers on the screen still refer to existing Mesh lines. Alternatively you can also work with textures that show different grids as explained above under "Test Pattern [262]".

The next chapter features a warping tutorial [264].

6.3.4.23.8.2 Practice: Warping Tutorial

This chapter describes step-by-step how to warp in Pandoras Box using the Mesh Editing Mode of the Preview. Please see the previous chapter General Warping Questions [259] which explains in detail what you need to know about projector setup, aspect ratio, test pattern and of course the terms "FFD" and "Mesh". The chapter before, Mesh Editing Mode [257], shows how to switch the Preview to the warping mode and work with it and also how to create a Canvas.

The task in this tutorial is, to create a good overlap and softedge with two projectors that project on a wall. There are two scenarios regarding the hardware. First, a Master is connected to a Client with two outputs connected to the projectors. This is a regular Master-Client setup. Then, a standalone setup is explained which means that only one system is used which is started as a Master. Here, the outputs toggle between showing the interface or showing the fullscreen content.
But first, let's start with explaining the general warping workflow.

The Warping Workflow

A good quality Mesh refers to a Mesh where the Mesh lines are uniformly distributed on the screen. For example, if the screen is 2m wide, and there are 20 horizontal Mesh segments, each segment should be 10cm wide. If this is not achieved sufficiently, and you project text that moves across the screen, it would scroll unevenly. Wherever there are smaller distances between Mesh lines, the text would be smaller too. Wherever there are larger Meshes, the text is enlarged.
If your content does not contain critical movements or visible geometrical forms, you can warp a little more rough. So before you start warping, or before you spend too much time within the last phase, check the content and decide how perfect the result really needs to be.

For some people it is quite hard to perceive equal distances. To fasten and ease the warping workflow, try to mark certain points on the screen as explained in the previous chapter under "Markers [260]". If you mark for example every 40cm with tape, it will be much easier to arrange the Mesh equally.
For the same reason it can be worth the time to create special test pattern [262]. This is definitely recommended when projecting on complex geometries and the content (that is used later on) refers to the geometry.

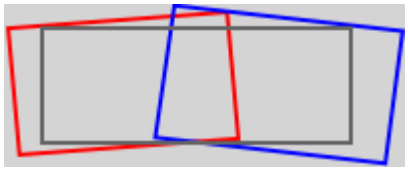
Keeping this in mind we can now start warping. The golden rule is always to warp as much as possible with the FFD, but not more than necessary. Or in other words: the **FFD is for the coarse adjustment and the Mesh for fine-tuning**.

Whilst moving the FFD points, match the Mesh outline (= content outline) as good as possible with your screen outline. At the same time keep an eye on the distance between the horizontal lines and between the vertical lines. As soon as you recognize that moving an FFD point helps within a small Mesh area but "destroys" an higher number of other Mesh areas it is better to finish with the FFD-phase. Decide for a final Mesh count and move on to moving Meshes. Here you will see that the better the FFD was adjusted, the less time needs to be spent for finishing.

When warping with overlapping Meshes, you are done with warping as soon as all Mesh lines overlay each other. This can be seen easily when both Meshes have a different color as the resulting color will be the sum, e.g. red+green=yellow. The closer the audience sees the projection, the more perfect the overlay needs to be.

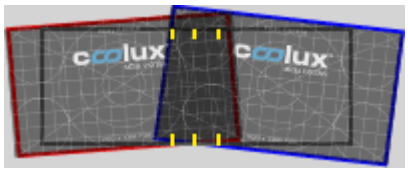
In general it is possible to do a rough warp and start programming with it in the timeline. Later on, when there is enough time or when it is sure the projector or screen will not move, you can go back in the Editing Mode and finalize it.

Warping in a Master-Client-Setup



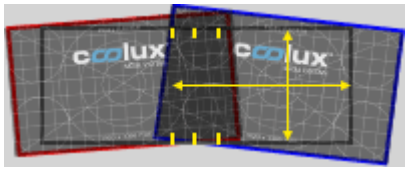
This part of the tutorial explains how to warp with a Master that is connected to a Client. The Client's outputs are connected to two projectors projecting on a wall where tape marks our projection area. Of course, under real circumstances you would try to adjust the projector's position better to have a higher final resolution and projection quality but for practicing warping this is absolutely fine.

Step 1: After setting up the projectors and resolution (see Theory: Projector and Lens Setup [259]), please start the Client and Master software (in the same version [125] and Domain [147] of course) and drag the Client into the project. From now one, we will only work with the Master's interface (in difference to warping with the Warper software that was needed before the Preview offered the Mesh Editing Mode).

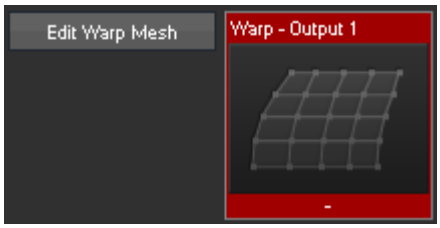


Step 2: Now, let's assign a test pattern to a Layer. In our example, we use HD projectors with a resolution of 1920px x 1080px, so we will use the test pattern from the path: "Stock Assets > Test Pattern > 1920 x 1080". As seen in the image, the circles (marking 25% of the image) overlap each other showing that the softedge area is more than enough. However, let's measure this just for practice (see, Theory: Overlap [260]). First, we need to place some markers. The tape strips mark the horizontal screen center and the overlap edges which we choose with the same distance as we have more than enough overlap and this way, both projection areas have the same size.

The right projector's width is 116cm, and the left is 109cm. If 1920px equal 116cm, 256px for the minimum soft edge equal 15.5cm (= 116cm / 1920px * 256px). Our overlap within the placed markers is 22cm wide, which is larger and thus alright.



Step 2b: As an optional step, we will also measure the on-screen area to show the workflow with the PB Camera set to this aspect ratio. The measured width is 88cm and the height is 57cm (as depicted with the yellow arrows). Select both Cameras in the Device Tree to see their Inspector information. Untick the option to "Adopt Output Aspect Ratio" and enter the new numbers according to the measurements. This is also possible when the content was prepared with a certain aspect ratio. In that case you just need to make sure, that the projector covers the screen according to the plan.
As explained in the Theory chapter: Aspect ratio [260], Pandoras Box now scales the Camera's view port area to fill the width of the test pattern. In our case, this results in a blank area at the top and bottom. As mentioned, we could now make a test pattern in the correct aspect ratio and output resolution which means (in px) 1920 x 1243 (= 1920 / 88 * 57) and this would fill out the new Camera view port. But as our final content will have the same size, we want to keep working with the default test pattern which requires one more setting. We can either scale the Layer or adjust the Z-distance from the Camera. Of course, you can adjust the parameters by eye. If you like to calculate this exactly, we need to figure out the factor as follows. Normally, the aspect ratio is 16:9. We changed the width to 88 which would require 49.5 (= 88 / 16 * 9) as a height to keep this aspect. However we entered 57, which means a multiplication factor of 1.15 (= 57 / 49.5). Alternatively, you could calculate 1243px / 1080px. The default camera distance is -3000px, so the new distance would be -3000px / 1.15 = -2605px. If you are using an older Player license that does not offer the distance parameter for a Camera, you can scale the Layer with 1.15 or calculate the FOV. Again, this step is optional.



Step 3: In this example, we will only use one Client with two outputs. The more Clients and outputs there are, the more it makes sense to rename them (with the shortcut [F2] or via the right-click menu). You can simply identify each Client and output quickly, especially when they have a unique name, for example "Server Stage proj" with the outputs "Left softedge", "Right softedge" or "Stage L" and "Stage R". In addition, the Canvas assets that will be generated in the next step, will carry the current Output name. As they are saved in the current project path, now is a good time to save the project.

Let's start with warping the left projector. Select the Output in the Device Tree [173] and press the "Edit Warp Mesh" button in the Device Control tab [171]. This will automatically...:

1- create an Editable Mesh with the size of your output resolution and a 3x3 FFD [259] grid, name it accordingly in the project folder "Warp Meshes" and save it in the project path.

2- assign the Editable Mesh to the according Output.

3- toggle the Preview to the Mesh Editing mode.

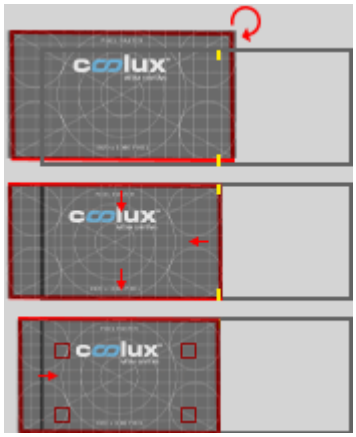
4- set the correct Editing Context [257] and View which is then scaled out slightly in order to make the selection of FFD points easier.

If you click the "Edit Warp Mesh" button a second time, only step 3 and 4 are executed. If you like to do this manually, please refer to the chapter "Mesh Editing Mode [257]".

As we warp on a flat wall, the default 3x3 FFD count is alright. In other cases please adopt the FFD grid to your needs before starting to warp. The paragraph "Setting up FFD Control Points and Mesh Segments [263]" in the previous chapter explains how to do this.

Step 4: In the next step we will use the keyboard and mouse to move the FFD and Mesh points. That means that you will use the devices connected to the Master but look at the output of the Client. To make the mouse selection more intuitive, the Client shows the mouse position as a gray circle. You will also see that unselected points are just outlined whilst selected ones are filled with the FFD or Mesh color. However, you might still want to adjust the Preview size in the Master. There are three possibilities. Firstly, you can simply adjust the pane or window size by dragging the empty area between the sections. Secondly, when you right-click on a tab (in our case, the Preview) you will see the option "Break out Pane". This will display the Preview tab in a separate window that can be re-sized and positioned in a fast and easy way. More information can be found in the chapter "Layout [313]". Lastly, you can maximize the Preview as explained below under "Warping in a Stand-Alone Setup" [268]. Press the icon "Toggle Maximized Preview" on the left side of the Preview or use the shortcut [Ctrl + Shift + F].

Step 5: Now, we will move the FFD points in such a way that the right edge (of the left projector) aligns with the marker for the overlap area and the top and bottom with the screen. The left edge depends on your chosen workflow and test pattern. As you see in the images, we turned the opacity for the right projector off and show both workflows.



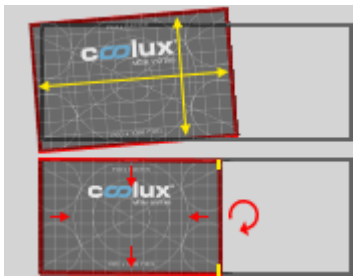
In case you have not followed the optional step with adjusting the Camera's aspect mode, your texture is larger than the screen meaning that the left edge should sit outside the screen so that the grid is even, as depicted by the four small red squares in the last (third) image.

To move the FFD points, you can either use the mouse or keyboard and either select single or multi points. The table at the end of the chapter Mesh Editing Mode [257] shows how to toggle modes, select and transform using both tools. Most people choose a combination of both, which also depends on the current projection of course.

A possible workflow for the left projection could be to go to the "Transform Sub Mesh" mode first and rotate the entire object clockwise. Then switch to the FFD mode, select the top three FFD points with the mouse and move them roughly to fit the top screen edge, do the same with the bottom points at the bottom edge and the right points at the overlap edge. Then re-do these three steps in a finer way using the keyboard. Then its time to get the grid even by moving the left three FFD points and most likely the inner FFD points that do not sit in a corner. We can refer to the grid of the test pattern image so the Mesh's visibility can be turned off. If you do not have a suitable image, choose a Mesh count with the aspect of the output. By the way, you can work on the FFD even though it's visibility is toggled off which helps you focus on the lines of the test pattern or Mesh.
When you are done with that, align the right projector in the same way. During the FFD-phase it makes sense, to switch off the other Output. When you are done, make both Meshes visible via the button "Toggle Wireframe Visibility for all Outputs"  and decide for a final Mesh count so that the overlap covers about 7 vertical Mesh lines. For an easier projection like ours where the surface is quite even, less lines are also alright.

As mentioned in the last chapter, it makes sense to change the Mesh color for different outputs. If you choose for example a red Mesh for one output and a green one for the other, it will become much easier to warp the overlap as both lines add up to a yellow color. Open the Mesh folder in the Project tab and click on the sub mesh named "Surface" to see the Sub Mesh Inspector [201]. There, you scroll down to the section "Editing Wireframe Color" and click in the color field.

Now, move the Mesh points so that the projectors overlap each other perfectly. If you alternate the Mesh you work in over and over, it is easier to get an even overlap.



Step 5b: Here, the Camera's aspect was adjusted to fit the measured or planned on-screen size. Thus, the Camera sees a smaller part of the Layer, or the Layer was assigned with a texture that fits the screen.

This makes it very easy to adjust the FFD; simply align all edges with the screen area and the squares in the test pattern will fit automatically. This is the only difference to the steps described above.

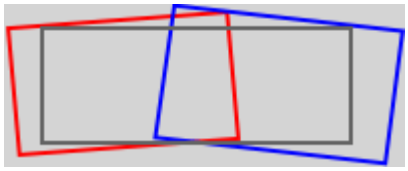
Step 6: As the last warping step, align the Cameras. To do that, select both Cameras in the Device Tree and open the Device Controls tab. Click the "Align [124]: Off" text in the bottom Status Bar, until it says "< >" or "><". Then, move the "X Offset" by 960px (=1920px / 2) from one Camera and the other one will move mirror-inverted until both camera images meet in the middle. For some projects it is a better alternative to move the "Viewpoint" and "Target X Pos" at the same time instead of using the "X Offset" parameter.

Step 7: Finally, the softedge can be adjusted. You can either make use of the softedge tools from the projector or from Pandoras Box. This depends on the quality of available softedge functions or parameters and also on the workload. If time is short for programming the show, you rather outsource parts to available projector technicians. The default softedge parameters in Pandoras Box can be found in the Output device in the section "Softedge" [688]. Advanced parameters can be added with a Softedge effect [611], or Softedge P-Curve effect [614].

Warping in a Stand-Alone-Setup

This paragraph explains the workflow when your Client is not connected to a remote Master but is started in stand-alone mode. In principle, the above explained information applies as well. So if something is not clear, please read the entire chapter.

As above, the example includes a Client with two outputs connected to two projectors that should be warped. At the end we will talk about the scenario where only one projector works as the content output that should be warped whilst the other output is connected to a screen displaying the interface permanently.



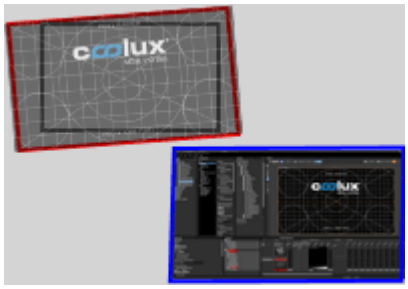
After setting up the projectors and their resolution (see Theory: Projector and Lens Setup 259), please start the Master mode of PB on your Client. From now one, we will only work with the Master's interface (in difference to warping with the Warper software that was needed before the Preview offered the Mesh Editing Mode). Please follow steps 2, the optional 2b and step 3 as described above to assign a test pattern image to a Layer and an Editable Mesh to the Output. If you like, you can already assign an Editable Mesh to each Output and color them in different colors. Open the Mesh folder in the Project tab and click on the sub mesh named "Surface" to see the Sub Mesh Inspector 201 where you scroll down to the section "Editing Wireframe Color" and click in the color field. But of course, you can do this later too.

Step 4: In the stand-alone case, we have to adjust the Preview, which is also our fullscreen content output, in a special way.
As we need to warp with an fullscreen image, we will switch the Preview to its full screen mode. However, the default setting is, that the maximized Preview uses only one output which is not handy when you like to align two outputs. To change this, please go to the Configuration tab > Local Preview 150 and enable the option "Show Maximized Preview on multiple Adapters".

Then press the icon "Toggle Maximized Preview"  on the left side of the Preview or use the shortcut [Ctrl + Shift + F]. The difference to the "Toggle Full Screen" icon located at the top (or shortcut [Ctrl + F]) is that we still see the Preview editing tools in the Button Bar.
Currently, you will always enter the "Layer Mode" when toggling the Preview, so please make sure that you switch to the "Mesh Editing Mode" by clicking the according icon  in the Button Bar.
Per default, the View is set automatically when you press the "Edit Warp Mesh" button in the Device Control tab for the output and is then zoomed out slightly in order to make the selection of FFD points easier. However, in our case this zoom is not desired so please reset it by pressing the "Reset" icon 

. Note that the "Camera interaction" icon is not tinted in orange anymore which indicates that the default View settings apply again.
Now you can set up the Mesh and FFD visibility for each output to suit your needs. Simply click on an output and the Button Bar is available offering all tools for the Preview. Don't forget to switch to the Mesh Editing Mode. where you will find these icons to toggle the visibility from the Gizmo, FFD, Mesh and Mesh on all outputs  . If you like to see the Layer on one Output only, for example because you like to start with one output first and the other one is distracting, you have several choices. First, you can go back to the embedded Preview view, open the Layer Inspector that shows the test pattern and untick the check box for one Output. Second, you can also assign the test pattern to a second Layer that is only rendered on the other Output and set the Opacity value whilst the Preview is maximized. Just toggle to the "Layer Mode" and open the Parameter Floater that offers the Opacity parameter. The third possibility is to leave the Layer render path settings as they are but zoom the second output temporarily. Click into the second output window and scroll with the mouse wheel which zooms out the current view and tints the "Camera interaction" icon orange again.

From now on, all steps apply as described above. The following list is just a quick summary:
- enable "Show Maximized Preview on multiple Adapters" option under Configuration > Local Preview
- assign a test pattern to a Layer (or two Layers when render path is changed in Layer Inspector)
- optionally, change the aspect mode in the Camera Inspector
- assign an Editable Mesh to each Output using the Device Control tab
- optionally, change the FFD and Mesh count and their color in the Sub Mesh Inspector
- enter maximized Preview [Ctrl + Shift + F]
- enter Mesh Editing Mode
- adjust visibility for the Mesh and FFD



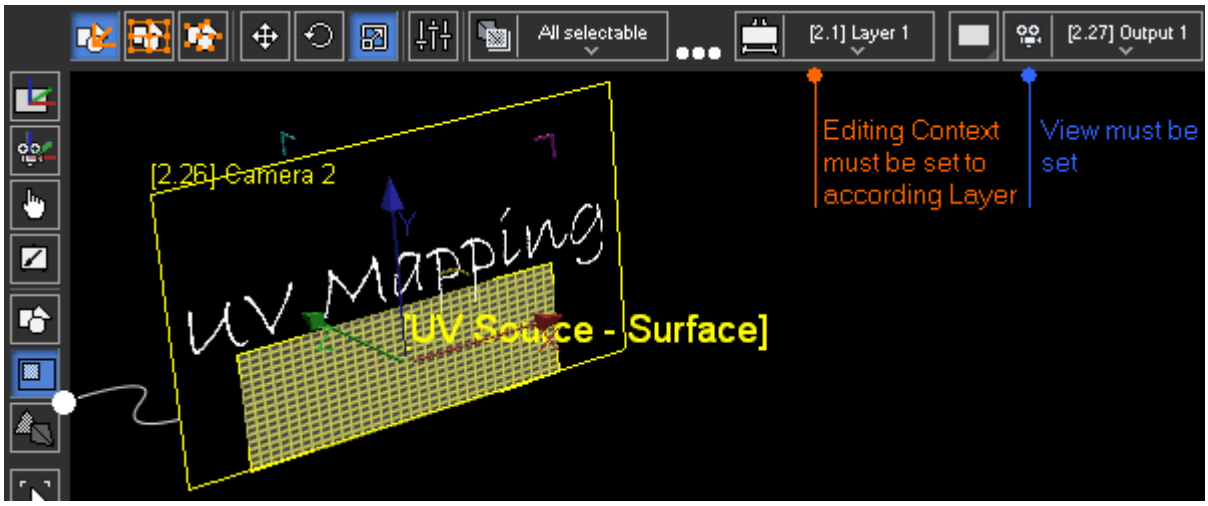
Lastly, this shows the case when you are working in a stand-alone setup where only one projector works as the content output that should be warped and the other output is connected to a screen displaying the interface permanently.

In fact, it is almost identical to the above case with two outputs to be warped. The only difference is, that you do not enable the option to maximize the Preview on all outputs that can be found in the Configuration tab > Local Preview [150]. However, there you also find the drop-down list that let's you assign the output that should be used for the maximized Preview. Likewise, the section Configuration > Render Engine [162] offers the settings to assign the full screen content output. First, enable the option "Full Screen is single" and then you can choose the desired output in the table row "Use for Full Screen".

The next chapter explains the Planar and Perspective UV Mapping Mode [269].

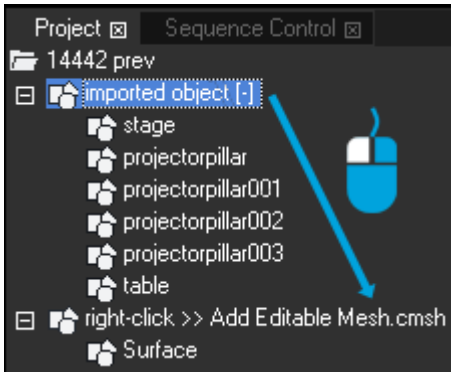
6.3.4.23.9 Planar and Perspective UV Mapping Mode

This chapter explains the Planar and Perspective UV Mapping Mode of the Preview which allows you to assign a UV map directly in Pandoras Box. For other topics regarding the Preview tab please see the introductory chapter [243]. The next chapter covers general mapping questions [274] and is of special interest for mapping beginners.



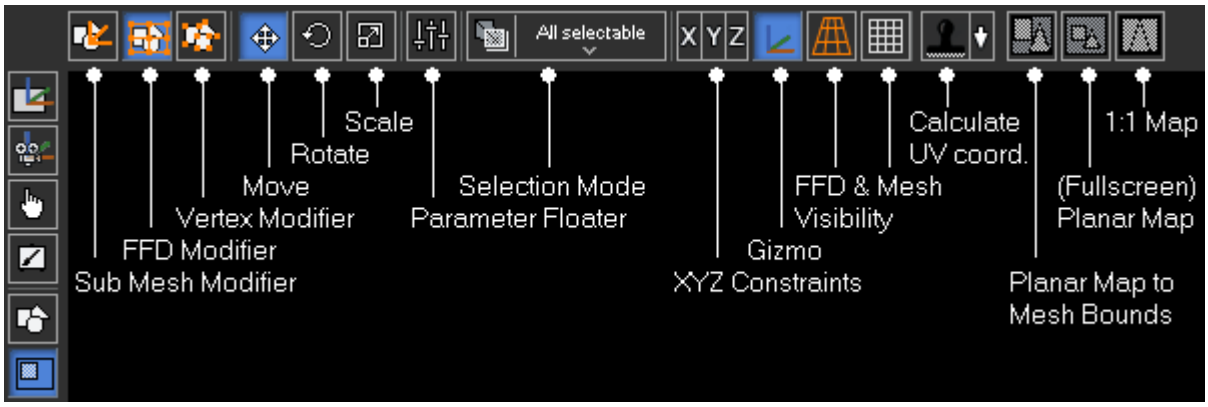
Per default, a newly created Mesh in Pandoras Box has a 1:1 map meaning that any (existing) deformation in the Mesh also deforms the UV source.

Pandoras Box offers two techniques to apply a new UV map to a Mesh or an object. There is the planar mapping and the perspective mapping for which you can choose the point of view. With both mappings you can "print" the texture on a 2D or 3D Mesh again without taking existing deformation into account.

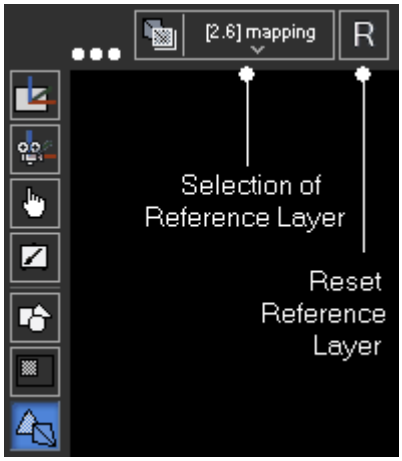


Before you get started with the UV mapping, assign a texture and an Editable Mesh to a Layer. To create an Editable Mesh, right-click in the Project tab and choose "Add Editable Mesh". If you need more than one Surface, right-click on the Editable Mesh and choose "Add Surface" adjust the size using the Surface Inspector or the Mesh Editing Mode.
In case you like to work with an imported object, simply right-click it and choose "Create Editable Mesh from Mesh". Alternatively you can drag it on a newly created Editable Mesh object as depicted to the left. Afterwards you can remove the "Surface" that was added per default.

Tools for the Mapping Modes



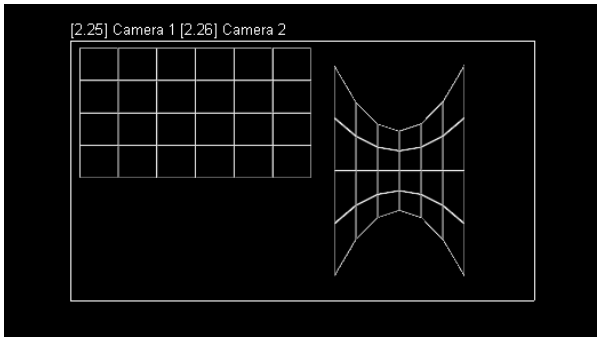
Most buttons are also part of the Mesh Editing Mode [257], please see this chapter for an explanation. The buttons starting with the stamp icon are explained in the example below.
For the Perspective UV Mapping Mode there is an additional drop-down menu to choose and reset a Reference Layer.



Planar UV Mapping Example

▼ This example shows how the planar UV mapping works

You can follow this example with any Editable Meshes. For now, it should be fine to program in a normal preview but later on you might want to switch to the maximized preview  . This button can be found on the left site at the bottom and toggles your Preview tab to a fullscreen window. All tools are available within the Button Bar, that you can toggle with the shortcut [T]. Thus, the maximized Preview gives you the best overview, allows to see more details of the image and adjust the UV Source grids better. You can also work with the Mesh Inspector to position, scale and rotate Meshes.

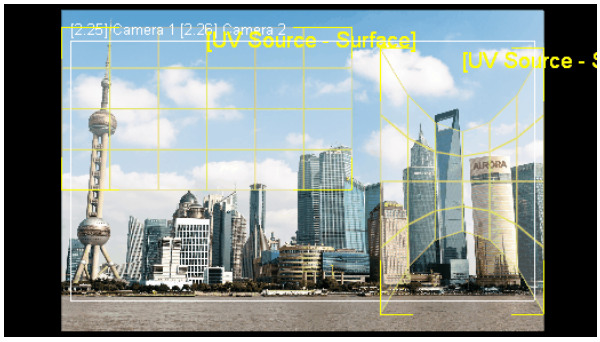


The planar mapping works for all Meshes: 2D or 3D, single or multiple Meshes. This example depicts one Editable Mesh with two Sub Meshes. One Sub Mesh was already deformed with the FFD handles as described in the Mesh Editing Mode [257].

Per default, when assigning an image (i.e. a texture) it covers the entire (Sub) Mesh and is then deformed along the Mesh lines.

Switch the Preview Mode to the  "Planar UV Mapping Mode". Make sure that the view you are working with is correct and that the Editing Context is set to the correct Layer.

Before starting to assign UV coordinates select both Surface UV Sources by clicking into the Preview window and pressing [Ctrl + A]. You should see both names on top of each other, e.g. [UV Source - Surface] and [UV Source - Surface (2)].



Now, press the button  "**Apply Planar Map to Mesh Bounds**". Both UV Source grids are arranged within the limits of the texture. Note that they fit either horizontally or vertically, i.e. the aspect ratio is preserved. The size, position and rotation of the grids does only represent what part of the texture is depicted on the Meshes. The texture itself is displayed in its pixel size.

Press the stamp button  to calculate the UV coordinates and toggle into the  Layer Mode. Note that the Meshes have their old size, position and rotation but the layer texture is shared differently.

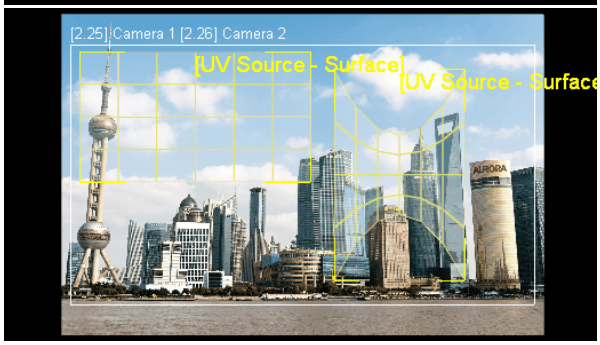
Planar Map to Mesh Bounds

Go back to the "Planar UV Mapping Mode". Select the first button  "Sub Mesh Modifier" (shortcut[M]) and then reposition, rotate or scale the UV grids with the according buttons  or shortcuts [1,2,3]. If needed you can use the Constraints  .

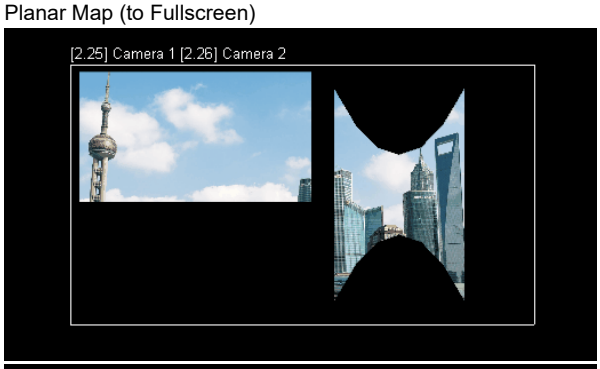
With the FFD or Mesh modifier you could influence the UV map in more detail. As the (drop-down)Selection Mode is set to "All selectable" you can transform each grid. If only one Mesh is selected, the other one will be locked.

Custom Planar Map

Always press the stamp button to calculate the UV coordinates. Switch to the Layer Mode to view the final result. Again, the Meshes have their old size, position and rotation but the layer texture is shared differently.



Planar Map (to Fullscreen)



Enter the "Planar UV Mapping Mode" again and select both grids. Select the second mapping button  "**Apply Planar Map**". This time the UV source grids are not scaled to fit the texture but represent the size of the according Meshes. The texture itself is always displayed in its pixel size.

1:1 Map

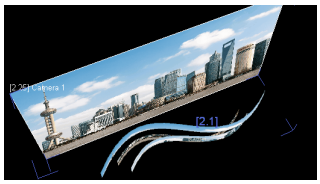
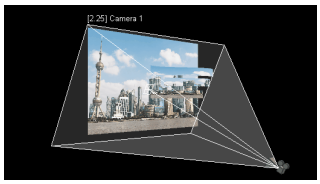
Last, try the third mapping button  "**Apply 1:1 Map**". Note that the UV source of the deformed Mesh is not deformed. In other words, the texture is first applied to the Mesh and then deformed according to the FFD grid. This was not the case in the other planar maps where the content was rather cut out following the mesh lines.

Perspective UV Mapping Example

In difference to the planar mapping, the perspective mapping allows to influence from where the texture is projected onto the Meshes.

▼ This example shows how the perspective UV mapping works

You can follow this example with any Editable Meshes. For now, it should be fine to program in a normal preview but later on you might want to switch to the maximized preview  . This button can be found on the left site at the bottom and toggles your Preview tab to a fullscreen window. All tools are available within the Button Bar [Ctrl + T] and thus, the maximized Preview gives you the best overview, allows to see more details of the image and adjust the UV Source grids better.

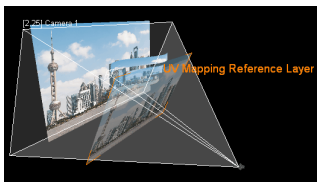



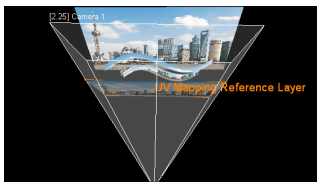
Scene with 2D plane and 3D wave and a Camera looking from a lower angle.
And the front view of the scene and the view from the Camera.

The perspective mapping works for all Meshes: 2D or 3D, single or multiple Meshes. This example uses an imported object that consists of a background plane and three waves in front of it. The images illustrates the scene. The camera position resembles the position of the audience.

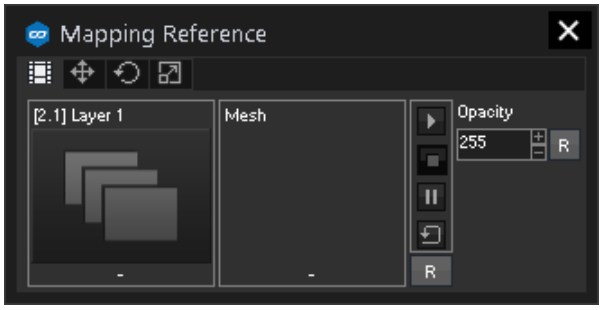
The imported object is converted to an Editable Mesh using the right-click menu in the Project tab.

Switch to the Preview mode "Perspective UV Mapping"

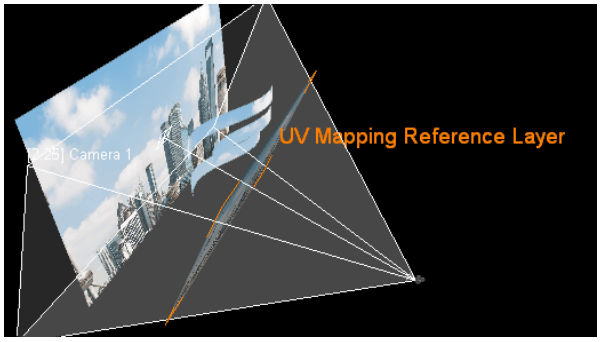




The view automatically zooms out to illustrate the mapping scene with the 3D objects, the camera position and a UV Mapping Reference Layer spanned in the opening angle of the Camera.



In addition, the Parameter Floater opens and shows the Layer's position, rotation and scaling. Per default,. the Reference Layer shares the texture from the Layer you are working with, but if needed you can drag any other media there from the Project tab.



Press the stamp button to calculate the UV coordinates and toggle into the Layer Mode.

Reset the Camera view with the R-button to view the scene again from the saved Camera position which resembles the position of the audience. As the texture was applied from this perspective and spreads over all objects evenly, you can not tell where one object starts or where it is deformed. In other words, the scene looks flat...

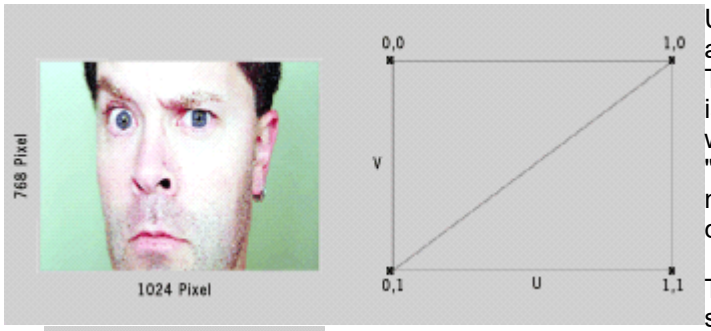
... but as soon as you leave this "sweet spot", your perspective changes. The further you go, the less you have the impression to see a seamless image covering all objects.

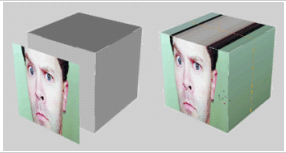
The next chapter covers general mapping questions [274] and is of special interest for mapping beginners.

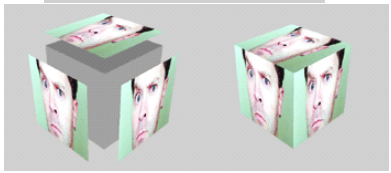
6.3.4.23.9.1 General Mapping Questions

This chapter covers some general mapping questions related to the Planar and Perspective UV Mapping Modes [269] in the Pandoras Box Preview.
What is a UV Map? What tools for texture mapping are provided by Pandoras Box?
For other topics regarding the Preview tab please see the introductory chapter [243].

UV Texture Mapping







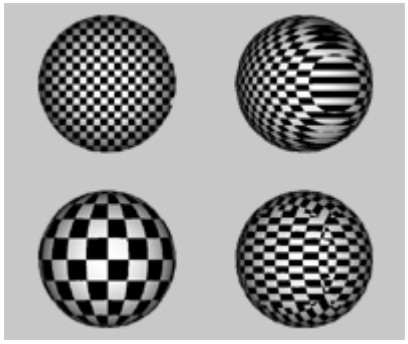
UV mapping is a process of defining how to represent a 2D image on a 3D model.
The 2D source image (in our case an image or video) is a so called texture. In contrast to "X", "Y" and "Z", which are the coordinates for the rendered 3D object, "U" and "V" are the coordinates of the texture. The UV map stores a defined U,V-coordinate for each X,Y,Z-coordinate.

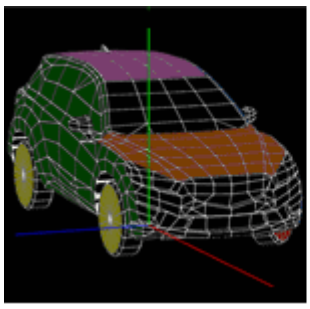
This creates the effect of painting the image onto the surface of the 3D object, or in other words, how to wrap or stretch the image around the object. As shown below there are different ways how to do that, hence the chosen UV map is stored as a property of the 3D model.

When working with objects modeled in a third party software [2177] like Maya or 3ds Max you will most likely also receive textures with them. In Pandoras Box, simply assign the object and the image or video to the same Layer. Pandoras Box loads the object, looks up its UV map and wraps the texture around the object accordingly. There are also effects that influence the UV map [631].
In case you have not received any texture but want to check whether there is a UV map, the fastest way is to assign any image and check whether it somehow covers the object. It is not important how many pixels your image has as the UV map stores UV-coordinates from 0 to 1. With the Warper tool [2129] you can also export the UV map [2143], e.g. to use it in another image editing program.

With Pandoras Box you can also create new UV maps. This works for 2D Editable Meshes created in Pandoras Box but also for imported 2D or 3D objects. You can assign a planar map or a perspective one.

There are several standard mapping techniques available to map a texture onto an object: planar, cubic





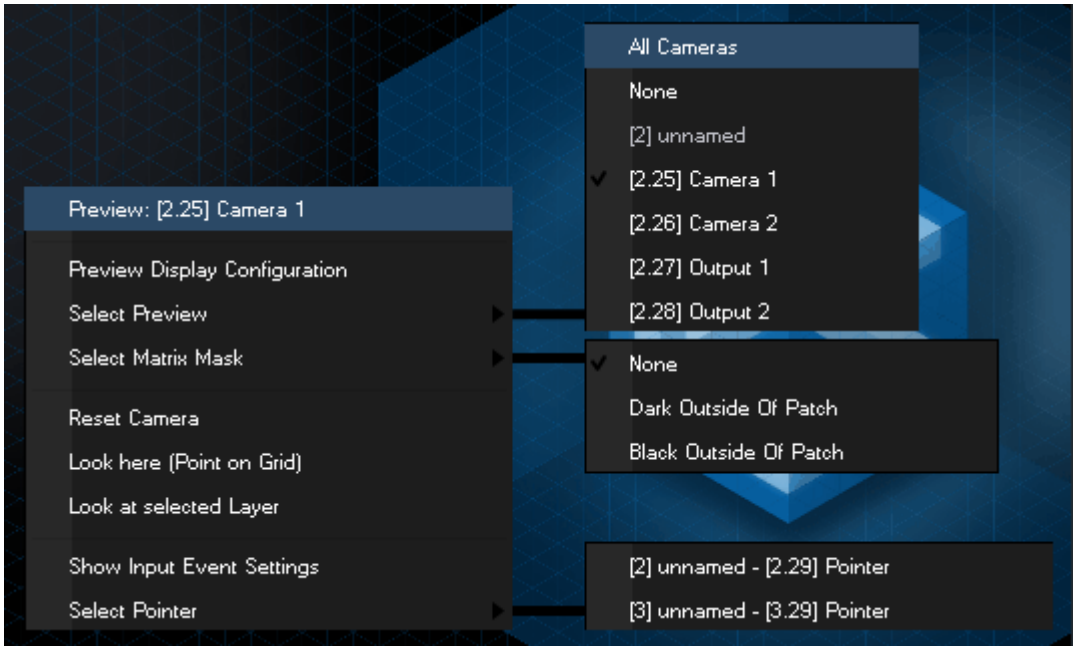
or box, cylindrical, spherical mapping etc. Just as an example, see the box to the left. First, it is assigned with a planar map, then with a cubic mapping that shares the texture multiple times without stretching it.

The car is an example how a complex UV map could look like.

The next chapter explains the Context Menu of the Preview [275].

6.3.4.23.10 Context Menu - Preview

This chapter explains the context menu of the Pandoras Box Preview tab. For other topics regarding the Preview tab please see the introductory chapter [243].



Right-click anywhere in the Preview to open the context menu. The Mesh and UV Mapping Editing Modes offer a few more commands than the other modes. In the Picking Mode, no context menu is available.

The **top line** of the context menu informs you which view is active, e.g the view from the Camera with Device ID 2.25. Below, the command "Select Preview" is explained.

Add Sub Mesh
This command is available in the Mesh Editing or UV Mapping Modes and adds a Sub Mesh to the Editable Mesh that is assigned to the Editing Context Layer.

Remove Sub Mesh
This command is available in the Mesh Editing or UV Mapping Modes and removes a Sub Mesh from the Editable Mesh that is assigned to the Editing Context Layer. Position your mouse over the Sub Mesh that you like to remove and open the context menu.

Reset Sub Mesh
This command is available in the Mesh Editing or UV Mapping Modes and offers several reset options for a Sub Mesh from the Editable Mesh that is assigned to the Editing Context Layer. Position your mouse over the Sub Mesh that you like to edit and open the context menu.
Reset **All** resets the transformation, FFD and Vertex Modifier as well as the UV Coordinates which means the mapping.
- In the Mesh Editing you can activate the Sub Mesh Modifier and change the position, scaling or rotation for a Sub Mesh. This also changes the parameters displayed in the Sub Mesh Inspector. The command **Reset Transformation** resets those changes.
- In the Mesh Editing you can activate the FFD and Vertex Modifier and change the FFD or Vertex points. The command **Reset Modifier** resets those changes. Alternatively, you can go into the Sub Mesh Inspector and reset the FFD and / or the Vertex grid.
- In the Planar and Perspective UV Mapping Modes you can change the UV Map of Editable Meshes. The command **Reset UV Coordinates** resets those changes but only for Editable Meshes created in Pandoras Box (not imported and converted ones). Alternatively, you can go into the Sub Mesh Inspector and click the button "Reset UV Coordinates".

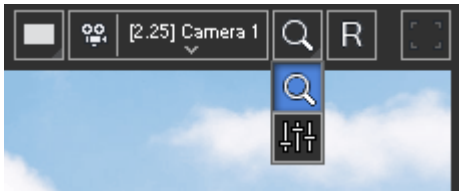
Preview Display Configuration
This opens the Configuration tab in a panel and displays the category "Preview Display" that includes for example the visibility for Gizmos and Wireframes.

Select Preview
Open the sub menu and choose one of the offered views to switch to it. Note that a Site must be included in the Preview [244] rendering. The so called "All Cameras" view is always available and allows previewing all included Sites at once. Furthermore the whole 3D space is shown instead of only the specific area seen by the Site's Camera or Output device. The so called "All Cameras" view is always available and allows previewing all included Sites at once. Furthermore the whole 3D space is shown instead of only the specific area seen by the Site's Camera or Output device.
Please note: Everything that is set up in the Output device of this previewed Site, i.e. a warping object, softedge etc. will NOT be shown in the All Cameras view! Please change to the Output's preview if you want to see the Output settings as well!

Select Matrix Mask

When working e.g. with LED walls that are controlled via Art-Net you can choose to preview the Art-Net output instead of the video (e.g. DVI) output. Whilst being in the output (!) view right-click and select a matrix mask. To use the matrix feature you first need to create a patch in the Matrix Patcher [2077] and export it to your Pandoras Box project. Drag the new format ".pbx file" on any(!) Output Layer [682]. Now, you may choose to preview it as described. **Dark Outside of Patch** dims the area not covered by the patch whilst the **Black** option does not display it at all. Choose **None** to see the entire area without the highlighted patch.

Reset Camera



This command resets the view, i.e. the Camera or Output position. Shortcut: [Ctrl + 0]
Note that only the "Zoom" Interaction Mode [248] is reset. When you toggle to "Parameters", view changes influence the parameters of the Camera or Output device which are not reset with this command. However, the R-button at the top right resets both modes.

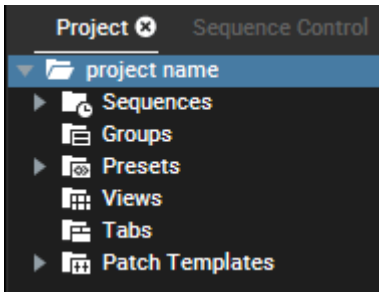
Look here (Point on Grid)
Moves the view so that the point where you clicked (projected to the Ground Grid) is in the center of the view port. This does not work for Camera and Output devices without the Target parameter.

Look at selected Layer
Moves the view so that the selected Layer is in the center of the view port. This does not work for Camera and Output devices without the Target parameter.

Show Input Event Settings
Shows the dialog "Input Event Settings" which includes the Picking settings for all Sites included in the Preview. See more in the chapter Picking Mode 252.

Select Pointer
This entry has a sub menu listing all Pointer Layers of your Sites. Choose the one you like to see in the Preview. Pointer Layers are of interest for the Picking Mode 252 etc.

6.3.4.24 Project



The Project tab holds all information regarding available:

- Media files (also called resources) and folder structure, see below for more information and the next chapter for the commands in the right-click menu [279]
- Sequences [281]
- Groups [281]
- Presets [283]
- Views [310]
- Tabs [288]
- Patch Templates [290]

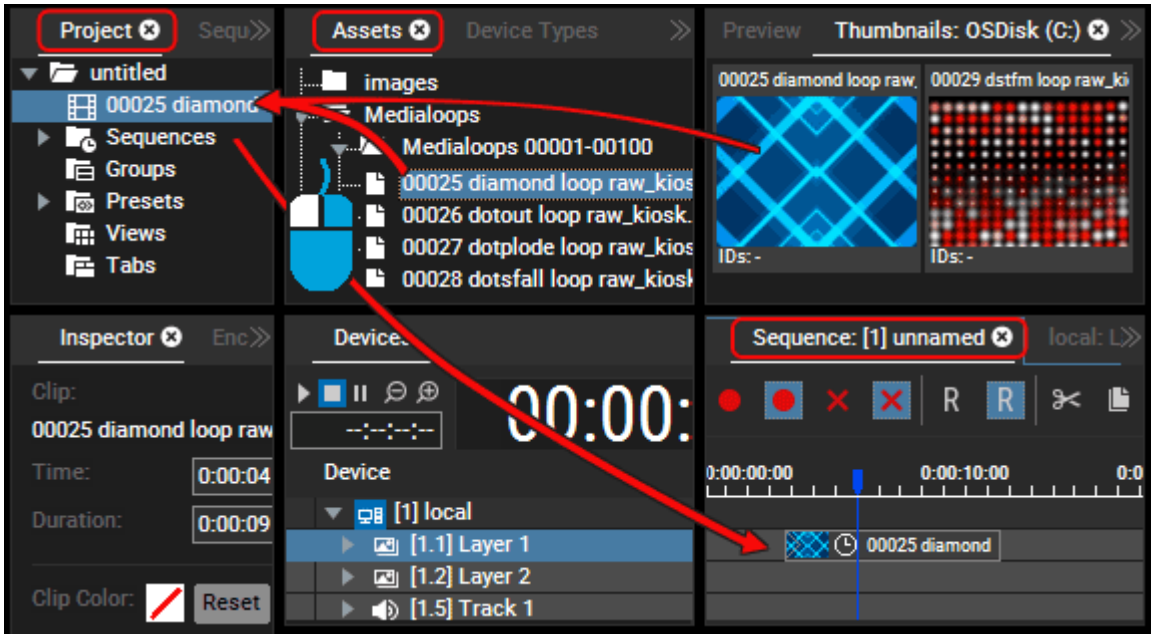
All devices, such as Servers and Compact Players, that are managed by the project are located and displayed in the Device Tree tab [173].

Adding Media Files to the Project

The Project tab's main folder lists all available media files. Only media that is part of the project can be used by layers. Add media files to the project by dragging them from the Assets tab [131] or the Thumbnails tab [310].

Pandoras Box creates links to the media file, it will not copy the file! The content chapter describes the supported media files, detailing supported formats as well. You may add Audio files [92], Still Images [93], Image Sequences [95], Videos [97], Matrix Patches [2077]. Other media types such as Text Inputs or Browser Assets are described in the next topic as they can be added via the context menu [279].

You may drag single files or entire folder(s), optionally including sub folders. Afterwards, you can drag content from the Project tab onto the Sequence as described in the chapter "Programming on the Timeline [296]".



Spreading Media Files

After adding content to the project it is automatically spread to all connected systems, if this default setting is not deactivated in the Configuration tab [139]. "Spreading" actually copies files to the other hard disk. If the Auto-Spread function is deactivated you may spread single files or folders with the according "Spread" command in the right-click menu. Content that was spread already is only copied again if the file has changed.

Again, Pandoras Box creates links, so if you move content to another folder, either on the Master system or a Client system, the links point to an non-existent file and it can not be displayed. In the Project tab, the so called inconsistent file is then marked with a red explanation mark. To view the file paths click the file and look into the File Inspector [191] for the File Location Table [193].

An exception to this rule is when you are using content from the project's "asset folder", also explained under "Assets [131]". As soon as a project is saved, a folder named "assets" is generated just next to the show file. If you copy content files to that folder and drag them from here into your Pandoras Box project the content links are relative. That means that you can move the project folder (including the *.pbb and asset folder) to every path wanted without the inconsistency effect and the need of re-linking the content.
This is especially of interest for users that use the "Bundle Project [119]" feature.

Working with Content and Watch Folders

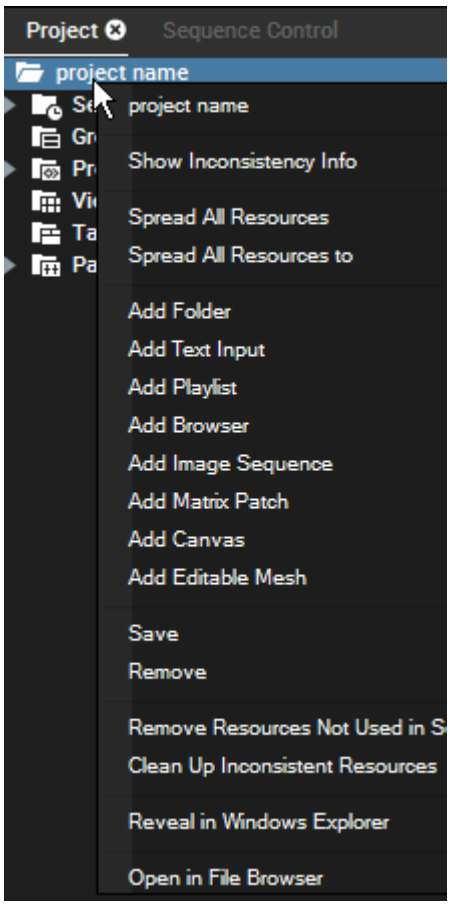
As Pandoras Box only creates links to files on the hard disk it is possible to create a folder structure in the project that is independent from the one on the hard disk. You may create content folders that only exist in the project, not on the hard disk. On the other hand you may drag entire folders from the hard disk to the project and the entire folder structure including all sub folders is used, but again, it is only linked. This means, that you are free to alter the folder structure.

Right-click in the Project tab, e.g. on the first folder, and choose "Add Folder". After selecting it, you may rename it with another right-click or the shortcut [F2].

Since version 5.5 you may turn a folder into a so called watch folder. This will automatically add a media file to your project if it is added to the watched folder on the hard disk. You may as well use the above described project's assets folder as a "Watch Folder" in Pandoras Box... Read more in the Folder Inspector [197].

6.3.4.24.1 Context Menu

Right-click in the Project tab, e.g. on the very first folder to open the context menu. Some commands are only available when clicking a single media file.



Show Inconsistency Info
This opens a new dialog that lists all media within the right-clicked folder that is inconsistent. Inconsistent media files might have a new name or path or became corrupted, in any case, Pandoras Box can not find or read them in the intended location. Read the File Inspector [193] for more information about how to solve inconsistencies.

Spread all Resources
This copies all files from the Master system to all connected Clients. If you have done this before, a file will only be re-spread if it has changed.

Spread all Resources to ...
In principle, this works like the "Spread all Resources" command. But here you can choose from a new menu whereto the file(s) are copied. "All Sites" means all connected Sites (e.g. Servers) that are part of the Device tree [173] . If you like to copy the file(s) to one site only, choose it from further down in the menu. "All Places" means all connected Masters and applies when working in a Multi-User environment [234].

Add Folder
This creates a new folder that only exists in the Project, not the hard disk. Now, you can drag content into it or turn it into a Watch Folder [197]. More information about Content Folders... [279]

In addition to the above described media types, Pandoras Box supports other media sources. Mostly they are created in the project itself and are stored in it.

Add Text Input
A Text Input displays text (in various formats) and is stored within the project. When the new text entry is selected, you can change its properties in the Text Inspector [203]. At the top there is a button to open the "Text Input Editor" tab where you can modify these and more settings.

Add Playlist
A Playlist plays back various media files (images and videos) one after another with an optional cross-fade on ONE layer. There is no need to program all files separately on the timeline. To modify the Playlist, e.g. add content or change its order, open it in the Playlist tab [239]. The Playlist is stored within the project.

Add Browser
A Browser is able to render a web page and is stored within the project. See its properties in the Browser Inspector [198].
Please note that the render and loading process might take up to 15seconds depending on the texture size and web site complexity. This is also subject to the internet connection.
A browser asset can be clicked as well, e.g. to execute included hyperlinks, if the feature Layer Picking [252] is activated in the respective Device Inspector [210]. A click input can be displayed by a Pointer Layer [665].

PB uses the browser engine "CEF" to load the web sites. CEF is an embedded browser engine based on Chromium and the current implementation is based on Chrome 63. If you like to check the Chromium version that is used, you can refer to web sites like https://whatsmybrowser.org; simply enter the URL in the Inspector of your PB Web Browser asset.

According to the Chromium version the framework supports HTML5, Javascript, CSS3 and Silverlight. Since Chromium 56 Flash is blocked. Again, you can use other web sites to check certain web features, for example https://caniuse.com. It is always recommended to test web sites before using them in a show to verify that the used techniques are working in the embedded Chromium version..

The support for native multiple touch events in CEF works since version 6.1.3. In former versions, custom JavaScript code is the only possibility for this feature.

CEF caches the web site first. This process takes some seconds and depends on the web site, but 2-15 seconds is normal. When the site is "ready", Pandoras Box displays it at once.

Add Image Sequence
An Image Sequence is an alternative to video files, see the differences in the according Content chapter [95]. The best way to add an image sequence is actually found in the Assets tab. Right-click on a folder and choose "Import as Image Sequence". Choose the frame rate in the pop-up. More options can be found in the Image Sequence Inspector [199].

Add Matrix Patch
A Matrix Patch feeds displays like LED walls that are connected via Art-Net instead of VGA, DVI or HDMI. You may add an existing Matrix Patch file using the Assets tab. If you create a new one in the Project tab, the Matrix Patcher [2077] opens with a new file. Matrix patch files are meant to be used on a Output Layer [682] or with an Aeon FX e.g. ReMap [580].

Add Canvas
A Canvas is a dynamic image file. It is not a static image like a bitmap for instance, nor is it a predefined video that has been rendered out beforehand. A Canvas is a still image texture but can receive drawing data, meaning that it can be modified in real-time without the need of saving the image itself! The Canvas Editing Mode [256] from the Preview allows to draw on the Canvas. Alternatively the drawing data can be sent from the Widget Designer and is generated through all possible tools, like a mouse or the AirScan. You may set up a brush size (dots, circles, or even another texture using it as a brush tip) and the color. This way you can apply the Canvas to a Layer to let your customer interact with your projection just for fun. Or you can use the Canvas to generate a live mask. More information about the settings in Widget Designer... [1245]
As a third option, you can use the Pandoras Box SDK [730] or solutions like Christie Mystique to draw on a Canvas and apply it then as a "Pixel Warp [686]" media to the output to warp it.

The Canvas needs to be stored on the hard disk.

Add Editable Mesh
An Editable Mesh is for warping in the preview. Find more info in the chapter about the The Mesh Editing Mode [257]
.
Note that the newly created Editable Mesh folder lists one Sub Mesh called "Surface" which is a planar 2D Mesh in fullscreen size with a 3x3 FFD [259] grid. The Editable Mesh needs to be stored on the hard disk.

Save (As)
This saves the current project either for the first time or with a different name and path if you like.

Rename
This renames the clicked item. Note that renaming a media file, renames it only in Pandoras Box, the real file (on the hard disk) stays unchanged.

Remove
This removes the clicked item from the Project.

Remove Resources Not Used in Sequence
This removes all media files from the project if they are not used in any of the Sequences. This is a handy command when bundling [120] a project and you like to make sure that no unnecessary media is bundled.

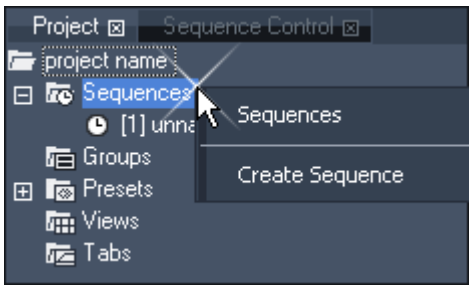
Clean Up Inconsistent Resources
This removes all media files from the project if they are inconsistent. Read the File Inspector [193] for more information about how to solve inconsistencies.

Reveal in Windows Explorer
This opens a new instance of the Windows File Explorer with the path of the clicked media file and selects it.

Show Used Content in Sequence
This selects the Container in the Sequence where the clicked media file is used in and displays its information in the Clip Inspector [206]. If the file was used more than once, a dialog opens and lists all containers with the number of Sequence, Site, Layer and their Time.

Open in File Browser
This opens a new File Browser tab [185] which shows the content of the respective folder from the Project (or Assets [131]) tab as thumbnails. You can open as many File Browsers as you like. If you are interested in changing the layout of the user interface, please see the chapter "Layout [313]". It describes how to add or break-out panes and how to save and load views.

6.3.4.24.2 Sequences



The Sequence folder in the Project tab [278] includes all available Sequences. Depending on the Pandoras Box version (please see "Product Overview [62]") you are using as the Master system, you are allowed to create more than one Sequence. To do so, right-click on the Sequence folder an choose "Create Sequence".

A new Sequence with a new ID is created. To rename it, right-click on the new Sequence. The other commands available in this context menu are the same as in the context menu [293] when you right-click in the time bar within the Sequence tab [292].

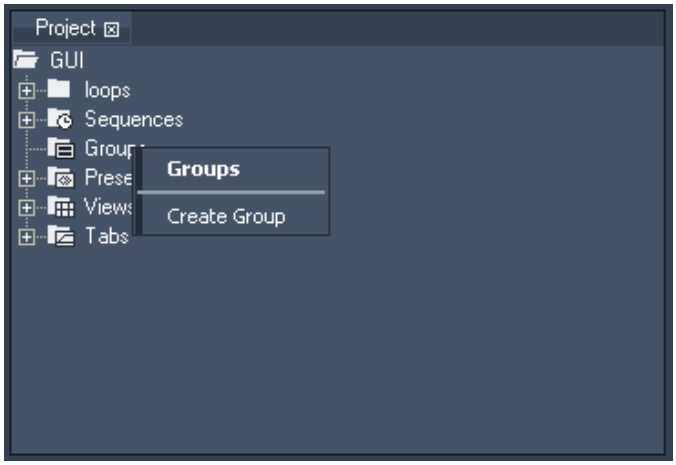
This tab displays all available Sequences as large thumbnails. Among others, it enables you, to toggle play / pause of various timelines with one click only.

To view the properties of a Sequence select it in the Project tab and look into the Sequence Inspector [204]. It shows all settings, e.g. timecode settings and the Master Sequence opacity.

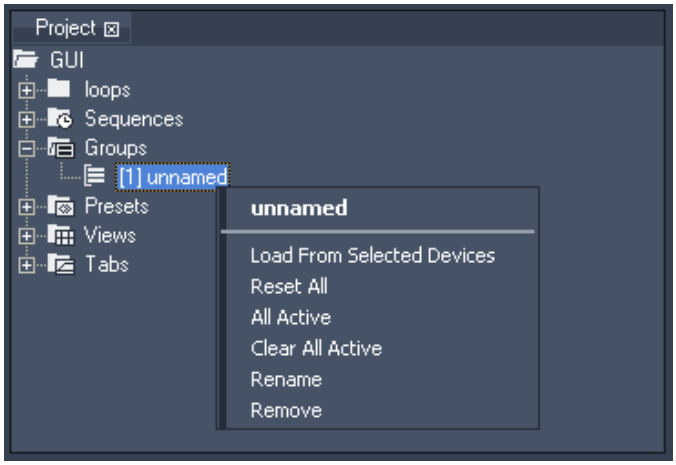
6.3.4.24.3 Groups

Within Pandoras Box groups define a multi-selection of devices according to their selection order.

To select multiple devices hold down the [Ctrl] key on the keyboard and left-click on the devices to select.
All selected device node icons will turn blue now.



In order to store this selection as a group for later editing, right-click on the group folder in the project tab and choose Create Group.



Once the group is created, you can recall the saved selection by simply clicking on the entry. Alternatively you can do this in the Groups tab [189].

If you right-click on the group name, the context menu gives access to more functions:

- Load From Selected Devices
This command will overwrite the actual group selection with the current selection in the device tree.

- Reset All
This command will reset all parameters for all devices stored within this group.

- All Active
This command will activate all parameters of all devices stored within this group.

- Clear All Active
This command will de-activate all parameters of all devices stored within this group.

- Rename
This command will allow you to rename the group.

- Remove
This command will delete the group from the project.

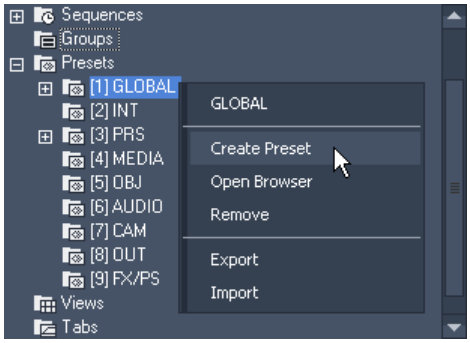
To select all devices that are stored within this group, just click on the group name in the project tab.

6.3.4.24.4 Presets

In Pandoras Box, presets can be used in two ways:
1. A preset can be a single snapshot taken from active values at one point in time. The snapshot may copy all parameters or it uses a predefined parameter filter, e.g. copy opacity values only.
2. A preset can hold an entire set of keys. The "source" keys must be stored in the timeline first before being turned into a preset. Again the recorded parameters may be chosen.

Both preset variations can be applied at any (other) point in time to any (other) layer. In this way one situation can be created and then be reused severally in the programming process.
In addition a preset can be modified in later programming and automatically change all according keys in the Sequence.

Creating a Snapshot Preset



After setting up active values do a right-click on the preset type you want to use and chose "Create Preset". You can do this either in the "Project Tab" or in the according preset browser (the Tab "Presets") 241.

The parameters that transfer their values to a preset will lose their active status.

Please note:
By default, presets taken this way look at all devices (no matter whether they are selected or not). If you wish to store values for selected devices only, create an empty preset first, then make a right-click on it in the project or preset tab and choose "Store active (Selected devices)".

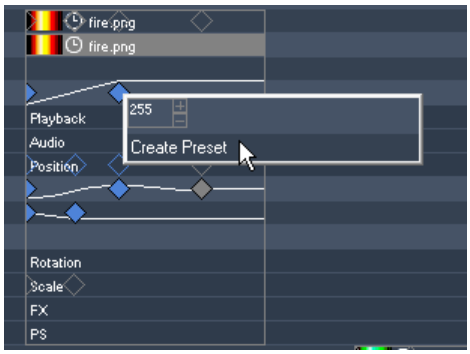
PRESET TYPE	PARAMETERS STORED
Global	All Parameters
INT	Opacity
PRS	Position, Rotation, Scale
MEDIA	File Selection, Transport Control,(Transport, Speed)
OBJ	Object Selection
AUDIO	Audio Parameters
CAM	Camera Control Settings
OUT	Output Control Settings
FX / PS	FX & Particle System parameters

The preset folder contains default preset banks for active value filtering during store operation. This means that by creating a preset in any of these categories only the parameters that match the category group will be stored.

Example:
When creating a preset in the GLOBAL bank, all active parameters will be stored within this preset.

When creating a preset in the PRS bank, only the active values of position, rotation and scale parameters will be stored in this preset. All other parameter keys are not stored and therefore will remain active.

Creating a Sequence Preset

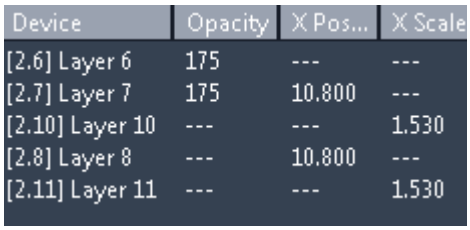


After creating a set of keys in the timeline choose all the keys you want to turn into a preset. In order to make a multi-selection use the [Shift] key. Now make a right-click on one of the keys and choose "Create Preset".

Please note:
Presets taken within the Sequence tab are always stored in the GLOBAL bank.

 Keys that belong to a preset will change their look from the standard diamond icon to an icon that has right angles along the top.

Applying and Pasting Presets



Device	Opacity	X Pos...	X Scale
[2.6] Layer 6	175	---	---
[2.7] Layer 7	175	10.800	---
[2.10] Layer 10	---	---	1.530
[2.8] Layer 8	---	10.800	---
[2.11] Layer 11	---	---	1.530

"Applying" a preset to one or more layers will activate those layer parameters that are stored inside the preset. The values are not inserted into the timeline as long as they are not stored intentionally.

Please note: "Applying" a Sequence preset will call the first value per parameter and ignore the following ones. E.g., assume a preset has three keys: X Pos = 6 @0sec and 8@2sec; Y Pos = 2 @1sec. The "Apply" command would lead to a selected device having two activated parameters: X Pos = 6 and Y Pos = 2. Importantly, the second X Pos key (with value 8) is ignored in this preset interpretation since only the "leftmost" key for each parameter is taken into account.

In order to apply a preset, choose the command from the presets right-click menu in the project or preset tab or double-click it in the project tab.

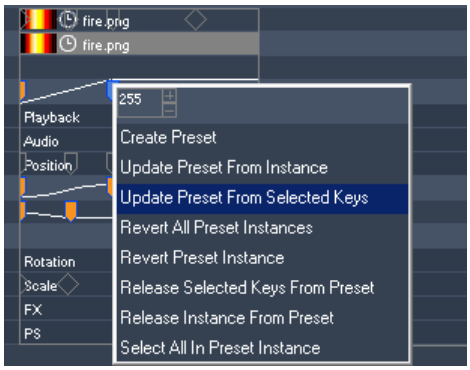
"Pasting" a preset to one or more layers will insert all preset keys into the timeline track (in the currently open Sequence at the time according to the nowpointer). These pasted keys are called an "instance" of the preset. The inspector informs you about the associated instance if you select a key in the Sequence.

In order to paste a preset, choose the command from the presets right-click menu in the project or preset tab or make a single left-click it in the preset tab.

Applying and pasting presets will always "start" with the selected device. E.g., assume a preset has three keys: Opacity for layer 1, X Pos for layer 2 and X Scale for layer 5. When being used while Layer 6 is selected, Layer 6 itself will obtain the opacity value, Layer 7 the position value and layer 10 the scale value. The lower left image depicts what happens if layer 6 and layer 7 are selected the same time.

You can, of course, repeat the "Apply" and "Paste Keys" commands as often as you like. I.e. the preset is a form of template which allows you to quickly create multiple instances.

Modifying Presets



If you change the time or value of a key that belongs to an instance, all associated keys are colored orange to show that the instance diverts from the preset as saved in the project tree.

Starting from a divergent instance, you have four main options:
1) apply the changes made in the instance to the master preset and all other instances
2) discard the changes and make the instance comply with the master preset again
3) transform the keys to be normal keys again, not being included in the instance any more
4) create an entirely new preset, starting from this situation

Note that, in the initial version of Pandoras Box that will have these temporally oriented presets, you can only edit a preset by first placing an instance of it in the Sequence. In other words, there is no editor available in which to manipulate the master version of the preset directly. However, such an editor is planned for future versions.

[Create Preset]
(4) Creates a new preset by mapping all selected keys. Note that non-selected keys would stay orange as their instance has still changed.

[Update Preset From Instance]
(1) Applies the changes made in the instance to the master preset. Note that you do not need to select all keys belonging to the instance, all together operate as the new source instance.
The current instance changes its color to grey, all other instances in the timeline change their values additively in order to comply with the new preset.

[Update Preset From Selected Keys]
(1) In contrast to the command above, this time not the entire instance operates as the new source for the preset, but the selected keys only.
This command is especially useful if an additional key (either before the first key or after the last key per parameter or one in a totally new parameter) should be added to the preset.

Please note:
An instance is always pasted at the time according to the nowpointer. It is always the very first key from the instance that is inserted at the nowpointer´s position and all other keys refer to this first key. If you change the preset regarding the first key, the other presets will change so that it is always the first key that holds the time the preset was once inserted. That could mean that other keys change their particular time.

[Revert All Preset Instances]
(2) Discard the changes made in one or more instance and make them all comply with to the master preset again.

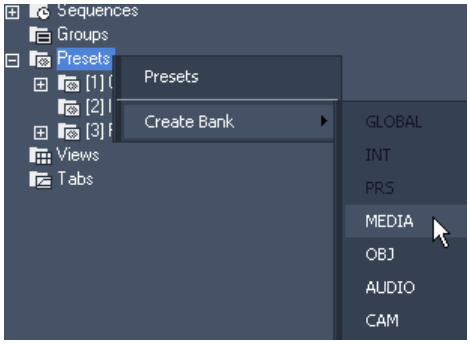
[Revert Preset Instance]
(2) Discard the changes made in the current instance and make this instance comply with the master preset again.

[Release Selected Keys From Preset]
(3) Transforms all selected keys to normal keys again. They are not included in the instance any more. Note that all other keys from the original instance will change they color to orange, as the instance has now changed.

[Release Instance From Preset]
(3) Transforms all keys belonging to the according instance to be normal keys again. They are not included in the instance any more.

[Select All In Preset Instance]
(-) Selects all keys belonging to the according instance.

Additional Commands

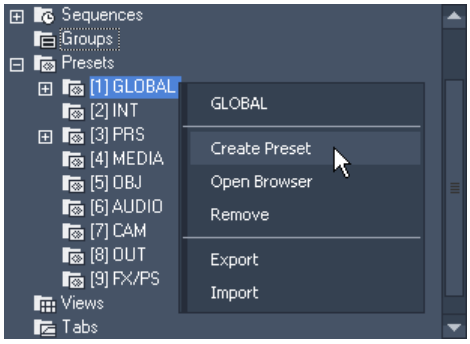


The context menu of the preset folder in the project tab provides the following option for organizing presets:

[Create Bank]
Re-creates a preset banks if it was removed previously.

For each preset folder you may open a designated browser [241] as an individual tab within the user interface. This tab can be stored in a view as well.

As well you may import previously saved presets to them.



The context menu of a preset type provides the following options for creating and organizing presets:

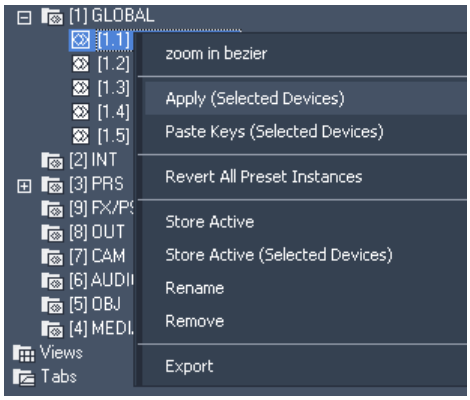
[Create Preset]
Creates a snapshot preset by looking at the current active values and copying them as a preset key (at zero time) according to the current preset bank.

[Open Browser]
Opens the Tab "Presets" [241].

[Remove]
Deletes the entire preset bank and all including presets.

[Export]
Saves all including presets as an XML file.

[Import]
Adds the previously exported presets to the preset bank.



The context menu of a preset provides the following options for creating, using and organizing presets:

[Apply (Selected Devices)] or double-click the preset
Applies a preset, starting at the first selected layer and activates those layer parameters that are stored inside the preset. The values are nor inserted into the timeline.

[Paste Keys (Selected Devices)]
Creates a new instance from the preset by pasting its keys the into the currently open Sequence starting at the first selected layer.

[Revert All Preset Instances]
Discard the changes made in one or more instance and make them all comply with to the master preset again.

[Store Active]
Stores all active parameter values to the preset that are allowed to be stored in this preset type.

[Store Active (Selected Devices)]
Stores all active values of the selected devices to the preset.

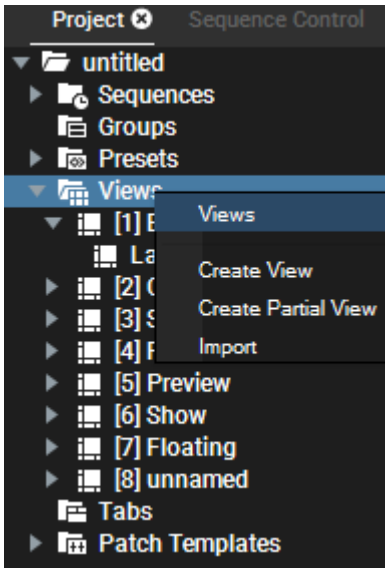
[Rename]
Renames the preset.

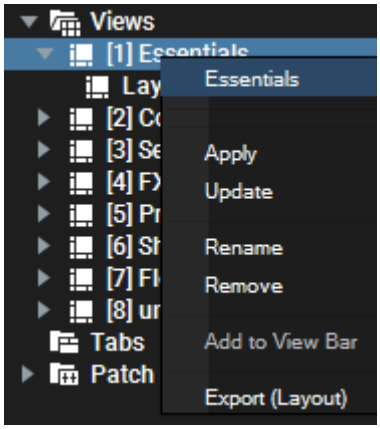
[Remove]
Removes the preset.

[Export]
Saves all including presets as an XML file.

Please refer to Presets tab [241] for further information.

6.3.4.24.5 Views





The "Views" entry in the Project tab, the View tab [310] and the View Bar [119] let you recall stored view layouts of the user interface with a single mouse click. This makes it very easy and fast to switch from one view, e.g. focusing on a large Preview window, to another view, e.g. displaying necessary Sequence tabs.

The View Bar shows seven views. The first, called "Essentials", is the default view. The others are named "Content, Sequence, FX, Preview, Show, Floating" and emphasize a particular use of the UI. Simply click on a View to change the currently shown tabs and their arrangement.

With the "Views" folder from the Project Tree, you can change existing views or create new ones. The chapter "Layout [313]" describes how you can influence the GUI layout: inserting new tabs and panes, breaking panes out, removing and resizing them.

To change an existing view, right-click on it and choose "**Update**". This saves the current layout.

When saving a new layout, you can now choose between two options from the right-click menu of the "Views" folder. With "**Create View**" you save the entire layout as usual (with a small structural change as explained further down). With "**Create Partial View**" you are saving only a specific part of the interface. A dialog asks which of the four sub-views should be included:
- "Layout" saves the layout of the UI including the pane arrangement, size, break-out state, which tabs are loaded and what they display. Its pretty much everything except of what the following three partial views save.
- "Preview" saves the layout of the Preview including the mode, number of preview windows and assigned cameras/outputs.
- "Sequence" saves the loaded Sequence ID.
- "Device Tree" saves which devices and sites are hidden or shown.

When creating a View (via "**Create View**"), all four sub-views can be found as sub entries in the Project tree. They can be applied, updated or removed separately. The seven new default views change only the "Layout" part, so that the currently loaded Sequence, Preview and Device Tree do not change.

You can add a new View to the View Bar to access it faster. You can either drag it there from the Project tab, or right-click it and choose "**Add to View Bar**" or make a right-click in the View Bar and choose "Add".

In the View's right-click menu, you can also find the option to "**Export**" a view, which saves the "Layout" sub view as a pbv file to the hard disc.
The right-click menu of the main Views folder, offers the corresponding "**Import**" option. By the way, the default views are saved in the installation path under `data/views`.

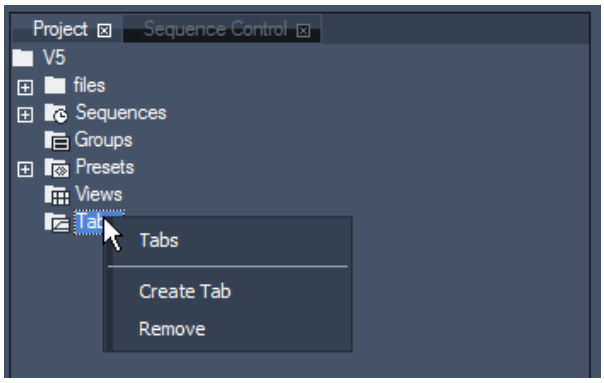
Views are saved within the project file.

6.3.4.24.6 Tabs

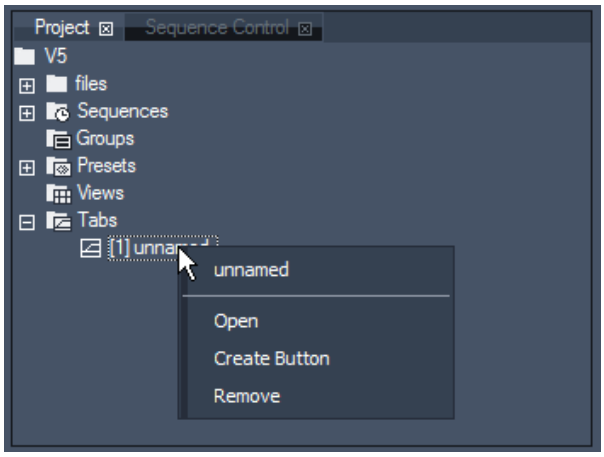
To access most of the common used context and main menu commands, custom tabs may be created to allow you a quick access via a single click from within a view layout.

Custom tabs consist of buttons that are especially useful for touchscreen operation with the user-interface as well as for direct access to your favorite features to speed up your workflow.

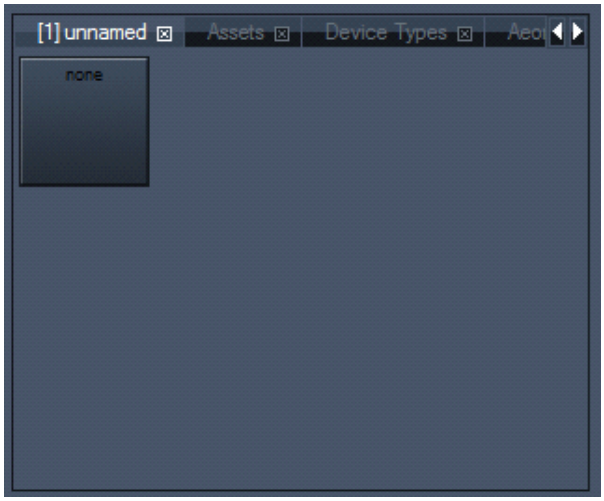
As tabs are part of your project, all settings you make here will be stored with each project you are working in.



To create a new tab, right-click in the project tab on the folder called "Tabs" and choose "Create Tab".

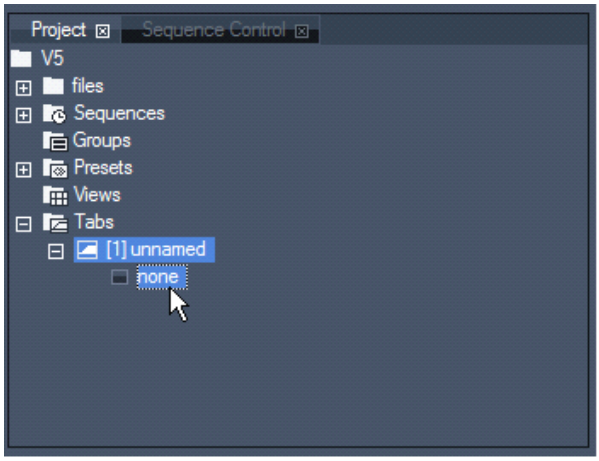


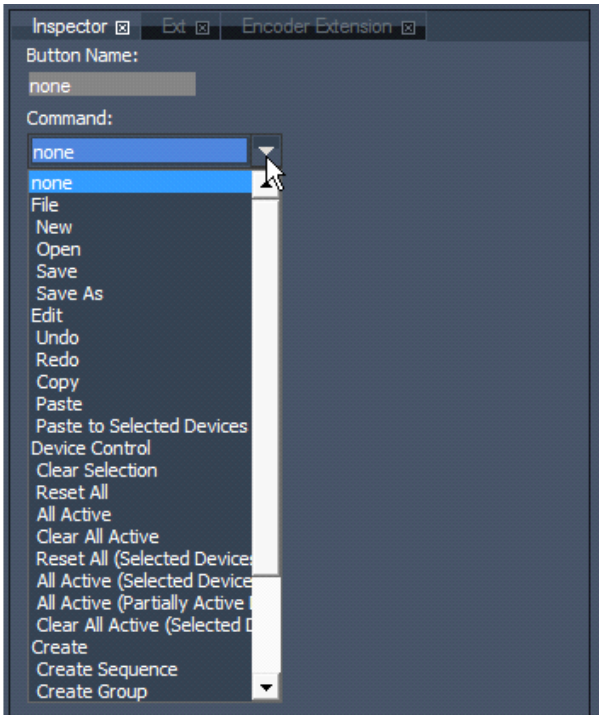
Once a new tab entry is created, you will need to add button items to the tab.
Do this by right-clicking on the new tab item and choose "create button".

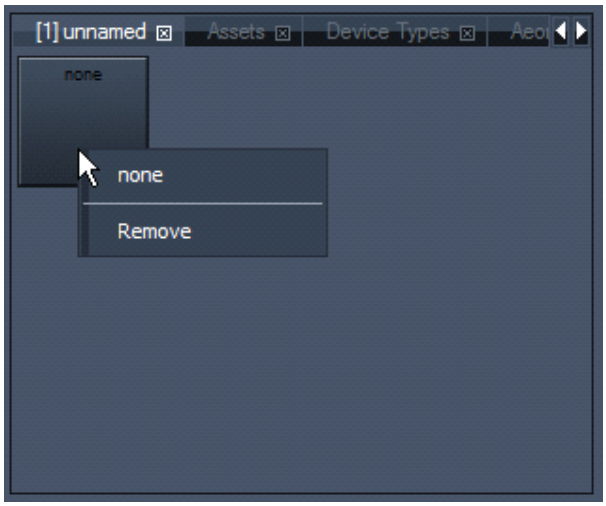


To open and show the new tab in the user interface right-click on the tab again and choose "open".
Now you should see a new tab with a single button in the user interface.

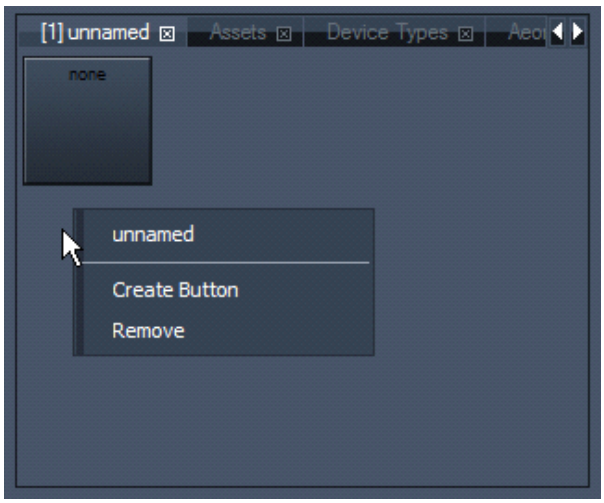
To assign a command to this button or to edit the commands of any created button select the button in the project tab. Go to the Button Inspector tab and select your favorite command from the drop down list. You may rename the button here as well.







To remove a button from the Button Tab, right-click on it and choose "Remove".



In the Button tab view you may right-click in any empty region to create new buttons.

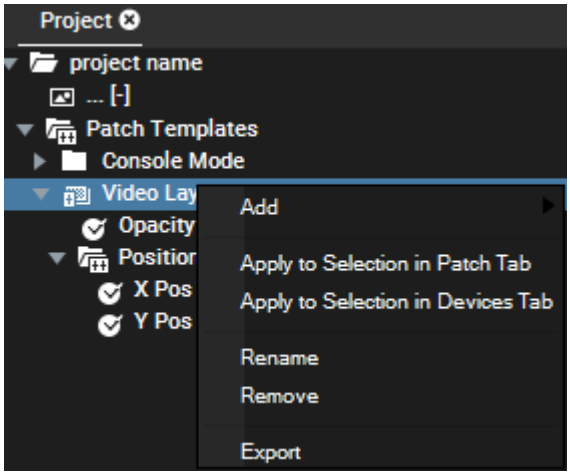
Choosing "Remove" from this context menu will remove the whole Button Tab from the user interface.

6.3.4.24.7 Patch Templates

In the Patch Templates folder in the Project Tree you can organize and edit your patch templates. A template lists all parameters that should be patched. You can save templates, export and import them and apply them to other Layers / Devices or Sites.

Per default, the Project tab includes one template called "Console Mode". Expand it to see that it includes the default patch for all kinds of Layers.

To add custom templates, you have two options. The first is described now, the second further down. Firstly, you can save one from a Site, Device or Layer that is already patched. Open the Patch tab [229] with a Site (or DMX fixture [183]) and choose "Create Template" after right-clicking a Layer (or Device or Parameter or Site). This saves the patched parameters to a template. Open the "Patch Templates" folder in the Project tab [278] and you will notice that it includes a new template with a name that depends on what kind of Layer or Device was saved. If you save a single parameter or entire Video Layer the template is called "Video Layer" and includes the according parameters. If you save a template from an Output Layer, a new template "Output" will appear, etc. Each entry can be expanded again to see which parameters are saved there. In the depicted example, the custom "Video Layer" template includes "Opacity", "X Pos" and "Y Pos" because these parameters were patched before.



In the Project tab, there are several context menus available. If you right-click on "Video Layer", for example, you will see the below options:

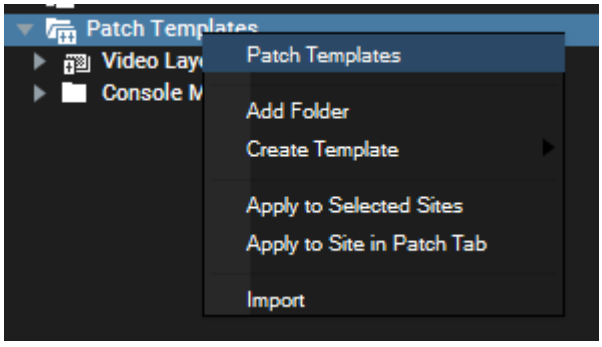
Add
If you like to add another Parameter, right-click on a Layer/Device entry, choose "Add" and select it from the list. To remove a Parameter from the template, right-click on it in the Project Tree and choose "Remove".

Apply to Selection in Patch Tab / Apply to Selection in Devices Tab
You can choose "Apply to Selection in Patch Tab" or "...Devices Tab" from the Template's context menu if you like to patch the same parameters in the Layer you selected in the according tab. Obviously you can only transfer a patch to the same Layer or Device type because the parameters must match. Remember to enter a new starting address in the Patch Tab, as described above. Either, enter your numbers in the "Addr" and "Universe" field for the Site or Layer to increase the address automatically. Or, right-click the Site or Layer in the Patch Tab and choose "Patch > Set Patch Start Values".

Rename
This allows you to choose another name for the entry.

Remove
This deletes the entry again.

Export
This opens a saving dialog where you can choose a name and path for a "pbt" file. Read below where to find the import option.



If you right-click on the main entry "Patch Templates", the following options are available:

Add Folder
If you like to organize your templates, right-click on the "Patch Templates" folder and choose "Add Folder". Now you can drag and drop other templates to it. If you add a Video Layer template and Output template for example, you can then right-click the folder and choose "Apply to Selected Sites" or "Apply to Site in Patch Tab" to patch all included Video Layers and Output Layers accordingly.

Create Template
Choose a Layer Type from the list that opens and add parameters via the "Add" command as described above. This is the alternative way to create templates, opposite to saving them from Sites that are already patched.

Apply to Selected Sites / Apply to Site in Patch Tab
This applies the included templates to the according Site. Note that each Layer Type should only exist once. If you have several templates, organize them first in folders and call this command from the folder's context menu.

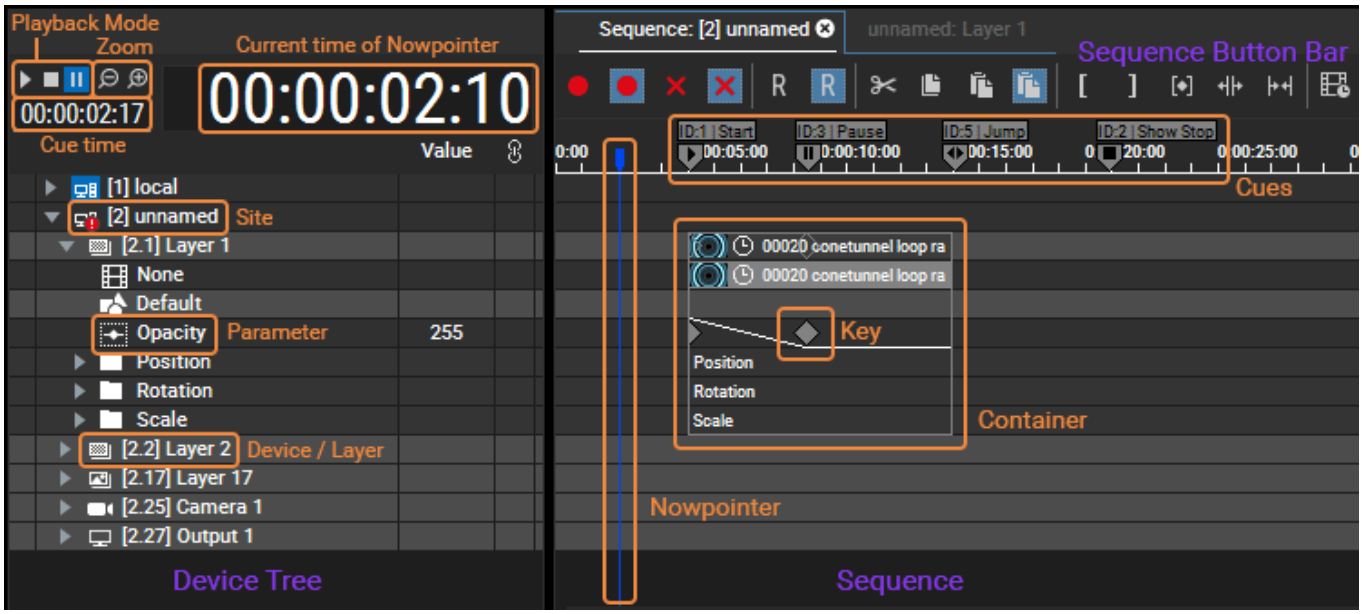
Import
You can import pbt files either to an added folder or the main "Patch Templates" folder. The according "Export" command is available when you right-click a Layer template or folder.

6.3.4.25 Sequence

The Sequence tab displays the timeline for the Sequence that is selected in the "Sequence" folder in the Project tab [278]. It allows to store containers with keys for the Devices seen in the Device Tree [173] and also other timeline elements like cues.

Overview

This illustration shows the most important Sequence items. One of the following chapter explains how to program and work [296] on the timeline.



Zooming

To zoom in/out the Sequence:
- press +/- on your main keyboard or
- click on the zoom tool (magnifying glass icons) on top of the Devices tab or
- hold the [Shift] key and left-click in the time bar (just above the timeline) and move the mouse up and down

Play/Pause/Stop

To start/pause the Sequence, press the space button on your keyboard or use the Sequence control buttons Play, Stop and Pause on top of the Devices tab.

Navigation

- Navigate through the Sequence by dragging the Nowpointer (blue vertical line) on the time bar.
- You can also enter the desired time code on top of the Devices Tab. The time code format is hours:minutes:seconds:frames but you can also just enter type the last numbers, e.g. "409" to get 00:00:04:09. You can also enter "++200" to jump 2 seconds ahead. If you have a numeric keypad, the [+] and [-] key activate the field to enter the time code directly.
- Use [Ctrl] and the [arrow right or left] key to jump from key to key (including the clip borders).
- Use [Ctrl + Alt + right or left arrow] to jump from cue to cue

The chapter Keyboard Shortcuts [314] lists all available shortcuts for Pandoras Box.

The next chapters:
- explain the commands in the Sequence Button Bar and right-click menues [293]
- tell you the most important things you need to know when programming and working [296] on the timeline
- for advanced users - give a closer look how to set up the underlying principle of value tracking (i.e. how default values are enforced [300] or not enforced when no keys are set)

If you select a Sequence in the Project tab (from the "Sequences" folder), you will see the Sequence Inspector, which options are described here [204].
There is also a Sequence Control tab [303], and a Sequence section [142] in the Configuration tab. seq [281]

6.3.4.25.1 Sequence Button Bar and Context Menues

This chapter describes all commands from the Sequence tab. The commands are available in the Button Bar above the timeline, or in the context menus that pop up when you right-click in the time bar or on a container. You will also find the according keyboard shortcut for a command if available. Please see the topic Keyboard shortcuts [314] for a complete list of all shortcuts available in Pandoras Box.

Sequence Button Bar

Above the Sequence tab you will find a button bar, offering the most used commands in an easy and fast accessible way. From left to right:

Store Active
This command stores all active values at the current time.
The keyboard shortcut is [Ctrl + Alt + S] and the command is also available in the context menu from the time bar or Device Tree [178] tab.

Store Active (Selected Devices)
In contrast to the above command, this one filters active values for those Devices (Layers) that are selected. The values are stored at the current time at the blue Nowpointer. All active values influencing other layers will stay active.
The keyboard shortcut is [Ctrl + Alt + D] and the command is also available in the context menu from the time bar or Device Tree [178] tab.

Clear All Active
This command de-activates the active status off all parameters. The values will be kept and not set back to default.
The keyboard shortcut is [Ctrl + Alt + C] and the command is also available in the context menu from the time bar or Device Tree [178] tab.

Clear Active Selected
This command De-activates the active status off all parameters of the selected Device(s). The values will be kept and not set back to default.
The command is also available in the context menu from the Device Tree [178] tab.

Reset All
This command resets all parameters to their default values and removes their active status.
The keyboard shortcut is [Ctrl + Alt + R] and the command is also available in the context menu from the time bar or Device Tree [178] tab.

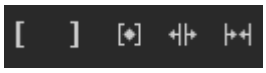
Reset Selected
This command resets all parameters of the selected Device(s) to their default values and removes their active status.
The command is also available in the context menu from the Device Tree [178] tab.

Cut
This command cuts selected keys, containers or cues. To select an item you can either click on it or draw a selection box around it. Hold the [Alt] key for selecting cues.
The keyboard shortcut is [Ctrl + X]

Copy
This command copies keys, containers or cues. To select an item you can either click on it or draw a selection box around it. Hold the [Alt] key for selecting cues.
The keyboard shortcut is [Ctrl + C]

Paste
This command pastes keys, containers or cues to the same Device(s) or current time of the Nowpointer.
The keyboard shortcut is [Ctrl + V]

Paste to Selected Devices
This command pastes keys and containers to the selected Device(s).
The keyboard shortcut is [Ctrl + Shift + V]



Trim Left
This command erases everything between the Nowpointer and the left clip boarder. The last key of a parameter before the Nowpointer will be moved to the new clip boarder time. The video clip will now be shortened at the beginning.
The command is also available in the Container's context menu.

Trim Right
This command erases everything between the Nowpointer and the right clip boarder. There will be a new key set at the right clip boarder containing the value a parameter had at the Nowpointer before.
The command is also available in the Container's context menu.

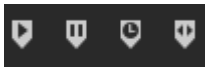
Trim To Clip Borders
If a clip container was scaled down, there could be hidden keys beyond the clip borders. The command Trim To Clip Borders will remove these hidden keys and sets new keys at the left and right clip border to ensure the wanted behavior in the visible part of the clip.
The command is also available in the Container's context menu.

Split Clip
You may split one clip at the Nowpointer's position.
The command is also available in the Container's context menu.

Merge Clips
If you have several containers selected within the same track, you may merge them. If they have different options regarding Lock to time or free-run, media file, mesh file etc., the first container in time overwrites other settings.
The command is also available in the Container's context menu.

Adopt Main Media Duration
This command adopts the duration of the clip if it was shortened/extended or replaced by another clip with different duration before.
The command is also available in the Container's context menu.

Lock to Time
There are two different modes how a container clip is played back, the mode is depicted by a small icon in the container.
Per default, "Lock To Time" is enabled. The clip icon is a clock. In this mode, the clip synchronizes to the timeline. If the Nowpointer is paused inside the container, the clip playback is also paused. If you have programmed video playback keys [649], the Sequence playback overrides them.
Otherwise the container is in "Free-run" mode. The clip icon is a running-stick-figure. In this mode, the clip playback does not depend on the Sequence playback mode. As soon as the Nowpointer enters the container, its video playmode keys (play once, play loop, pause, stop) take effect.
To switch the mode, you can also use the "Lock To Time" option in the Clip Inspector [206].



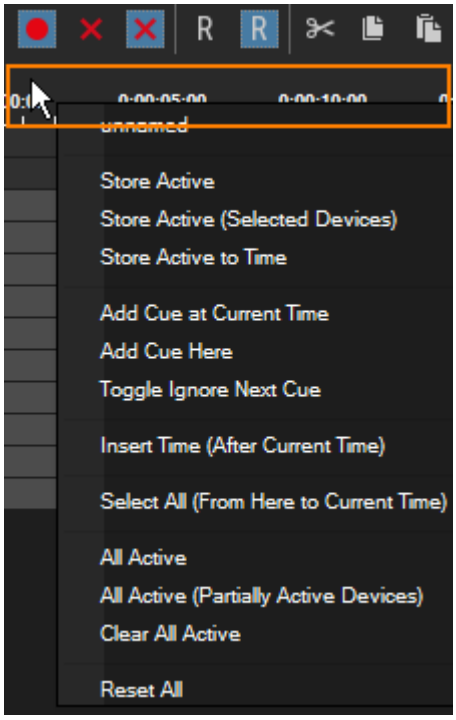
Add Cue at Current Time
These buttons add a Play, Pause, Wait, or Jump Cue at the current time, indicated by the blue Nowpointer. The paragraph "Cues [298]" of the next chapter explains the difference between those Cues. The added Cue is automatically selected and its properties are displayed in the Inspector tab [208].

The command (to add a Play Cue) is also available in the context menu from the time bar.

Zoom out / in
These buttons zoom the Sequence out or in. Alternatively, you can:
- press +/- on your main keyboard or
- click on the zoom tool (magnifying glass icons) on top of the Devices tab or
- hold the [Shift] key and left-click in the time bar (just above the timeline) and move the mouse up and down

Toggle Multi-User Blind Mode
This button is only available for Sub Masters in a Multi-User [234] environment. It activates the "Blind Mode" which detaches the Active Values and Playback status (including the time for the Nowpointer), and allows to use the partly synchronized Master as a Blind Preview system or a Blind Programming system. For more information please refer to the Multi-User topic [234].

Context Menu from Time Bar



When right-clicking on the depicted timeline area (with the Cues and times) a context menu pops up. Most of its commands are already explained above as they are also available in the Button Bar. The following ones can only be found in the context menu.

Store Active to Time
A small dialog opens and asks for the time whereto all active values will be stored. The time can be entered in the format H:MM:SS:FF or shortened to SS:FF or even SFF. So, for example 3 seconds can be 0:00:03:00 or 03:00 or 300. Click Enter or the button [Ok] to close the dialog and save the active values.

Add Cue Here
This command adds a Cue at the position of the mouse pointer (instead of the current time from the Nowpointer). The Cue is automatically selected and its properties are displayed in the Inspector tab [208].

Toggle Ignore Next Cue
Click here to ignore the next (upcoming) Cue in the Sequence or to respect this Cue again if it was set to be ignored. An ignored Cue is displayed in orange.
The keyboard shortcut is [Ctrl + Shift + I]

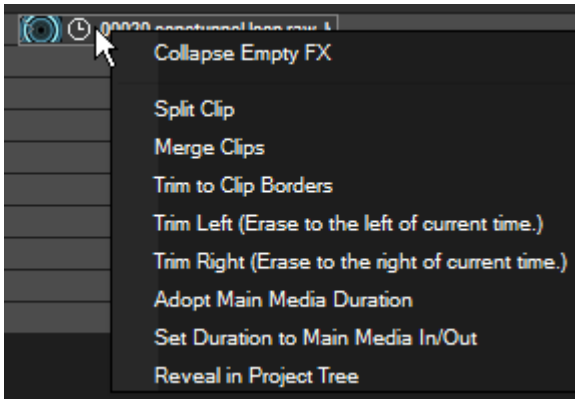
Insert Time (After Current Time)
A small dialog opens and asks for the time interval which will be inserted after the current time indicated by the blue Nowpointer. The time can be entered in the format H:MM:SS:FF or shortened to SS:FF or even SFF. So, for example 3 seconds can be 0:00:03:00 or 03:00 or 300. Click Enter or the button [Ok] to close the dialog and insert the time.
Please note that the time of the "Jump Target" of jump cues will be adjusted automatically if the target time and the jump cue are in front of the current time. In other cases the target time keeps its value.

Select All (From Here To Current Time)
This selects all keys, containers and cues that lay between the time where the mouse pointer clicked and the Nowpointer.

All Active
This command sets all parameters of all Devices to active.
The keyboard shortcut is [Ctrl + Alt + A]

All Active (Partially Active Devices)
This command sets all parameters of partially active devices to active.

Context Menu from Container



When right-clicking on a container that is saved in the timeline already, a context menu pops up. Most of its commands are already explained above as they are also available in the Button Bar. The following ones can only be found in the context menu.

Collapse Empty FX
When there are effects added to the layer without being used, this command collapses expanded sub folders from effects.

Set Duration to Main Media In/Out
This is of interest when you have shortened the duration of the media, e.g. by modifying the In- or Outpoint. The command will set the container length according to the resulting duration between the In- and Outpoint.

Reveal in Project Tree
This selects the file assigned as the Main Media in the Project tab which in return also shows according information in the Inspector.

6.3.4.25.2 Programming on the Timeline

This topic explains how to generate keys and containers in the sequence and how to edit them, i.e. select, move and change them. If you are new to Pandoras Box, please check the introductory chapter "Sequence [292]" for the general explanation how to navigate in the Sequence tab and what is meant when talking about a Site, Container, Key etc.
At the end, there is also an explanation of teh advanced technique that toggles the key mode [299].

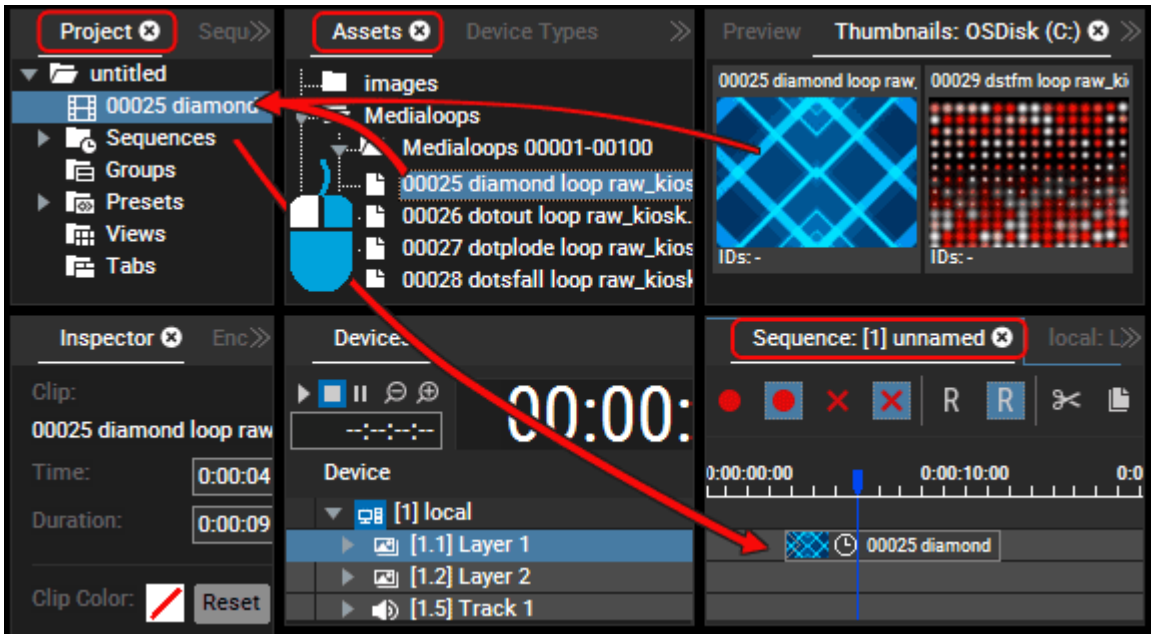
The shortcut [Ctrl + Z] will undo steps of sequence editing, except for Toggle Key mode and active values.

Creating Containers

Before working in the Sequence tab, we first need to have media files available in the Project tab [278]. These are our sources. Please open the Assets tab [131] (or Thumbnails tab [310]) and drag them there. More information can be found in the linked chapters.

Next, we need some Layers in the Device Tree. These are our "outputs" to which the sources can be assigned to. Please follow the instructions from the chapter "Device Tree" [173] as the steps vary between a stand-alone and Master-Client setup and also depend on whether the Clients are started or not. Include the Site to the Preview (right-click menu) and toggle it open (double-click) to see the Layers in the Device Tree and the according tracks in the Sequence.

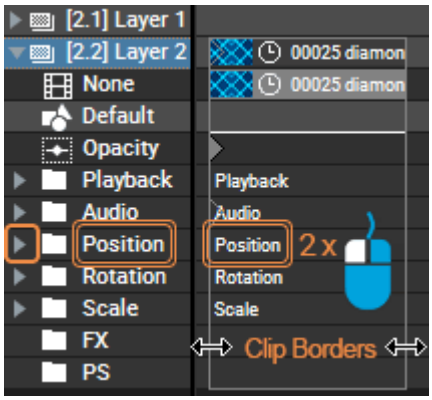
To add containers to your timeline / sequence, simply drag a media file from the Project tab onto a layer track in the Sequence tab. This results in a container that holds the information to load this file (when the Nowpointer hits the container) and to display it with an Opacity value of 255 which is 100%. When having a look in the Preview, you should see that the media file is there whenever the Nowpointer is "in" the container and not there, if the Nowpointer is outside.



As an alternative to the above workflow, you can also work with active values [127]. Select another layer and double-click on a media file in the Project tab. Note that the Layer and Site are highlighted in red, which indicates that there are "unstored" active values. In the Preview you should see the media file right away because active values overwrite the current values from the sequence.

Now, we want to store the active values to the timeline, which will result in a second container where each active value is stored as a parameter key. Position the Nowpointer and click the first button in the Sequence Button Bar above the timeline or use the shortcut [Ctrl + Alt + S]. See the last chapter [293] for other ways to call the "Store active" command(s).

Now, we got two containers created in two different ways but with the same result. As said above, containers include stored layer information. Each parameter change is stored with a so called key at a certain time that holds the parameter value. The key for the "Media" parameter is special because there can only be one (main) media per container. The "Opacity" key is a regular key. Double-click the container to toggle it open and display the included parameter tracks with their keys.

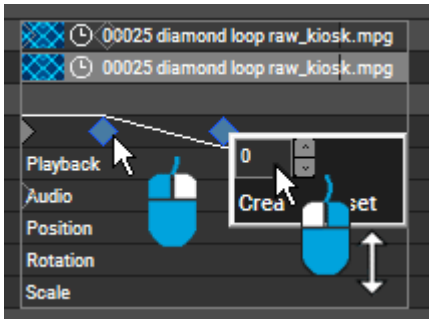


On the level of "Layer 2" you see the clip summary container which also visible when the container is collapsed to one line. Below that, there are the clip container and the keys of different parameters. Most of the parameters are integrated in sub folders, like "Position". To toggle them open, click on the arrow key in front of the folder's name or just double-click on the folder in the Device Tree or the according track in the Sequence tab.

The chapter Site, Layer and Parameter Structure [176] shows how to toggle the visibility of parameters.

Later, we will add some keys and edit them, but first we will have a look how to shift Containers or their Clip borders on the left and right edge.

How to Add, Select and Edit Keys



Following the two workflows we have seen above when creating containers, Keys too, can be added and changed in two ways.

Firstly, simply right-click in the desired parameter track, e.g. next to the existing Opacity key. This creates a new key.
To change its value, right-click it again to access its parameter handle. Left-click into the number field and drag it up or down to change the value. Or, double-click the numbers and enter a new value.
You can also select a key with a left click and change its value in the key inspector tab [209]. Once, the value is changed you will notice that the line that connects two keys adopts to the new value. The line shows the parameter value behavior over time.

The second way to create keys is to work with active values. Open the Device Controls tab [171] and move the Opacity fader. Note that it turns red to indicate the active value state. Now, back in the Sequence tab, place the Nowpointer somewhere between the clip borders and store the value as explained above. If you place the Nowpointer outside, a new Clip would be created but without any media. By the way, a key is overwritten when you store a new active value at its time. For more advanced users: The tab Active Values [127] gives an overview of all current active values. And the Layer Inspector [211] includes the option to exclude a Layer from the "Store Active" command.

Most operators work with both workflows depending on the situation. Storing active values values makes most sense when storing many parameters at once, e.g. when working also with position [651], scale [653], effect [322] parameters etc. Changing a value from an existing key is mostly more comfortable via the first way.

To replace the content of a clip container without active values, drag and drop the new content from the project tab onto the clip. If the new clip has a different length, you may adopt the clip container to the new media duration. Right-click onto the clip summary container and choose "Adopt Main Media" or click the according button in the Sequence Button Bar.

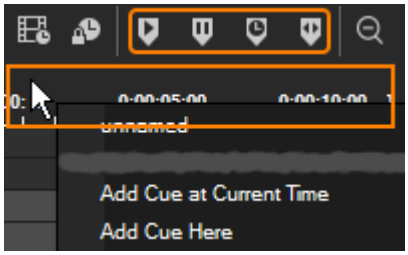
To select a single key, just click on it or drag a window across it. To select multiple keys, drag a window across the region of keys.
Once selected, the keys may be moved (left click and drag selection), copied ([Ctrl + C]) and pasted ([Ctrl + V]) within the same layer or device. To copy a key within the container borders drag it whilst holding [Ctrl].

To copy a selection of keys from one device to another, use [Ctrl + C] to copy and [Ctrl + Shift + V] to paste to a new selection of devices.
Please note: If copying parameters from one device to a different device, only those parameters which are shared by both devices can be copied.

With the shortcut [Ctrl + right or left arrow], the Nowpointer jumps from key to key (including the clip borders).

More information for advanced users: In the image, the line shows, that we created a linear fade out. In case you like to change the type of a key, e.g. to a Bezier key, select it (or multiple ones), look at the key inspector tab [209] and change the "Curve Type". The Curve Editor [169] allows even more fine-tuned value editing and opens when a parameter name is double-clicked in the Device Tree. The linked topic also shows how the different curve types behave.
Last, if you are interest in working with "Key Presets", the chapter "Presets [283]" explains how to do this.

How to Add, Select and Edit Cues



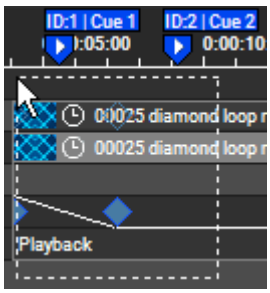
Cues can be used as markers in the timeline or to perform actions like pausing the timeline (for some time) or jumping to another time or cue.

To add a Cue, you can either right-click in the time bar to open the context menu and choose "Add Cue Here" which will add a Play Cue at the time where you clicked. On the other hand, "Add Cue at Current time" adds a Cue to the Nowpointer's time. The Cue is automatically selected and its properties are displayed in the Inspector tab [208] where the type (see below) can be changed if needed.

Alternatively you can place the Nowpointer at a certain time and choose one of the Cue buttons in the Sequence Button Bar [293].

The Play Cue does not change the sequence playback mode, it can be understood as a simple marker. PB can be controlled externally [706], e.g. via DMX or Widget Designer [786], and mostly there are commands that tell the Sequence to jump to a certain Cue number.
The Pause Cue pauses the timeline if the Nowpointer hits it whilst the Stop Cue stops and resets the timeline to 0:00:00.
The Wait Cue pauses the Nowpointer for a certain time, which is displayed above the Device Tree.
The Jump Cue sends the Nowpointer to a certain timecode and can also be programmed with an interval.

Even more information can be found in the Cue Inspector [208].



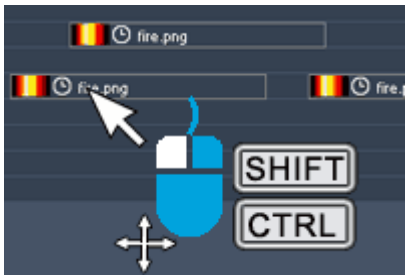
To select a single Cue, simply click on it. To select another one, hold the [Ctrl] or [Shift] key and click it.

To select both, cues and keys hold down the [Alt] key whilst drawing a selection box as seen in the left image.

With the shortcut [Ctrl + Alt + right or left arrow], the Nowpointer jumps from cue to cue.

The tab "Cues [168]" gives an overview of all Cues and allows fast changes.

How to Select and Move Containers



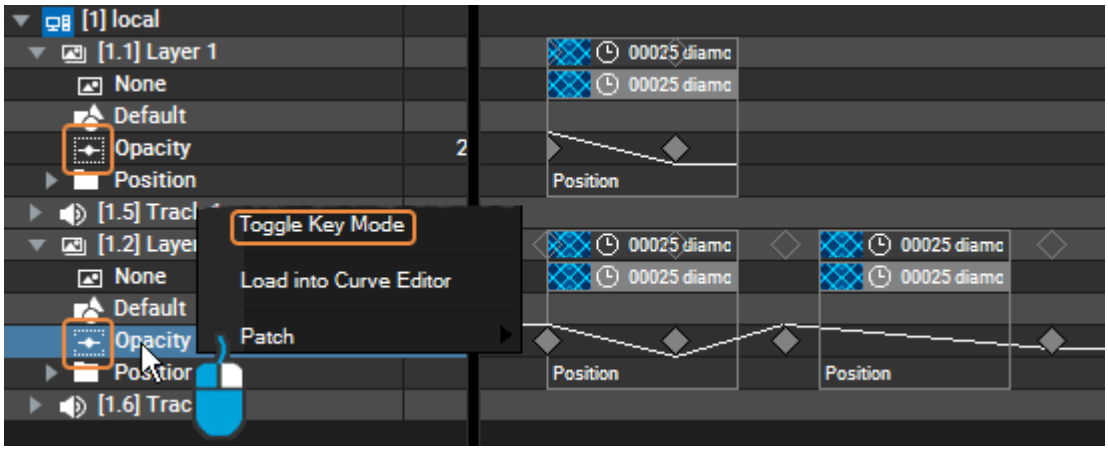
Clip containers can be positioned horizontally as well as vertically by selecting the summary clip container and moving it with the mouse. Moving the mouse left and right will change the container's time. You will notice a small pop-up informing you about the relative time, the container has been shifted.
Since version 6.1.3 it is easier and more intuitive to drag containers and key frames precisely to another element on the timeline. If you drag one or several containers or key frames in the Sequence [292] tab, you will notice that they snap to cues and to the borders of other containers. When moving multiple containers, the container you clicked on to drag the selection, is the only one that will snap to other elements. Key frames can also snap to other key frames within the same container. Hence, you can align containers and keys much faster without positioning the Nowpointer first (which was always "magnetic"). You can snap to the beginning or end of all containers, might they be on other Layers or Sites. You can only snap to elements in Layers that are shown and not hidden.
Whenever a snapping target is active, you will see a white dashed vertical line and arrows which indicate the snapping partners, e.g. the beginning of your dragged container and the end of another container.
Zooming in and out could help finding the required snap target faster. To disable snapping temporarily, hold the [Shift] and [Alt] key whilst dragging. To disable it permanently, go to Configuration tab > User > Sequence [142] and uncheck "Snap Containers".

Moving the mouse up and down results in shifting the container(s) to another layer. The container changes the color to warn if it is another layer type and parameters will be discarded. If you want to make sure that the container stays on the same layer, press [Shift] whilst dragging the container. To copy the container, press [Ctrl] instead of [Shift].

Clip containers can be stretched in time by selecting the clip container borders and moving them with the mouse. Again, a pop-up displays the relative time and you can snap to cues and to the borders of other containers. You can also select the container and change its value in the clip summary inspector tab [206].

Advanced Technique: Toggle Key Mode

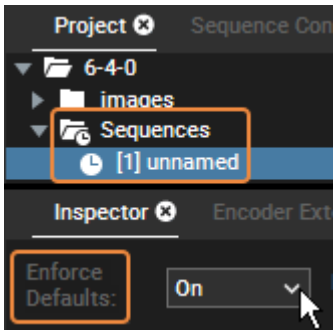
If you want a parameter value to be applied for the whole sequence or many Containers at once and not to be set up for each single clip, you can take it out of the "clip orientated" key mode. Simply right-click on the parameter in the Device Tree and choose "Toggle Key Mode". This command cannot be undone! Confirm with "OK" and the clip boarders are no longer effective for this parameter.



You can toggle the parameter back to the clip orientated key mode at any time. In that case Pandoras Box will add new keys at any clip border.

The next chapter explains the advanced technique that changes how default parameter values for all parameters in a Sequence are defined.

6.3.4.25.3 Enforcing Default Values



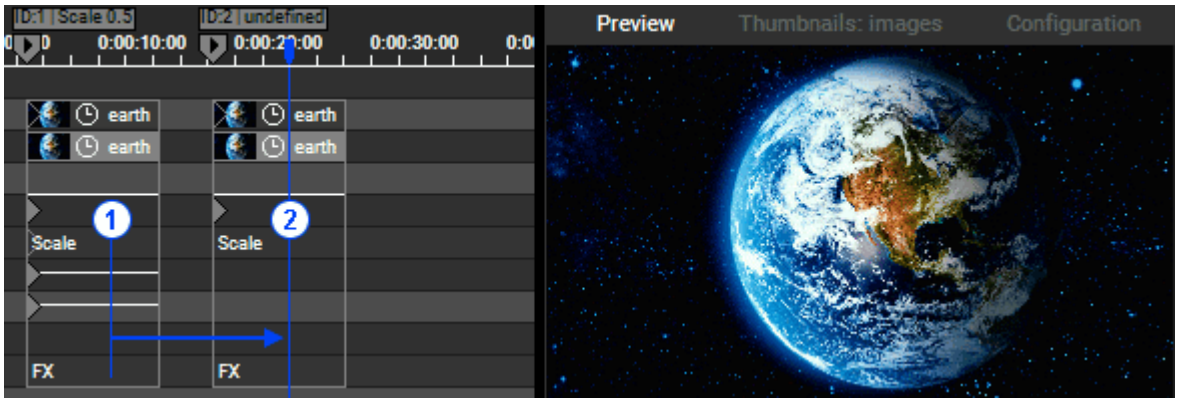
This chapter is for advanced users and gives a closer look how to set up the underlying principle of value tracking [300], i.e. how default values are enforced or not enforced when no keys are set in a parameter track on the timeline.

The "Enforce Defaults" option can be found in the Sequence Inspector [204]. To see it, open the "Sequences" folder in the Project tab first and select a sequence there. You can choose between three states: On, Off, Only Mesh/Media.

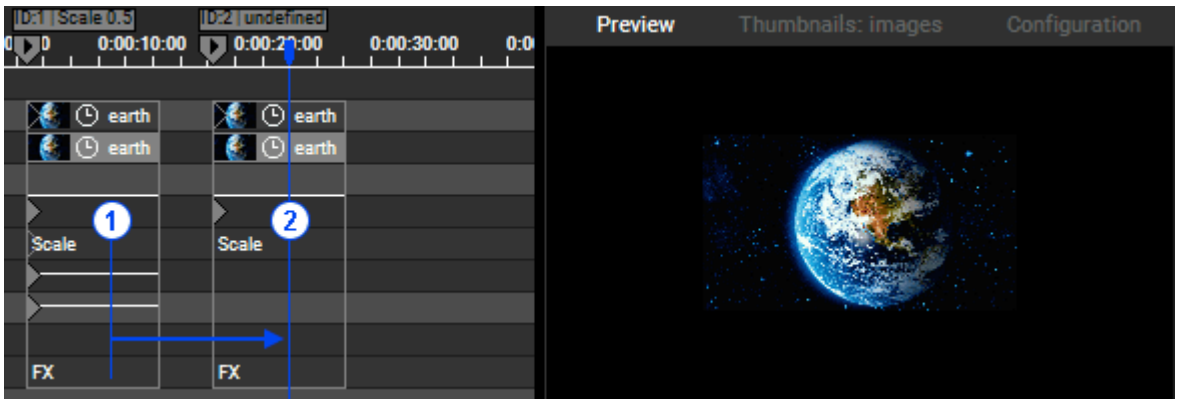
To see where a change of this option is effective, we must first understand the default behavior of Pandoras Box' timeline.
In Pandora's Box timeline programming you normally determine the value of a parameter by setting a key. In some situations, however, a certain parameter may not have a key associated with it that explicitly determines its value at a certain point in time. There is for example the time span inside a clip when no key is set or when the first key is set later in time.

Per default, the "Enforce Defaults" option in the Sequence Inspector is set to "On" which means that Pandoras Box refers back to the default values in those situations. The default value for "Position" is "0px" and for "Scale" it is "1.0", for example. In other words, whenever a parameter is not defined per key, it is defined by the default value. This behavior ensures that each frame in the timeline is associated with exactly one definite state for all parameters, even though not all of them may have explicitly set values. So if you click into the timeline at a certain timecode, a frame is rendered. If you then jump to other places in the timeline and return back to this certain frame, the output will always be exactly the same as before.

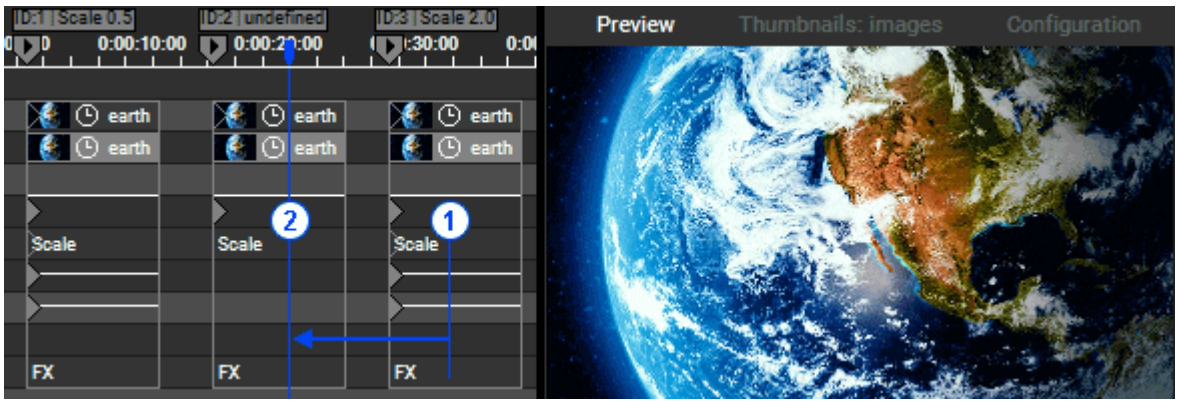
The following example visualizes this behavior. At the beginning of the first clip, the Scale keys set the parameters to "0.5". The second clip has no Scale keys. With "Enforce Default = On", the Scale parameter is reset to "1.0" whenever the Nowpointer is in that clip.



If the "Enforce Defaults" option is turned "Off" the system does not apply default values any more in those situations. If keys define a parameter, they of course still count. But all undefined parameters retain to the value that was set before the situation was encountered, hence they stay what they where and do not change to the default values.
Following with the example: if the Nowpointer is set into the first clip and then into the second one, the output now looks different as the Scale parameters are NOT reset to "1.0" but retain "0.5".

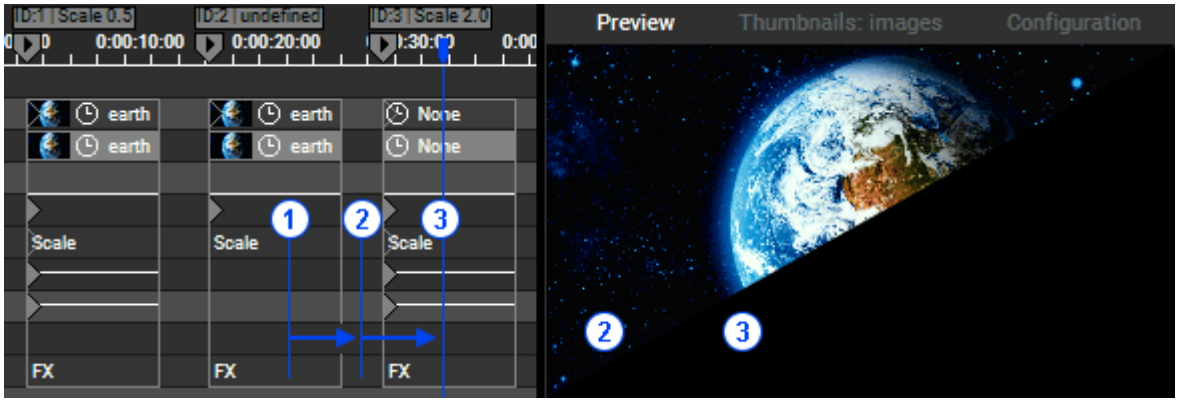


This shows, that a frame in a timeline is not "static" any more but can be rendered differently. It depends on the way how the Nowpointer got there, on the "path" it took to reach a particular timeline frame. Let's extend our example to visualize this "path dependency". A third clip is added and holds Scale keys with the value "2". If you now set the Nowpointer into the new clip and then back to the clip with undefined Scale parameters, the output looks different to before because the Scale parameters now retain "2.0".



The default setting "Enforce Defaults = On" which means path independence is usually a good idea, especially when working with a single timeline. It makes sure that a frame in the timeline is rendered exactly in the same way, no matter in which order the timeline was played.
For some cases however, path dependence is a solution that let's you solve your task in an efficient way. Especially when working with multiple Sequences, the setting "Enforce defaults = Off" makes a lot of sense as it allows to program parameters of one Layer on multiple timelines. You could use one Sequence for defining the Media keys and another for defining other parameters like Position or Scale. In this and similar scenarios, it is necessary for each timeline to only apply those values that are explicitly set in it via keys and nothing else.

A disadvantage of the setting "Enforce defaults = Off" is that unloading a media file from a Layer becomes more complicated. As we have seen, a parameter value does not reset when it is undefined, it rather stays what it was. In case you like to clear the media file from the "Media" or "Mesh" parameter when the Nowpointer leaves the Container this behavior is not helpful. Turning the opacity off is not a good workaround, because the media files are still loaded on the Layer and hence, consume resources which becomes important when working with high resolution files and / or close to the performance limit. A better solution is to set a key which clears the media parameter as is loads explicitly "nothing". This is called a "None" key.
Let's create one, in our example from above. Click on the media key from the third Container and hit the "Reset" button in the Inspector, or select the Container and hit "Reset" in the section "Main Media Key". If you like to create one from scratch, first reset all layer parameters (e.g. via the Reset button(s) in the Sequence button bar [293]) and than make a right-click in the Media or Mesh parameter track. The example shows that the media file still rendered when the Nowpointer stands between two clips but cleared when the Media key "None" is evaluated.



Now this is when the setting "Enforce defaults = Only Mesh/Media" becomes interesting which combines aspects of the other two Enforce Defaults modes. In undefined situations, it enforces only the Media/Mesh default values (just like the "On" mode) but no other parameter (just like the "Off" mode).

Think of the example with multiple Sequences controlling parameters of the same Layer. It is a good idea to say "Enforce defaults = Only Mesh/Media" for that Sequence that define the Mesh / Media Parameters as each Container loads files and the empty space between Containers clears them. Other Sequences that control other parameters would be set to "Enforce defaults = Off.

If you play around with multiple Sequence and the different Enforce Defaults modes, it might become confusing when media files are cleared and when not. So as a summary, this table shows what command is send from a Sequence when its "Enforce Defaults" option is set to On, Off or Only Mesh/Media. For the Nowpointer, it depends whether it plays inside a Container (with a certain media key, with an explicit "None" key or without a key at all) or whether it plays outside a Container whereas it makes a difference whether there is a Container before or after on that track or whether it is an entirely empty track which is absolutely undefined.

	Media key	"None" key	Empty Clip	Outside of Clip	Empty track
On	media	clear	clear	clear	nothing
Off	media	clear	nothing	nothing	nothing
M/M	media	clear	nothing	clear	nothing

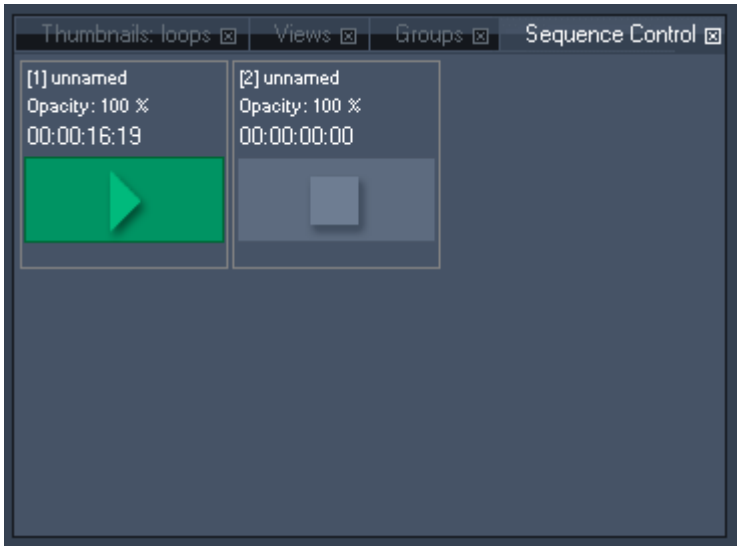
6.3.4.26 Sequence Control

Use the sequence control tab to monitor the status of available sequences.

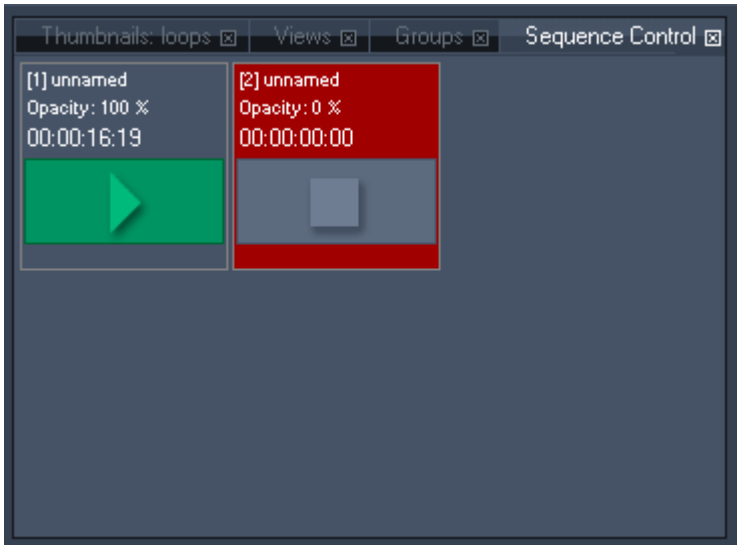
All created sequences in the project's sequence folder are displayed in the sequence control tab.

The sequence tapper shows the following information:

- ID and name of the sequence
- the percentage value of the sequence opacity
- timecode of the nowpointer's position
- current play mode (play, pause or stop)

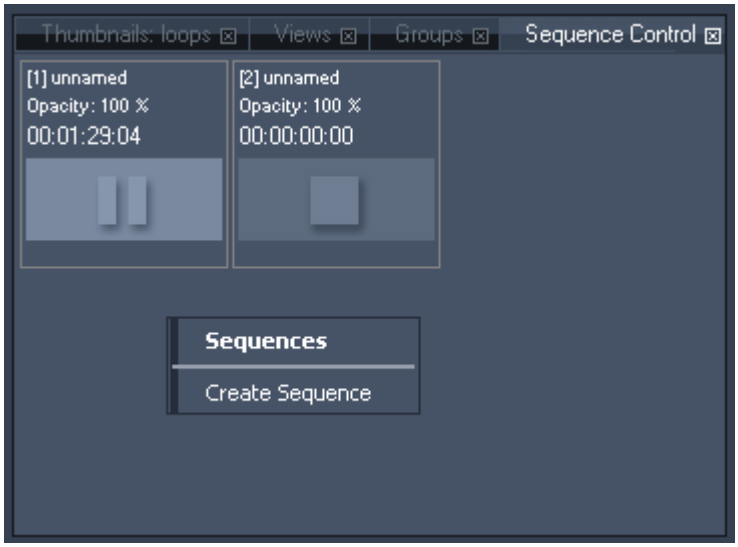


If the sequences opacity is set to zero, the sequence tapper background will turn red:

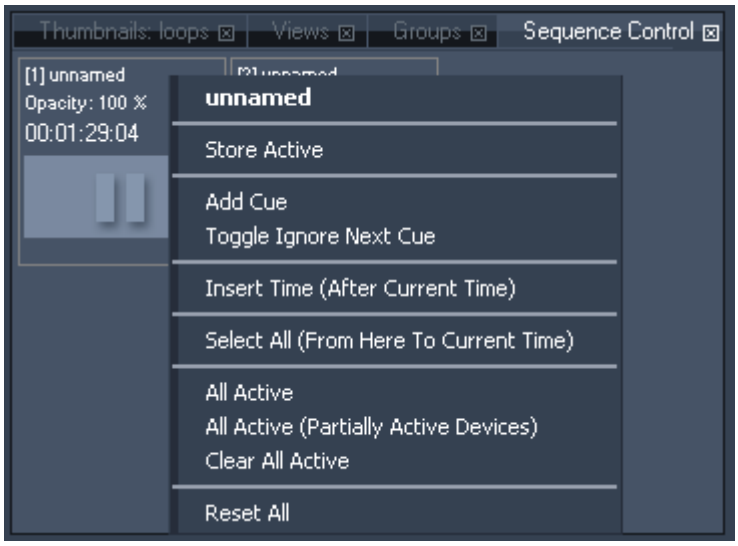


By clicking on a sequence item in the sequence control tab you may set the selected sequence to Play or Pause.

To create a new sequence you may right-click in any empty region of the sequence control tab. The newly created sequence will automatically be loaded into the Timeline Tab.



Once a sequence is shown in the sequence control tab, you may right-click on the sequence item at any time and choose one of the following available commands:



- Store Active
This command stores all active values at the current time.

- Add Cue
Adds a Cue at the position of the mouse pointer. See the Cue Inspector [208] for further information.
Please note: it is recommended to have a spacing of at least 10 frames in between two cues.

- Toggle Ignore Next Cue
Click here to Ignore the Next Cue in the timeline (or use the shortcut [Ctrl + Shift + I]) or to respect this cue again if it was set to be ignored.

- Insert Time (After Current Time)
Enter the amount of seconds that will be inserted after current time.

- Select All (From Here To Current Time)
All keys, containers and cues from the mouse pointer up to current time will be selected.

- All Active
This command sets all parameters of all devices to active.

- All Active (Partially Active Devices)
This command sets all parameters of partially active devices to active.

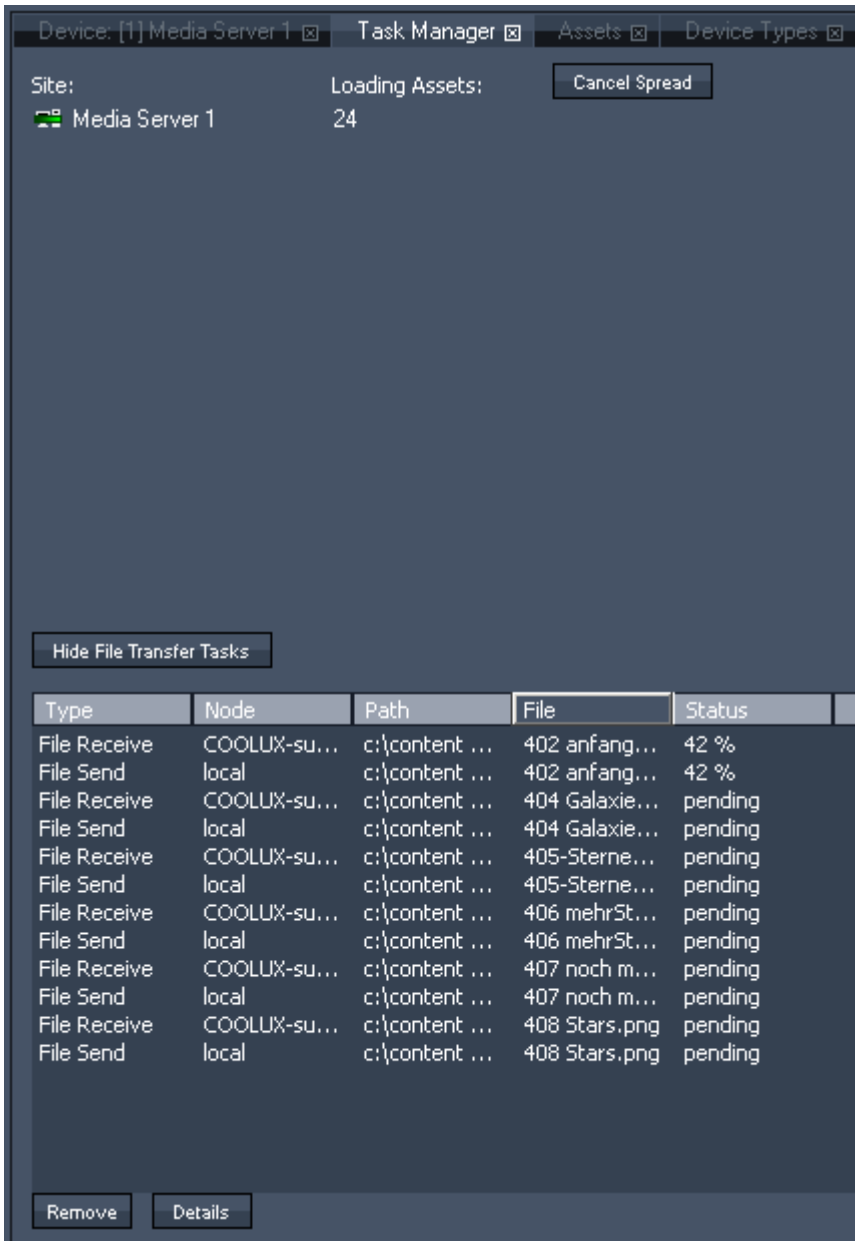
- Clear All Active
This command clears all active parameters.

- Reset All
This command resets all active values.

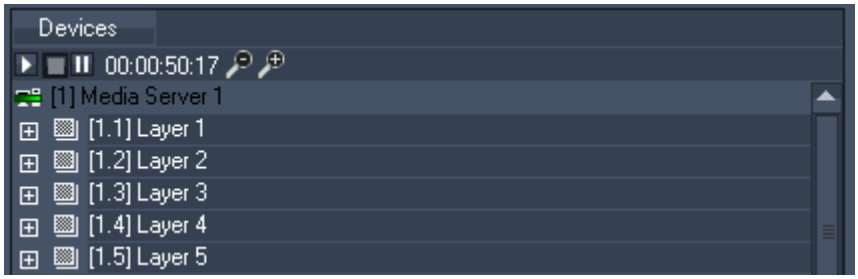
6.3.4.27 Task Manager

To open the taskmanager tab, navigate to Tabs - Taskmanager or press [Ctrl + Shift + M] on your keyboard. The taskmanager lets you monitor if a site is still loading files and if all file send and receive actions that were initiated by a file or folder spread process.

If you click on Show File Transfer Tasks, you will get a list with the status of all file transfers.



The site icon with a green stripe indicates that the site is still active receiving or loading assets.
The same icon will also be shown in the Devices Tab.



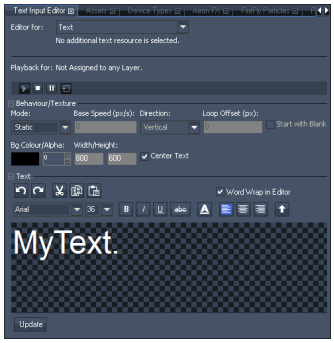
The number of assets to load is shown next to the site name in the taskmanager. As soon as the loading process is done "complete" will be displayed and the site icon turns back to the normal one.

Clicking on [Cancel Spread] will cancel the spreading process.

If you encounter any network problems or issues of spreading files, individual items may be selected from the list and removed to continue with other spread operations.

If you encounter issues that you cannot solve yourself by removing and/or restarting the project, please collect the "details" description of the task that is causing the issue and report it to the support team.

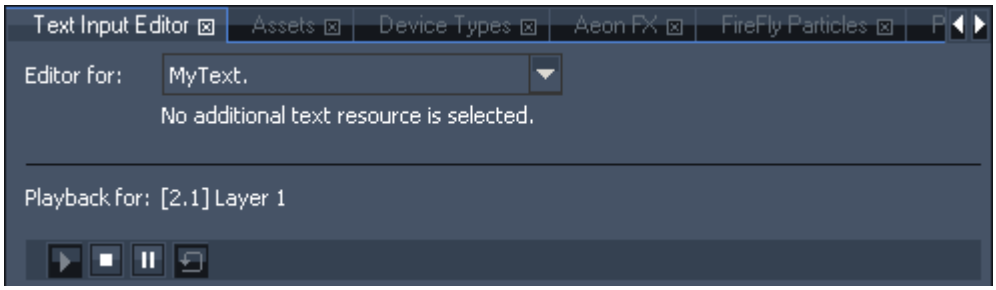
6.3.4.28 Text Input Editor



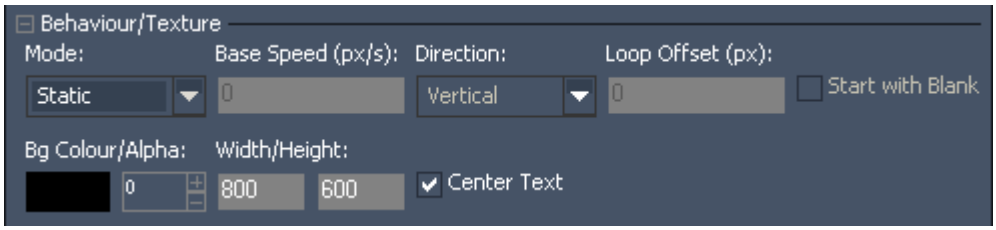
Create your Text Input by right-clicking on a folder in the Project tab [278] > Add Text Input. Select it to see its properties in the Text Inspector [203]. It also offers the button "Open in Resource Editor" which opens the tab "Text Input Editor". Alternatively, you can click on 'Tabs' in the Toolbar > Text Input Editor.

In the Text Input Editor you may set up all options regarding the content of the selected Text Input. It also allows to transfer style properties to other Text Inputs.

Please note, that .net framework 4.0 is needed to display this tab.



At the top of the depicted Editor you see in the drop-down list that only one Resource named "MyText." is selected. If multiple Text Inputs were selected, the section "Behavior / Texture" applies to all of them.
Afterwards, it is displayed, that the according Text Input is assigned to the first Video Layer of the second Site and that the Playback command [649] (from the Layer) is set to Play Loop. The playback is needed when the Text Mode is set to Scrolling or Streaming and the container is changed from Lock To Time [206] to free run mode. If the container is locked to time, and the sequence is playing, the text will "play", i.e. move also. Mind that playing back is only possible on Video Layers. If you like to use scrolling text on another Layer, please use the Aeon FX "UV Scroll [631]".



- Mode
--Static
Your text is displayed on a fixed position according to the Layer's Position parameters.
--Scrolling and Streaming
Your text will move according to the following parameters.
For the Scrolling Mode, you enter text, click "Update" and instantly the entire text will be displayed as scrolling text. Depending on the playback mode, the text will scroll once or endlessly. The old text will be kept in the Editor window. If you like to show new text, enter it and press "Update". In case, your text is scrolling endlessly (Play Loop) the old text will change abruptly to the new one.
For the Streaming Mode, you enter text, click "Commit" and after a couple of seconds, your entire text will scroll once through the window. Note, that it cannot scroll endlessly. The old text will be deleted from the Editor window. As soon as you enter new text and click "Commit", it stands in a pending line. Only when the old text has moved out of the window, the new one can be displayed. This means on the one hand that you can commit unlimited text entries and PB takes care, that one follows the other. This is done by rendering multiple textures at the same time. Which also means on the other hand, that textures can be empty if now pending text is available. This explains why you have to wait for the duration of one texture width / height before recently committed text becomes visible. If you like to clear pending text click the "Pending" button underneath the Editor. If you like to clear pending text and the one currently displayed, click "All".

Please note that Widget Designer can send text to a Pandoras Box Text Input regardless from its mode. You can use the Text Input control [926] or the PB Text Output node [1225].

- Base Speed
Enter the speed for Scrolling and Streaming text. It is measured in pixel per second. When entering negative numbers the direction will be reversed.

- Direction
Choose how Scrolling and Streaming text should move: "Horizontal" for right-left movement and "Vertical" for down-up. Note that the direction can be reversed with a negative Base Speed.

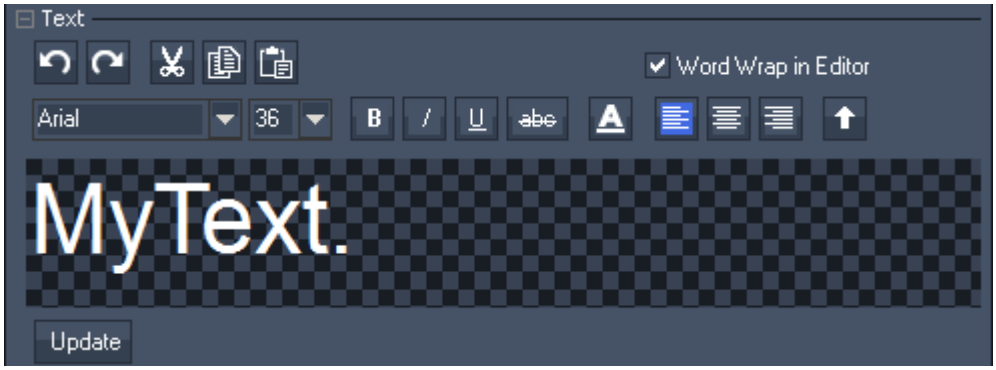
Loop Offset
Influence whether Scrolling and Streaming text should have an offset, i.e. blank space before the start of a new text entry.

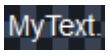
Start with Blank
Decide whether Scrolling text should start with blank space before its first letter. Use the option "Offset

BG Color / Alpha
Choose a background color (for all text modes) and its opacity (0 = transparent)

Width / Height
Set up how large the texture for a text entry should be. Note that this influences the Base Speed and Font Size. The larger the Font Size the smoother font edges look and the larger the texture needs to be.

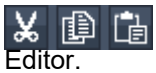
Center Text
This centers Static Text vertically.





Enter your text into the Editor. If you have chosen a Static or Scrolling text, click "Update" to display your changes. In case you are working with a Streaming text, click "Commit". See above explanation for further detail.


Undo your last steps or Redo undone ones.

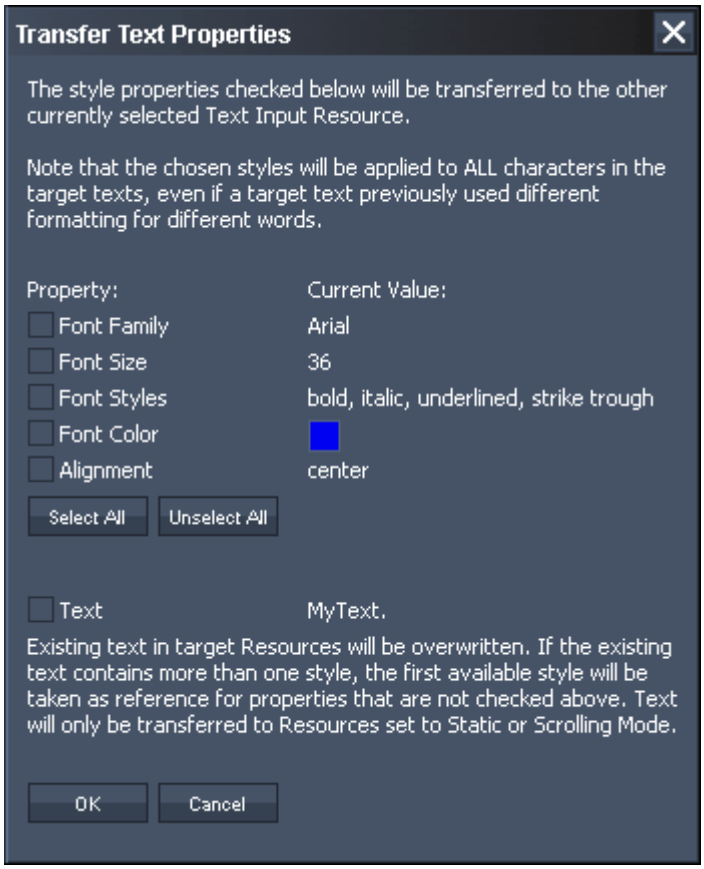

Cut or Copy selected text and Insert it with the last button. The shortcut [Ctrl + A] selects all text in the Editor.

"Word Wrap in Editor" creates a new line if your text is longer than one line in the Editor.

Choose a Font and the Font Size for selected text. Note that the Font Size also depends on the texture width/height.

Choose a Font Style for selected text: Bold, Italic, Underlined, Strike through, Font Color (opens a Color dialog)


Align Left, Align Center and Align Right are used to position your text in regards to the texture width.



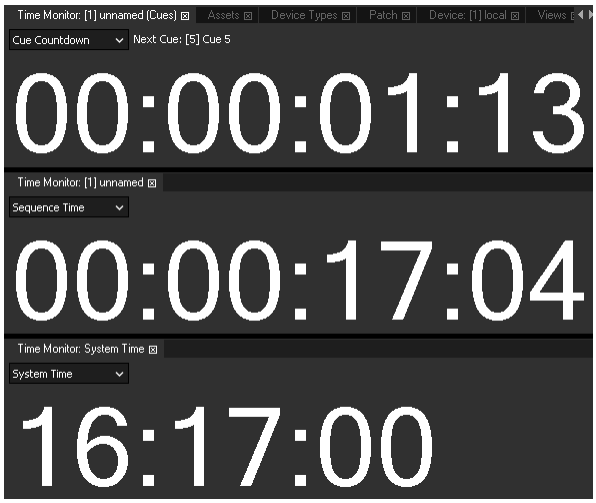
The last button  opens the "Transfer Text Properties dialog".

In case you have selected multiple Text Inputs in the Project tab, you can choose with the drop-down list (at the very top) which text is the "master" or "source". Per default, all "Text" options apply only to the master text. With this dialog however, you can choose to transfer the current options to all other selected Text Inputs too. In case you have set up more than one style in your master text, only the first one can be transferred.

Select which property you like to copy and press "Ok".

If you like to create multiple copies of your current Text Input, please right-click on it in the Project tab and choose "Clone".

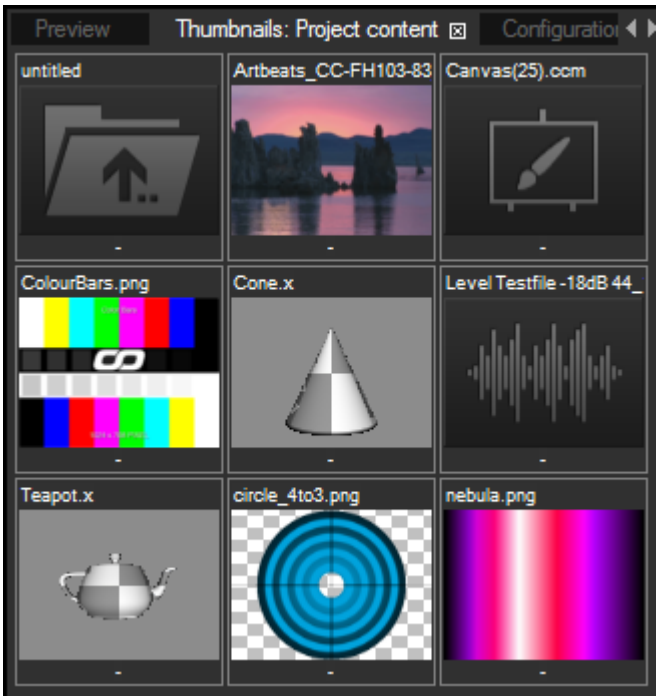
6.3.4.29 Time Monitor



The Time Monitor tab in Pandoras Box displays the Sequence Time, the System Time or the Countdown for the next cue. The tab is part of the View [287] called "Show". In general, to open it, please click on "Tabs" in the Toolbar and choose "Time Monitor".

To choose a time, select an entry from the drop-down list. You can also open three Time Monitors and display all time editions as displayed in the left image. Of course you can adjust the size of the new tab, or as known from previous versions, break it out [313] the main user interface. The new layout can be saved to a view [287].

6.3.4.30 Thumbnails



Whenever you click on a folder in the Project or Assets tab, its content is displayed as thumbnails in the Thumbnails tab. In other words, the Thumbnails tab is context sensitive as its changing any time when a folder is selected.

All content types are displayed: audio, image, video, mesh files etc. A video thumbnail displays a frame from the middle of the video, not the first frame as many videos start in black. 3D media is displayed with a default texture to identify the UV mapping orientation. Audio and other files are displayed with an icon.

Pandoras Box version 6.4.0 introduced a new File Browser tab [185]. Right-click on a folder in the Project [278] tab or in the Assets [131] tab, and you will see the option to open this folder in a new File Browser [185]. It opens in a new tab and shows the content from the respective folder as thumbnails. In difference to the Thumbnails tab, this file overview is more static as it keeps displaying the same content. Another difference is that you may open as many File Browsers as you need.

Files from a Assets tab folder that are displayed in the Thumbnails tab or File Browser can be dragged into the Project tab. After adding files to the project you can assign them to a Layer as described below.
Files from a Project tab folder that are displayed in the Thumbnails tab or File Browser can be dragged into the Sequence [292] tab. You may also double-click a thumbnail in order to assign the file to the currently selected Layer. Please be aware that all thumbnail images are stored locally. If the media is not present on all Clients you may have to spread the media to all Clients in order to assign it to the selected devices. This is described in the Assets chapter [131] in more detail.

6.3.4.31 Views

The Views tab lets you recall stored view layouts of the user interface with a single mouse click. Nowadays, using the View Bar [119] is more convenient.

The Views tab shows all entries that are also shown in the folder "Views" [287] in the Project Tree. The right-click menu simply allows, to remove existing Views.

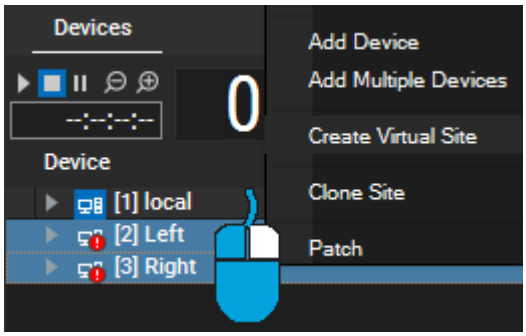
6.3.4.32 Virtual Site

This topic explains the feature "Virtual Site" which allows to combine multiple Sites (i.e. Pandoras Box Clients) in terms of merging them to one device called Virtual Site.

In the Device Tree, a Virtual Site, offers you so called Virtual Layers. If you merged for example three PB Clients, the Virtual Layer 1 will control the Video Layer 1 of three Clients. So instead of programming or copying three Containers, you only do that once. At the same time, you can program on individual Layers, which is described below.

Whilst the "Groups" feature [281], which allows to simultaneously control multiple Layers via active values, is a good solution for a short time, the Virtual Site has more advantages when looking for a permanent solution. It is often used when two or more Sites play back the same content in the same way most of the time.

Creating and Programming a Virtual Site

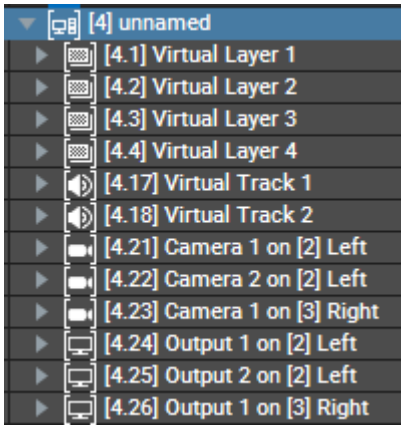


First, add your Pandoras Box systems to the Device Tree, either from the Device Types tab [183] or from Assets [131] in case they are connected already.

Now, select the Sites that should be combined, right-click one and choose "Create Virtual Site".

Please note: If the Sites that you like to add to a Virtual Site contain Sequence information (i.e. Containers), it will be deleted in the course of the Virtual Site creation and this action cannot be undone! Active values will also be reset. That is the reason why it is recommended to create a Virtual Site before starting any other programming. If you however merge later but still like to keep (some of) your Containers, simply copy them first to a Site that you add to the project temporarily and from there to the Virtual Site.

Layers of a Virtual Site



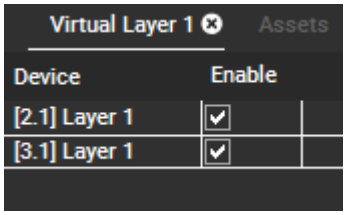
As you see in the example, the Sites with ID 2 and 3 do not appear in the Device Tree any more because they were merged to the Virtual Site which has ID 4. All so called Virtual Layers control Video Layers from both Sites, same applies for Audio Tracks and other layer types except Camera and Output Layers which are kept as "single" layers as warping, softedging etc. is always done individually. If you renamed those Layers before, the names are forwarded to the Virtual Site.

If you now select Virtual Layer 1 for example and change a parameter, the Video Layer 1 of all included Sites will show that change. Of course this includes active values and Sequence programming likewise. If you like to assign content, make sure, that all Clients have the content, which means that it was spread everywhere via automatic Auto-Spreading [155] or manually if you deactivated the feature.

If you add a layer to a Virtual Site it is automatically added for all Sites combined in it. Simply right-click the Virtual Site in the Device Tree and choose a layer type. All other options from the context menu are explained in the chapter Context Menus - Device Tree [178].

The next paragraph explains how to control individual layers.

The Virtual Site Tab



Please select the Virtual Site in the Device Tree and have a look in the Inspector. Click the button "Open Virtual Site Tab" or open it alternatively via the Tabs menu [122]. Remember that you can save custom views [287] of the interface.

The Virtual Site tab shows you the names and IDs of all merged Sites. If you select a Site there, you can see the all related settings in the Inspector [210] which is not only of interest for entering an IP to connect to.

Now, select a Virtual Layer to see the according information in the Virtual Site tab. If you like to remote control layers, in most cases you need the IDs of the underlying layers.

The "Enable" check box is of high interest. If you disable a layer, the Virtual Layer would not forward parameter changes to it anymore. That way you can program on individual Sites even though they are part of a Virtual Site. Depending on your show you might need layers which control all combined Sites but in certain scenes for example it makes more sense to program them individually. It is a good idea to rename layers in the Device Tree, sort them or use the feature to tint [211] them in order to keep track which layers are are controlled.

Dissolving a Virtual Site

If you like to have separated Clients again, simply right-click the Virtual Site in the Device Tree and choose "Dissolve Virtual Site". The Device Tree now shows all previously merged Sites. Again, please bare in mind, that active values or information stored to the timeline are reset or deleted. As described above, you can save Containers by copying them first to a temporarily added Site.

6.3.5 Layout

The user interface layout can be customized and stored in "Views [287]" to be recalled at any time during operation. The "Layout" part of a view is determined by the layout of the panes and the tabs they contain. The default view (Essentials) for example consists of six panes, the top left pane displays the Project tab.

The user-interface can be freely arranged by adding or removing panes and by resizing them.

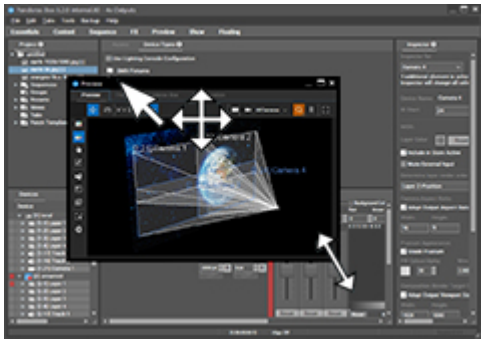
If you right-click *between* two existing panes as seen in the image to the left, or if you click between a pane and the main frame's border the context menu offers you to insert a new pane. The command "Insert Pane ..." completes either with "Above, Below, To Right or To Left" according to the position where you clicked.

A new placeholder is inserted in the main frame, as a next step you may drag an existing tab into it. Click on the title of a tab and drag it into the title of the new pane.

To adjust the pane size, left-click between two panes and drag the mouse.

If you right-click into the title of an existing pane, you will see additional options to split the pane. If you choose to split it horizontally the existing pane will be divided into two panes, one upon the other. A vertical cut will give you two panes side by side. You may choose where you like to keep the existing tab.

The command "Break Out Pane" will release the pane from the fixed structure - it will break the pane out of the main frame. Now you can move and scale the new window independently. Note that the window will always be on-top of the GUI window.



A broken out pane offers the same options as an integrated pane: you may drag tabs into it, insert new panes or split them horizontally and vertically. You may as well remove it completely. When deciding to keep the tabs they will be moved to the next available pane.

To re-integrate a broken out pane, make a right-click into its title and choose the option "Return To Main Frame Next To Tab". Two new sub dialogs let you choose whereto position it exactly. For example you may decide to insert the broken out Inspector left to the Device Tree, it's default position.

Please note:
The Device Tree and Sequence tab can not be moved or be broken out; these two tabs are always linked to each other.

There are seven default views displayed in the View Bar at the top of the interface. Simply click on them to assign a new view. You can also save new views and add them to the View Bar (see "View Tab [310]").

The "Tabs [288]" menu from the main menu bar shows all available tabs.

6.3.6 Pandoras Box Keyboard Shortcuts

General	F1	Launch help
	F2	Rename (e.g. in tabs: Project or Device Tree)
	Ctrl + Z	Undo last step
	Ctrl + Shift + Z	Redo undone step
	Ctrl + left mouse click	Add another item to the current selection (e.g. in tabs: Project, Asset, Device Tree, Sequence)
	Shift + left mouse click	Add another item and all items in between to the current selection
	Ctrl + Pause	Lock / unlock GUI (keyboard, mouse, Jog/Shuttle)
File Menu	Ctrl + N	New project
	Ctrl + O	Open project
	Ctrl + S	Save project
	Ctrl + Shift + S	Save project as
	Ctrl + Alt + Shift + S	Save project copy
	Ctrl + Shift + B	Bundle project
	Ctrl + W	Close project
	Ctrl + Q or Alt + F4	Exit application
Assets, Project, Playlist and Video Export	F5 or Ctrl + Shift + A	Refresh file tree in Assets tab
	Shift + S	Spread all resources
	Ctrl + Alt + Y	Toggle sync of selected Live Inputs in Project tab
	Ctrl + Alt + K	Toggle 4K mode of selected Live Inputs in Project tab
	Ctrl + A	Select all entries in a Playlist [239]
	Ctrl + Q	Stop the current Video Export [135]
Preview	Scroll	Zoom (changes parameter "Viewpoint Z Pos")
	Drag (middle mouse button / wheel)	Pan (changes parameters "Viewpoint" and "Target" X- and Y-axis)
	Alt + Drag	Rotate (changes parameters "Viewpoint" X, Y and Z Pos)
	Alt + Shift + Drag	only in Camera Mode [251], changes "Target" with constraint
	Shift + Drag	only in Cam. Mode [251], changes "Viewport"+"Target" with constraint
	Ctrl + 0	Apply default view to Camera or Output when in Zoom mode [248]
	Ctrl + F	Toggle local Fullscreen
	Esc	Leave Fullscreen (when enabled in Configuration > Render Engine [162])
	Ctrl + Shift + F	Toggle maximized Preview
	T	Show / Hide Toolbar in maximzed Preview
	Ctrl + P	Open dialog Parameter Floater [250]
	1,2,3	Apply Move, Scale, Rotation mode
	X,Y,Z	Set axis constraint for X-, Y- and Z-axis
	M, F, V	Select Sub Mesh, FFD or Vertex
	← / → / ↑ / ↓ / left mouse click	Select single Sub Mesh, FFD or Vertex point
	Ctrl + ← / → / ↑ / ↓ or left mouse click	Add another control point to the current selection
	(Alt +) Shift + ← / → / ↑ / ↓	Warp Mesh according to selected Sub Mesh, FFD or Vertex points ([Alt] key for 10 steps at once)
	Alt + ← / → / ↑ / ↓	Move current selection

Device Tree, Sequence, Cue List and Curve Editor	Space	Play / Pause
	Esc	Clear current Device / Layer selection
	Left mouse drag in the timeline (+ Alt)	Select keys and containers in the Sequence (and cues in the time bar)
	Ctrl + X	Cuts keys, containers and cues
	Ctrl + C	Copy keys, containers and cues
	Ctrl + V	Paste keys, containers and cues to the same Device/ current time
	Ctrl + Shift + V	Paste keys and containers to the selected Device
	⊞ / ⊟ on the main keyboard; Shift + left mouse drag up/down in the time bar (above the timeline)	Zoom the Sequence time
	Shift + ⊞ / ⊟	Zoom the parameter value in the Curve Editor tab [169]
	← / →	Set the Nowpointer to the previous / next frame
	Ctrl + ← / →	Jump to previous / next key (including clip borders)
	Ctrl + Alt + ← / →	Jump to previous / next cue
	Shift + ← / →	Move selected keys and containers to previous / next frame
	Ctrl + left mouse drag	Drags a copie of selected keys, containers and cues
	Shift + left mouse drag	Drags selected container(s) without changing Layers
	Alt + left mouse drag	Drags selected container(s) without changing the time
	Alt + Shift + left mouse drag	Drags selected container(s) without snapping to other elements
	Ctrl + Shift + I	Toggle "Ignore Next Cue" for current Sequence
	Ctrl + Alt + A	Mark all values as (red) active values
	Ctrl + Alt + C	Clear all active values but do not reset them
	Ctrl + Alt + R	Reset all values
	Ctrl + Alt + S	Store active values as keys to the timeline
	Ctrl + Alt + D	Store active values for selected Device(s) only
	Ctrl + U	Cue List tab: Show selected Cue in the Inspector and Sequence
	Ctrl + J	Cue List tab: Jump with Nowpointer to selected cue

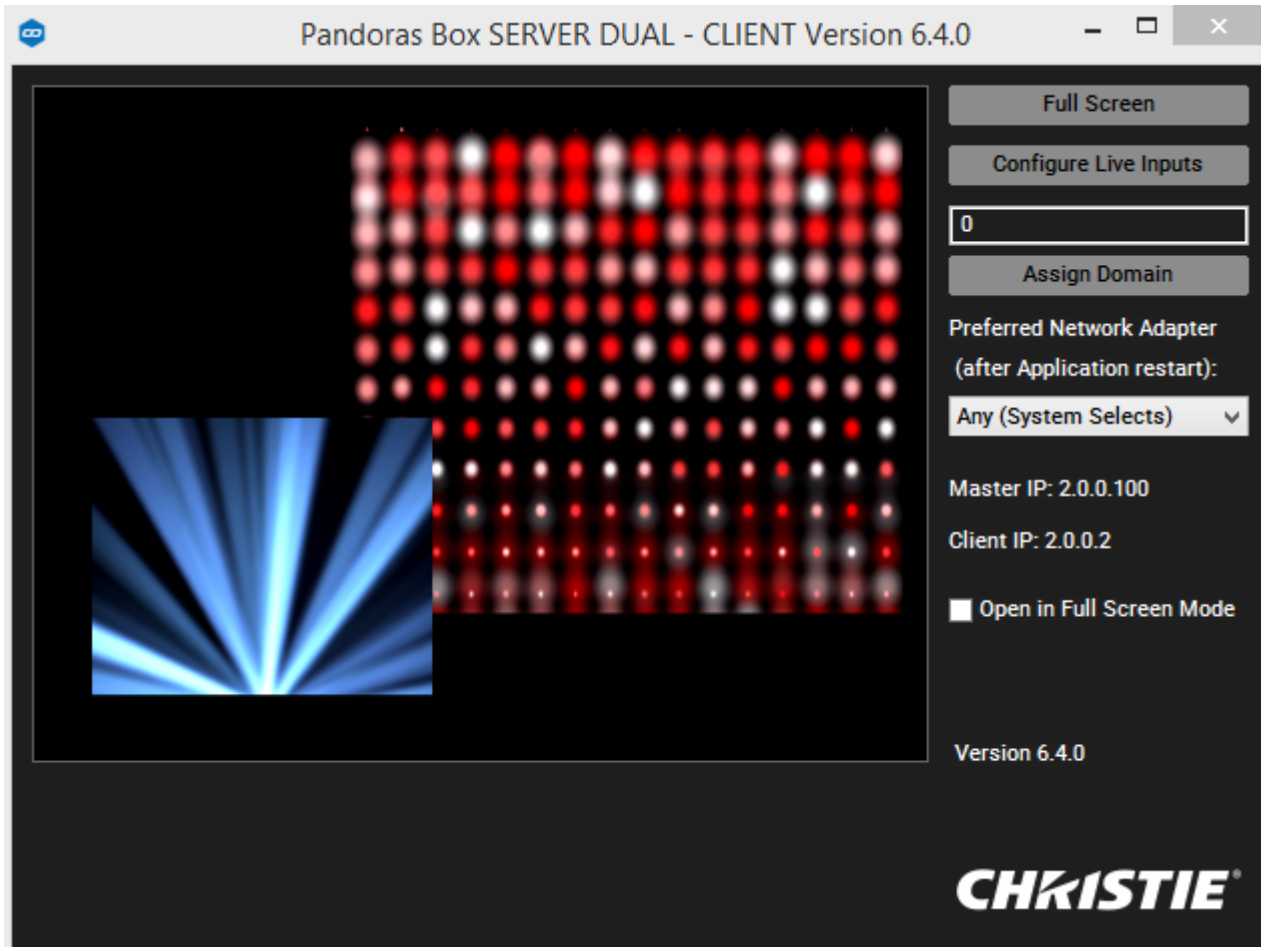
Hide / Show parameters for the selected Layer(s) in the Device Tree	U	Only parameters used in sequence		
	A	Show all parameters	H	Hide all parameters
	M	Media	O	Object (Mesh)
	I	Opacity (Intensity)	B	Blend Mode
	V	Playback	X	Audio
	P	Position		
	R	Rotation	T	Rotation Pivot
	S	Scale	D	Scale Pivot
	F	Effects	G	Particle System

Open / Show Tab	Ctrl + R	Project	Ctrl + M	Inspector
	Ctrl + T	Assets	Ctrl + I	Device Viewer
	Ctrl + Shift + M	Task Manager	Ctrl + D	Active Values
	Ctrl + Shift + R	Thumbnails	Ctrl + K	Configuration
	Ctrl + Shift + L	Device Controls	Ctrl + E	Sequence

Kiosk Mode	Ctrl + O	Open file
	Ctrl + Shift + O	Open (previously saved) Playlist
	Ctrl + S	Save Playlist
	Ctrl + Shift + S	Save Playlist as
	Space	Play / Pause Playlist
	Ctrl + F	Toggle Fullscreen
	Esc	Leave Fullscreen

6.4 User Interface - Client

This chapter explains the user interface of a Pandoras Box Client system. The Master's interface [117] is covered in the previous chapter. The chapter Master / Client Remote Setup [67] explains the general difference between a PB Master and PB Client, how to connect them and how to include a Client device in your Master project.



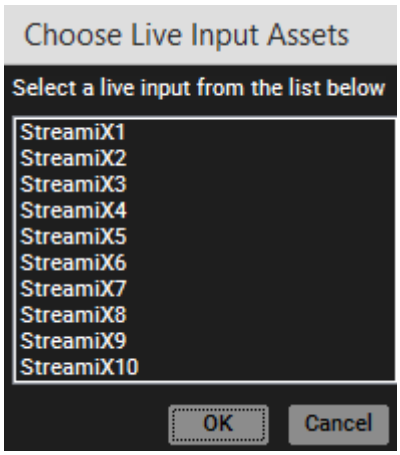
The Client user interface allows you to set up as follows:

- Toggle Full Screen by clicking "Full Screen" or alternatively press [Ctrl + F]
- Click "Configure Live Inputs" and a configuration window [317] will open (please see more details about live inputs and how to use them in Pandoras Box in the chapter "Input Card Settings" [1947])
- Change the domain channel and confirm it by clicking "Assign Domain" (please see the chapter "Network [147]" to read more about the Domain channel and how to change it in the Master software)
- Choose a dedicated network adapter in the drop-down list
- Check "Open in Full Screen Mode" and the next time starting the Client it will open in full screen-mode

See the following chapters for more information about the Client:
Client Live Input Configuration [317]
Client Keyboard Shortcuts [317]

More settings regarding the render engine (e.g. Full Screen is single) can be found in the Master interface:
Configuration tab > Render Engine [162]

6.4.1 Client Live Input Configuration



Click the "Configure Live Inputs" button in the main Client window and choose one of the available live inputs or software screen grabbers / capture tools and click OK. Pandoras Box supports most input devices that are conform and support DirectShow. Depending on the installed component, the according driver window will then show up and you can set up the available parameters of the input device.

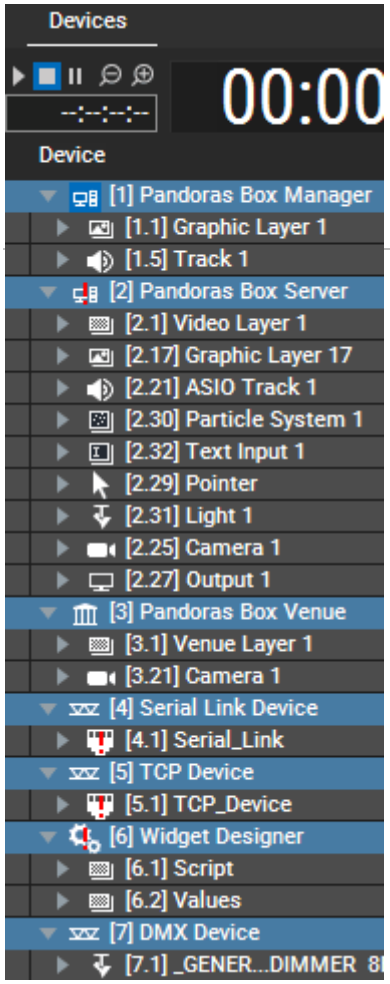
If you are using an input card from Christie or Deltacast, please use the Master to configure them [195]. See more details about live inputs and how to use them in Pandoras Box in the chapter "Input Card Settings" [1947].

StreamiX Live Inputs are described in this chapter. [724]

6.4.2 Client Keyboard Shortcuts

Ctrl + F Toggle Fullscreen

6.5 Devices and Layer Parameters



This chapter explains all different devices and layer types that can be controlled via the timeline.
Please see the topics Sequence [292], Device Tree [173] and Device Control tab [171] if you are rather interested in learning how to actually add devices to the device tree or how to work within the Sequence, i.e. how to change and store values, program timelines etc.

Layers in Pandoras Box Systems

Layer Type	function
Video [647]	full motion video (and audio) playback; image sequences or static images
Audio [661]	ASIO playback, synchronized to the Master clock
Particle [185]	optimized for the use of Particle systems (*)
Pointer [665]	displaying the mouse pointer or touch inputs
Light [666]	illumination of other layers including shadow
Notch [671]	rendering and controlling of Notch scenes
Camera [675]	set up the viewpoint / look-at-point onto your 2D / 3D composition; first render path / Composition Pass and source for the Output Pass
Output [682]	controls the overall output of the Client including keystone and softedge; second render path / Output Pass

To add a specific Layer type, right-click the Site and choose "Add Device" or "Add Multiple Devices". Since version 8, all Layers except the Pointer Layer, can be added unlimited times. However, as soon as a Site should be manifested on a connected Client, the maximum number of Output Layers is bound to the number of available "Pandoras Box Software Licenses". Multiple licenses can be stacked by combining dongles or they can be applied to a single dongle.
Per default, the source of an Output Layer is a camera texture. If you add a new Output Layer, a Camera Layer is generated automatically and cannot be removed unless the according Output Layer is removed. However, you can add more Cameras than Outputs, e.g. to have different viewpoints from a 3D composition.

(*) The Particle Layer has significantly reduced layer parameters and one Particle System with a Particle Emitter already applied, hence its ready to be used. You can of course drag Particle Systems to a normal Video Layer but the Particle Layer provides an easier workflow.

The Venue Site [694] device is for pure (pre-)visualization. It renders your programming (of Clients) on 3D objects that form a virtual copy of your stage.

See the following chapters:
Video Processing Pipeline [320] – Get to know the layer and rendering structure (i.e. first and second render paths) in Pandoras Box
Working with Effects [322] - Learn how to add effects to layers and work with them

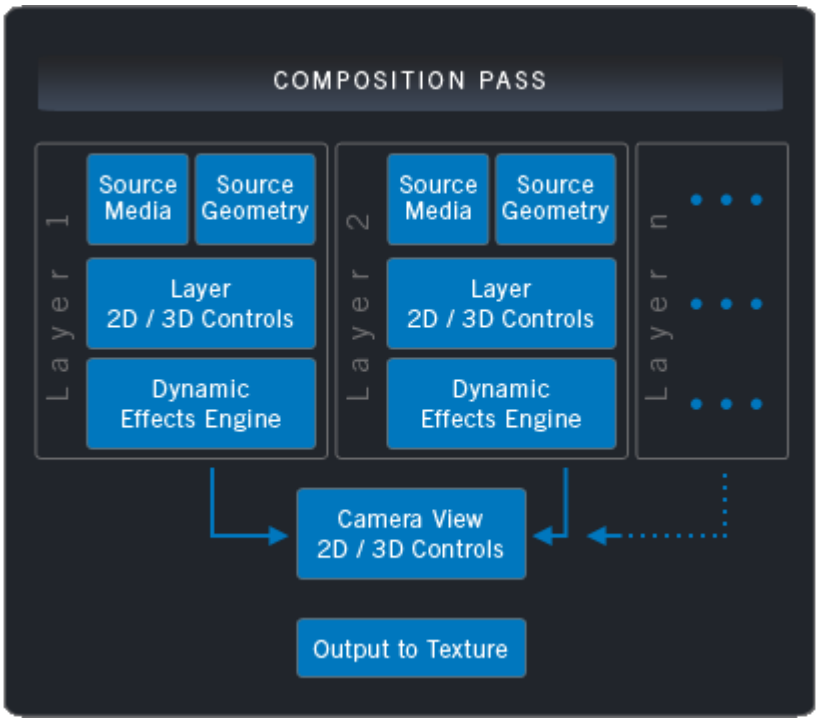
Controlling External Devices

One of the most important features of Pandoras Box is its ability to control a wide range of devices based on various output control protocols. Beside Pandoras Box Clients other devices such as DMX, RS232 / 422 or any TCP/IP device might be remote controlled from any timeline as well. This allows perfectly synchronized show control. The topic External Control [706] goes more into detail.

	function
Serial Link Device [692]	remote control Serial Link or other external RS232 / 422 or TCP/IP devices
TCP Device [693]	remote control TCP/IP devices, e.g. projectors or matrix switches
Widget Designer Device [696]	sending commands and values to Widget Designer application
DMX Devices [691]	remote control external DMX device via DMX / Art-Net

6.5.1 Video Processing Pipeline

This chapter explains the fundamental video processing pipeline in Pandoras Box. It is separated in two render passes
- the (layer) composition pass, most important for the creation process
- the output pass, most important for the technical setup



Step 1, the Creation Process:

By default, all Video [647], Pointer [665], Light [666] and Notch Layers [671] are part of the Composition Pass. Here, you do your layer composition - setting the media, the video playback, position, rotation and scale parameters as well as the visual effects. The render history is determined by the Z Position respectively, if it is not set, by the layer arrangement whereas the topmost layer in the Device Tree tab is rendered first and thus overlaid by other layers. (For even more options see Camera Inspector [218] > "Determine layer render order")

The end of the Composition Pass is determined by adjusting the camera's perspective (see Camera Layer) [675] onto your 3D composition. Then the "Output to Texture" process generates a so called render target. There is a render target for each camera in the Device Tree.
As said above, per default, all mentioned Layers are seen by all cameras. However, the Layer Inspector [211] > "Render Layer in Pass" settings allow to adjust this. You can for example exclude a Layer from specific Cameras, or send it to an Output directly.

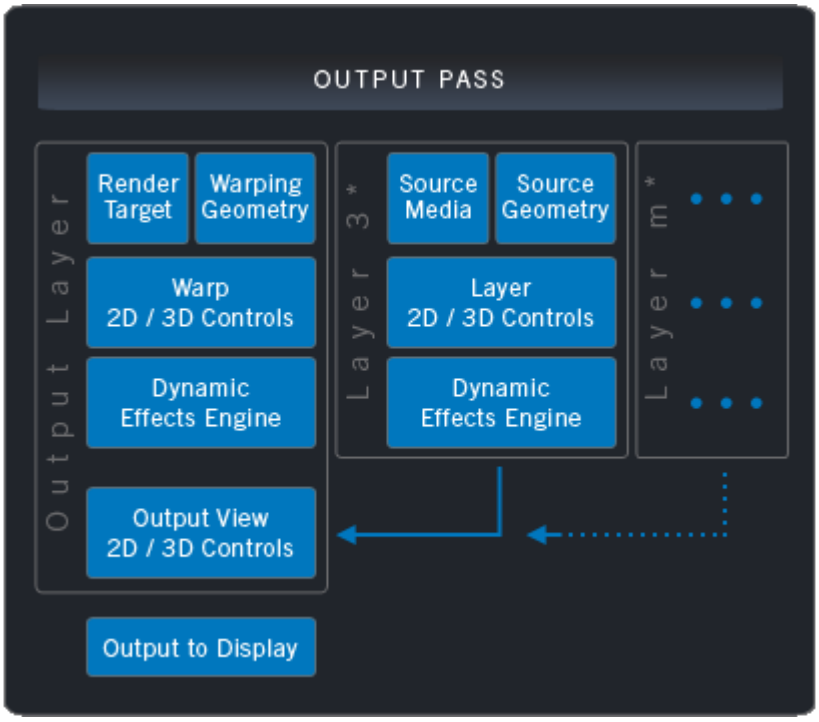
The final texture, the render target, can be forwarded in various ways. In most cases it is simply processed by the next Output pass. This is the case when the Camera Layer is linked to an Output Layer [682] and its render state [689] says "Render".

Further it is forwarded:
- to the graphics card's output, thus your connected display, in case the output's render state [689] is set to bypass
- to the matrix patch [682]
- to be rendered for the video export [135]

Camera setup


Render Target 1

Render Target 2



Step 2, the Technical Setup:

Here, you match the texture to your technical setup.
By default, the source media is simply the render target from the assigned Camera Layer.

In the output pass, the controls of the Output Layer [682] allow to set up:
- a warping object if the texture is projected on a bended screen,
- a matrix patch,
- softedge or keystone adjustments,
- a color correction on the whole output (via effects [322])
- additional perspective settings.

As said above, the Layer Inspector [211] offers Render settings which allow to send a Layer directly to the Output Pass. In the left image, such a Layer is marked with a star (*). In that case, a layer is not part of the Render Target and not influenced by warp, softedge, keystone and FX settings. Only the perspective settings (named "Output View Controls" in the left image) apply. In short, the feature allows to place Layers outside the warp or softedge area

At the end of the output pass, the "Output to Display" process forwards a texture to the so called back buffer respectively to the graphics card and thus your connected display.
For the above depicted example to possible outputs could be:

for output 1, a warp object has been applied and softedge settings were used

for output 2, keystone and softedge settings were used

6.5.2 Working with Effects

This chapter describes, how to work with effects. The Pandoras Box Software License allows to apply an unlimited number of effects to a Layer, e.g. a Video Layer or Output Device. All effects and animations are automatically synchronized across the system network.

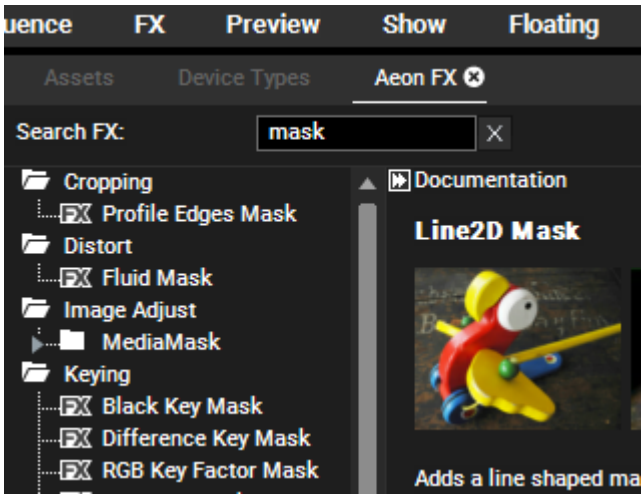
Adding, Editing and Removing Effects 323
FX Order 325
Copy / Paste FX 326
FX List 327
Blacklevel Compensation 646

6.5.2.1 Adding, Editing and Removing FX

This chapter explains how to add an effect to a Layer, access its parameters and how to remove it again. The effects themselves can be found in the chapter "FX List" [327]. The following chapters explain how to re-order [325] effects or copy [326] them.

Adding Effects

Generally, layers come without optional effects, except if the site/project is started in Lighting Console Configuration [1005]. If you have a look into the Device Tree [173] of a plain Video Layer, the FX folder is empty and the Device Controls [171] tab shows no FX parameters.



To add an effect to a Video Layer, or Output device, please open the "Aeon FX" tab, which is part of the View [287] called "FX". In case you closed it, please click on "Tabs" in the Toolbar and choose "Aeon FX".

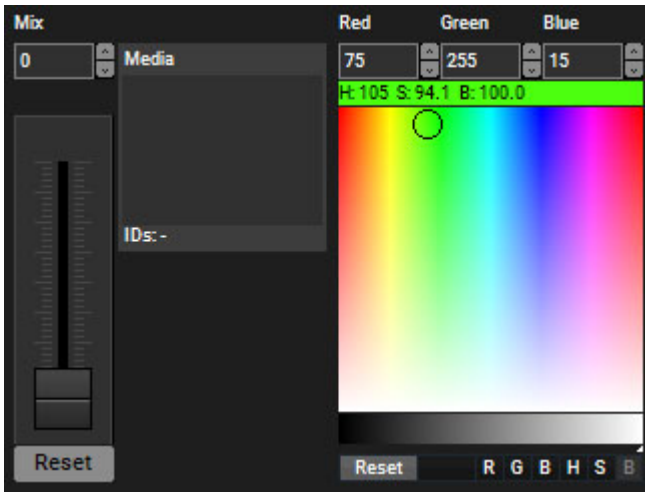
You can browse through the FX folders or use the "Search FX" field. If you click on an effect, the right side shows more information about it.

The "FX Explorer [129]" allows to try and toggle between various effects. Simply, right-click an effect to launch it.

To add an effect to a Video or Output Layer. There are multiple work flows available:
- select a layer first (multi-selection is possible with the [Ctrl] key) and then double-click the desired effect
- choose the effect and drag and drop it onto the Layer in the Device Tree tab
- choose the effect and drag and drop it onto the Device Controls tab, into an empty area right to the default parameter faders.

Now, the Device Tree and Device Controls tab show the new effect next to the standard parameters. The new section is always titled with the name of the effect.
You can add more effects if you like. As effects build upon each other, the effect order is important. The chapter FX Order [325] explains more about this.

Editing and Controlling Effects



Depending on your effect, different parameters can be set. The default values are set in a way that they do not affect the layer automatically.

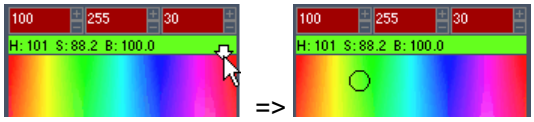
To apply the effect, set the "Mix" fader to 255. "Mix" defines how much the effect is mixed to the image / video file.

Most effect parameters are fader controls which are explained in the chapter "Changing Parameter Values [171]". Some effects also allow to assign a media or mesh. If a color can be adjusted, you will see a standard color picker field.

How to use the Color Picker
The Color Picker is based on the RGB (Red, Green, Blue) color model and includes faders for HSB (Hue, Saturation, Brightness).

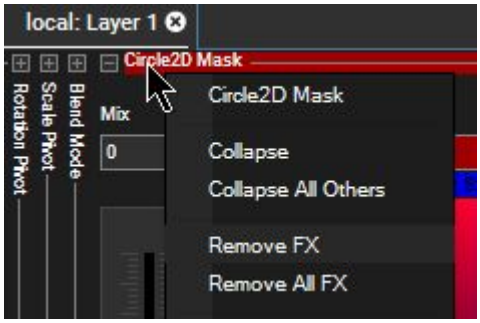
You can pick a color directly from the color field. First click into the white area to increase the brightness, and then click into the color field. The horizontal color fader and the color field depend on the parameter at the bottom of the color picker (R,G,B,H,S or B) which is "B" (Brightness) per default.

You can also enter values for red, green and blue manually above the color field. Note that the bar between the field and the values show the corresponding values for HSB and is colored in the chosen color instantly. If you want to see the color in the color field (marked with a black outlined circle), click the white arrow.



Of course, you can also control effect parameters using the SDK or Widget Designer [786], e.g. through interactive devices like the AirScan [1988] or sensors like Phidgets [1383] and more. To do this, you need to know the exact parameter name which is case-sensitive and all spaces count. Further, the effect name is followed by the pipe "|" character and then the effect parameter name, e.g. Line 2D Mask|Mix or Line 2D Mask|Red. The effect names and parameter are shown in the Device Control tab but also in the Device Tree when unfolding entries, which helps, if a name is quiet long.

Removing Effects



There are two ways to remove an effect from a layer.
You can either use the Device Control tab, right-click on the effect's name and choose "Remove FX" or "Remove All FX".

Or you can use the Device Tree. Right-click an effect to remove it or right-click a layer for the option to "Remove All FX".

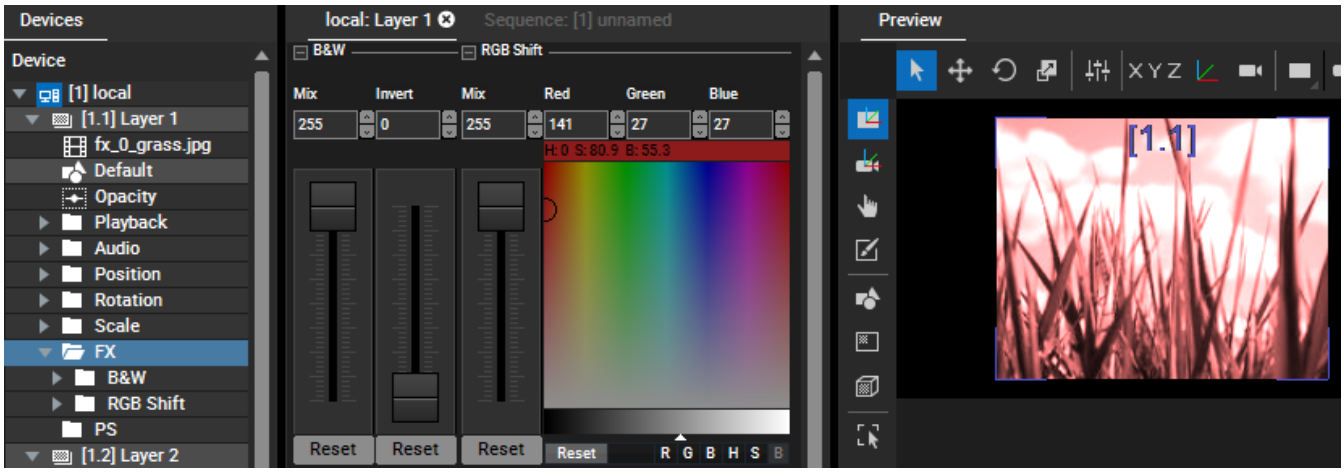
If you saved values from the effect to the timeline(i.e. there are Containers with Keys on the Layer) a warning pops up as those keys will be automatically deleted.

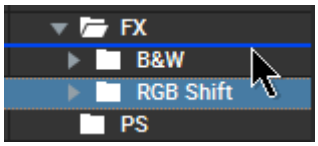
6.5.2.2 FX Order

This chapter explains how to re-order effects and why the effect order is important. The effects themselves can be found in the chapter "FX List" 327.

If you apply several effects to a layer, they build upon each other according to their order of appearance. The order FX1>FX2 achieves a different result than FX2>FX1.

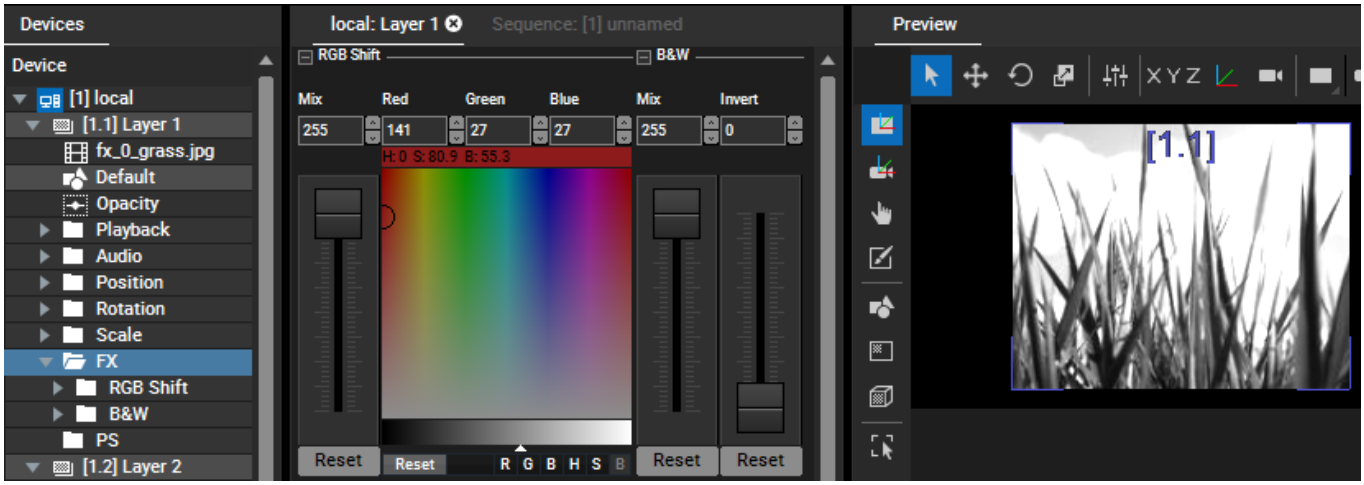
The first example shows the result when first a "B&W" effect is applied and then "RGB Shift" which gives the image a red tone.





To reorder an effect, simply drag it in the Device Tree to a new position (marked with the blue line) within the FX folder.

As you see, the result looks different because the "B&W" effect is applied now at the end of the FX chain.



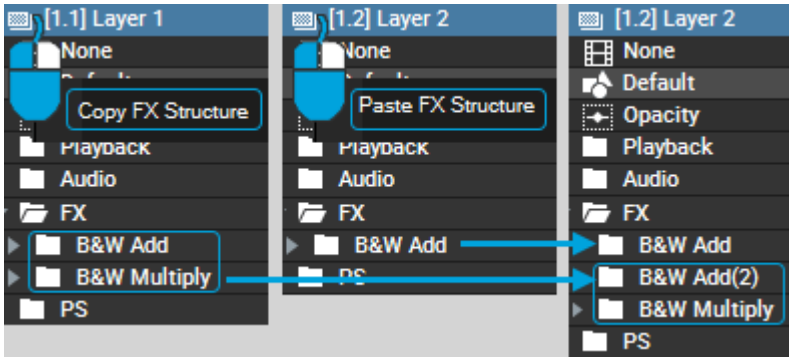
6.5.2.3 Copy / Paste of FX

This chapter explains how to copy effects from one layer to another via the command "Copy FX" and alternatively via containers that contain FX keys.

Copy and Paste the FX Structure

To copy effects from one layer to another one, simply right-click the first and choose "Copy FX Structure" and "Paste FX Structure" on the second.

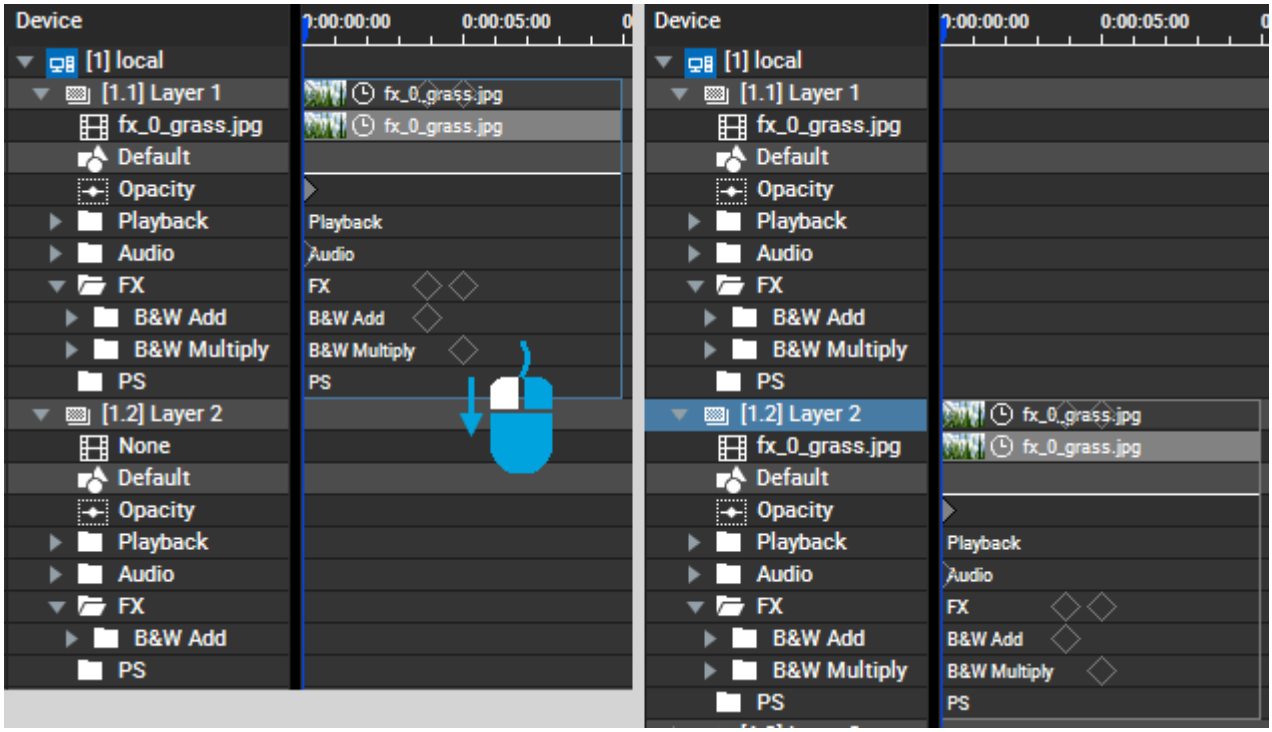
The FX structure (i.e. order) from the first layer is added to the second layer even though it might already have (same or other) effects. The example shows that two effects are pasted to "Layer2" but as "B&W Add" already exists, the pasted one becomes "B&W Add(2)". Note that active values are also copied.



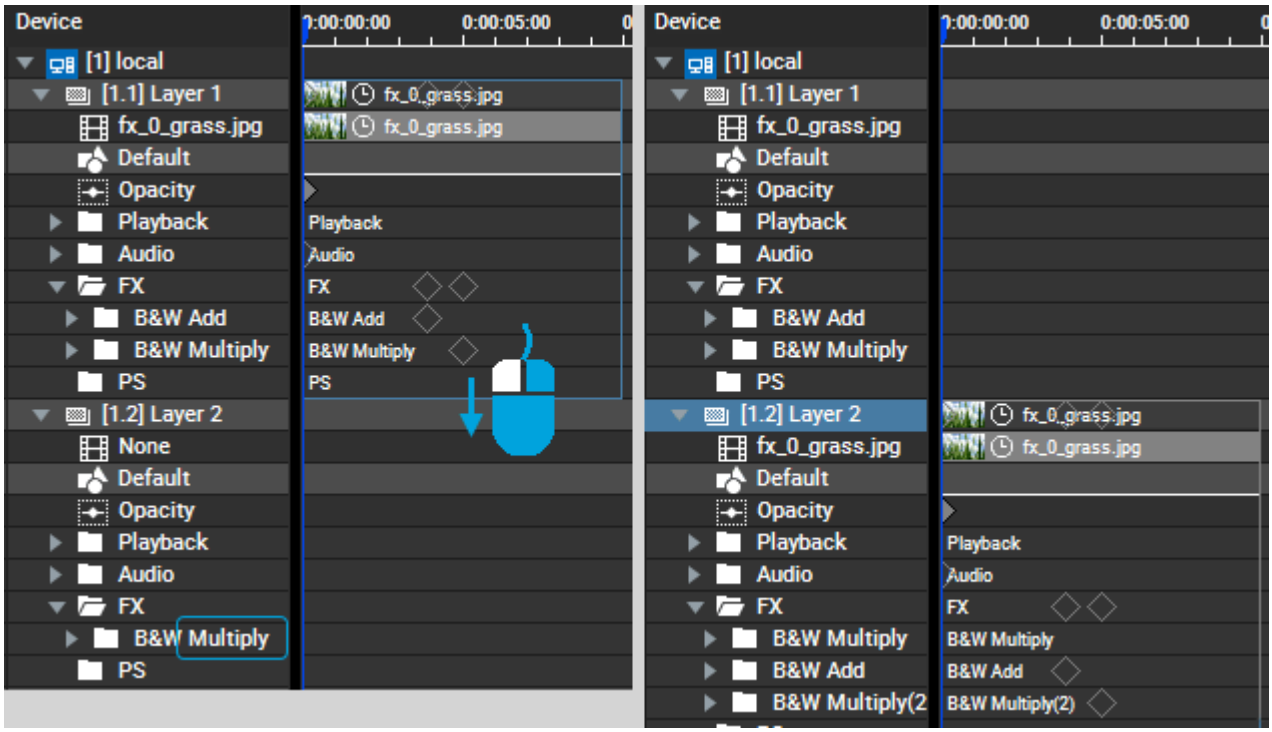
Moving Containers with FX Keys

Effects are added automatically to a layer when you move or copy Containers to it that contain FX Keys. If the target layer has already an FX with the same name (or in case of several FXs, the FX order matches) the Keys are copied to the existing effects.

Let's take the example from above before pasting the FX structure: If you copy a Container from Layer1 (with Keys for both effects!) to Layer2, only "B&W Multiply" is added there.



However, if Layer2 had other effects or the same ones in a different order, the result would be different. In the following example, "B&W Multiply" is applied to Layer2, but in Layer1 it follows the other one and that is the reason why both effects are added to Layer2 in case a Container is copied or moved.



6.5.2.4 FX List

The following tables list all effects in the order of their according FX theme collection. WD und PB

Animation 339
Rain 339

Anti-Aliasing 340
Anti-Aliasing 1 340
Anti-Aliasing 2 340
Anti-Aliasing 3 341

Blur 342
Blur - Downsampling 1 342
Blur - Downsampling 2 342
Blur - Downsampling 3 343
Blur XY Alpha Glow Color 344
Blur XY Alpha Glow Pre-Multiplied 345
Blur XY Alpha 346
Blur XY 347
Dilate Alpha 348
Dilate 349
Erode Alpha 350
Erode 351
GaussianBlur On-Off 351
GaussianBlur 352
Glow 352
Radial Blur Alpha Color 353
Radial Blur Alpha 354

Blur 342
Radial Blur 355
Refresh Delay 355
Render Delay 356
Render Draw Media 357
Render Draw 358
Video Delay 358

Camera Projection 359
Perspective Projection Mapping 359

Color Effects 360
B&W Add 360
B&W Multiply 360
B&W 361
BiTone 361
BiTone Luma 362
Color Stripes H 362
Color Stripes V 363
ColorFade 363
ColorScroll Add 363
ColorScroll Multiply 364
ColorScroll Only 364
Emboss Angle 365
Emboss Edges 365
Emboss 366
Granite 366
Invert 367
LinearLight 367
RGB Inversion 367
Sepiatone 368
Solarize 368
Tone Mapping 369
TriTone 369

Color Effects - Crop 370
B&W Add Crop Pixel 370
B&W Crop Pixel 370
B&W Multiply Crop Pixel 371
BiTone Crop Pixel 372
BiTone Luma Crop Pixel 373
Color Stripes H Crop Pixel 373
Color Stripes V Crop Pixel 374
Color Fade Crop Pixel 374
ColorScroll Add Crop Pixel 375
ColorScroll Multiply Crop Pixel 375
ColorScroll Only Crop Pixel 376
Emboss Angle Crop Pixel 376

Color Effects - Crop 370
Emboss Crop Pixel 377
Emboss Edges Crop Pixel 377
Granite Crop Pixel 378
Invert Crop Pixel 378
Linear Light Crop Pixel 379
RGB Inversion Crop Pixel 379
Sepiatone Crop Pixel 380
Solarize Crop Pixel 380
Tone Mapping Crop Pixel 381
TriTone Crop Pixel 382

Color Effects - Softborder 383
B&W Add Softborder 383
B&W Multiply Softborder 383
B&W Softborder 384
BiTone Softborder 384
BiTone Luma Softborder 385
Color Stripes H Softborder 385
Color Stripes V Softborder 386
Color Fade Softborder 386
ColorScroll Add Softborder 387
ColorScroll Multiply Softborder 387
ColorScroll Only Softborder 388
Emboss Angle Softborder 388
Emboss Edges Softborder 389
Emboss Softborder 389
Invert Softborder 390
Linear Light Softborder 390
RGB Inversion Softborder 391
Sepiatone Softborder 391
Tone Mapping Softborder 392
TriTone Softborder 393

Color Shift 394
BW Media ColorShift Scroll 394
BW Media ColorShift 395
Color Inversion 395
ColorOnly 396
RGB Add 396
RGB Channel Inversion 397
RGB Hue 397
RGB INV Add 398
RGB INV Multiply 398
RGB Multiply 399
RGB Replace 399
RGB Screen 400
RGB Shift 400

Color Shift 394
RGB XY Shift 401
RGB-BGR Add 401
RGB-BGR Multiply 402
RGB-BRG Add 402
RGB-BRG Multiply 403
RGB-GBR Add 403
RGB-GBR Multiply 404
RGB-GRB Add 404
RGB-GRB Multiply 405
RGB-RBG Add 405
RGB-RBG Multiply 406
RGBA Modulo 406

Compositing 406
Render History 408
Target Add 408
Target Buffer 409
Target Color Burn 409
Target Darken 409
Target Difference 410
Target Lighten 410
Target Multiply 410
Target Reflect 411
Target Screen 411
Target Subtract 411

Cropping 412
Crop Edges INV Softborder In 412
Crop Edges INV 413
Crop Edges Softborder In 414
Crop Edges Softborder 415
Crop Edges 416
Crop Radial 417
Cropper HV RGBA 418
Cropper HV 419
Frame RGBA 420
Frame 421
Framed Cropper HV 422
Iris RGBA 423
Iris 424
Profile Edges Alpha 425
Profile Edges Mask 426
Softborder RGBA 427
Softborder 428

Distort 429
Displace 429

Distort 429
Distort 429
Fluid Color 430
Fluid Mask 431
Fluid Paint 432
Interactive Waves Buffer RH 433
Interactive Waves Buffer 434
Interactive Waves RH 435
Interactive Waves 436
Lens Barrel Distortion Color Corr 437
Lens Barrel Distortion 437
Lens Pincushion Distortion 438
Magnify Glass 438
PolarCoordinates 439
Ripple Animation 439
Ripple 440
Swirl Angle 440
Texture Morph 441
Waves 442

Fractals 443
Julia Fractal 443
Mandelbrot Fractal 443

FX Collections 444
Server FX 16bit 9Ch 444
Player FX 16bit 4Ch 444

Generator 445
RND Noise Add 445
RND Noise Divide 445
RND Noise Mix 445
RND Noise Multiply 446
RND Noise Subtract 446

Geometry 447
Inflate 447
Lens Barrel Distortion - Vertex 447
Lens Pincushion Distortion - Vertex 448
Morph A-B 448
Morph A-B-C 448
Rotate Local Quaternion 449
Rotate Local XYZ 449
Rotation Speed Local XYZ 450
XYZ Push 450
XYZ Squeeze 451

Gradients 452
Gradient Linear Horizontal 452

Gradients 452
Gradient Linear Vertical 452
Gradient Linear 453
Gradient Quad 16bit 454
Gradient Quad Black Lift 16bit 455
Gradient Quad Black Lift 456
Gradient Quad 457
Gradient Radial 458
Gradient Spiral 458
Gradient Swirl 459
Gradient Triangle 16bit 460
Gradient Triangle Black Lift 16bit 461
Gradient Triangle Black Lift 462
Gradient Triangle 463

Image Adjust 464
BasicColorCorrection 464
Brighten 464
Burn 465
CMYK Adjust 466
ColorContrastBrightness Multiply 465
ColorContrastBrightness 466
Gamma RGB 467
HighMidLowLights 467
HSV Adjust 468
HSV by Hue 469
HSV-HS for V range 470
Levels RGB 470
Levels 471
Posterize 471
RGB Add 471
RGB Clipper 472
RGB CTB 472
RGB CTO 473
RGB Gamma 473
RGB Min Max 474
RGB Multiply 474
RGB Squisher 475
RGB to HSL 475
RGB to Video Colors 16-235 475
Video Colors 16-235 to RGB 476
Video RGB CC 476
Video YUV to RGB 477
YUV(bt709)2RGB 477
YUV2RGB 477

Image Adjust - Crop 478
CCB Crop Edges (px) 478

Image Adjust - Crop 478
High Mid Low Crop Pixel 479
Levels Crop Edges (px) 480
Levels RGB Crop Edges (px) 480
RGB Add Crop Edges (px) 481
RGB Clipper Crop Edges (px) 482
RGB CTB Crop Edges (px) 482
RGB CTO Crop Edges (px) 483
RGB Gamma Crop Edges (px) 483
RGB Min Max Crop Edges (px) 484
RGB Multiply Crop Edges (px) 485
RGB Squisher Crop Edges (px) 486

Image Adjust - Iris 487
CCB Iris 487
High Mid Low Iris 488
Levels Iris 489
Levels RGB Iris 490
RGB Add Iris 491
RGB Clipper Iris 492
RGB CTB Iris 493
RGB CTO Iris 493
RGB Gamma Iris 494
RGB Min Max Iris 495
RGB Multiply Iris 496
RGB Squisher Iris 497

Image Adjust - Mediamask 498
CCB MediaMask 498
High Mid Low MediaMask 499
Levels MediaMask 499
Levels RGB MediaMask 500
RGB Add MediaMask 500
RGB Clipper MediaMask 501
RGB CTB MediaMask 501
RGB CTO MediaMask 502
RGB Gamma MediaMask 502
RGB Min Max MediaMask 503
RGB Multiply MediaMask 503
RGB Squisher MediaMask 504

Image Adjust - Softborder 505
CCB Softborder 505
High Mid Low Softborder 506
Levels RGB Softborder 507
Levels Softborder 507
RGB Add Softborder 508
RGB Clipper Softborder 508

Image Adjust - Softborder 505
RGB CTB Softborder 509
RGB CTO Softborder 509
RGB Gamma Softborder 510
RGB Min Max Softborder 510
RGB Multiply Softborder 511
RGB Squisher Softborder 511

Keying 512
Alpha Adjust 512
Black Key Mask 512
Black Key 513
Difference Key Mask 513
Difference Key 514
Difference Matte 515
Luma Key 516
RGB Key Factor Mask 517
RGB Key Factor 518
RGB Key Mask 519
RGB Key 520
Spill Reduce Blue AVG 521
Spill Reduce Blue by Green 522
Spill Reduce Blue 523
Spill Reduce Green AVG 524
Spill Reduce Green by Blue 525
Spill Reduce Green 526
White Key Mask 526
White Key 527

Lighting 527
1 Light Mat-Spec-Color 530
1 Light Shadow Catcher 531
1 Light 532
2 Lights Mat-Spec-Color 532
2 Lights Texture Projection Adaptive 533
2 Lights 533
3 Lights Mat-Spec-Color 533
3 Lights 534
Global Light 534
Light Texture Projection 535
Lighting 536
Texture Projection 537

Lighting - Texture Projection 538
Texture Projection Add 538
Texture Projection Alpha Mix AB 539
Texture Projection Alpha Mix 540
Texture Projection Blend AB 541

Lighting - Texture Projection 538
Texture Projection Mix AB 542
Texture Projection Mix 543

LUT 544
LUT Camera Arri 544
LUT Creative 544
Custom 3D LUT Filter 544
LUT Filter 1 544
LUT Filter 2 545

Masking 546
Alpha Black Fill RGB 546
Alpha Fill RGB 547
Alpha Left Right - Retain Resolution 549
Alpha Left Right 548
Alpha Media As Mask Invert 549
Alpha Media As Mask 550
Alpha Replace 550
Alpha Top Bottom 551
BW Mask 551
BW Media As Mask Invert 552
BW Media as Mask PreMul 553
BW Media As Mask 554
WB Mask 554

Mirror 555
Horizontal Flip 555
HV Flip 555
Mirror 2D 556
Vertical Flip 556

Overlay 557
Horizontal Dual Media Overlay 557
Horizontal Quad Media Overlay 558
Horizontal Triple Media Overlay 559
Media Overlay Add 560
Media Overlay Darken 561
Media Overlay Divide Inv 562
Media Overlay Divide 563
Media Overlay Hue 564
Media Overlay Lighten 565
Media Overlay Mix 566
Media Overlay Multiply 567
Media Overlay Screen 568
Media Overlay Subtract 569
Quad Media Overlay 570
Vertical Dual Media Overlay 571
Vertical Quad Media Overlay 572

Overlay 557
Vertical Triple Media Overlay 573

Pattern 574
BinaryPattern 574
Checkerboard - Odd Even 574
Checkerboard-Transparency 575
Checkerboard 575
Grid 576
LED 576
Lines Horizontal 577
Lines Vertical 577
Pixels 578
Simple Outline 579

Remap 580
ReMap Overlay 580
ReMap 581

Shadows 582
Drop Shadow Alpha 582

Shapes 583
Circle2D Mask 583
Circle2D 584
Draw Point Alpha 585
Draw Point RGB 586
Ellipse2D Mask 587
Ellipse2D 588
EQ Media 588
EQ Waveform 590
EQ 591
Line Horizontal 592
Line Vertical 592
Line2D Mask 593
Line2D 594
MetaBall2D Add 595
MetaBall2D Multiply 596
MetaBall2D Outline Glow 597
Quad 16bit 598
Quad Black Lift 16bit 599
Quad2D Mask 600
Quad2D 601
Rectangle 602
Rectangle2D Mask 603
Rectangle2D 604
Triangle2D Mask 605
Triangle2D 606

Sharpen 607
Edges 607
Outlines 608
Sharpen Alpha Edges (Fill Color) 608
Sharpen Alpha Edges 609
Sharpen 609

Snapshot 610
Render Feedback 610
Render Freeze 610
Video Freeze 610

Softedge 611
Alpha Softedge 611
RGB Softedge 611
Softedge Bottom Alpha 612
Softedge Bottom RGB 612
Softedge Left Alpha 612
Softedge Left RGB 612
Softedge Right Alpha 613
Softedge Right RGB 613
Softedge Top Alpha 613
Softedge Top RGB 613

Softedge - Softedge P-Curve Alpha 614
Alpha Softedge P-Curve 614
Softedge P-Curve Bottom Alpha 614
Softedge P-Curve Left Alpha 614
Softedge P-Curve Right Alpha 615
Softedge P-Curve Top Alpha 615

Softedge - Softedge P-Curve Alpha Gamma 616
Alpha Softedge P-Curve Gamma 616
Softedge P-Curve Gamma Bottom Alpha 617
Softedge P-Curve Gamma Left Alpha 617
Softedge P-Curve Gamma Right Alpha 617
Softedge P-Curve Gamma Top Alpha 617

Strobe 618
Alpha Flash 618
Alpha Pulse 618
RND Alpha Flash 619
Sync Color Strobe 619

Stylize 620
ASCII Replace 620
Film Look 620
Wireframe Edges 621

Stylize 620
Wireframe 622

Transition 623
Alpha 16bit 623
Dissolve 16bit 623
Transition Crop Media Mask 624
Transition Media Mask 624
TransitionFX 625

UV Mapping 631
AutoAspect Scroll Horizontal (infinite) 631
AutoAspect Scroll Horizontal (steady speed) 631
Matrox Dual Head Split H 632
Matrox Dual Head Split V 635
Matrox Tripple Head Split H 634
Matrox Tripple Head Split V 635
Resize 635
Rotate Tiled 636
Rotate 637
Texture Zoom 637
Tiling 638
Underscan 638
UV Offset 639
UV Remap 4x 639
UV Remap 641
UV Scroll 642
UV Wrap Blend Horizontal 642
UV Wrap Blend Vertical 643

Warp 644
Warp 644
Warp Target 645

6.5.2.4.1 Animation

Rain

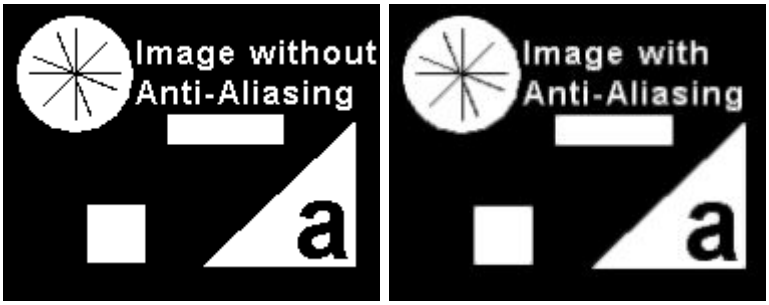
Adds animated raindrops. Test

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
SpeedX	Speed of Rain horizontal	0-65535	32768
SpeedY	Speed of Rain vertical	0-65535	32768
Size	Size of raindrops	0-65535	32768

6.5.2.4.2 Anti-Aliasing

Anti-Aliasing 1

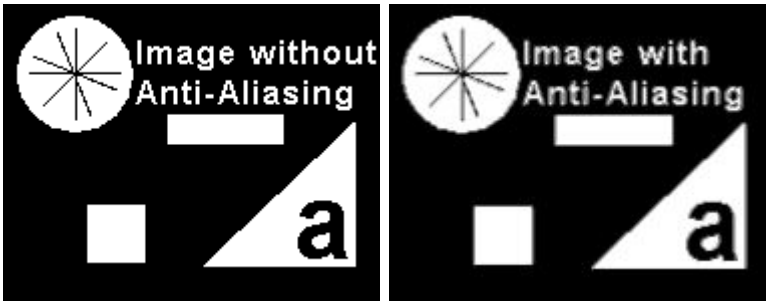


All anti-aliasing effects improve the image quality by smoothing pixelated or jagged edges within the image. There are various methods to achieve this. This anti-aliasing effect applies an directionally localized anti-aliasing (DLAA) algorithm that you can adjust with a "Mix" parameter.
An alternative to anti-aliasing are the Blur [342] effects, especially the downsampling ones.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Anti-Aliasing 2



All anti-aliasing effects improve the image quality by smoothing pixelated or jagged edges within the image. There are various methods to achieve this. This anti-aliasing effect applies a normal filter anti-aliasing (NFAA) algorithm that you can adjust with a "Mix" and "Scale" parameter.
An alternative to anti-aliasing are the Blur [342] effects, especially the downsampling ones.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scale	Scale	0-1000	250

Anti-Aliasing 3

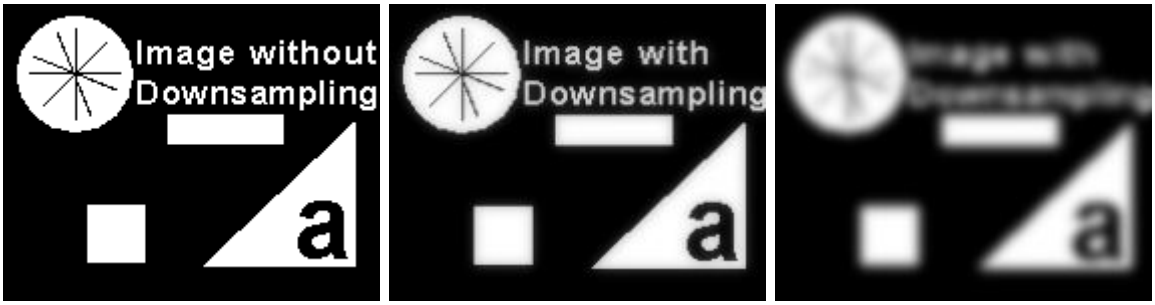
All anti-aliasing effects improve the image quality by smoothing pixelated or jagged edges within the image. There are various methods to achieve this. This anti-aliasing effect applies a super sampling anti-aliasing (SSAA) algorithm that you can adjust with a "Mix", "Width" and "Softness" parameter.
An alternative to anti-aliasing are the Blur [342] effects, especially the downsampling ones.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Width	Width	0-1000	175
Softness	Softness	0-1000	250

6.5.2.4.3 Blur

Blur - Downsampling 1

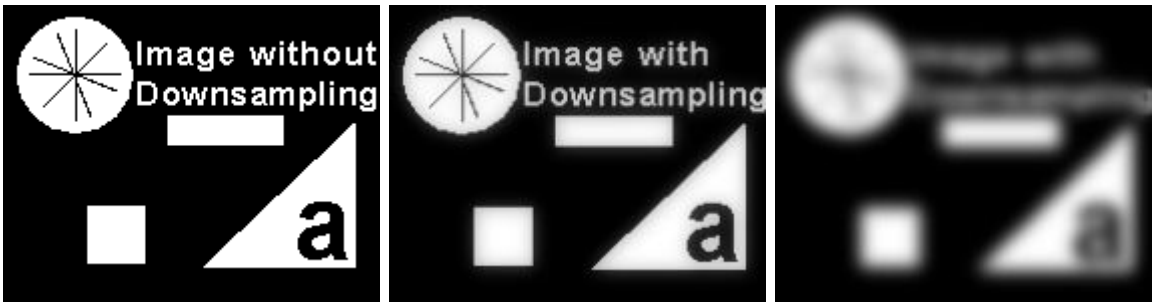


All Downsampling effects improve the image quality by smoothing pixelated or jagged edges within the image and can be used alternatively to an anti-aliasing effect [340].
This applies a downsampling blur algorithm with a texture resolution of 75%. The middle image shows the effect with the "Mix" set to a third, in the last image it is set to full.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Blur - Downsampling 2

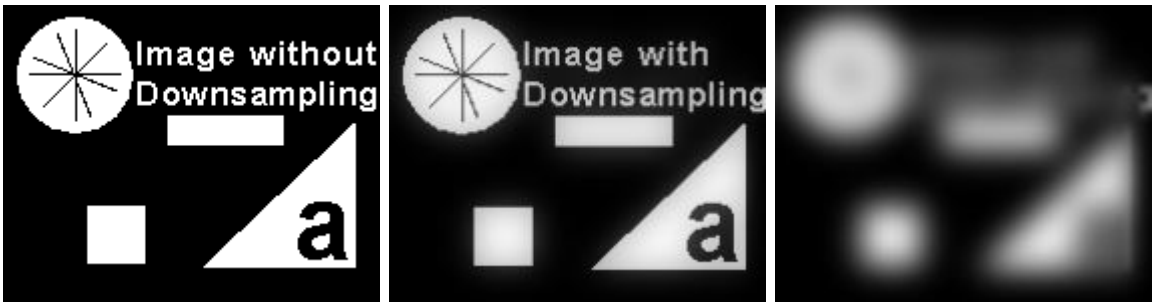


All Downsampling effects improve the image quality by smoothing pixelated or jagged edges within the image and can be used alternatively to an anti-aliasing effect [340].
This applies a downsampling blur algorithm with a texture resolution of 50%. The middle image shows the effect with the "Mix" set to a third, in the last image it is set to full.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Blur - Downsampling 3

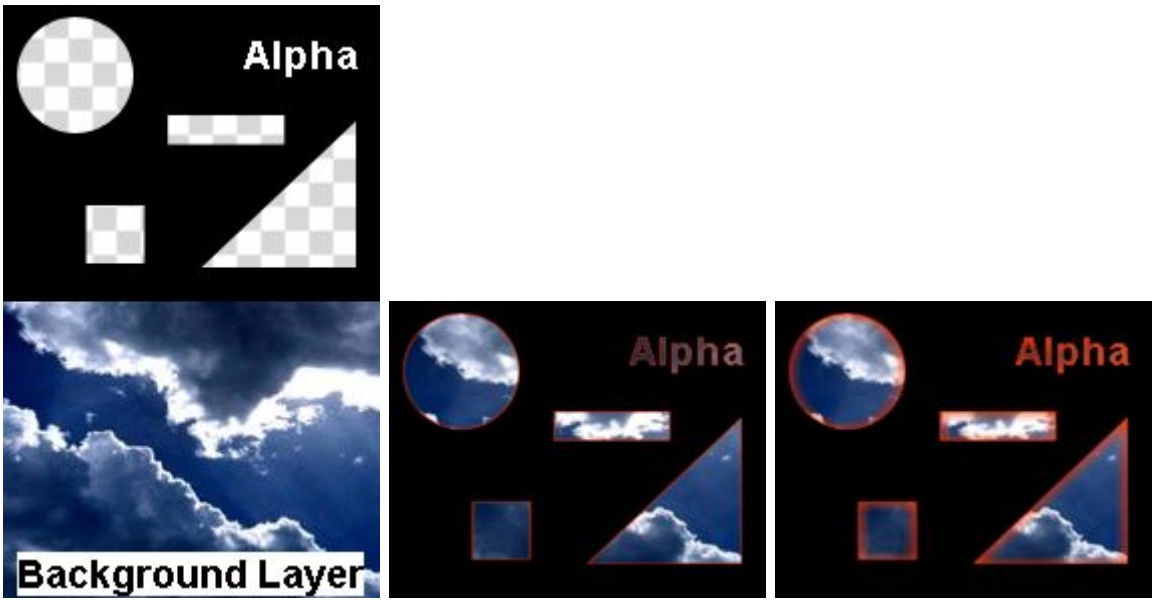


All Downsampling effects improve the image quality by smoothing pixelated or jagged edges within the image and can be used alternatively to an anti-aliasing effect [340].
This applies a downsampling blur algorithm with a texture resolution of 25%. The middle image shows the effect with the "Mix" set to a third, in the last image it is set to full.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Blur XY Alpha Glow Color

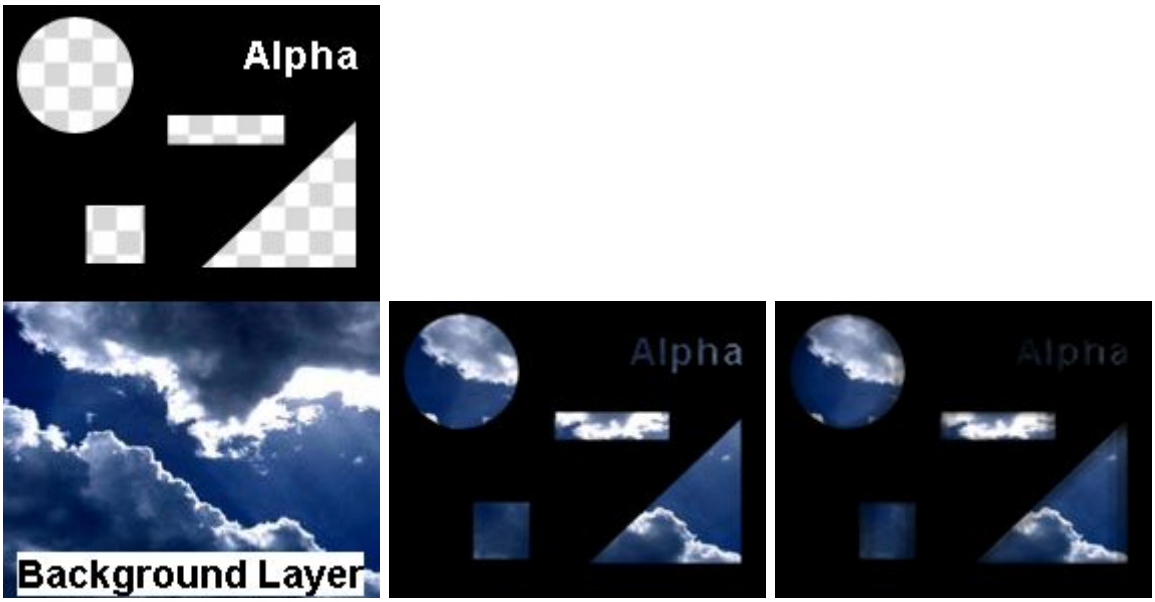


Overlays those pixels of the layer that are alpha (transparent) and adjacent to opaque (non see-through) pixels. The overlay line can be colored and blurred which creates the impression of a glowing line. You can adjust the amount as well es the X- and Y-offset of the blur.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	32
X	X Offset	0-255	255
Y	Y Offset	0-255	255
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128

Blur XY Alpha Glow Pre-Multiplied

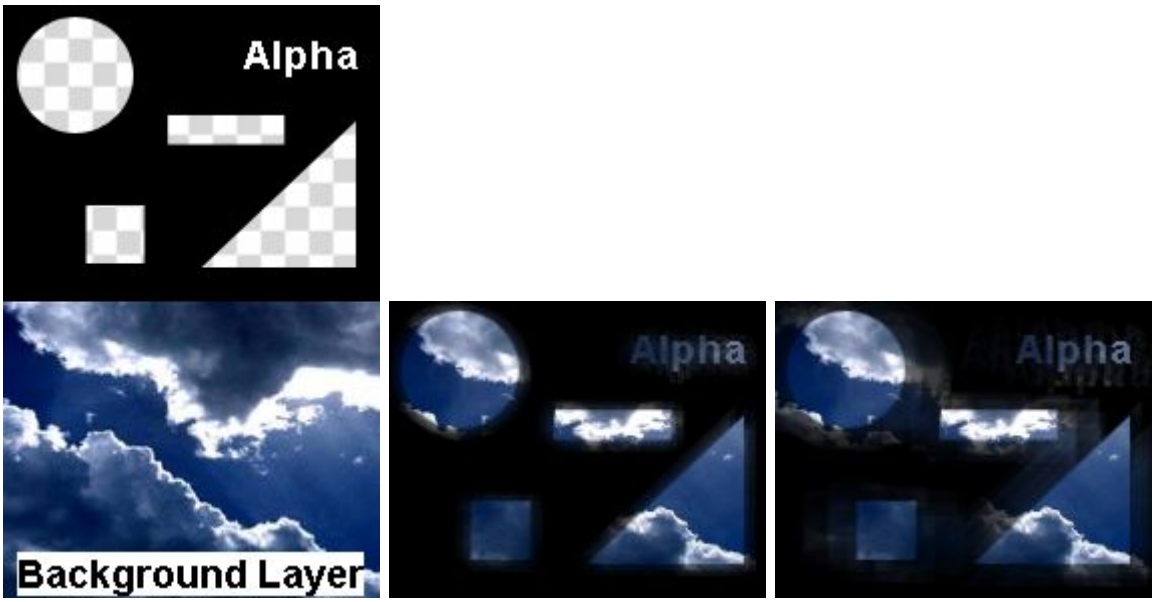


Overlays those pixels of the layer that are alpha (transparent) and adjacent to opaque (non see-through) pixels. In difference to the above effect "Blur XY Alpha Glow Color" 344 the overlay is simply black. You can adjust the amount as well es the X- and Y-offset of the blur.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine 162". The chapter "FX Order 325" explains how to drag effects in the Device Tree 173 tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	32
X	X Offset	0-255	255
Y	Y Offset	0-255	255

Blur XY Alpha



Blurs the alpha (transparent) areas of the layer, all opaque (non see-through) areas stay sharp. You can adjust the amount as well es the X- and Y-offset.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	132
X	X Offset	0-255	132
Y	Y Offset	0-255	122

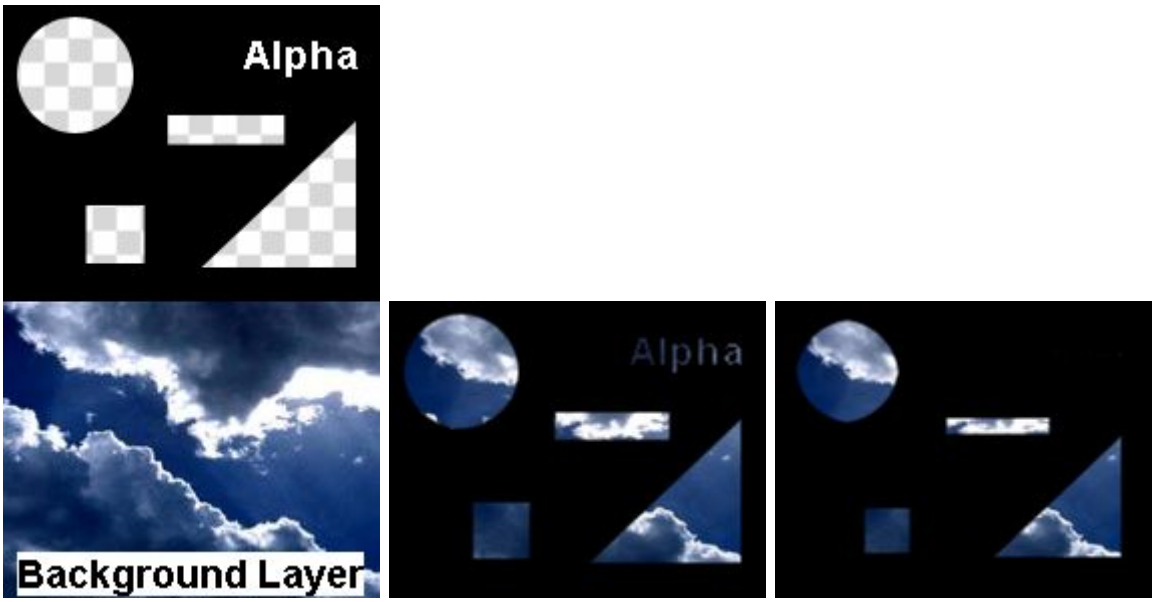
Blur XY

Blurs all RGBA pixels of the layer. You can adjust the amount as well es the X- and Y-offset.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of Blur	0-255	132
X	X Offset	0-255	132
Y	Y Offset	0-255	122

Dilate Alpha



This effect looks for opaque (non see-through) areas of the layer that are (partially) surrounded by alpha (transparent) areas. Then it enlarges the size of the opaque areas according to the "Amount" parameter by repeating the opaque pixels.
This effect might be of interest when working with a generated mask (e.g. keying a specific color) and there are small areas left that should be either transparent or opaque. This appears as so called noise. A combination of a dilate and erode effect can reduce the noise.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	132

Dilate

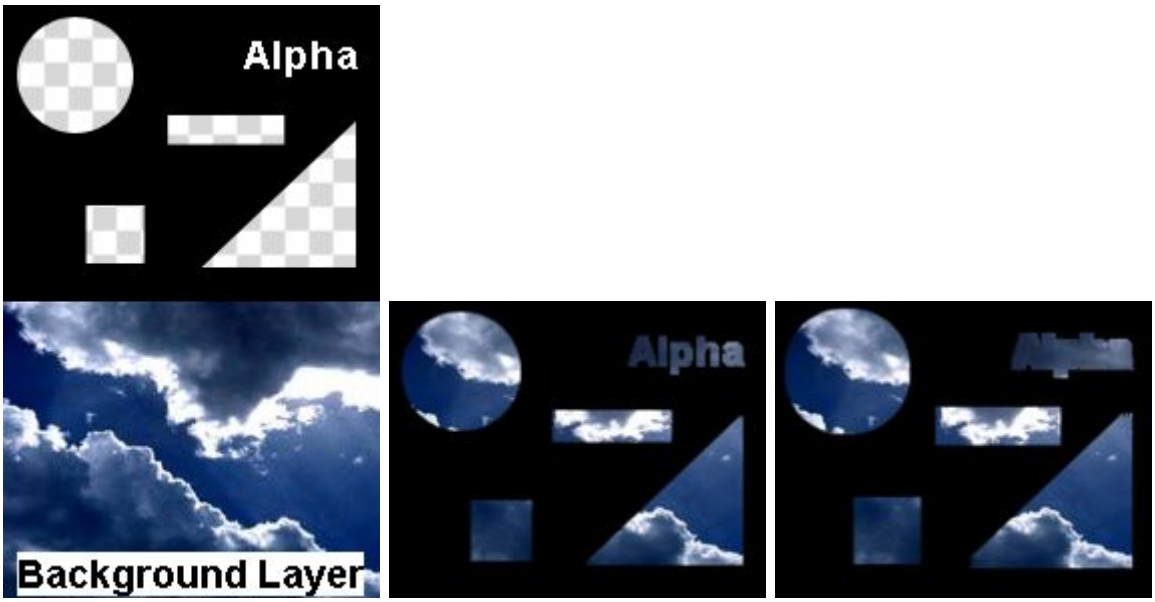
This effect looks for areas in the layer that are brighter than those areas that (partially) surrounded them. Then it enlarges their size according to the "Amount" parameter by repeating the brighter pixels.

This effect might be of interest when working with a generated mask (e.g. coloring regions of interest in black or white) and there are small areas left that should be the opposite color. This appears as so called noise. A combination of a dilate and erode effect can reduce the noise.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	132

Erode Alpha



This effect looks for opaque (non see-through) areas of the layer that are (partially) surrounded by alpha (transparent) areas. Then it reduces the size of the opaque areas according to the "Amount" parameter by repeating the adjacent transparent pixels.
This effect might be of interest when working with a generated mask (e.g. keying a specific color) and there are small areas left that should be either transparent or opaque. This appears as so called noise. A combination of a dilate and erode effect can reduce the noise.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	132

Erode

This effect looks for areas in the layer that are darker than those areas that (partially) surrounded them. Then it enlarges their size according to the "Amount" parameter by repeating the darker pixels.

This effect might be of interest when working with a generated mask (e.g. coloring regions of interest in black or white) and there are small areas left that should be either black or white. This appears as so called noise. A combination of a dilate and erode effect can reduce the noise.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine 162". The chapter "FX Order 325" explains how to drag effects in the Device Tree 173 tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	132

GaussianBlur On-Off

This is a switchable Gaussian Blur effect.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine 162".

Parameter	Description	Value Range	Default
On/Off	On/Off	-	-
Blur	Blur	0-100	0

GaussianBlur

This adds a Gaussian Blur effect to the layer.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Blur	Blur	0-100	0

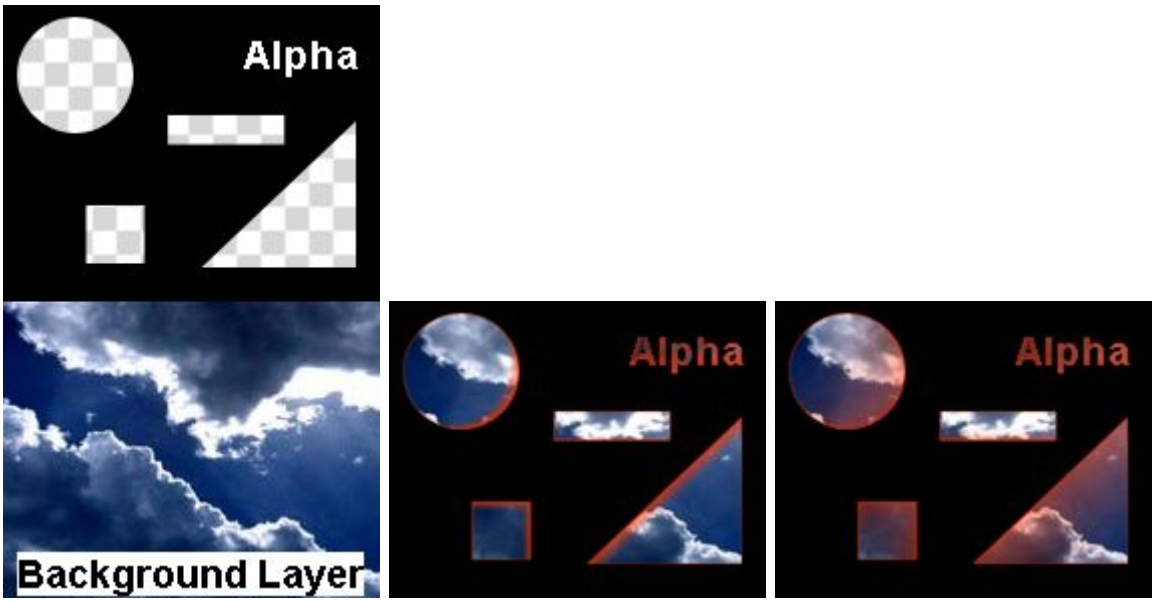
Glow

This effect looks for bright pixels in the layer texture and adds brightness to surrounding pixels, thus they appear to glow.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Luminance	Luminance	0-65535	60000
Mid	Mid	0-255	46
Cut-Off	Cut-Off	0-255	200
Bloom	Bloom	0-255	190
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255

Radial Blur Alpha Color

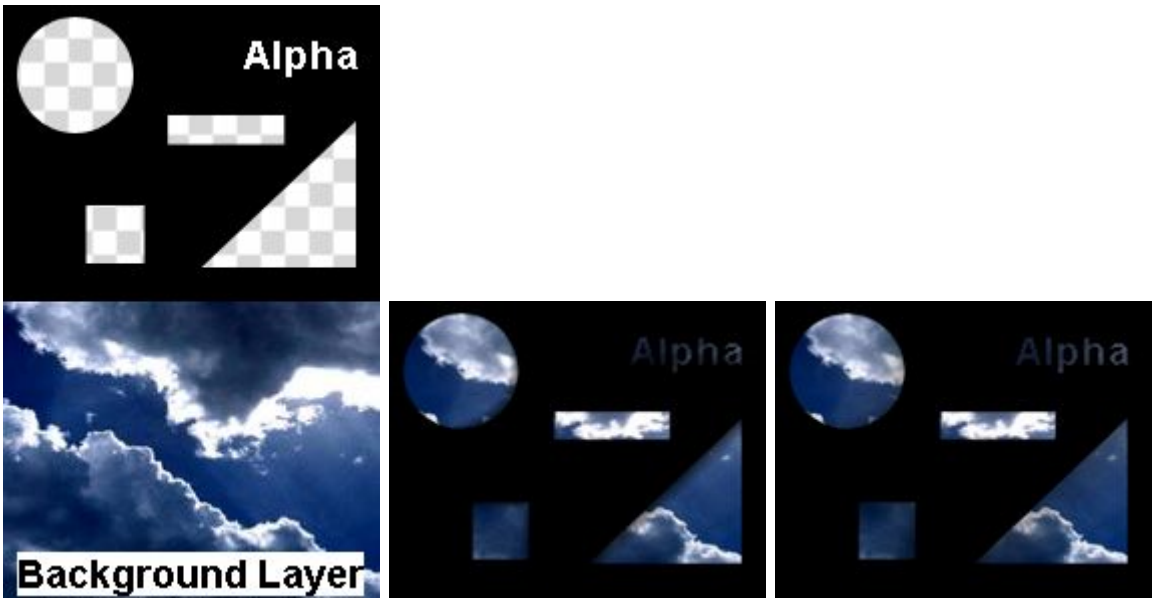


Overlays those pixels of the layer that are alpha (transparent) and adjacent to opaque (non see-through)pixels. The overlay line can be colored and blurred with a radial blur which creates the impression of a glowing shadow in one direction. You can adjust the amount as well es the X- and Y-direction and the intensity of the radial blur.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	64
X	X	0-65535	32768
Y	Y	0-65535	32768
Intensity	Intensity	0-255	128
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128

Radial Blur Alpha



Blurs the alpha (transparent) areas of the layer with a radial blur effect. All opaque (non see-through) areas stay sharp. You can adjust the amount as well es the X- and Y-direction and the intensity of the radial blur.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	64
X	X Direction	0-65535	32768
Y	Y Direction	0-65535	32768
Intensity	Intensity	0-255	128

Radial Blur

Blurs all RGBA pixels of the layer with a radial blur. You can adjust the amount as well es the X- and Y-direction and the intensity.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of Blur	0-255	64
X	X Position of Center Point	0-65535	32768
Y	Y Position of Center Point	0-65535	32768
Intensity	Amount of Blur	0-255	128

Refresh Delay

This adds a delay of one frame when applied to any Output device [682].
This is of interest when working with multiple displays that are mounted underneath each other and have a hard edge. With sensitive displays and sensitive content it can look better when the second signal is delayed by one frame which is the time it takes to process the first frame.

Note that this effect can not be applied twice to an Output to generate an delay of two frames!

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	255

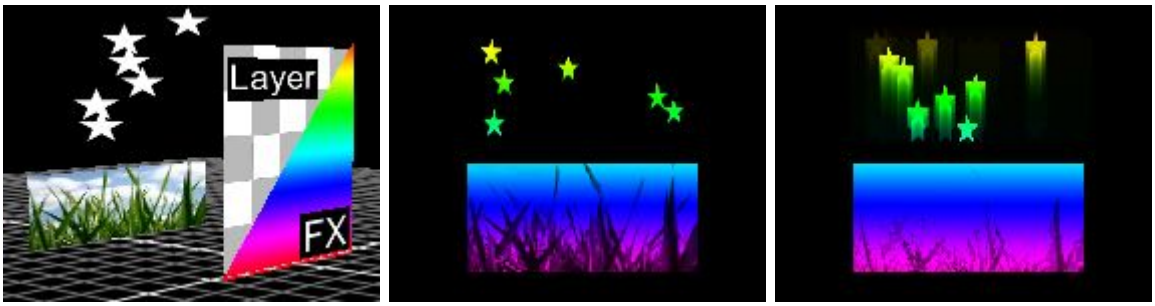
Render Delay

This effect is based on the Layer's Z Position or render history as explained in the manual chapter "Compositing" [406]. The layer with the assigned "Render Delay" effect makes a composition of all layers behind it. In the example, the "render" layer is positioned in the foreground, its main media is a transparent image. In the background there is a layer with a running timecode video and a particle system [185] (also with a transparent media!).
All background pixels are delayed depending on the effect parameter "Delay".

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Delay	Amount of Delay	0-65535	32768

Render Draw Media



This effect is based on the Layer's Z Position or render history as explained in the manual chapter "Compositing" [406]. The layer with the assigned "Render Draw Media" effect makes a composition of all layers behind it. In the example, the "render" layer is positioned in the foreground, its main media is a transparent image whilst the effect's media is a rainbow image. In the background there is a grass layer and a particle system [185] (also with a transparent media!).
Depending on the effect parameter "Clear" the background pixels add up to the foreground texture. E.g. the particle's movement draws onto the rainbow effect texture; the second image shows a low "Clear", the third an increased "Clear" time. Another drawing result could be achieved by moving the grass layer up and down or by using a video with moving pixels.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	255
Clear	Clear	0-65535	32768
Media	Media file as source for overlay or mask	-	-

Render Draw

This effect is based on the Layer's Z Position or render history as explained in the manual chapter "Compositing" [406]. The layer with the assigned "Render Draw Media" effect makes a composition of all layers behind it. In the example, the "render" layer is positioned in the foreground, its main media is a transparent image. In the background there is a grass layer and a particle system [185] (also with a transparent media!). Depending on the effect parameter "Clear" the background pixels add up to the foreground texture. The particle's movement draws onto the transparent layer texture; the second image shows a low "Clear", the third an increased "Clear" time. Another drawing result could be achieved by moving the grass layer up and down or by using a video with moving pixels.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Clear	Clears drawn areas	0-65535	32768

Video Delay

Delays a Video File as if it blends over several frames.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Delay	Amount of delay	0-65535	60000

6.5.2.4.4 Camera Projection

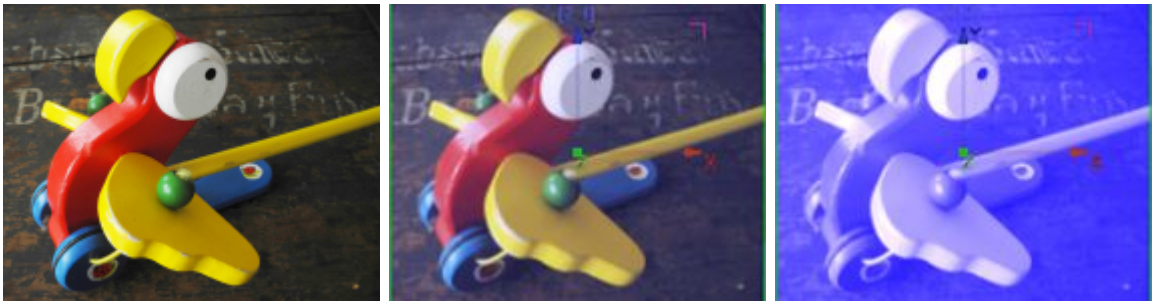
Perspective Projection Mapping

This effect is the only one that can (and must) be applied to a Camera layer [675]. It allows special tracking setups, e.g. when a camera movement is tracked for a real/virtual TV studio setup. According to the (real) movement of the camera Pandoras Box renders a cut-out of a virtual space which is then displayed as a background in the (real) setup. Therefore it is seen by the (real) camera which results in a recorded mix of real and virtual elements. Only from the real camera's viewpoint the background is in the correct perspective. This setup is an alternative to keying applications with a green or blue screen.
The tracking data changes the parameters of a PB camera layer. The effect parameters on the other hand are static, i.e they are setup once as fixed values. The four coordinates represent the corners of the screen whereon the background is projected and can also be understood as a "window" through which the (PB) camera sees and renders the background.
An example for this effect is the setup of the coolux booth on NAB 2013 seen here:
https://www.youtube.com/watch?v=7ectw1Z9BvA

Parameter	Description	Value Range	Default
GenPerspTargetPt1X	GenPerspTargetPt1X	-999.999 - +999.999	-8
GenPerspTargetPt1Y	GenPerspTargetPt1Y	-999.999 - +999.999	-5
GenPerspTargetPt1Z	GenPerspTargetPt1Z	-999.999 - +999.999	0
GenPerspTargetPt2X	GenPerspTargetPt2X	-999.999 - +999.999	8
GenPerspTargetPt2Y	GenPerspTargetPt2Y	-999.999 - +999.999	-5
GenPerspTargetPt2Z	GenPerspTargetPt2Z	-999.999 - +999.999	0
GenPerspTargetPt3X	GenPerspTargetPt3X	-999.999 - +999.999	-8
GenPerspTargetPt3Y	GenPerspTargetPt3Y	-999.999 - +999.999	5
GenPerspTargetPt3Z	GenPerspTargetPt3Z	-999.999 - +999.999	0

6.5.2.4.5 Color Effects

B&W Add



Adds Black and White values to the RGB colors, the result can be inverted.
Uses the color picker, with saturation set to 0 by default.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Color Picker	0-255	128
Green	Color Picker	0-255	128
Blue	Color Picker	0-255	128
Invert	Inverts the RGB levels	0-255	0

B&W Multiply

All RGB colors are multiplied by B&W color mix value, the result can be inverted.
Uses the color picker, with saturation set to 0 by default.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Color Picker	0-255	128
Green	Color Picker	0-255	128
Blue	Color Picker	0-255	128
Factor	Color Multiply Factor	0-255	64
Invert	Inverts the RGB levels	0-255	0

B&W

Turns the RGB image into Black and White, the result can be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Invert	Inverts the RGB Levels	0-255	128

BiTone

Turns the RGB image into just two colors, using Threshold. Both colors can be set by color pickers.

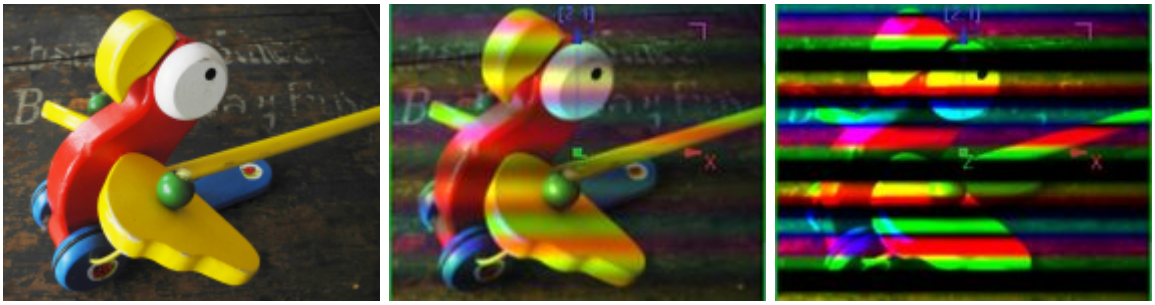
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red 1	Color Picker for Color 1	0-255	255
Green 1	Color Picker for Color 1	0-255	255
Blue 1	Color Picker for Color 1	0-255	255
Red 2	Color Picker for Color 2	0-255	0
Green 2	Color Picker for Color 2	0-255	0
Blue 2	Color Picker for Color 2	0-255	0
Threshold	Defines the Threshold	0-255	128

BiTone Luma

Colorizes the image based upon its luminance values in two colors, each for high (H) and low (L) levels, using color pickers.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Color Picker for high luminance value	0-255	128
Green H	Color Picker for high luminance value	0-255	128
Blue H	Color Picker for high luminance value	0-255	128
Red L	Color Picker for low luminance value	0-255	128
Green L	Color Picker for low luminance value	0-255	128
Blue L	Color Picker for low luminance value	0-255	128

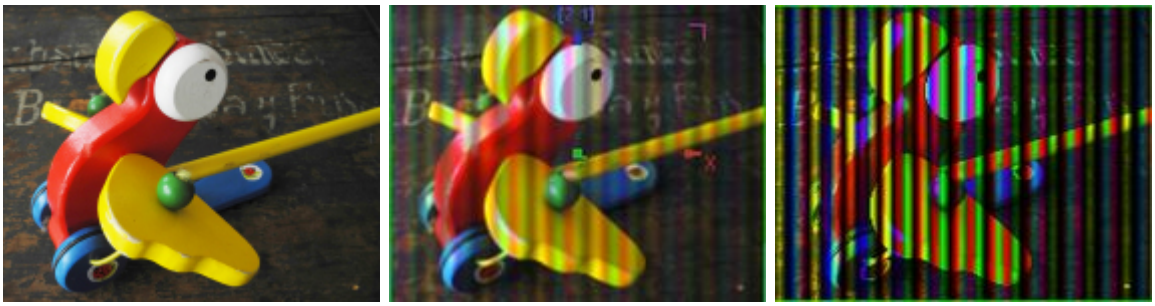
Color Stripes H



Adds horizontal colored stripes.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-65535	1024

Color Stripes V



Adds vertical colored stripes.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-65535	1024

ColorFade

Fades all RGB colors to full RGB or CMY color saturation.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
R / C	Fades into Red (up) / Cyan (down)	0-255	128
G / M	Fades into Green (up) / Magenta (down)	0-255	128
B / Y	Fades into Blue (up) / Yellow (down)	0-255	128

ColorScroll Add

Applies a color-changing effect to the layer texture by adding the RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0

ColorScroll Multiply

Applies a color-changing effect to the layer texture by multiplying the RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0

ColorScroll Only

Applies a color-changing effect to the layer texture by overlaying the original RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0

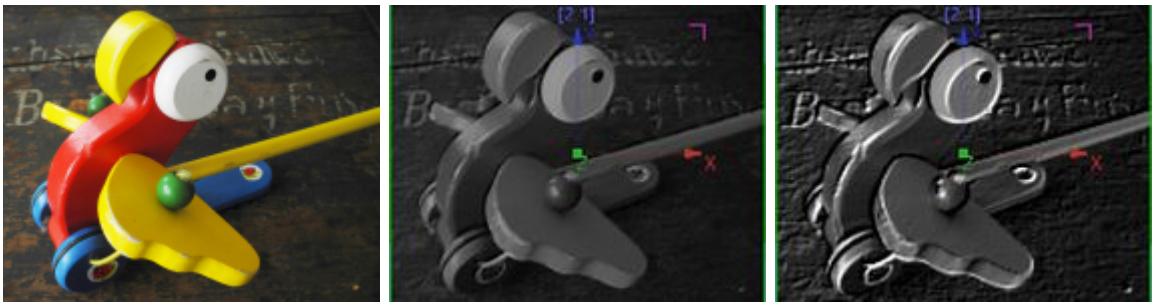
Emboss Angle

Emboss with control for Factor and Angle.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Factor	Amount of Emboss	0-255	128
Angle	Angle of Emboss	0-255	128

Emboss Edges

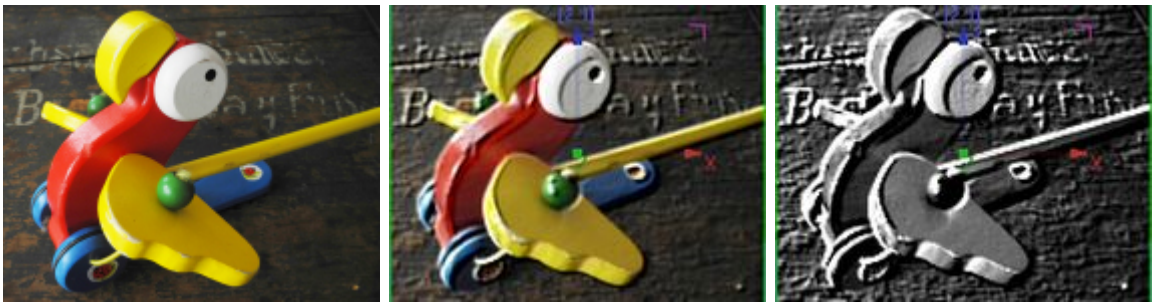


Emboss with control for Threshold.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Threshold	Level of Threshold	0-255	128

Emboss



Emboss with control for Threshold.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine 162". The chapter "FX Order 325" explains how to drag effects in the Device Tree 173 tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Threshold	Level of Threshold	0-255	144
Factor	Factor of Emboss	0-255	196

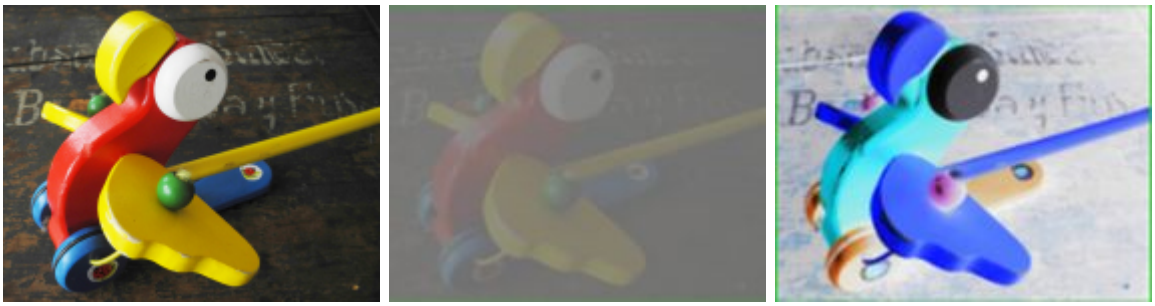
Granite

Granite Effect.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine 162". The chapter "FX Order 325" explains how to drag effects in the Device Tree 173 tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Level of Granite pattern	0-255	255

Invert



Inverts the RGB levels.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

LinearLight

Linear light reflexion over the whole image.

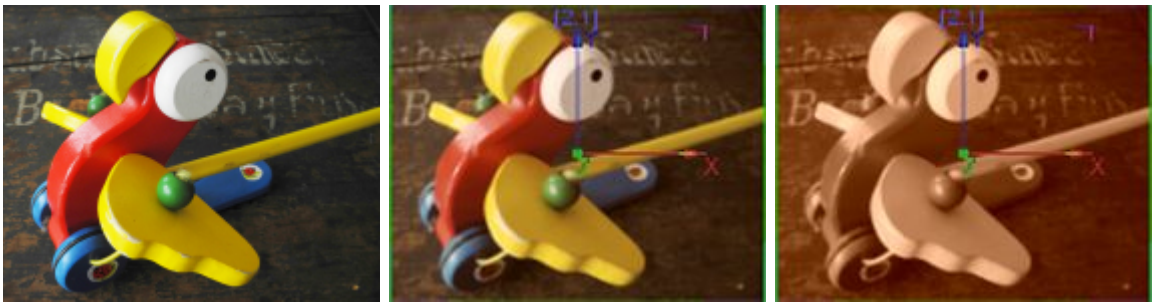
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

RGB Inversion

Inverts the RGB levels with the possibility of setting RGB levels, Threshold and Radius.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Threshold	Threshold	0-255	255
Radius	Radius	0-255	128
Invert	Inverts the applied effect.	0-255	0

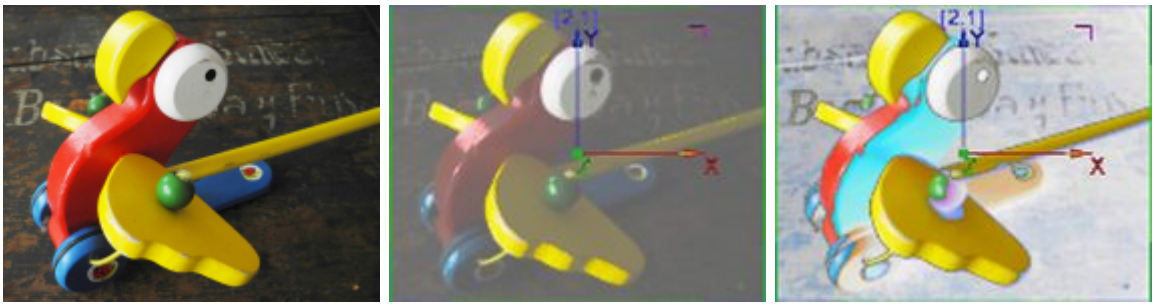
Sepiatone



Tints the texture in sepia tones, amount is adjustable.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of sepia tone	0-255	128

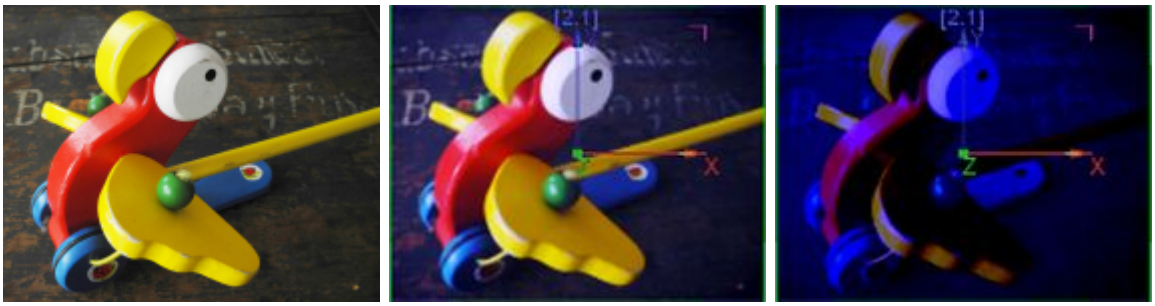
Solarize



Solarize inverts first dark then brighter colors.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of solarization	0-255	128

Tone Mapping



Allows to set Fog, Exposure, Gamma, Vignette and Blueshift to create specific photo style.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Defog	Defog	0-255	2
Fog Red	Fog Red	0-255	255
Fog Green	Fog Green	0-255	255
Fog Blue	Fog Blue	0-255	255
Exposure	Exposure	0-255	144
Gamma	Gamma	0-255	32
VignetteCenter X	VignetteCenter X	0-65535	32768
VignetteCenter Y	VignetteCenter Y	0-65535	32768
VignetteRadius	VignetteRadius	0-65535	65535
VignetteAmount	VignetteAmount	0-65535	64
BlueShift	BlueShift	0-255	0

TriTone

Colorizes the image based upon its luminance values in a mix of three colors, each for high (H), middle (M) and low (L) levels, using color pickers.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Color Picker for high luminance value	0-255	128
Green H	Color Picker for high luminance value	0-255	128
Blue H	Color Picker for high luminance value	0-255	128
Red M	Color Picker for middle luminance value	0-255	128
Green M	Color Picker for middle luminance value	0-255	128
Blue M	Color Picker for middle luminance value	0-255	128
Red L	Color Picker for low luminance value	0-255	128
Green L	Color Picker for low luminance value	0-255	128
Blue L	Color Picker for low luminance value	0-255	128

6.5.2.4.6 Color Effects - Crop

B&W Add Crop Pixel

Applies the "B&W Add" FX within an adjustable rectangular area with hard edges. It adds Black and White values to the RGB colors, the result can be inverted.
Uses the color picker, with saturation set to 0 by default.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

B&W Crop Pixel

Applies the "B&W" FX within an adjustable rectangular area with hard edges. It turns the RGB image into Black and White, the result can be inverted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Invert	Inverts the RGB levels or the effect	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

B&W Multiply Crop Pixel

Applies the "B&W Multiply" FX within an adjustable rectangular area with hard edges. All RGB colors are multiplied by B&W color mix value, the result can be inverted.
Uses the color picker, with saturation set to 0 by default.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128
Factor	Factor	0-255	64
Invert	Inverts the RGB levels or the effect	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

BiTone Crop Pixel

Applies the "Bitone" FX within an adjustable rectangular area with hard edges. It turns the RGB image into just two colors, using Threshold. Both colors can be set by color pickers.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red 1	First color picker / Level of red	0-255	255
Green 1	First color picker / Level of green	0-255	255
Blue 1	First color picker / Level of blue	0-255	255
Red 2	Second color picker / Level of red	0-255	0
Green 2	Second color picker / Level of green	0-255	0
Blue 2	Second color picker / Level of blue	0-255	0
Threshold	Threshold	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

BiTone Luma Crop Pixel

Applies the "BiTone Luma" FX within an adjustable rectangular area with hard edges. It colorizes the image based upon its luminance values in two colors, each for high (H) and low (L) levels, using color pickers.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Color Stripes H Crop Pixel

Applies the "Color Stripes H" FX within an adjustable rectangular area with hard edges. It adds horizontal color stripes in different sizes.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-65535	1024
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Color Stripes V Crop Pixel

Applies the "Color Stripes V" FX within an adjustable rectangular area with hard edges. It adds vertical color stripes in different sizes.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-65535	1024
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Color Fade Crop Pixel

Applies the "Color Fade" FX within an adjustable rectangular area with hard edges. It fades all RGB colors to full RGB or CMY color saturation.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
R/C	R/C	0-255	128
G/M	G/M	0-255	128
B/Y	B/Y	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

ColorScroll Add Crop Pixel

Applies the "ColorScroll Add" FX within an adjustable rectangular area with hard edges. It applies a color-changing effect to the layer texture by adding the RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0
Bottom (px)	Bottom (px)	0-8192	0

ColorScroll Multiply Crop Pixel

Applies the "ColorScroll Multiply" FX within an adjustable rectangular area with hard edges. It applies a color-changing effect to the layer texture by multiplying the RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

ColorScroll Only Crop Pixel

Applies the "ColorScroll Only" FX within an adjustable rectangular area with hard edges. It applies a color-changing effect to the layer texture by overlaying the original RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Emboss Angle Crop Pixel

Applies an emboss effect with different angles in an adjustable rectangular area with hard edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Factor	Factor	0-255	128
Orientation	Orientation	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Emboss Crop Pixel

Applies an emboss effect in an adjustable rectangular area with hard edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Threshold	Threshold	0-255	144
Factor	Factor	0-255	196
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Emboss Edges Crop Pixel

Applies an emboss effect with a threshold in an adjustable rectangular area with hard edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Threshold	Threshold	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Granite Crop Pixel

Applies the "Granite" FX within an adjustable rectangular area with hard edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	255
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Invert Crop Pixel

Applies the "Invert" FX within an adjustable rectangular area with hard edges. It inverts the RGB levels.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Linear Light Crop Pixel

Applies the "Linear Light" FX within an adjustable rectangular area with hard edges.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB Inversion Crop Pixel

Applies the "RGB Inversion" FX within an adjustable rectangular area with hard edges. It inverts the RGB levels with the possibility of setting RGB levels, Threshold and Radius.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128
Threshold	Threshold	0-255	255
Radius	Radius	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Sepiatone Crop Pixel

Applies the "Sepiatone" FX within an adjustable rectangular area with hard edges. It tints the texture in sepia tones, amount is adjustable.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Solarize Crop Pixel

Applies the "Solarize" FX within an adjustable rectangular area with hard edges. It inverts first dark then brighter colors.

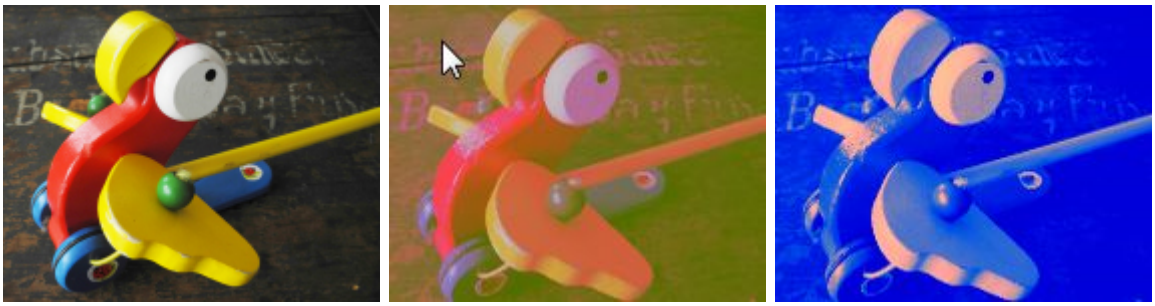
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Tone Mapping Crop Pixel

Applies the "Tone Mapping" FX within an adjustable rectangular area with hard edges. It allows to set Fog, Exposure, Gamma, Vignette and Blueshift to create a specific photo style.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Defog	Defog	0-255	2
Fog Red	Fog Red	0-255	255
Fog Green	Fog Green	0-255	255
Fog Blue	Fog Blue	0-255	255
Exposure	Exposure	0-255	144
Gamma	Gamma	0-255	32
VignetteCenterX	VignetteCenterX	0-65535	32768
VignetteCenterY	VignetteCenterY	0-65535	32768
VignetteRadius	VignetteRadius	0-65535	65535
VignetteAmount	VignetteAmount	0-255	0
BlueShift	BlueShift	0-255	64
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

TriTone Crop Pixel



Applies the "TriTone" FX within an adjustable rectangular area with hard edges. It colorizes the image based upon its luminance values in a mix of three colors, each for high (H), middle (M) and low (L) levels, using color pickers.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red M	Red M	0-255	128
Green M	Green M	0-255	128
Blue M	Blue M	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

6.5.2.4.7 Color Effects - Softborder

B&W Add Softborder

Applies the "B&W Add" FX within an adjustable rectangular area with soft edges. It adds Black and White values to the RGB colors, the result can be inverted. Uses the color picker, with saturation set to 0 by default.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

B&W Multiply Softborder

Applies the "B&W Multiply" FX within an adjustable rectangular area with soft edges. All RGB colors are multiplied by B&W color mix value, the result can be inverted.

Uses the color picker, with saturation set to 0 by default.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128
Factor	Factor	0-255	64
Invert	Inverts the RGB levels or the effect	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

B&W Softborder

Applies the "B&W" FX within an adjustable rectangular area with soft edges. It turns the RGB image into Black and White, the result can be inverted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Invert	Inverts the RGB levels or the effect	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

BiTone Softborder

Applies the "Bitone" FX within an adjustable rectangular area with soft edges. It turns the RGB image into just two colors, using Threshold. Both colors can be set by color pickers.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red 1	First color picker / Level of red	0-255	255
Green 1	First color picker / Level of green	0-255	255
Blue 1	First color picker / Level of blue	0-255	255
Red 2	Second color picker / Level of red	0-255	0
Green 2	Second color picker / Level of green	0-255	0
Blue 2	Second color picker / Level of blue	0-255	0
Threshold	Threshold	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

BiTone Luma Softborder

Applies the "BiTone Luma" FX within an adjustable rectangular area with soft edges. It colorizes the image based upon its luminance values in two colors, each for high (H) and low (L) levels, using color pickers.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

Color Stripes H Softborder

Applies the "Color Stripes H" FX within an adjustable rectangular area with soft edges. It adds horizontal color stripes in different sizes.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-65535	1024
Size	Size	0-255	16
Corner	Corner	0-255	0

Color Stripes V Softborder

Applies the "Color Stripes V" FX within an adjustable rectangular area with soft edges. It adds vertical color stripes in different sizes.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-65535	1024
Size	Size	0-255	16
Corner	Corner	0-255	0

Color Fade Softborder

Applies the "Color Fade" FX within an adjustable rectangular area with soft edges. It fades all RGB colors to full RGB or CMY color saturation.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
R/C	R/C	0-255	128
G/M	G/M	0-255	128
B/Y	B/Y	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

ColorScroll Add Softborder

Applies the "ColorScroll Add" FX within an adjustable rectangular area with soft edges. It applies a color-changing effect to the layer texture by adding the RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0
Corner	Corner	0-255	0

ColorScroll Multiply Softborder

Applies the "ColorScroll Multiply" FX within an adjustable rectangular area with soft edges. It applies a color-changing effect to the layer texture by multiplying the RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0
Corner	Corner	0-255	0

ColorScroll Only Softborder

Applies the "ColorScroll Only" FX within an adjustable rectangular area with soft edges. It applies a color-changing effect to the layer texture by overlaying the original RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Scroll Speed	Scroll Speed	0-65535	0
Corner	Corner	0-255	0

Emboss Angle Softborder

Applies an emboss effect with different angles in an adjustable rectangular area with soft edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Factor	Factor	0-255	128
Orientation	Orientation	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

Emboss Edges Softborder

Applies an emboss effect with a threshold in an adjustable rectangular area with soft edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Threshold	Threshold	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

Emboss Softborder

Applies an emboss effect in an adjustable rectangular area with soft edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Threshold	Threshold	0-255	144
Factor	Factor	0-255	196
Size	Size	0-255	16
Corner	Corner	0-255	0

Invert Softborder

Applies the "Invert" FX within an adjustable rectangular area with soft edges. It inverts the RGB levels.

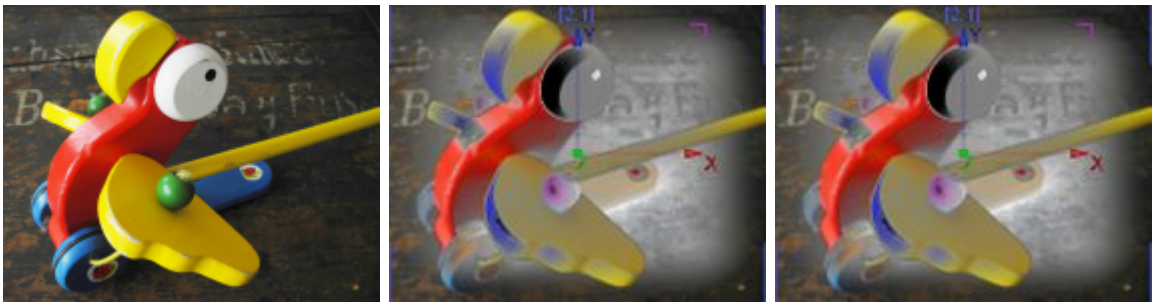
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

Linear Light Softborder

Applies the "Linear Light" FX within an adjustable rectangular area with soft edges.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB Inversion Softborder



Applies the "RGB Inversion" FX within an adjustable rectangular area with soft edges. It inverts the RGB levels with the possibility of setting RGB levels, Threshold and Radius.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	128
Blue	Color picker / Level of blue	0-255	128
Threshold	Threshold	0-255	255
Radius	Radius	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

Sepiatone Softborder

Applies the "Sepiatone" FX within an adjustable rectangular area with soft edges. It tints the texture in sepia tones, amount is adjustable.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

Tone Mapping Softborder

Applies the "Tone Mapping" FX within an adjustable rectangular area with soft edges. It allows to set Fog, Exposure, Gamma, Vignette and Blueshift to create a specific photo style.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Defog	Defog	0-255	2
Fog Red	Fog Red	0-255	255
Fog Green	Fog Green	0-255	255
Fog Blue	Fog Blue	0-255	255
Exposure	Exposure	0-255	144
Gamma	Gamma	0-255	32
VignetteCenterX	VignetteCenterX	0-65535	32768
VignetteCenterY	VignetteCenterY	0-65535	32768
VignetteRadius	VignetteRadius	0-65535	65535
VignetteAmount	VignetteAmount	0-255	0
BlueShift	BlueShift	0-255	64
Size	Size	0-255	16
Corner	Corner	0-255	0

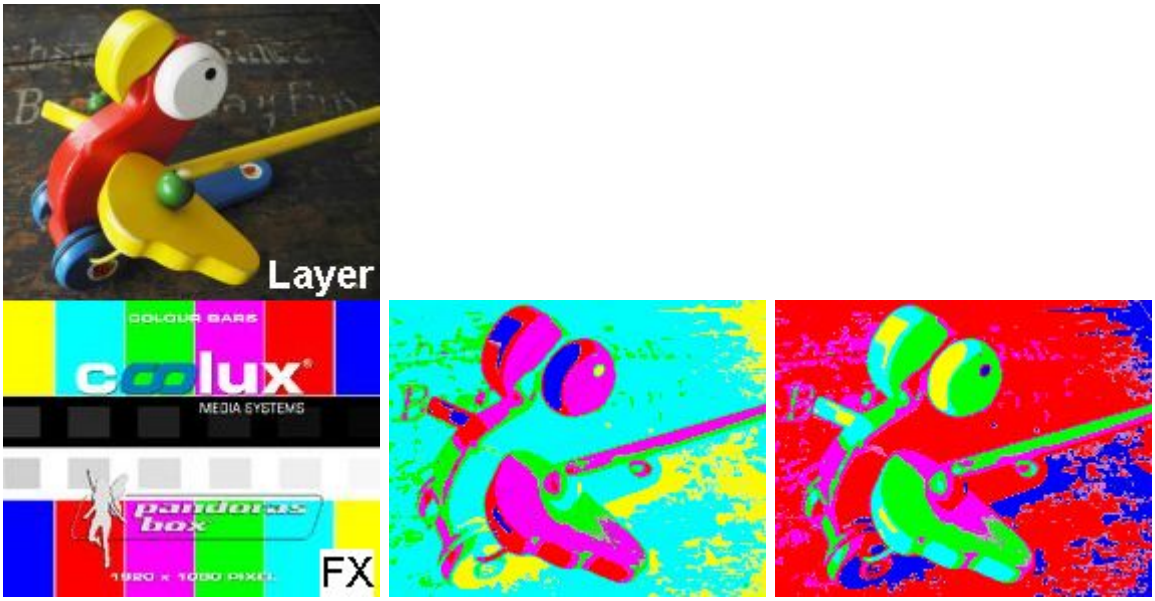
TriTone Softborder

Applies the "TriTone" FX within an adjustable rectangular area with soft edges. It colorizes the image based upon its luminance values in a mix of three colors, each for high (H), middle (M) and low (L) levels, using color pickers.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red M	Red M	0-255	128
Green M	Green M	0-255	128
Blue M	Blue M	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

6.5.2.4.8 Color Shift

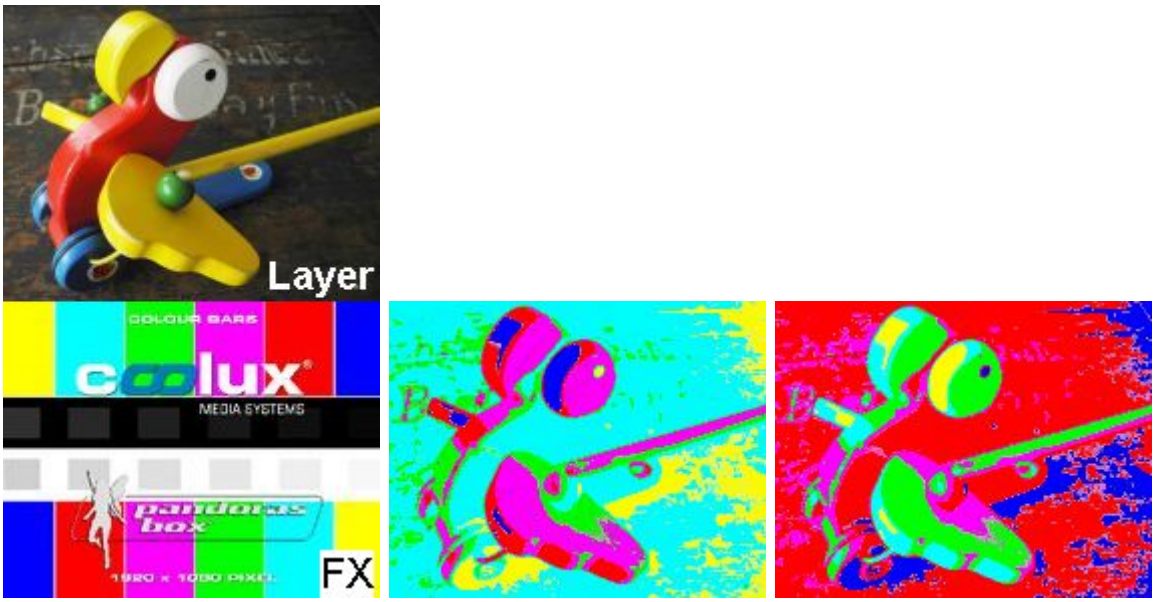
BW Media ColorShift Scroll



This effect mixes the layer texture with color values of (one horizontal line of) the effect media. The overlaid color is taken from the very left position for black values and moves further to the right the brighter the original pixel is. In the middle image the "Line" parameter is set to "0", so only the colors from the top line are taken. Dark original pixels are now yellow or cyan, the most brightest ones are red or even blue. For the last image the "Line" parameter is set to "255", so the bottom line from the FX media is applied.
With the "Scroll" parameter the overlaid line scrolls with the according speed from top to bottom; press "Reset" to match "0" to the top line again.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Line	Line	0-255	0
Scroll	Scroll	0-65535	0
Media	Media file as source for overlay or mask	-	-

BW Media ColorShift



This effect mixes the layer texture with color values of (one horizontal line of) the effect media. The overlaid color is taken from the very left position for black values and moves further to the right the brighter the original pixel is. In the middle image the "Line" parameter is set to "0", so only the colors from the top line are taken. Dark original pixels are now yellow or cyan, the most brightest ones are red or even blue. For the last image the "Line" parameter is set to "255", so the bottom line from the FX media is applied.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Line	Line	0-255	0
Media	Media file as source for overlay or mask	-	-

Color Inversion

Inverts the image based upon its RGB values and allows to influence the color, threshold and feather.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Threshold	Threshold	0-255	255
Feather	Feather	0-255	128

ColorOnly

Turns the image into a single color and allows to set an alpha value to define the transparency of the layer.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Alpha	Set the transparency of the layer.	0-255	128

RGB Add

Adds RGB colors by the RGB color mix value, allows inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Invert	Inverts the RGB Levels	0-255	0

RGB Channel Inversion

Allows to invert the image channel by channel (RGB & alpha)

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
R	Red level inversion	0-255	0
G	Green level inversion	0-255	0
B	Blue level inversion	0-255	0
A	Alpha level inversion	0-255	0

RGB Hue

Changes all color values of the layer texture to the chosen hue / color. Black and white pixels are not influenced.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0

RGB INV Add

Allows adding RGB colors by the color mix values after inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	0
Green	Green level, Color Picker	0-255	0
Blue	Blue level, Color Picker	0-255	0

RGB INV Multiply

Allows multiplying with RGB colors by the color mix values after inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255

RGB Multiply

All RGB colors are multiplied by RGB color mix values. Multiply can be factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Factor	Color Multiply Factor	0-255	64
Invert	Inverts the RGB levels	0-255	0

RGB Replace

With this effect you can pick one color that should be replaced and you can fine tune the color range with the "Threshold" and "Radius" parameters. Then pick a new color that replaces it.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Old	Red Old	0-255	0
Green Old	Green Old	0-255	0
Blue Old	Blue Old	0-255	0
Threshold	Threshold	0-255	64
Radius	Radius	0-255	32
Red New	Red New	0-255	0
Green New	Green New	0-255	0
Blue New	Blue New	0-255	0

RGB Screen

This applies the screen blend mode, i.e. it negates the RGB values of the layer texture, then multiplying them and negates them again. The result is a brighter image.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

RGB Shift

With this effect you can tint your image with a certain color. In detail, the original RGB color values are increased with the chosen RGB value, as long as their according RGB share is grater than 0.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255

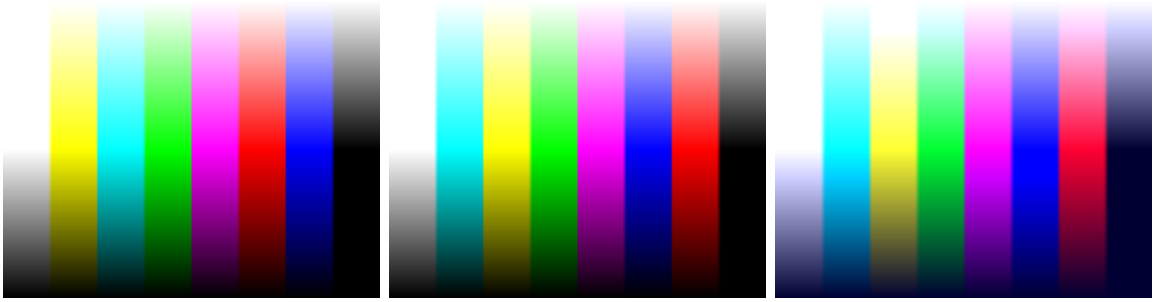
RGB XY Shift

Allows shifting all RGB colors in X and Y direction.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
R X	Red level shift in X direction	0-65535	32768
R Y	Red level shift in Y direction	0-65535	32768
G X	Green level shift in X direction	0-65535	32768
G Y	Green level shift in Y direction	0-65535	32768
B X	Blue level shift in X direction	0-65535	32768
B Y	Blue level shift in Y direction	0-65535	32768

RGB-BGR Add

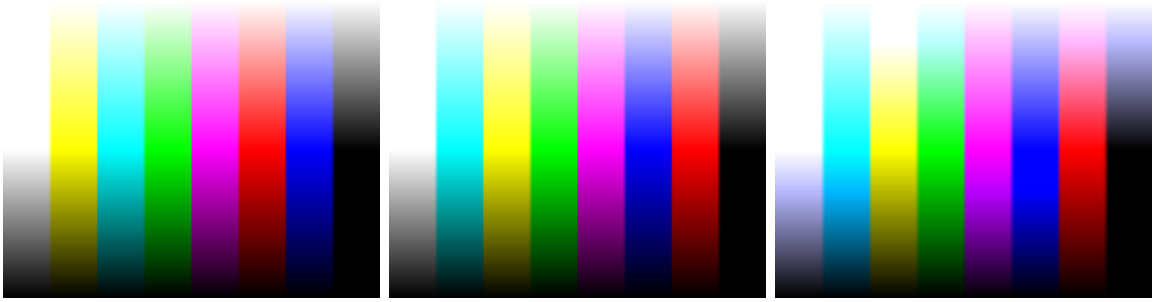


This effect shifts the RGB channels as follows:
Red -> Blue
Green stays
Blue -> Red

Then, the color mix value can be added and the colors may be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Invert	Inverts the result	0-255	0

RGB-BGR Multiply



This effect shifts the RGB channels as follows:
Red -> Blue
Green stays
Blue -> Red

Then, the channels can be multiplied with the color mix value, factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Factor	Factor of Multiplication	0-255	64
Invert	Inverts the result	0-255	0

RGB-BRG Add

This effect shifts the RGB channels as follows:
Red -> Blue
Green -> Red
Blue -> Green

Then, the color mix value can be added and the colors may be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Invert	Inverts the result	0-255	0

RGB-BRG Multiply

This effect shifts the RGB channels as follows:
Red -> Blue
Green -> Red
Blue -> Green

Then, the channels can be multiplied with the color mix value, factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Factor	Factor of Multiplication	0-255	64
Invert	Inverts the result	0-255	0

RGB-GBR Add

This effect shifts the RGB channels as follows:
Red -> Green
Green -> Blue
Blue -> Red

Then, the color mix value can be added and the colors may be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Invert	Inverts the result	0-255	0

RGB-GBR Multiply

This effect shifts the RGB channels as follows:
Red -> Green
Green -> Blue
Blue -> Red

Then, the channels can be multiplied with the color mix value, factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Factor	Factor of Multiplication	0-255	64
Invert	Inverts the result	0-255	0

RGB-GRB Add

This effect shifts the RGB channels as follows:
Red -> Green
Green -> Red
Blue stays

Then, the color mix value can be added and the colors may be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Invert	Inverts the result	0-255	0

RGB-GRB Multiply

This effect shifts the RGB channels as follows:
Red -> Green
Green -> Red
Blue stays

Then, the channels can be multiplied with the color mix value, factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Factor	Factor of Multiplication	0-255	64
Invert	Inverts the result	0-255	0

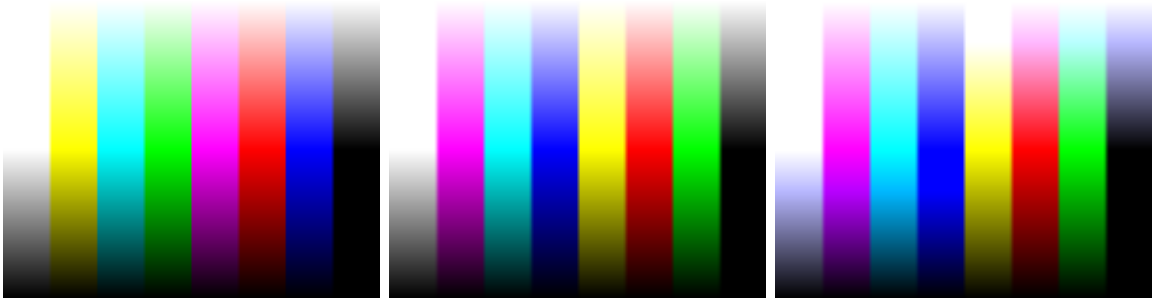
RGB-RBG Add

This effect shifts the RGB channels as follows:
Red stays
Green -> Blue
Blue -> Green

Then, the color mix value can be added and the colors may be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Invert	Inverts the result	0-255	0

RGB-RBG Multiply



This effect shifts the RGB channels as follows:
Red stays
Green -> Blue
Blue -> Green

Then, the channels can be multiplied with the color mix value, factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Factor	Factor of Multiplication	0-255	64
Invert	Inverts the result	0-255	0

RGBA Modulo

Modulo on RGB channels.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Alpha	Alpha level	0-255	255

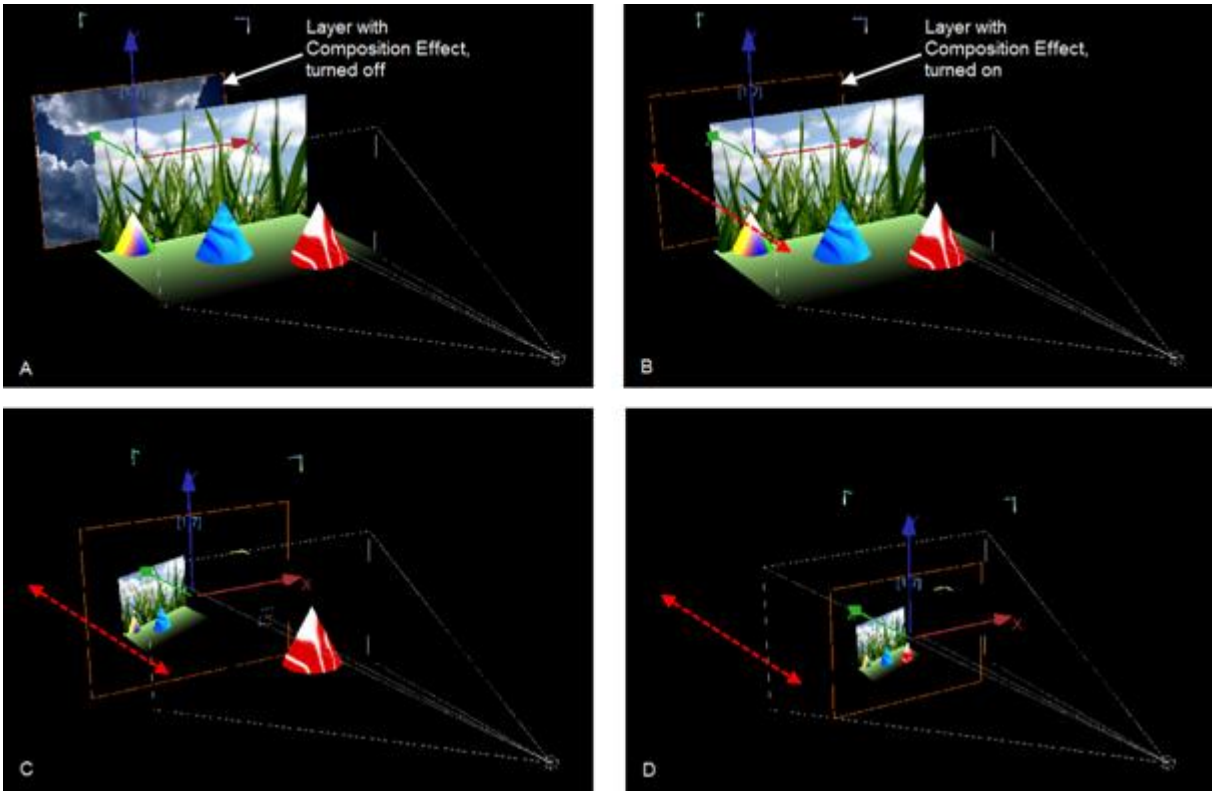
6.5.2.4.9 Compositing

++ General information for Compositing Effects ++

The Compositing Effects allow rendering a whole composition onto one layer, this is the so called Renderhistory. This way the composition can e.g. be scaled or positioned somewhere else on the output without the need of scaling or positioning every single layer.

Please note:
Everything in the 3D space that gets behind the layer with the composition effect will be rendered onto this layer. In doing so the composition does not longer exist as 3D composition, but is transferred into a flat 2D texture.

The example below shows how the Compositing Effects works.



A: There is a layer with grass on it, a bottom layer and three cones. Each item stays in a different Z position. The Compositing Effect will be applied to the layer in the background (with clouds on it).

B: The Compositing Effect is applied to the background layer. The layer now shows black, because there is nothing behind it that could be rendered.

C: The background layer was moved in Z position. It is now in front of the grass, the bottom layer and the first two cones, but behind the red cone. Everything except the red cone is now rendered as texture on the background layer.

D: The background layer is moved further to the front. Every item of the composition is now behind it and will be rendered as its texture.

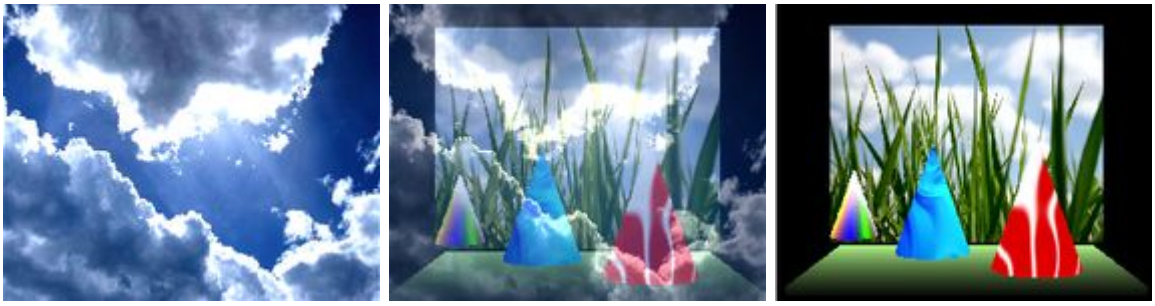
When looking at the scene above by previewing the Output (and not in Global Cam Mode), you will see this result:

E: When seen from the front side, the composition looks like in the beginning.
F: When scaling and rotating the background layer, it is now visible, that the composition is turned from a 3D composition to just a flat texture on the layer, from now on called "Rendering".

The difference between the single Compositing Effects is the way the composition behind the layer is rendered and mixed with the texture being originally on the layer.

The three images for each Compositing Effect show:
Left: Effect's Mix Value=0; Center: Effect's Mix Value =128; Right: Effect's Mix Value =255

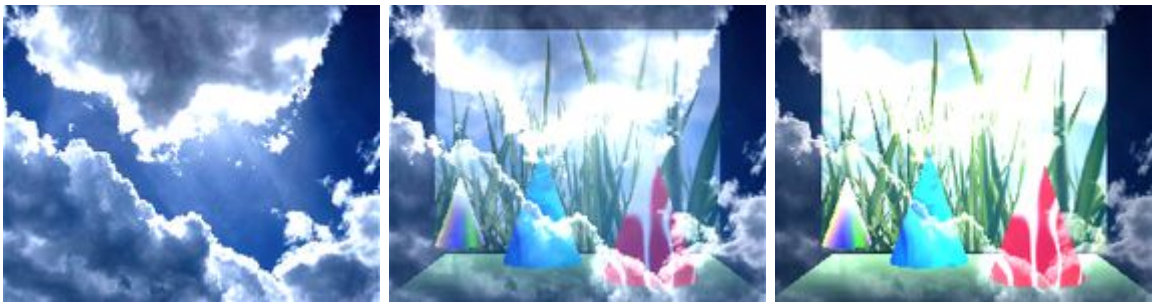
Render History



Fades between the Target's texture and the Rendering.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

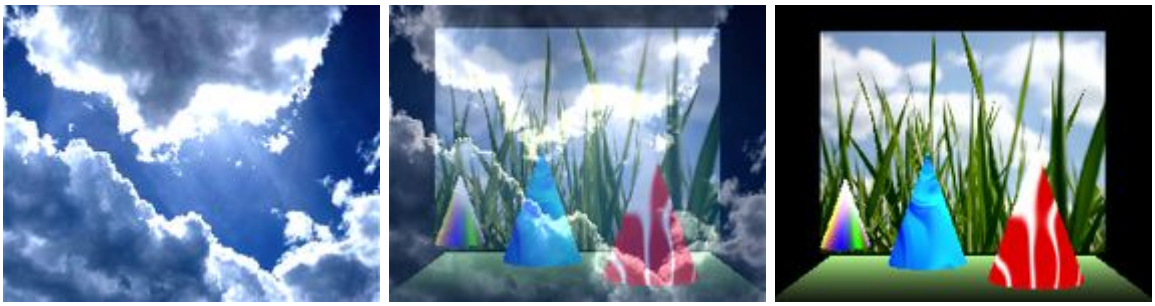
Target Add



Adds the pixel values of the Target's texture with the ones of the Rendering. In case of values above 255 (in the case of RGB), white is displayed.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

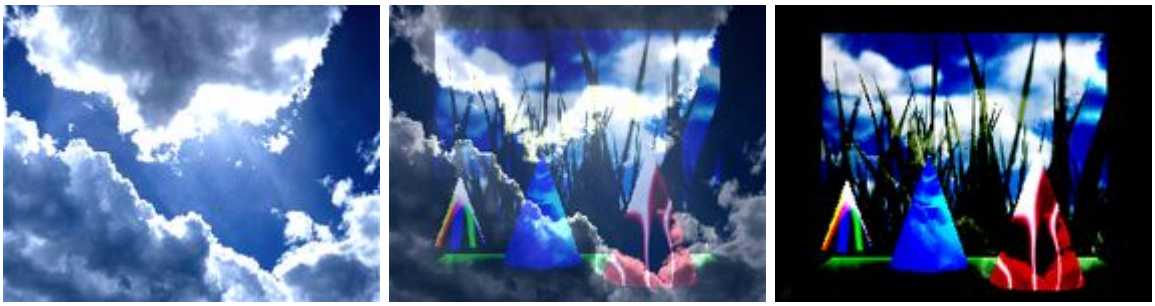
Target Buffer



Example above: left: Mix=0; center: Mix=128; right: Mix=255

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

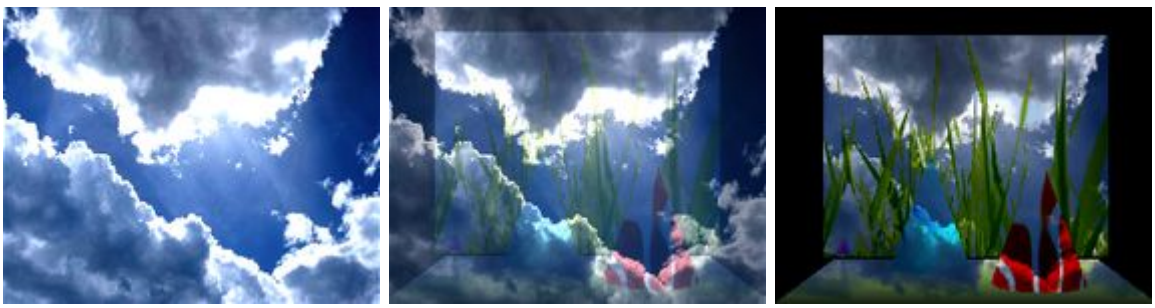
Target Color Burn



The Color Burn effect divides the inverted Rendering by the Target layer, and then inverts the result. This darkens the Target layer increasing the contrast to reflect the color of the Rendering. The darker the Rendering, the more its color is used. Blending with white produces no difference.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

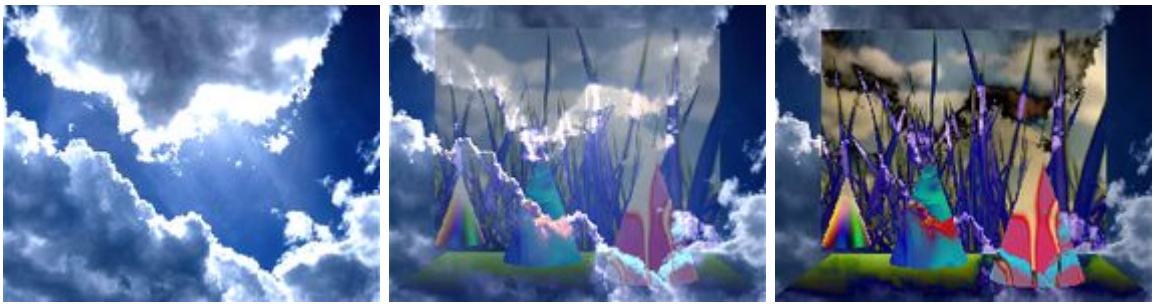
Target Darken



Darken takes the darkest value for each pixel from both, the Rendering and the Target layer.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

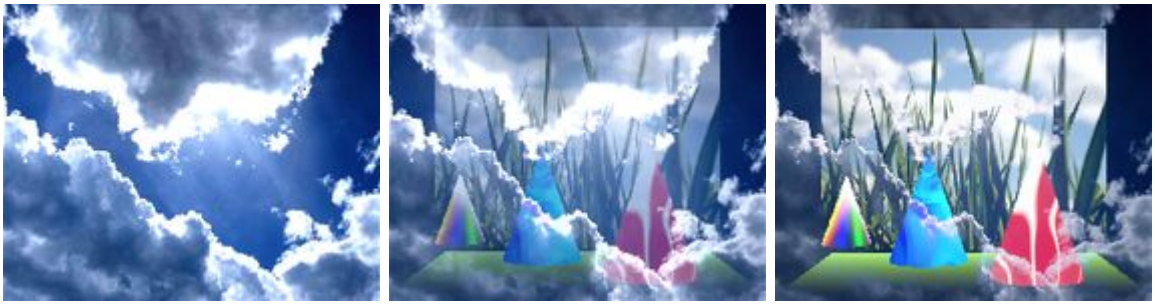
Target Difference



Difference subtracts the Target layer from the Rendering or the other way round; to always get a positive value. Blending with black produces no change, as values for all colors are 0. (The RGB value for black is 0,0,0). Blending with white inverts the picture.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

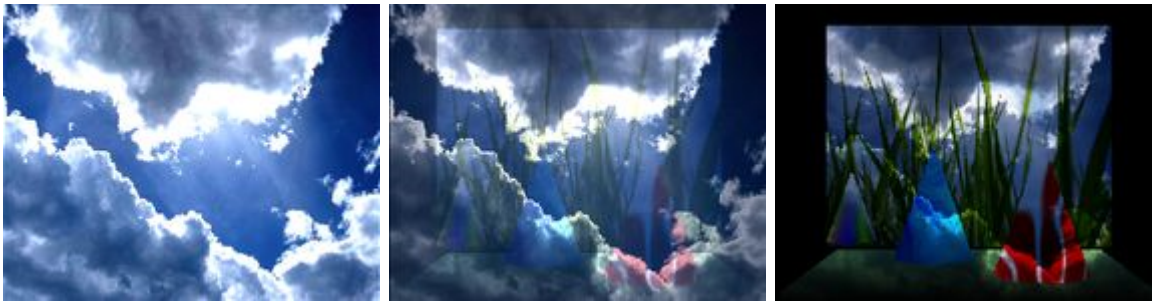
Target Lighten



Lighten takes the lightest pixel from both, the Rendering and the Target layer.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

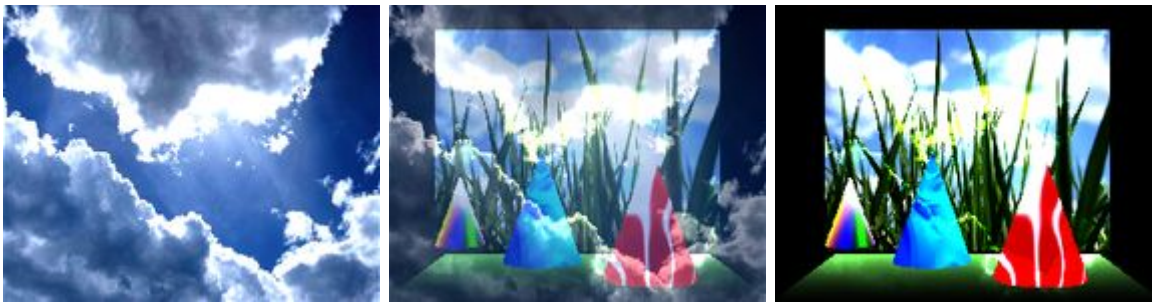
Target Multiply



Multiplies each pixel of the Target texture with the Rendering. The result is a darker picture.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

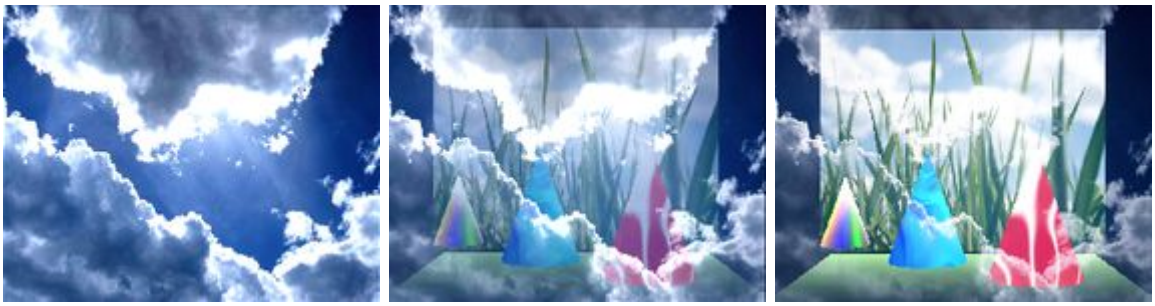
Target Reflect



Reflects the Target's texture.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

Target Screen

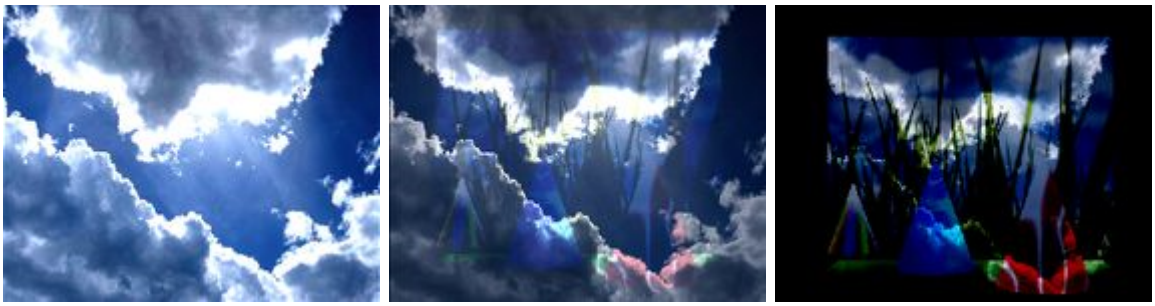


With Target Screen effect the values of the pixels in the two layers (Target and Rendering) are negated, multiplied, and then negated again. This is in some way the opposite of multiply.

The result is a brighter picture.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

Target Subtract

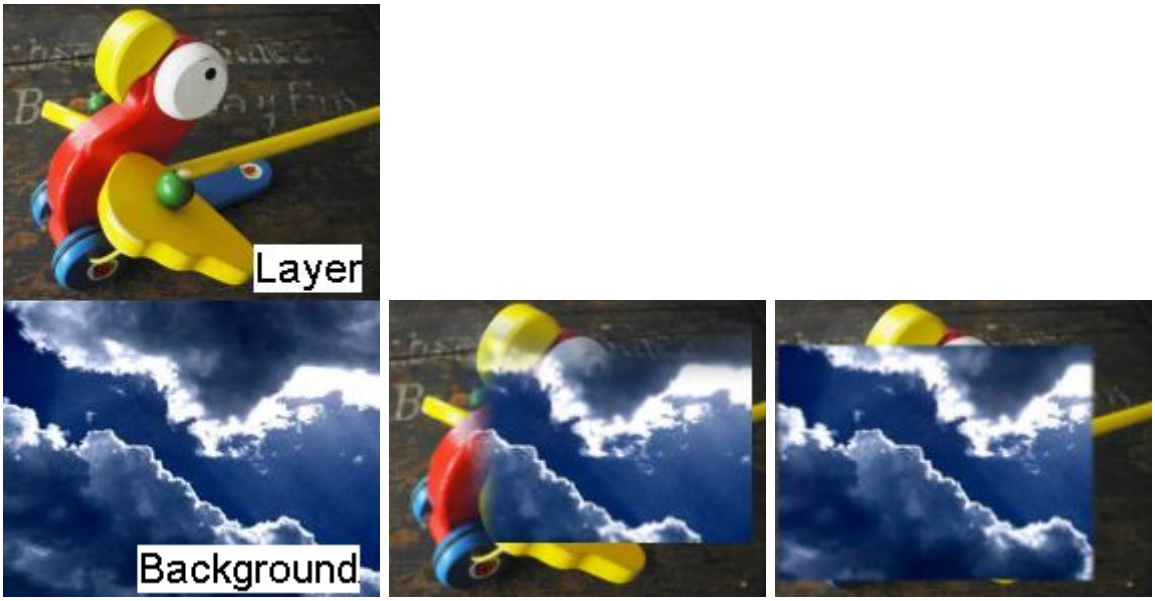


This Effect subtracts pixel values of the Target's texture with the Rendering. In case of negative values, black is displayed.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

6.5.2.4.10 Cropping

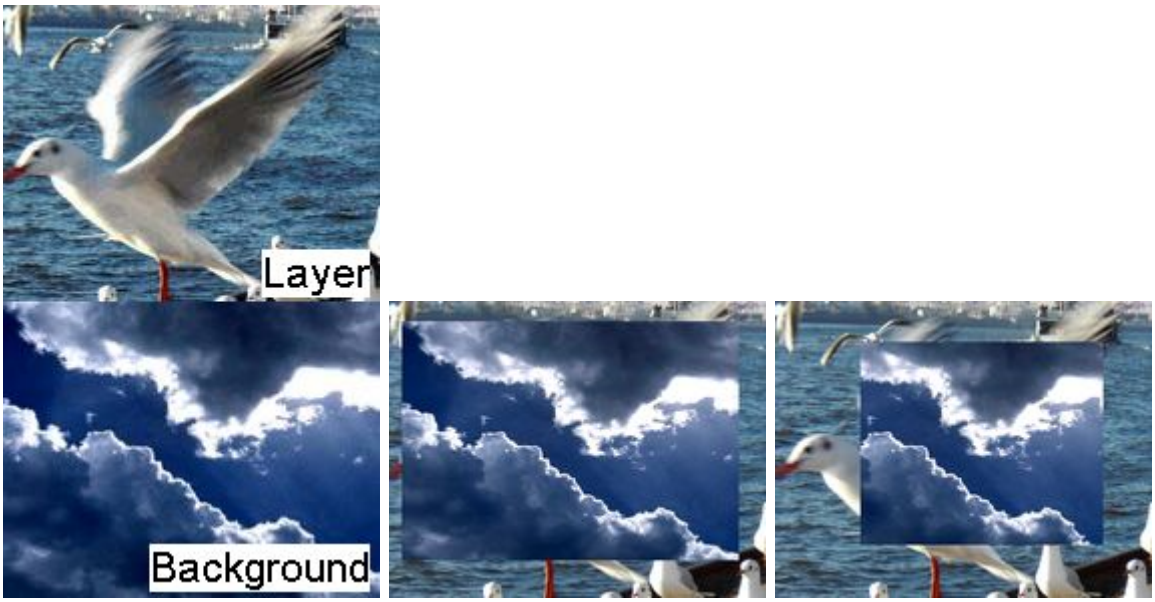
Crop Edges INV Softborder In



Crops all 4 edges individually into transparency and inverts them. In addition you can set a softborder individually for each side which goes inwards.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Left	0-65535	0
Right	Right	0-65535	0
Top	Top	0-65535	0
Bottom	Bottom	0-65535	0
Softness	Softness	0-65535	0
Left Soft	Left Soft	0-65535	0
Right Soft	Right Soft	0-65535	0
Top Soft	Top Soft	0-65535	0
Bottom Soft	Bottom Soft	0-65535	0

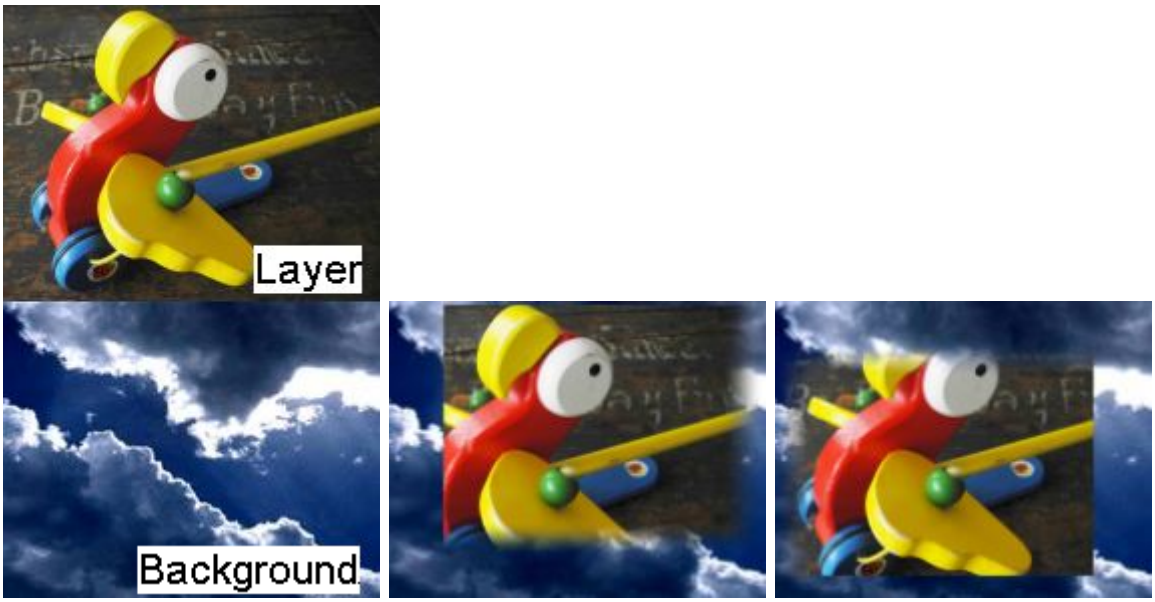
Crop Edges INV



Crops all 4 edges individually into transparency and inverts them.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Left	Crops left edge inverted	0-65535	0
Right	Crops right edge inverted	0-65535	0
Top	Crops top edge inverted	0-65535	0
Bottom	Crops bottom edge inverted	0-65535	0

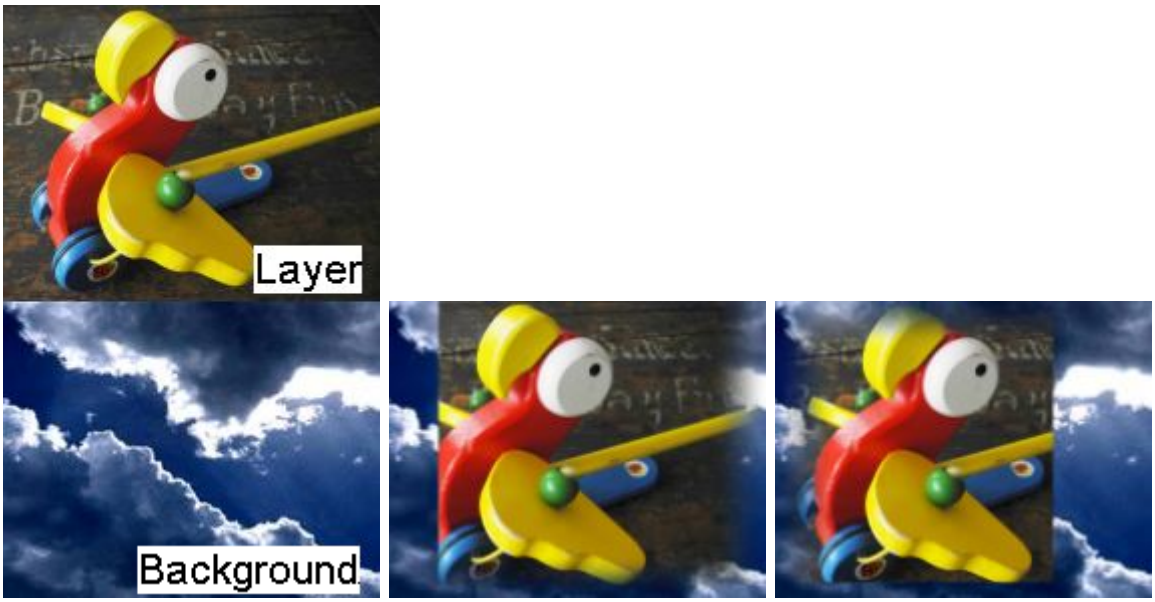
Crop Edges Softborder In



Crops all 4 edges individually into transparency. In addition you can set a softborder individually for each side which goes inwards.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Left	0-65535	0
Right	Right	0-65535	0
Top	Top	0-65535	0
Bottom	Bottom	0-65535	0
Softness	Softness	0-65535	0
Left Soft	Left Soft	0-65535	0
Right Soft	Right Soft	0-65535	0
Top Soft	Top Soft	0-65535	0
Bottom Soft	Bottom Soft	0-65535	0

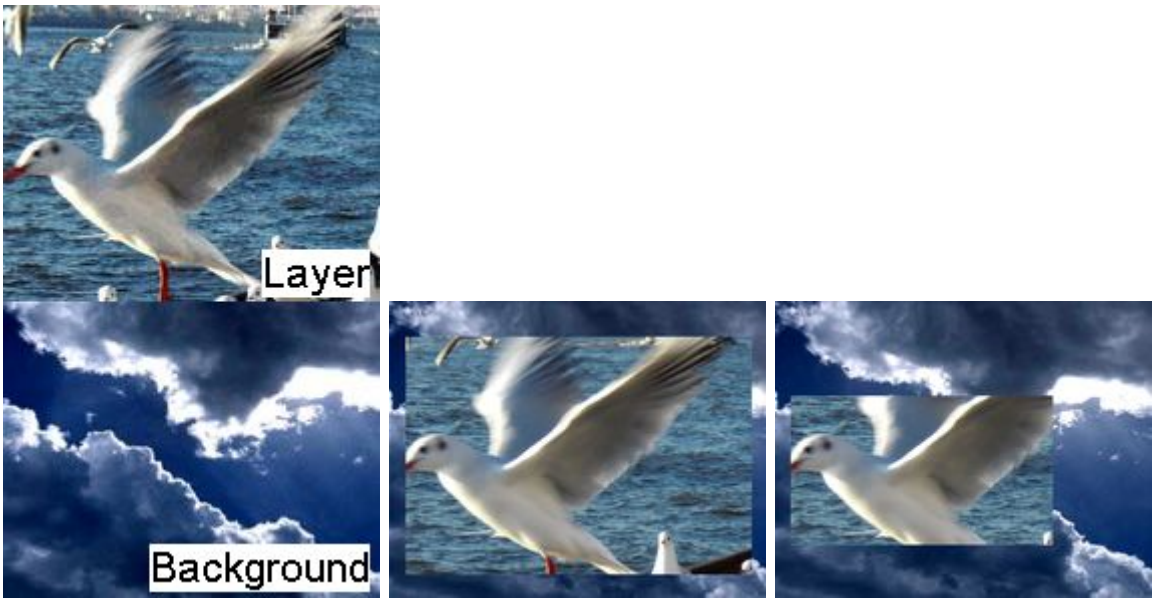
Crop Edges Softborder



Crops all 4 edges individually into transparency. In addition you can set a softborder individually for each side which goes outwards.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Left	0-65535	0
Right	Right	0-65535	0
Top	Top	0-65535	0
Bottom	Bottom	0-65535	0
Left Soft	Left Soft	0-65535	0
Right Soft	Right Soft	0-65535	0
Top Soft	Top Soft	0-65535	0
Bottom Soft	Bottom Soft	0-65535	0

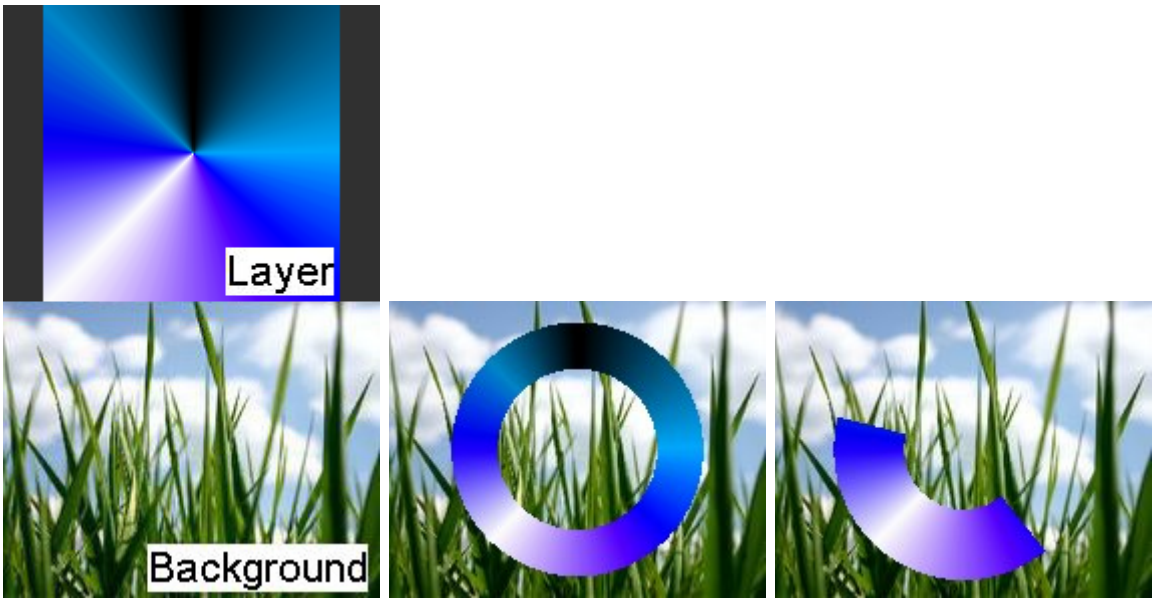
Crop Edges



Crops all 4 edges individually into transparency.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Left	Crops left edge	0-65535	0
Right	Crops right edge	0-65535	0
Top	Crops top edge	0-65535	0
Bottom	Crops bottom edge	0-65535	0

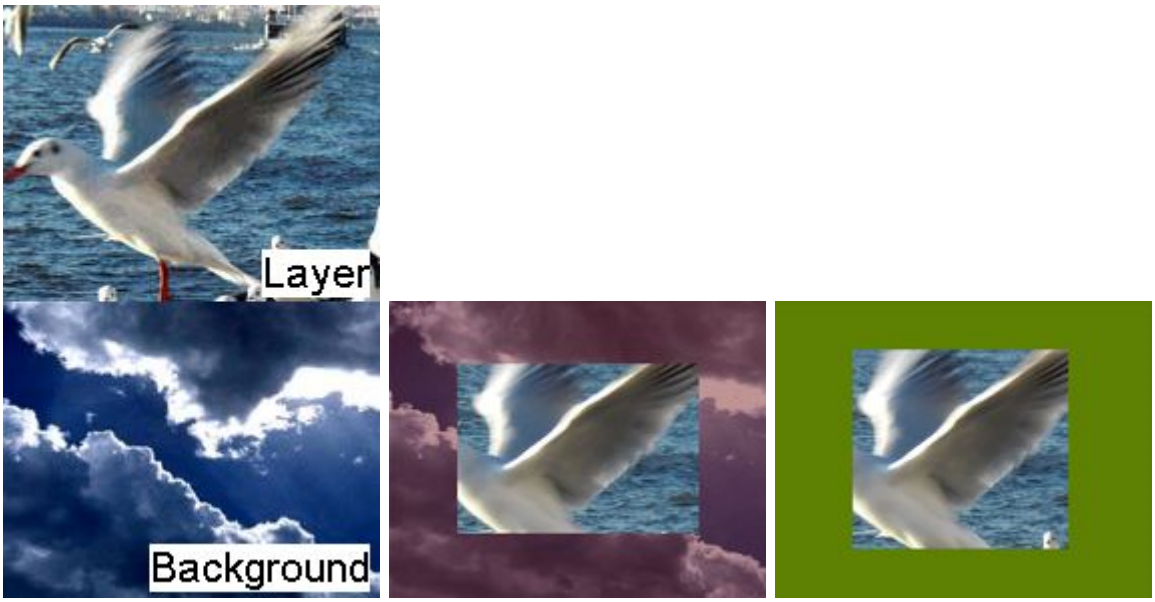
Crop Radial



This crops a texture into a ring shape by cutting the inner and outer part in a circular way. If the original texture is not a square, the ring's aspect mode can be adjusted. Optionally, the ring can then be cropped into a section by defining a starting and ending angle. The effect can be used for creating ring section or donut diagrams, for example.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Inner Size	Cropped size of inner part of texture	0-255	64
Outer Size	Cropped size of outer part of texture	0-255	128
Start Angle	Start angle for non-cropped circle	0-36000	0
End Angle	End angle for non-cropped circle	0-36000	36000
Aspect	Aspect Ratio of circle. Default value 128 applies the images aspect ratio to the iris.	0-255	128
Invert	Inverts the applied effect.	0-255	0

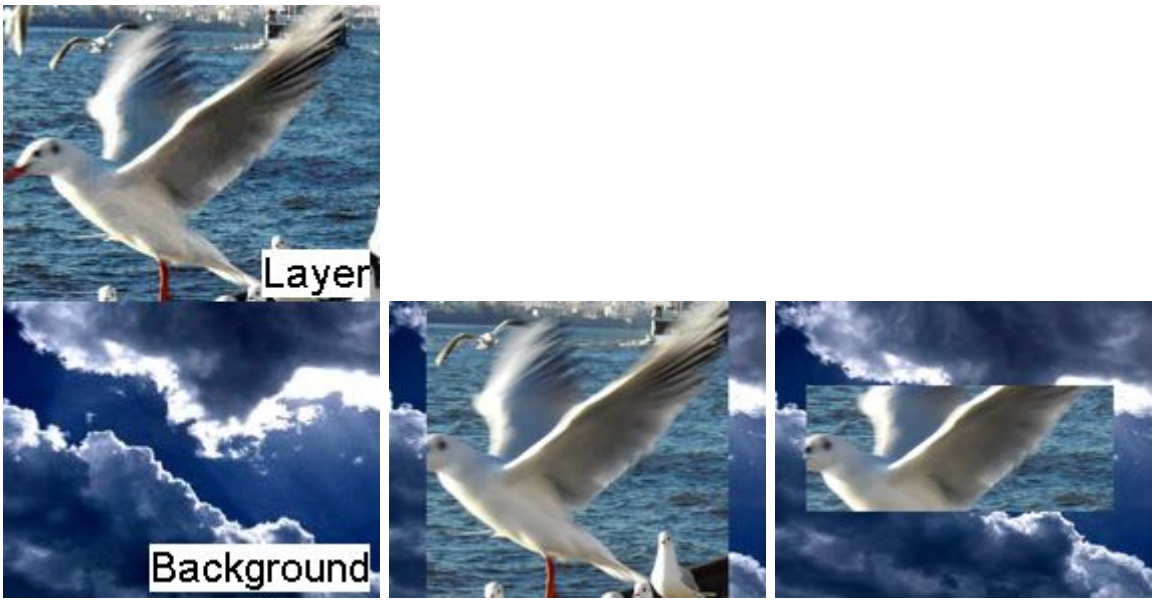
Cropper HV RGBA



Crops both horizontal edges and independently both vertical edges. The color and transparency of the cropped area can be set with the color and alpha mix parameters.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
H	Horizontal even crop	0-65535	0
V	Vertical even crop	0-65535	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Alpha	Alpha level of cropped area	0-255	128

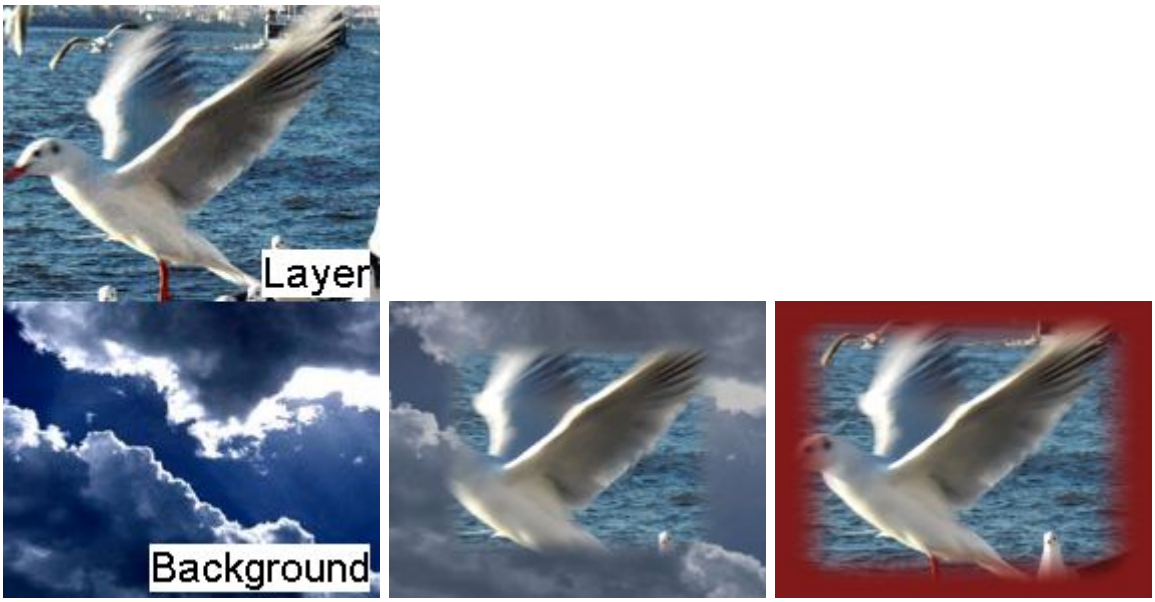
Cropper HV



Crops both horizontal edges and independently both vertical edges. The color of the cropped area is transparent.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
H	Horizontal even crop	0-65535	0
V	Vertical even crop	0-65535	0

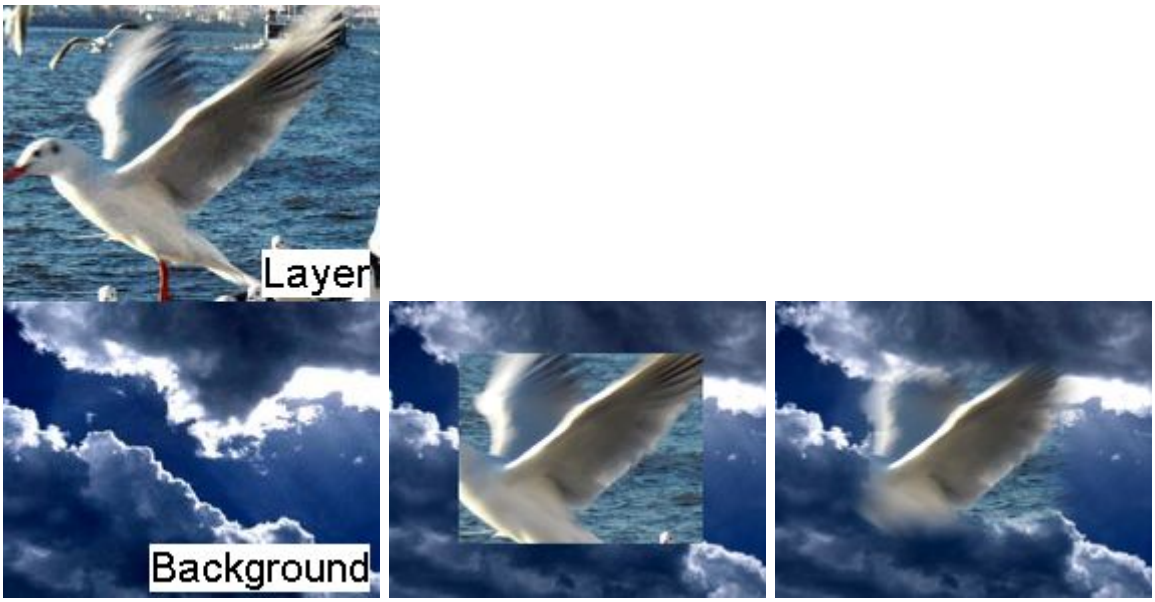
Frame RGBA



Crops all 4 edges equally at the same time creating the impression of framing the layer. The frame size and inwards softborder are adjustable. The color and transparency of the frame can be set with the color and alpha mix parameters.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of frame	0-255	8
Feather	Softness of inner frame border	0-255	8
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Alpha	Alpha level of cropped area	0-255	128

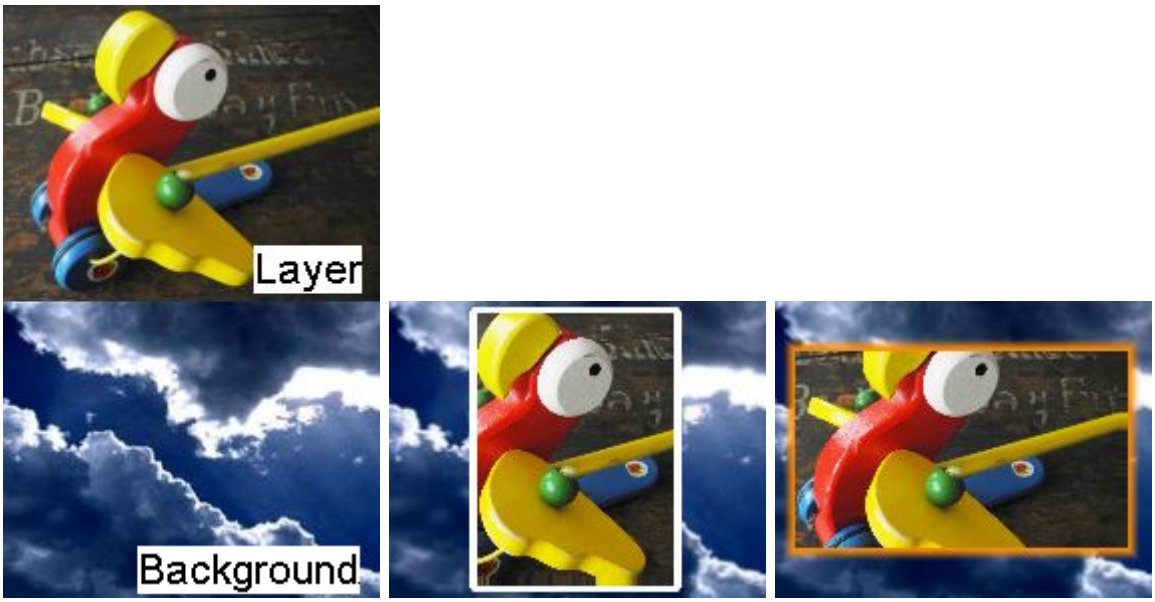
Frame



Crops all 4 edges equally at the same time creating the impression of framing the layer. The frame size and inwards softborder are adjustable. The color of the frame is transparent.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of frame	0-255	8
Feather	Softness of inner frame border	0-255	8

Framed Cropper HV

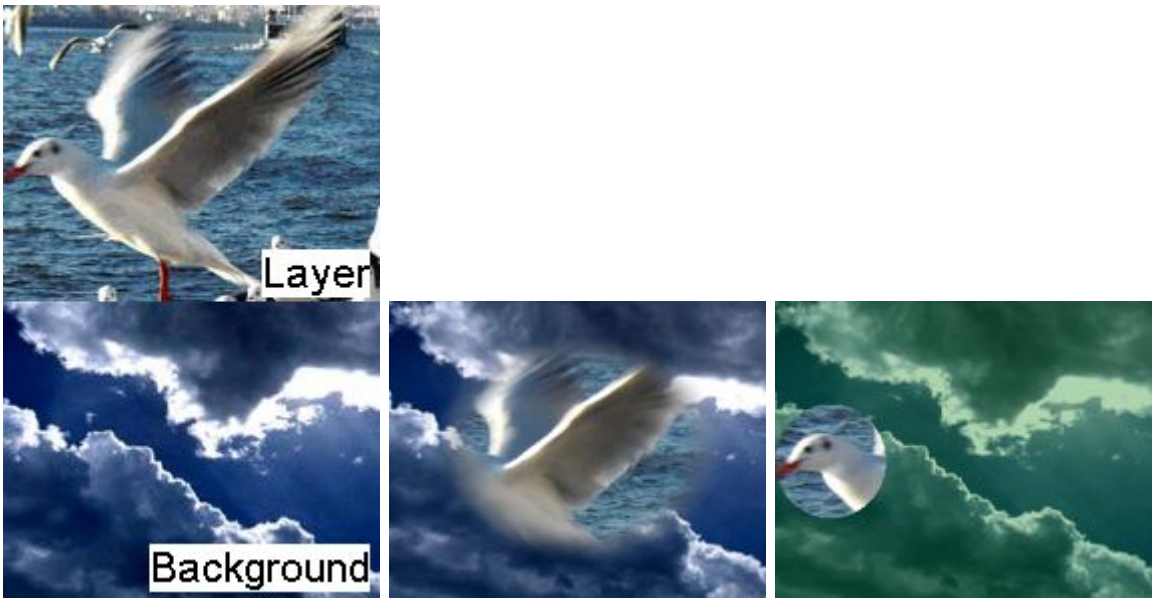


Crops both horizontal edges and independently both vertical edges. The cropped image can be surrounded with a frame for which you can set up the border width in pixels and the roundness of the corners. The color, transparency and softness of the frame can be set with the color, alpha mix and soft parameters.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Crop H (px)	Horizontal even crop	0-4096	0
Crop V (px)	Vertical even crop	0-4096	0
Frame (px)	Size of frame in pixel	0-4096	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Alpha	Alpha level of cropped area	0-255	128
Soft	Softness of frame	0-255	0
Round	Roundness of frame corners	0-255	0

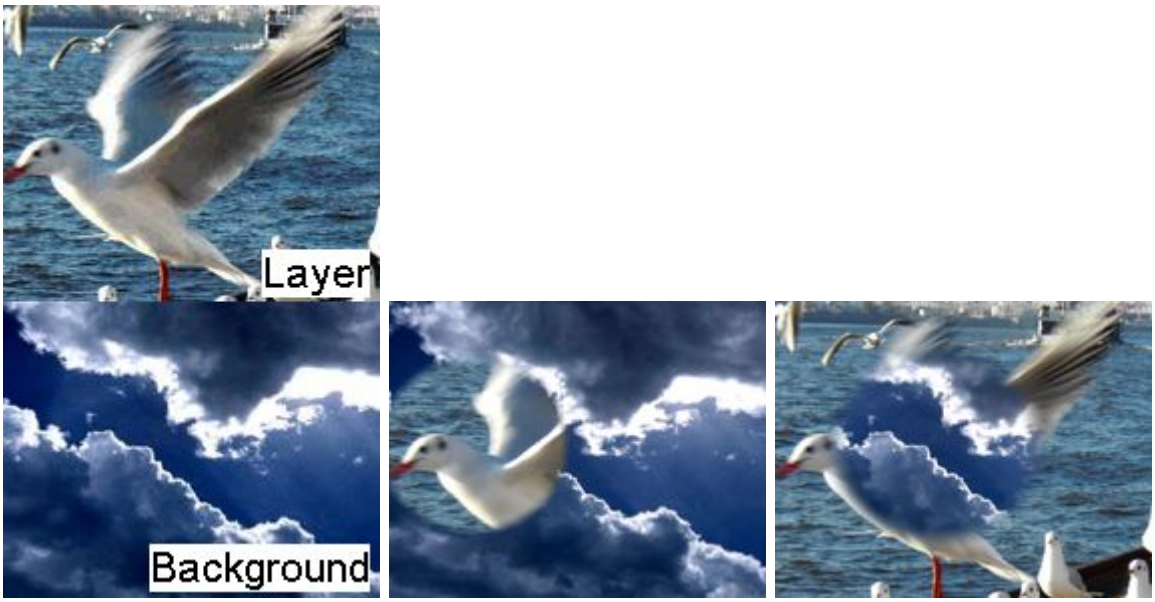
Iris RGBA



Applies an iris shaped cut-out to the layer creating the impression of putting the layer into a circular frame. The frame size, softborder, position and aspect ratio are adjustable. The color and transparency of the frame can be set with the color and alpha mix parameters. The frame effect can be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of frame	0-255	128
Softness	Softness of iris	0-255	64
U	Position of Iris horizontal	0-65535	32768
V	Position of Iris vertical	0-65535	32768
Aspect	Aspect Ratio of Iris. Default value 128 applies the images aspect ratio to the iris.	0.000 – 128.000	128
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Alpha	Alpha level of the iris.		
Invert	Invert Iris		

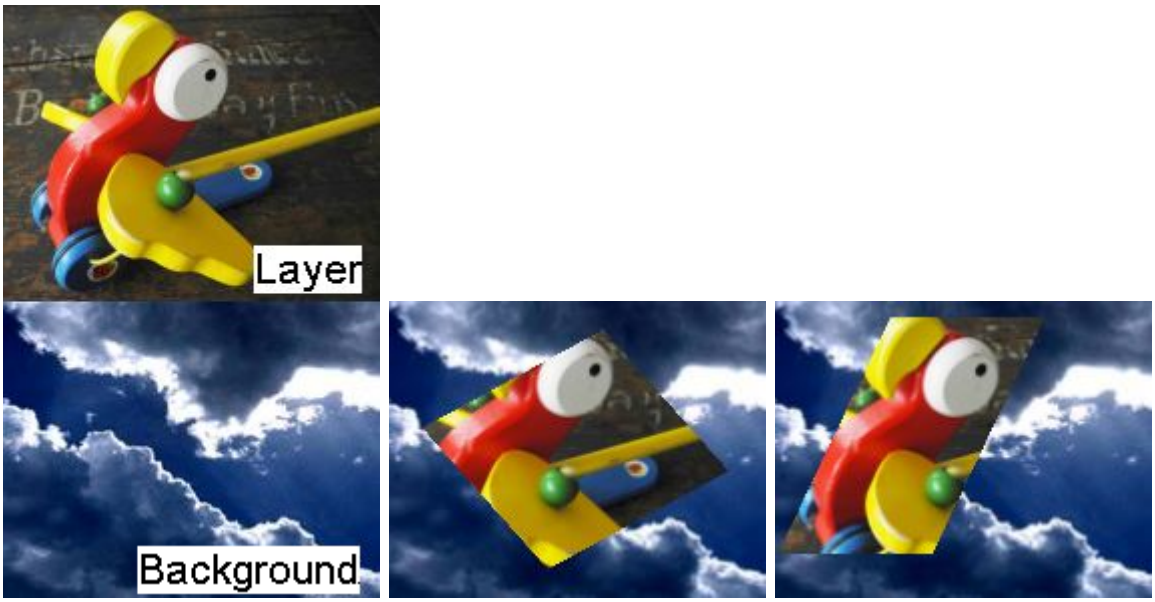
Iris



Applies an iris shaped cut-out to the layer creating the impression of putting the layer into a circular frame. The frame size, softborder, position and aspect ratio are adjustable. The frame is transparent and can be inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of frame	0-255	128
Softness	Softness of iris	0-255	64
U	Position of Iris horizontal	0-65535	32768
V	Position of Iris vertical	0-65535	32768
Aspect	Aspect Ratio of Iris. Default value 128 applies the images aspect ratio to the iris.	0-255	128
Invert	Invert Iris	0-255	0

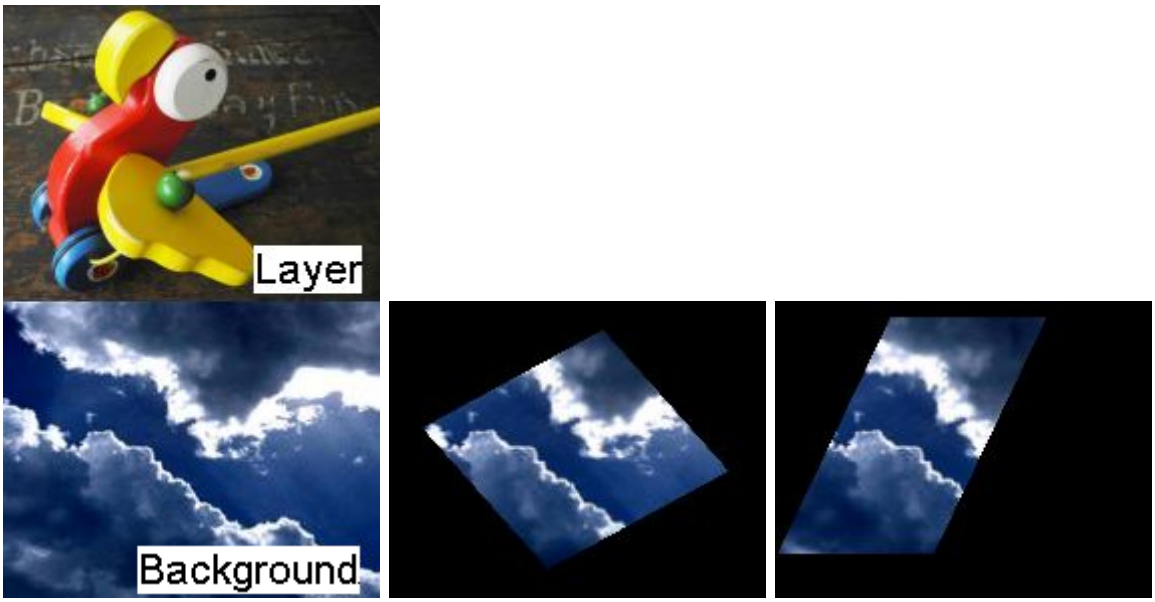
Profile Edges Alpha



Crops all 4 edges individually into transparency. You can choose for each side how much it should be cropped and rotated.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Left	0-65535	0
Left R	Left R	0-65535	32768
Right	Right	0-65535	0
Right R	Right R	0-65535	32768
Top	Top	0-65535	0
Top R	Top R	0-65535	32768
Bottom	Bottom	0-65535	0
Bottom R	Bottom R	0-65535	32768

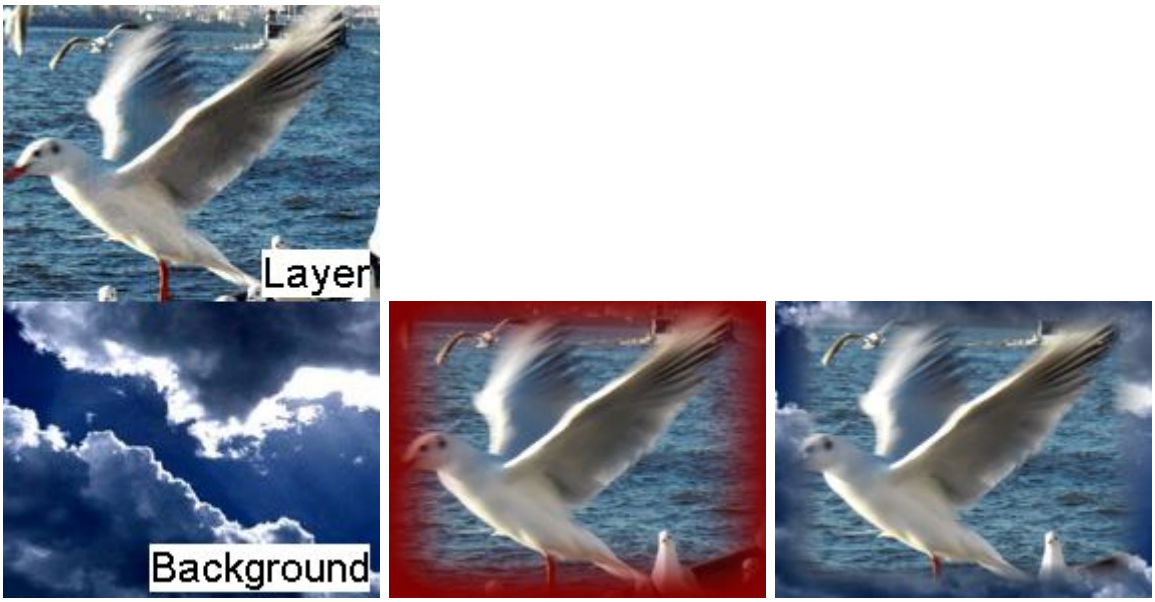
Profile Edges Mask



Turns the layer texture into a transparent and black mask. Crops all 4 edges individually into black. You can choose for each side how much it should be cropped and rotated.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Left	0-65535	0
Left R	Left R	0-65535	32768
Right	Right	0-65535	0
Right R	Right R	0-65535	32768
Top	Top	0-65535	0
Top R	Top R	0-65535	32768
Bottom	Bottom	0-65535	0
Bottom R	Bottom R	0-65535	32768

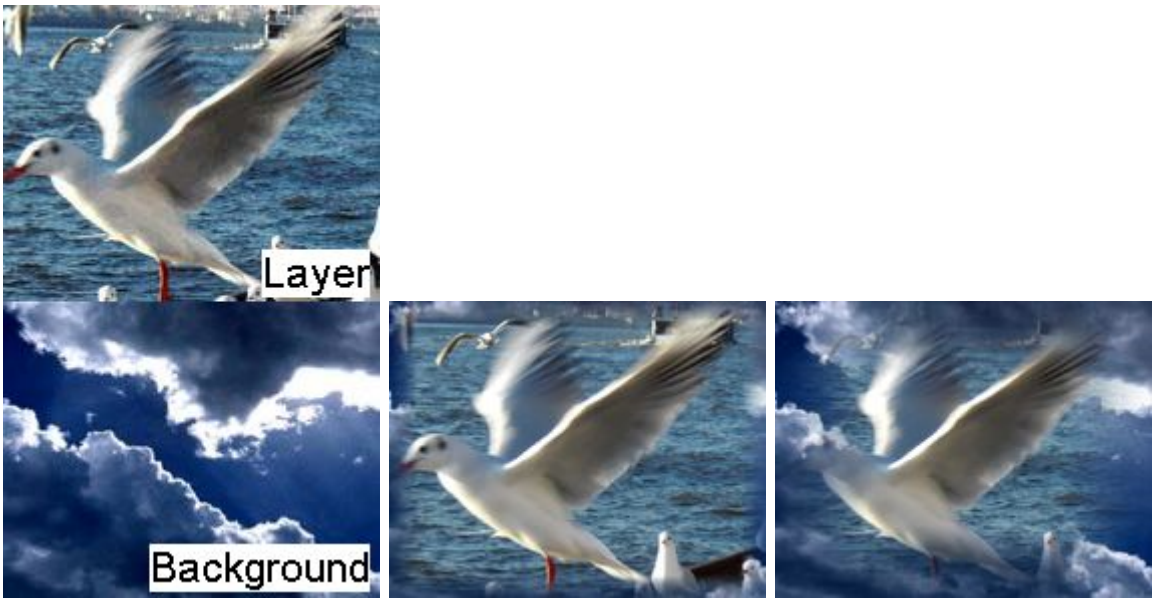
Softborder RGBA



Adds a softborder to all edges. The border size and roundness of the corners can be adjusted. The color and transparency of the border can be set with the color and alpha mix parameters.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of softborder	0-255	16
Corner	Softness of corner	0-255	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Alpha	Alpha level of the border area	0-255	128

Softborder



Adds a softborder to all edges. The border size and roundness of the corners can be adjusted. The color of the border is transparent.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of softborder	0-255	16
Corner	Softness of corner	0-255	0

6.5.2.4.11 Distort

Displace

Displaces the pixels of the target image.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Offset	Amount of displacement	0-65535	32768

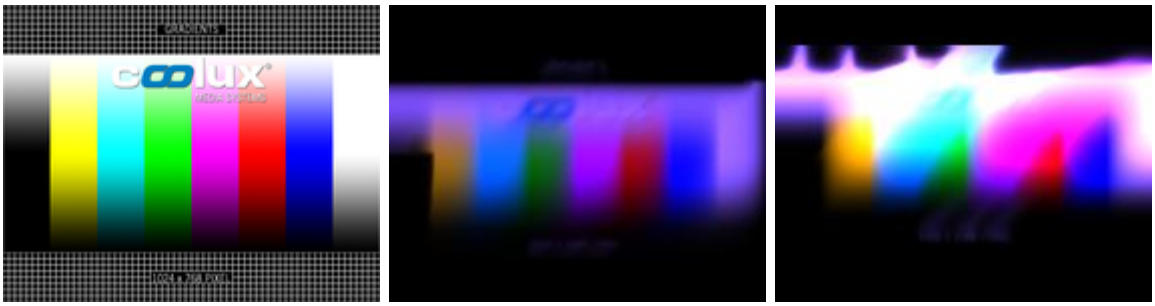
Distort

Distorts the image, using factorized UV mapping offsets.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
U Offset	Offset of U Mapping	0-65535	32768
V Offset	Offset of V Mapping	0-65535	32768
U Factor	Factor of U Mapping	0-65535	32768
V Factor	Factor of V Mapping	0-65535	32768

Fluid Color

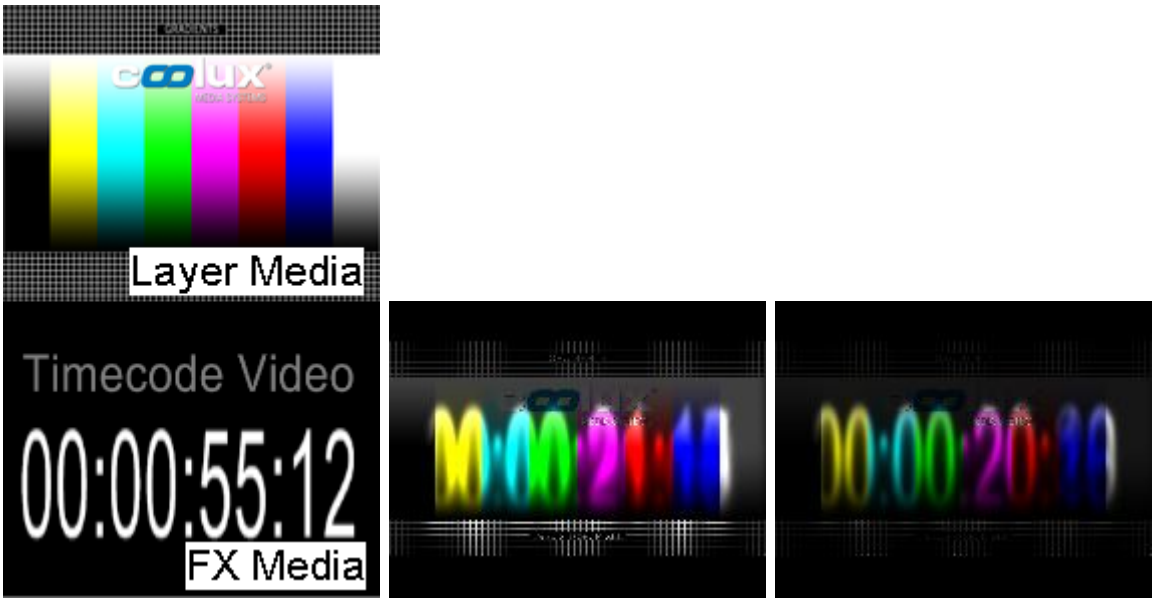


This effect multiplies the FX texture with a color and adds a fluid effect which is best seen when applied to a Video Layer.
According to the Offset and Velocity parameters, the frames of this result change there position which creates the impression, the frames would fly into a (changing) direction.
With the "Density difference" fader you can change how many frames are multiplied into the Layer texture and with "Velocity Factor" how far frames move. The last image shows a higher velocity than in the second image.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Vel Dir	Vel Dir	0-65535	32768
Offset	Offset	0-65535	0
Offs. Freq.	Offs. Freq.	0-65535	32768
Vel Input	Vel Input	0-65535	6553
Vel Factor	Vel Factor	0-65535	5242
Vel Cutoff	Vel Cutoff	0-65535	13000
Vel Diff	Vel Diff	0-65535	63000
Den Diff	Den Diff	0-65535	63000
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255

Fluid Mask

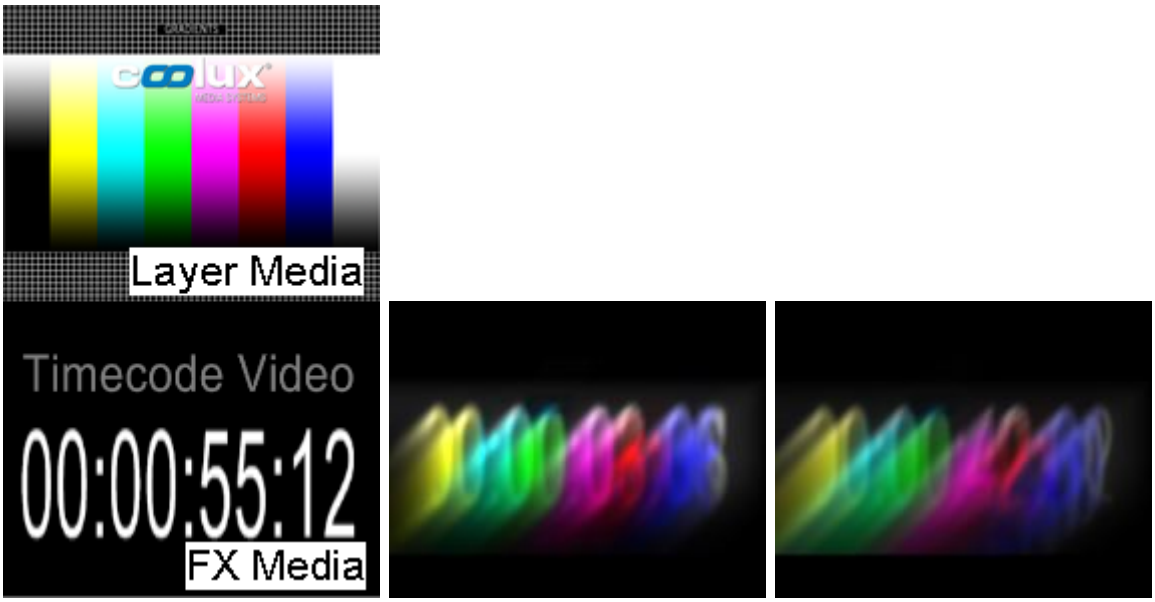


This effect multiplies the FX texture with the Layer texture and adds a fluid effect which is best seen when applied to a Video Layer.
According to the Offset and Velocity parameters, the frames of this result change there position slightly which creates the impression, they are blurred fluidly.
With the "Density difference" fader you can change how many frames are multiplied into the Layer texture. The last image shows a lower difference, resulting in a darker image and in a clearer FX media. "Invert RGB" inverts the FX texture before its mixed to the Layer.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Media file as source for overlay or mask	-	-
Vel Dir	Vel Dir	0-65535	32768
Offset	Offset	0-65535	0
Offs. Freq.	Offs. Freq.	0-65535	32768
Vel Input	Vel Input	0-65535	6553
Vel Factor	Vel Factor	0-65535	5242
Vel Cutoff	Vel Cutoff	0-65535	13000
Vel Diff	Vel Diff	0-65535	63000
Den Diff	Density Difference	0-65535	63000
Inv. RGB	Invert RGB levels from FX media	0-255	0
Inv. Alpha	Inverts transparency from FX media	0-255	0

Fluid Paint



This effect multiplies the FX texture with the Layer texture and adds a fluid effect which is best seen when applied to a Video Layer.
According to the Offset and Velocity parameters, the frames of this result change there position which creates the impression, the frames would fly into a (changing) direction.
With the "Density difference" fader you can change how many frames are multiplied into the Layer texture and with "Velocity Factor" how far frames move. The last image shows a higher velocity than in the second image.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Media file as source for overlay or mask	-	-
Vel Dir	Vel Dir	0-65535	32768
Offset	Offset	0-65535	0
Offs. Freq.	Offs. Freq.	0-65535	32768
Vel Input	Vel Input	0-65535	6553
Vel Factor	Vel Factor	0-65535	5242
Vel Cutoff	Vel Cutoff	0-65535	13000
Vel Diff	Vel Diff	0-65535	63000
Den Diff	Den Diff	0-65535	63000

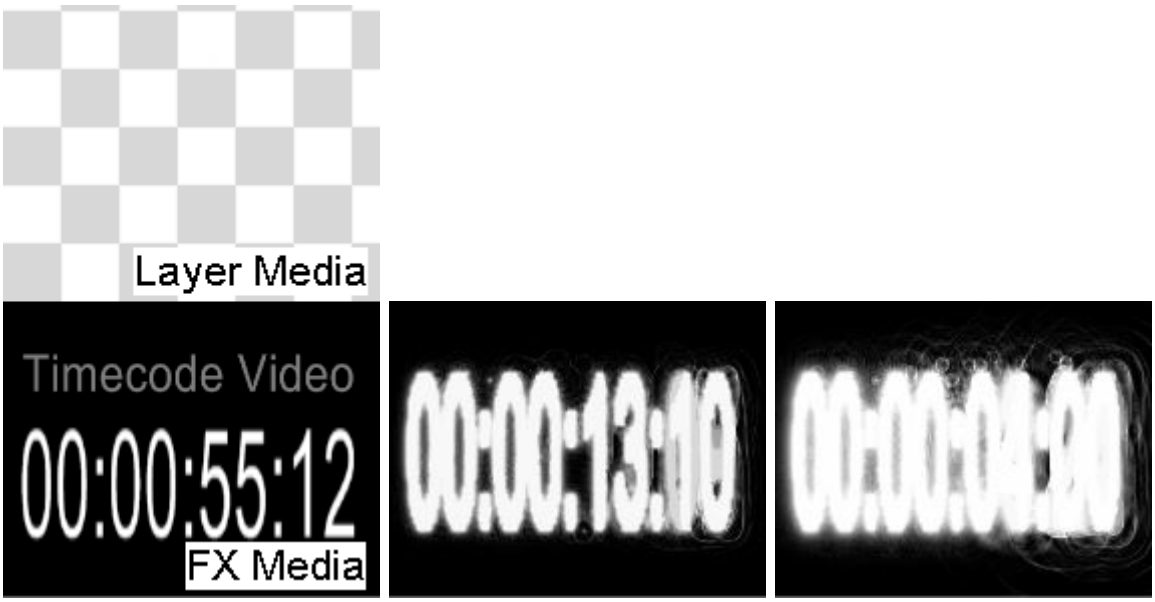
Interactive Waves Buffer RH

This effect is based on the Layer's Z Position or render history as explained in the manual chapter "Compositing" [406]. The layer with the assigned effect makes a composition of all layers behind it and displays it ontop its own texture. In the example, the "render" layer is positioned in the foreground, its main media is a transparent image. In the background there is a Video Layer showing a running timecode and a particle system [185] (also with a transparent media!).
Whenever a pixel from the composition changes, it causes waves as if a rain drop deformed the rendering texture.
The parameter "Strenght" influences the wave speed / frequency, whilst "Delay" influences the amplitude, i.e. how far the waves go).

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Strength	Strength	0-255	0
Delay	Delay	0-65535	60000

Interactive Waves Buffer



The main media from the layer is a transparent image whilst the effect's media is video showing a running timecode. Whenever a pixel from the timecode video changes, it causes waves as if a rain drop deformed the rendering texture.
The parameter "Strenght" influences the wave speed / freuquency, whilst "Delay" influences the amplitude, i.e. how far the waves go).

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Strength	Strength	0-255	0
Delay	Delay	0-65535	60000
Media	Media file as source for waves	-	-

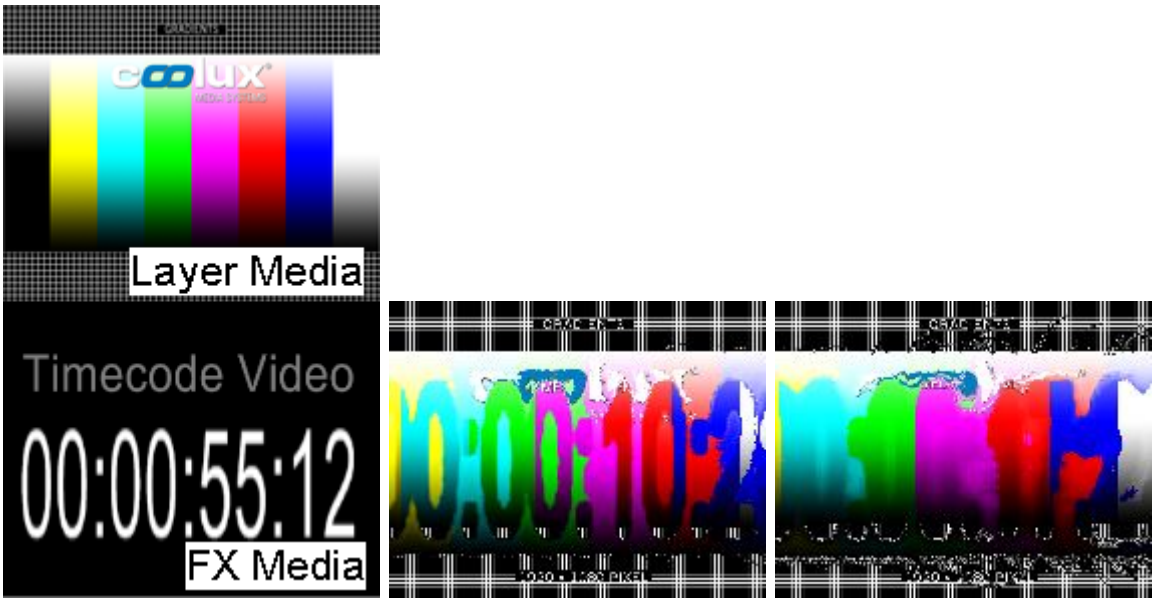
Interactive Waves RH

This effect is based on the Layer's Z Position or render history as explained in the manual chapter "Compositing" [406]. The layer with the assigned effect makes a composition of all layers behind it and mixes it into its own texture. In the example, the "render" layer is positioned in the foreground, its main media is a grass image. In the background there is a Video Layer showing a running timecode and a particle system [185] (also with a transparent media!).
Whenever a pixel from the composition changes, it causes waves as if a rain drop deformed the rendering grass texture.
The parameter "Strenght" influences the wave speed / frequency, whilst "Delay" influences the amplitude, i.e. how far the waves go). "Offset" offsets the grass texture in the waves.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Strength	Strength	0-255	0
Offset	Offset	0-65535	32768
Delay	Delay	0-65535	60000

Interactive Waves

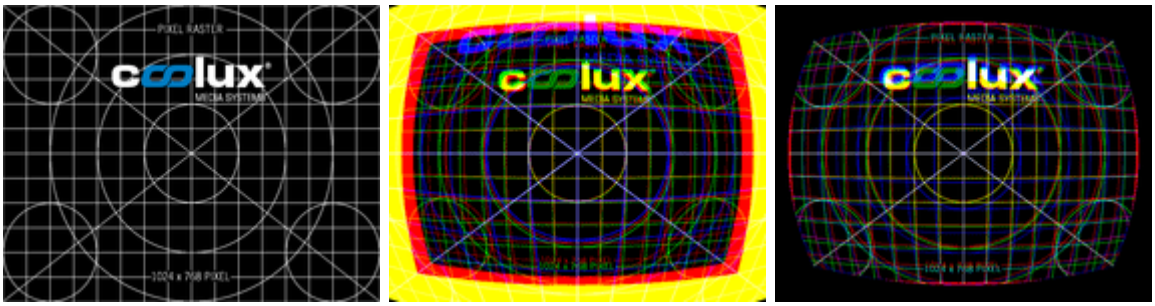


The main media from the layer is a color gradients image whilst the effect's media is video showing a running timecode. Whenever a pixel from the timecode video changes, it causes waves as if a rain drop deformed the rendering color gradient texture.
The parameter "Strenght" influences the wave speed / freuquency, whilst "Delay" influences the amplitude, i.e. how far the waves go). "Offset" offsets the grass texture in the waves.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Strength	Strength	0-255	0
Offset	Offset	0-65535	32768
Delay	Delay	0-65535	60000
Media	Media file as source for waves	-	-

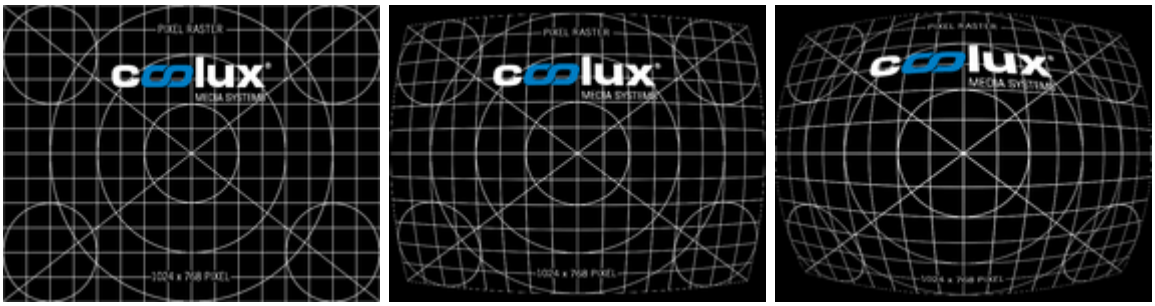
Lens Barrel Distortion Color Corr



Applies a barrel distortion to a layer or output texture. In addition it applies a correction for chromatic aberration which is for example of interest when working with a device like an Oculus Rift.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Lens X	Horizontal lens position	0-65535	32768
Lens Y	Vertical lens position	0-65535	32768
Scale X	Horizontal scale	0-65535	32768
Scale Y	Vertical scale	0-65535	32768
ScaleIn X	Horizontal scale	0-65535	32768
ScaleIn Y	Vertical scale	0-65535	32768
Coef. 0	Coef. 0	0-65535	65535
Coef. 1	Coef. 1	0-65535	32768
Coef. 2	Coef. 2	0-65535	32768
Coef. 3	Coef. 3	0-65535	32768

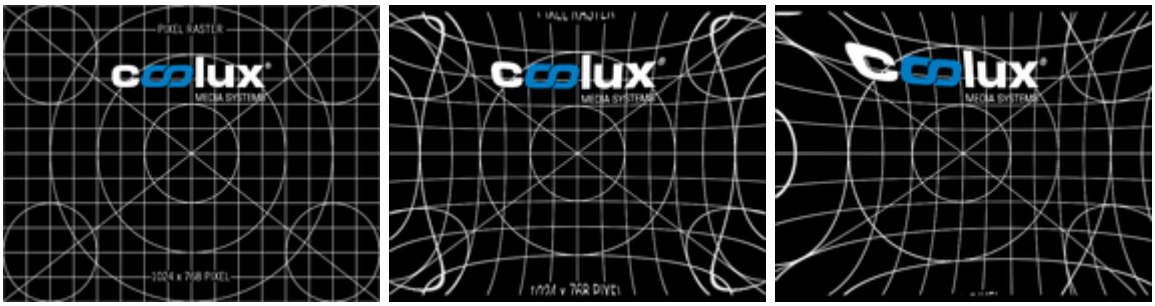
Lens Barrel Distortion



Applies a barrel distortion to a layer or output texture.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Lens X	Horizontal lens position	0-65535	32768
Lens Y	Vertical lens position	0-65535	32768
Scale X	Horizontal scale	0-65535	32768
Scale Y	Vertical scale	0-65535	32768
ScaleIn X	Horizontal scale	0-65535	32768
ScaleIn Y	Vertical scale	0-65535	32768
Coef. 0	Coef. 0	0-65535	65535
Coef. 1	Coef. 1	0-65535	32768
Coef. 2	Coef. 2	0-65535	32768
Coef. 3	Coef. 3	0-65535	32768

Lens Pincushion Distortion



Applies a pincushion distortion to a layer or output texture.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Lens X	Horizontal lens position	0-65535	32768
Lens Y	Vertical lens position	0-65535	32768
Scale X	Horizontal scale	0-65535	32768
Scale Y	Vertical scale	0-65535	32768
ScaleIn X	Horizontal scale	0-65535	32768
ScaleIn Y	Vertical scale	0-65535	32768
Coef. 0	Coef. 0	0-65535	65535
Coef. 1	Coef. 1	0-65535	32768
Coef. 2	Coef. 2	0-65535	32768
Coef. 3	Coef. 3	0-65535	32768

Magnify Glass

Magnifies an area of the layer image and allows to define the position and size of this area as well as the used zoom factor.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
X	X Position of magnifier	0-65535	32768
Y	Y Position of magnifier	0-65535	32768
In	Size of completely magnified area	0-65535	16448
Out	Size of area where magnification ends	0-65535	17733
Magnify	Amount of magnification	0-65535	32768
Aspect	Aspect ratio of magnifier	0-65535	42662

PolarCoordinates

Polar Coordinates Effect. Can be used to either treat current coordinates as Cartesian and Resample to Polar or vice versa. Created by Florian Mosleh www.s2gfx.com

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine 162". The chapter "FX Order 325" explains how to drag effects in the Device Tree 173 tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Operation	Operation	0-255	0

Ripple Animation

Distorts the texture with a ripple animation. Amplitude, frequency, motion speed and direction as well as size and position can be set up.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amplitude	Defines the height of the ripple's amplitude	0-65535	16384
Frequency	Defines the ripple's frequency (=amount of ripples coming out from its center)	0-65535	32768
Motion	Defines the speed and direction of the ripple's motion	0-65535	16448
Pos U	U Position	0-65535	17733
Pos V	V Position	0-65535	32768
Scale U	U Scale	0-65535	16384
Scale V	V Scale	0-65535	16384

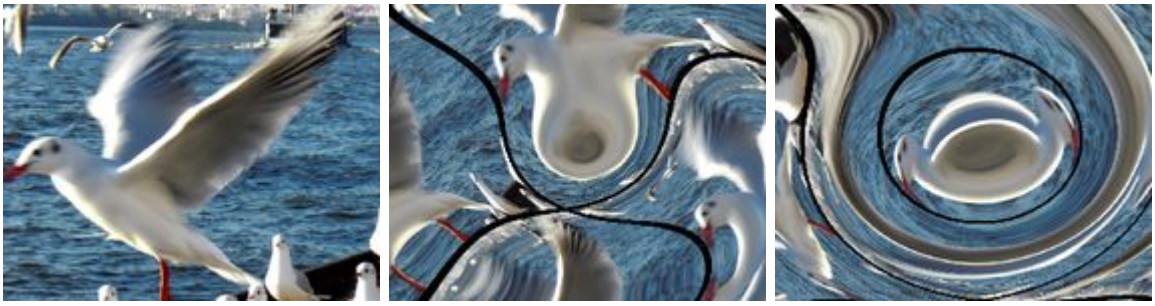
Ripple

Distorts the texture with a non-animated ripple effect. Amplitude, frequency, phase as well as size and position can be set up.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amplitude	Defines the height of the ripple's amplitude	0-65535	16384
Frequency	Defines the ripple's frequency (=amount of ripples coming out from its center)	0-65535	32768
Phase	Define the phase of the ripple's curve	0-65535	16448
Pos U	U Position	0-65535	17733
Pos V	V Position	0-65535	32768
Scale U	U Scale	0-65535	16384
Scale V	V Scale	0-65535	16384

Swirl Angle

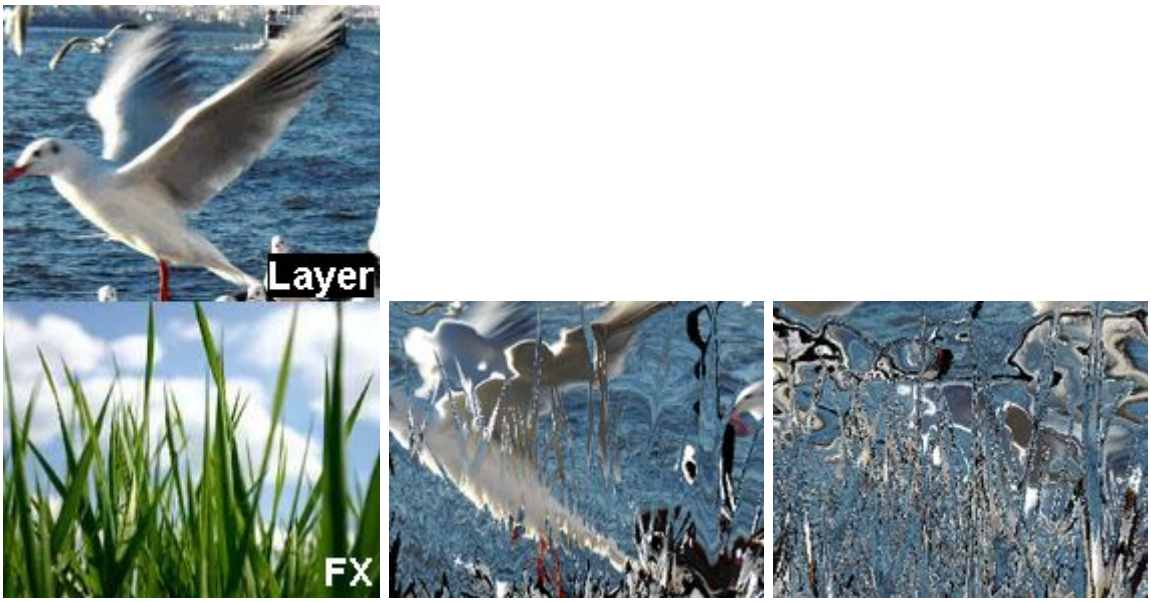


Distorts the texture with a non-animated swirl effect. Twist, Depth and Amount can be set up.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Twist	Amount of Twist	0-65535	128
Depth	Depth of the Swirl	0-65535	4096
Amount	Amount of swirl waves	0-65535	2048

Texture Morph



Morphs the image by using an additional media file as texture.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Media	Add the image file that should be used as texture for the morph effect		
Mix	Level of Effect	0-255	0
Factor	Amount of Texture Morphing	0-255	64

Waves

Modifies the texture with non-animated waves.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

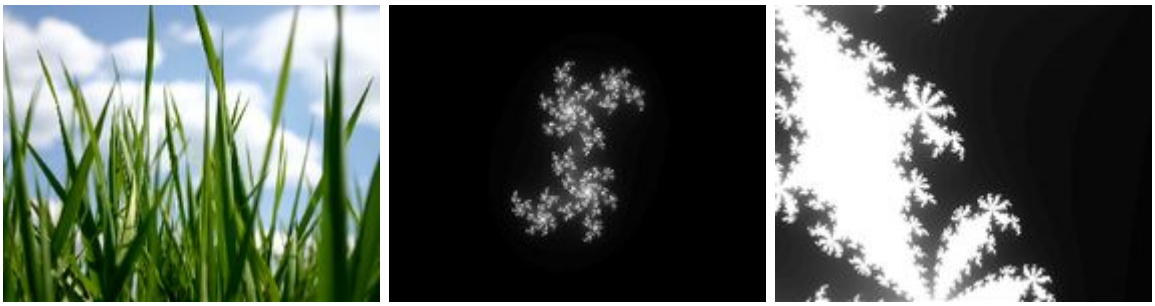
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of Waves	0-65535	64000
U Offset	U Mapping Offset (in X direction)	0-65535	32768
V Offset	V Mapping Offset (in Y direction)	0-65535	32768

6.5.2.4.12 Fractals

++ General information for Fractal Effects ++

A fractal is a geometric pattern that is repeated at ever smaller scales to produce irregular shapes and surfaces that cannot be represented by classical geometry. Fractals are used especially in computer modeling of irregular patterns and structures in nature.

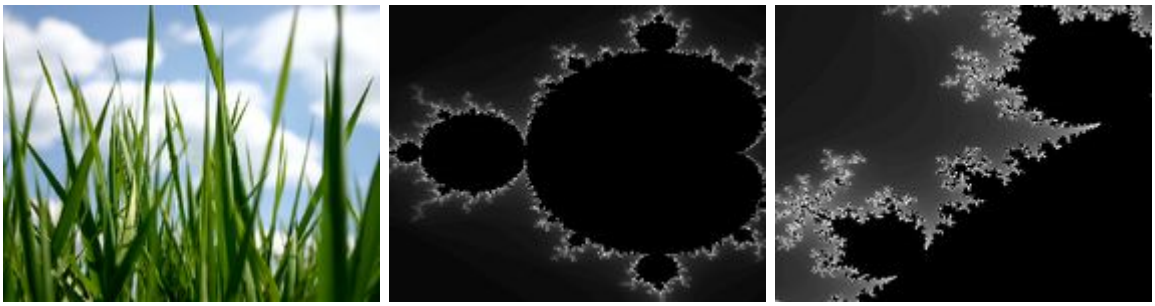
Julia Fractal



Shows a Julia Fractal that can be modified by Zoom, Position and Seed Parameters.
The texture being originally on the layer does not influence this effect except during blending.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Zoom	Zoom into the fractal	0-65535	32768
Pos U	Pos U	0-65535	32768
Pos V	Pos V	0-65535	32768
Seed X	Seed X	0-65535	25000
Seed Y	Seed Y	0-65535	25000

Mandelbrot Fractal



Shows a Mandelbrot Fractal that can be modified by Zoom, Aspect Ratio and Position Parameters.
The texture being originally on the layer does not influence this effect except during blending.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Zoom	Zoom into the fractal	0-65535	32768
Aspect	Defines the Aspect Ratio of the Fractal	0-65535	16384
Pos U	Pos U	0-65535	32768
Pos V	Pos V	0-65535	32768

6.5.2.4.13 FX Collections

Server FX 16bit 9Ch

This "Effect Collection" contains almost 150 different effects that can be chosen with the "FX Name" drop-down list. All effects are also available as separate, single effects, so if necessary please look up their description by looking up the effect itself.
The 9 available channels will adopt to an effect parameter depending on which effect you chose. Channels not needed will be labeled "n/a". Each channel has a resolution of 16 bit.

When starting your project in "Lighting Console Mode" or adding Clients with that option to your Device Tree, three "Server FX" effects will be added to each Video Layer and patched automatically, see Patch tab 228.

Player FX 16bit 4Ch

With the introduction of the v8 License Model 62, the "Player FX Collection" should only be used due to legacy reasons as the "Server FX Collection" offers more effects and channels.

This "Effect Collection" contains almost 120 different effects that can be chosen with the "FX Name" drop-down list. All effects are also available as separate, single effects, so if necessary please look up their description by looking up the effect itself.
The 4 available channels will adopt to an effect parameter depending on which effect you chose. Channels not needed will be labeled "n/a". Each channel has a resolution of 16 bit.

6.5.2.4.14 Generator

RND Noise Add

Adds the RGB values of a random noise texture (animated) with the RGB values of the layer's texture. Size of noise can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of noisy items	0-65535	64

RND Noise Divide

Divides the RGB values of a random noise texture (animated) with the RGB values of the layer's texture. Size of noise can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of noisy items	0-65535	64

RND Noise Mix

Fades into a random noise texture (animated). Size of noise can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of noisy items	0-65535	64

RND Noise Multiply

Multiplies the RGB values of a random noise texture (animated) with the RGB values of the layer's texture. Size of noise can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of noisy items	0-65535	64

RND Noise Subtract

Subtracts the RGB values of a random noise texture (animated) with the RGB values of the layer's texture. Size of noise can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size of noisy items	0-65535	64

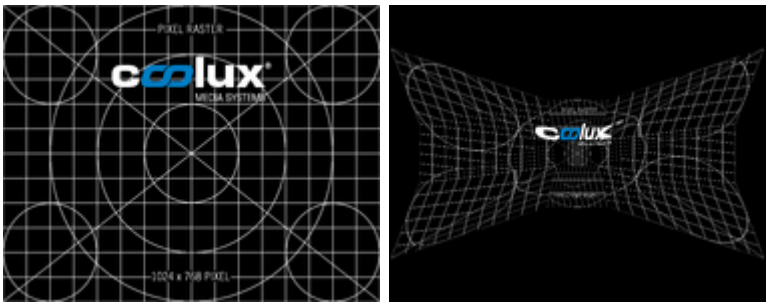
6.5.2.4.15 Geometry

Inflate

Inflates the layer in XYZ, best visible when having a 3D object applied as Mesh. Using just the 2D layer texture (without a mesh!) results in inflating the layer in X&Y, and moving the layer in Z Position closer to the camera.

Parameter	Description	Value Range	Default
Amount	Amount of Inflation	0-65535	0

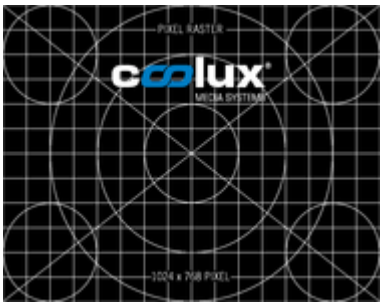
Lens Barrel Distortion - Vertex

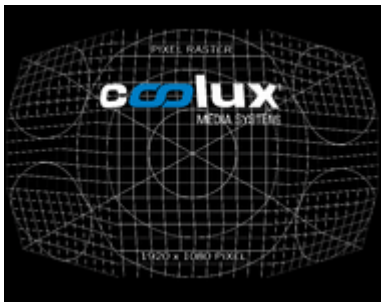


Applies a barrel distortion to a layer or output.

Parameter	Description	Value Range	Default
Lens X	Horizontal lens position	0-65535	32768
Lens Y	Vertical lens position	0-65535	32768
Scale X	Horizontal scale	0-65535	32768
Scale Y	Vertical scale	0-65535	32768
ScaleIn X	Horizontal scale	0-65535	32768
ScaleIn Y	Vertical scale	0-65535	32768
Coef. 0	Coef. 0	0-65535	65535
Coef. 1	Coef. 1	0-65535	0
Coef. 2	Coef. 2	0-65535	0
Coef. 3	Coef. 3	0-65535	0

Lens Pincushion Distortion - Vertex





Applies a barrel distortion to a layer or output.

Parameter	Description	Value Range	Default
Lens X	Horizontal lens position	0-65535	32768
Lens Y	Vertical lens position	0-65535	32768
Scale X	Horizontal scale	0-65535	32768
Scale Y	Vertical scale	0-65535	32768
ScaleIn X	Horizontal scale	0-65535	32768
ScaleIn Y	Vertical scale	0-65535	32768
Coef. 0	Coef. 0	0-65535	65535
Coef. 1	Coef. 1	0-65535	0
Coef. 2	Coef. 2	0-65535	0
Coef. 3	Coef. 3	0-65535	0

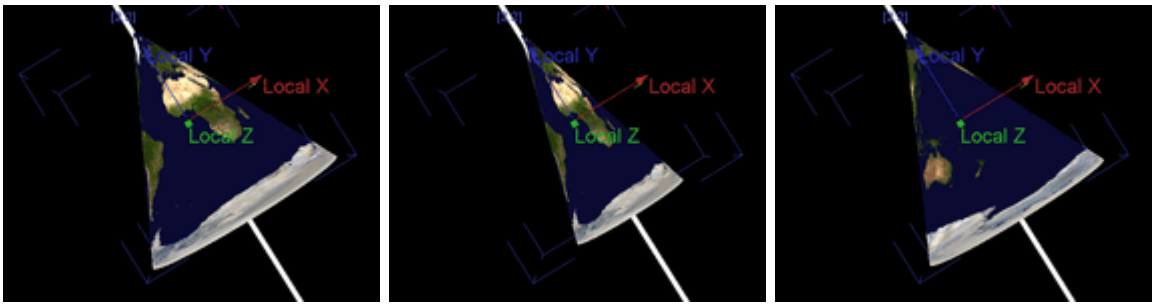
Morph A-B

This effect allows deforming an object by moving its vertices from one assembly to another. Please refer to this live warping tutorial [2174] in the manual.

Morph A-B-C

This effect allows deforming an object by moving its vertices from one assembly to a second and a third one. Please refer to the live warping tutorial [2174] in the manual.

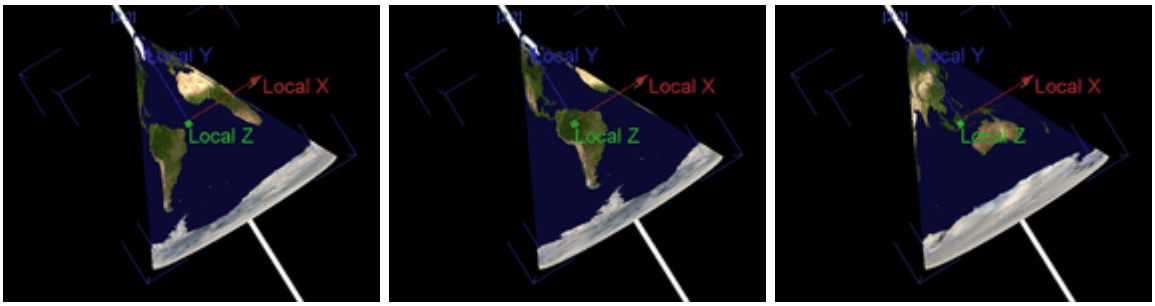
Rotate Local Quaternion



The rotation set up with the default XYZ rotation parameters from a layer depends on the rotation pivot's position. It rotates the object as well as the rotation pivot point.
In contrary, the "Rotate Local Quaternion" parameters apply locally, independent from the pivot's position but using its orientation. It rotates only the object not the rotation pivot point.
The "Rotate Local Quaternion" applies a fixed rotation value based on quaternion parameters in difference to the standard summarization of Euler XYZ rotation angles.

Parameter	Description	Value Range	Default
QX	QX	-1 - +1	0
QY	QY	-1 - +1	0
QZ	QZ	-1 - +1	0
QW	QW	-1 - +1	0

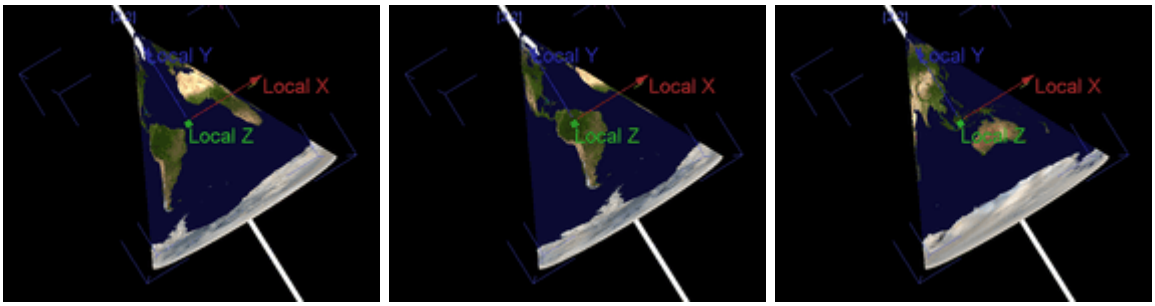
Rotate Local XYZ



The rotation set up with the default XYZ rotation parameters from a layer depends on the rotation pivot's position. It rotates the object as well as the rotation pivot point.
In contrary, the "Rotate Local XYZ" parameters apply locally, independent from the pivot's position but using its orientation. It rotates only the object not the rotation pivot point.
The "Rotate Local XYZ" applies a fixed rotation value, whilst "Rotation Speed Local XYZ" applies a constant rotation over time.

Parameter	Description	Value Range	Default
X	Rotation of local X Axis	0-65535	32768
Y	Rotation of local Y Axis	0-65535	32768
Z	Rotation of local Z Axis	0-65535	32768

Rotation Speed Local XYZ



The rotation set up with the default XYZ rotation parameters from a layer depends on the rotation pivot's position. It rotates the object as well as the rotation pivot point.
In contrary, the "Rotate Local XYZ" parameters apply locally, independent from the pivot's position but using its orientation. It rotates only the object not the rotation pivot point.
The "Rotate Local XYZ" applies a fixed rotation value, whilst "Rotation Speed Local XYZ" applies a constant rotation over time.

Parameter	Description	Value Range	Default
X	X	0-65535	32768
Y	Y	0-65535	32768
Z	Z	0-65535	32768

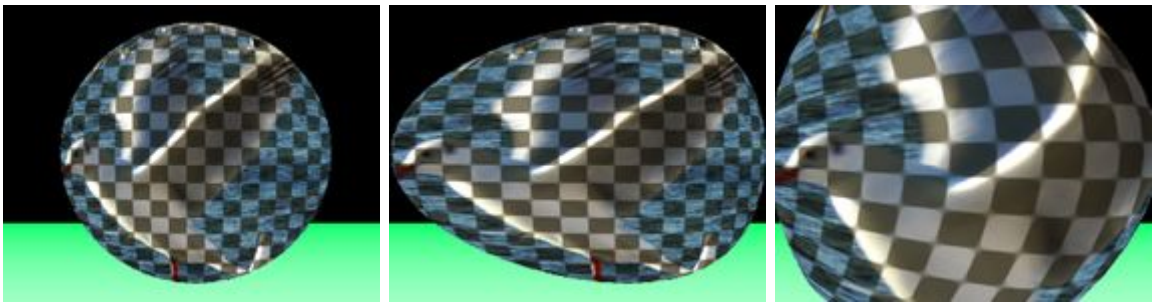
XYZ Push

Pushes the layer texture in XYZ, depending on the settings of XYZ pivots.

Example above:
Left: Effect is not applied
Center: Y Offset is applied; Y Pivot is centered (default value)
Right: X Offset is applied; X Pivot is moved to right edge of the layer

Parameter	Description	Value Range	Default
X Off	X Offset	0-65535	32768
Y Off	Y Offset	0-65535	32768
Z Off	Z Offset	0-65535	32768
PX	Position of X Pivot	0-65535	32768
PY	Position of Y Pivot	0-65535	32768
PZ	Position of Z Pivot	0-65535	32768

XYZ Squeeze

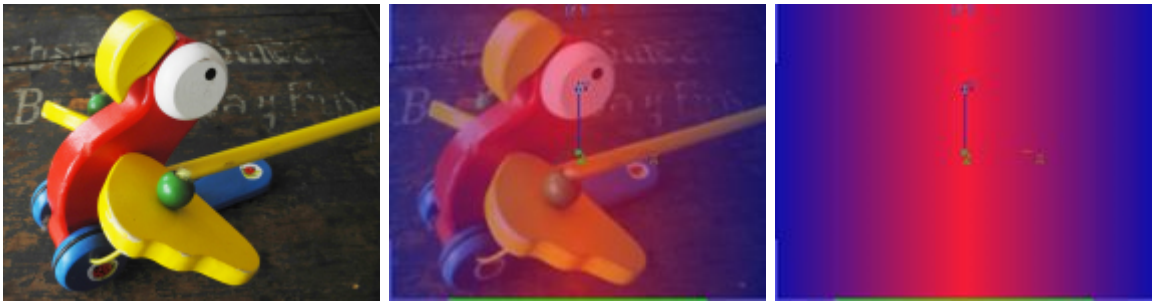


Squeezes the layer texture in XYZ, depending on the settings of XYZ pivots.

Parameter	Description	Value Range	Default
X Off	X Offset	0-65535	0
Y Off	Y Offset	0-65535	0
Z Off	Z Offset	0-65535	0
PX	Position of X Pivot	0-65535	32768
PY	Position of Y Pivot	0-65535	32878
PZ	Position of Z Pivot	0-65535	32768

6.5.2.4.16 Gradients

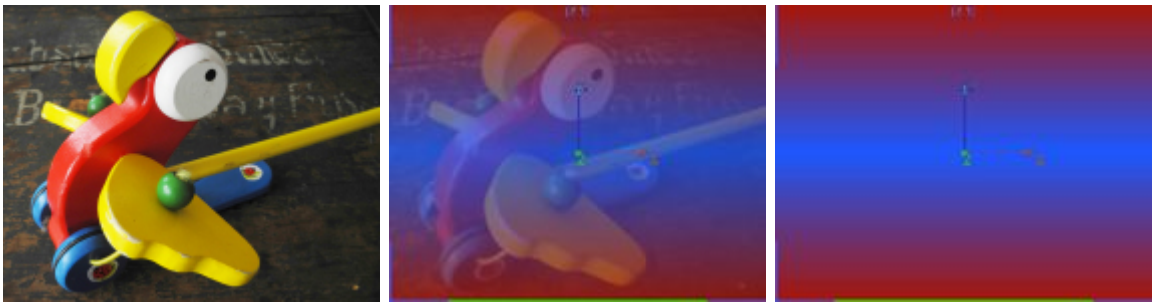
Gradient Linear Horizontal



Allows turning the layer texture into a linear gradient with the possibility of choosing the colors for the horizontal area. Alpha layers for both areas are adjustable.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Red C	Red C	0-255	0
Green C	Green C	0-255	0
Blue C	Blue C	0-255	0
Alpha C	Alpha C	0-255	255

Gradient Linear Vertical



Allows turning the layer texture into a linear gradient with the possibility of choosing the colors for the vertical area. Alpha layers for both areas are adjustable.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Red C	Red C	0-255	0
Green C	Green C	0-255	0
Blue C	Blue C	0-255	0
Alpha C	Alpha C	0-255	255

Gradient Linear

Allows turning the layer texture into a linear gradient with the possibility of choosing the colors for the higher and lower area. Alpha levels for both areas are adjustable.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Red level of higher color, Color Picker	0-255	255
Green H	Green level of higher color, Color Picker	0-255	255
Blue H	Blue level of higher color, Color Picker	0-255	255
Alpha H	Alpha level of higher color	0-255	255
Red L	Red level of lower color, Color Picker	0-255	0
Green L	Green level of lower color, Color Picker	0-255	0
Blue L	Blue level of lower color, Color Picker	0-255	0
Alpha L	Alpha level of lower color	0-255	255

Gradient Quad 16bit

Allows turning the layer texture into a gradient with four different areas with the possibility of choosing 16 bit colors and alpha channels. Size and softness of the quadrangle (four-sided figure) are adjustable.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	65535
Red 1	First color picker / Level of red	0-65535	65535
Green 1	First color picker / Level of green	0-65535	65535
Blue 1	First color picker / Level of blue	0-65535	65535
Alpha 1	Alpha 1	0-65535	65535
Softness 1	Softness 1	0-65535	250
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-65535	65535
Green 2	Second color picker / Level of green	0-65535	0
Blue 2	Second color picker / Level of blue	0-65535	0
Alpha 2	Alpha 2	0-65535	65535
Softness 2	Softness 2	0-65535	250
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-65535	0
Green 3	Third color picker / Level of green	0-65535	65535
Blue 3	Third color picker / Level of blue	0-65535	0
Alpha 3	Alpha 3	0-65535	65535
Softness 3	Softness 3	0-65535	250
Pt4 X	Horizontal position of the fourth point	0-65535	0
Pt4 Y	Vertical position of the fourth point	0-65535	0
Red 4	Fourth color picker / Level of red	0-65535	0
Green 4	Fourth color picker / Level of green	0-65535	0
Blue 4	Fourth color picker / Level of blue	0-65535	65535
Alpha 4	Alpha 4	0-65535	65535
Softness 4	Softness 4	0-65535	250

Gradient Quad Black Lift 16bit

Allows turning the layer texture into a gradient with four different areas with the possibility of choosing 16 bit colors. Size and softness of the quadrangle (four-sided figure) are adjustable. The gradient colors are added to the layer texture but only to dark values. The higher the "Threshold" parameter the lighter the colors get that are influenced by the gradient colors.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	65535
Red 1	First color picker / Level of red	0-65535	10000
Green 1	First color picker / Level of green	0-65535	10000
Blue 1	First color picker / Level of blue	0-65535	10000
Softness 1	Softness 1	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-65535	10000
Green 2	Second color picker / Level of green	0-65535	10000
Blue 2	Second color picker / Level of blue	0-65535	10000
Softness 2	Softness 2	0-65535	0
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-65535	10000
Green 3	Third color picker / Level of green	0-65535	10000
Blue 3	Third color picker / Level of blue	0-65535	10000
Softness 3	Softness 3	0-65535	0
Pt4 X	Horizontal position of the fourth point	0-65535	0
Pt4 Y	Vertical position of the fourth point	0-65535	0
Red 4	Fourth color picker / Level of red	0-65535	10000
Green 4	Fourth color picker / Level of green	0-65535	10000
Blue 4	Fourth color picker / Level of blue	0-65535	10000
Softness 4	Softness 4	0-65535	0
Threshold	Threshold	0-65535	30000

Gradient Quad Black Lift

Allows turning the layer texture into a gradient with four different areas with the possibility of choosing 8 bit colors. Size and softness of the quadrangle (four-sided figure) are adjustable. The gradient colors are added to the layer texture but only to dark values. The higher the "Threshold" parameter the lighter the colors get that are influenced by the gradient colors.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	65535
Red 1	First color picker / Level of red	0-255	30
Green 1	First color picker / Level of green	0-255	30
Blue 1	First color picker / Level of blue	0-255	30
Softness 1	Softness 1	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-255	30
Green 2	Second color picker / Level of green	0-255	30
Blue 2	Second color picker / Level of blue	0-255	30
Softness 2	Softness 2	0-65535	0
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-255	30
Green 3	Third color picker / Level of green	0-255	30
Blue 3	Third color picker / Level of blue	0-255	30
Softness 3	Softness 3	0-65535	0
Pt4 X	Horizontal position of the fourth point	0-65535	0
Pt4 Y	Vertical position of the fourth point	0-65535	0
Red 4	Fourth color picker / Level of red	0-255	30
Green 4	Fourth color picker / Level of green	0-255	30
Blue 4	Fourth color picker / Level of blue	0-255	30
Softness 4	Softness 4	0-65535	0
Threshold	Threshold	0-255	75

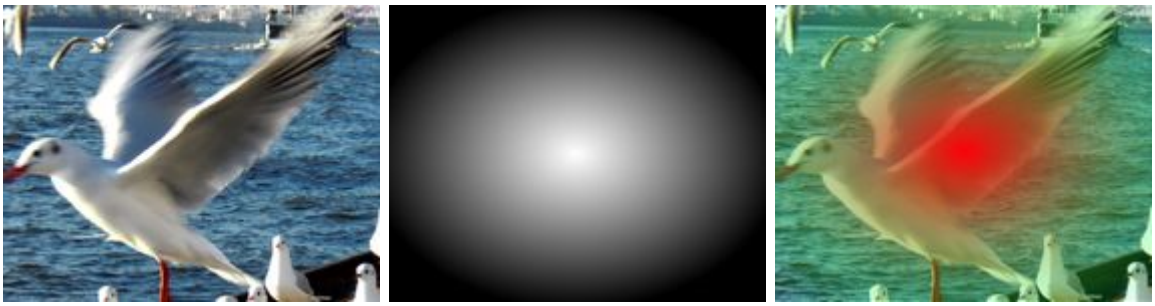
Gradient Quad

Allows turning the layer texture into a gradient with four different areas with the possibility of choosing 8 bit colors and alpha channels. Size and softness of the quadrangle (four-sided figure) are adjustable.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	65535
Red 1	First color picker / Level of red	0-255	255
Green 1	First color picker / Level of green	0-255	255
Blue 1	First color picker / Level of blue	0-255	255
Alpha 1	Alpha 1	0-255	255
Softness 1	Softness 1	0-65535	250
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-255	255
Green 2	Second color picker / Level of green	0-255	0
Blue 2	Second color picker / Level of blue	0-255	0
Alpha 2	Alpha 2	0-255	255
Softness 2	Softness 2	0-65535	250
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-255	0
Green 3	Third color picker / Level of green	0-255	255
Blue 3	Third color picker / Level of blue	0-255	0
Alpha 3	Alpha 3	0-255	255
Softness 3	Softness 3	0-65535	250
Pt4 X	Horizontal position of the fourth point	0-65535	0
Pt4 Y	Vertical position of the fourth point	0-65535	0
Red 4	Fourth color picker / Level of red	0-255	0
Green 4	Fourth color picker / Level of green	0-255	0
Blue 4	Fourth color picker / Level of blue	0-255	255
Alpha 4	Alpha 4	0-255	255
Softness 4	Softness 4	0-65535	250

Gradient Radial



Allows turning the layer texture into a radial gradient with the possibility of choosing the colors for the higher and lower area. Alpha levels for both areas are adjustable.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Red level of higher color, Color Picker	0-255	255
Green H	Green level of higher color, Color Picker	0-255	255
Blue H	Blue level of higher color, Color Picker	0-255	255
Alpha H	Alpha level of higher color	0-255	255
Red L	Red level of lower color, Color Picker	0-255	0
Green L	Green level of lower color, Color Picker	0-255	0
Blue L	Blue level of lower color, Color Picker	0-255	0
Alpha L	Alpha level of lower color	0-255	255

Gradient Spiral

Allows turning the layer texture into a spiral gradient with the possibility of choosing the colors for the higher and lower area. Alpha levels for both areas are adjustable. The amount of the spiral is influenced by the opacity value of the layer as well.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Red level of higher color, Color Picker	0-255	255
Green H	Green level of higher color, Color Picker	0-255	255
Blue H	Blue level of higher color, Color Picker	0-255	255
Alpha H	Alpha level of higher color	0-255	255
Red L	Red level of lower color, Color Picker	0-255	0
Green L	Green level of lower color, Color Picker	0-255	0
Blue L	Blue level of lower color, Color Picker	0-255	0
Alpha L	Alpha level of lower color	0-255	255

Gradient Swirl

Allows turning the layer texture into a swirl gradient with the possibility of choosing the colors for the higher and lower area. Alpha levels for both areas are adjustable.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Red level of higher color, Color Picker	0-255	255
Green H	Green level of higher color, Color Picker	0-255	255
Blue H	Blue level of higher color, Color Picker	0-255	255
Alpha H	Alpha level of higher color	0-255	255
Red L	Red level of lower color, Color Picker	0-255	0
Green L	Green level of lower color, Color Picker	0-255	0
Blue L	Blue level of lower color, Color Picker	0-255	0
Alpha L	Alpha level of lower color	0-255	255

Gradient Triangle 16bit

Allows turning the layer texture into a gradient with three different areas with the possibility of choosing 16 bit colors and alpha channels. Size, softness and rotation angle of the triangular (3-sided figure) are adjustable. The gradient colors are added to the layer texture.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Red 1	First color picker / Level of red	0-65535	65535
Green 1	First color picker / Level of green	0-65535	65535
Blue 1	First color picker / Level of blue	0-65535	65535
Alpha 1	Alpha 1	0-255	255
Softness 1	Softness 1	0-65535	250
Pt2 X	Horizontal position of the second point	0-65535	32768
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-65535	65535
Green 2	Second color picker / Level of green	0-65535	0
Blue 2	Second color picker / Level of blue	0-65535	0
Alpha 2	Alpha 2	0-255	255
Softness 2	Softness 2	0-65535	250
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-65535	0
Green 3	Third color picker / Level of green	0-65535	65535
Blue 3	Third color picker / Level of blue	0-65535	0
Alpha 3	Alpha 3	0-255	255
Softness 3	Softness 3	0-65535	250
Angle	Angle	0-360	180

Gradient Triangle Black Lift 16bit

Allows turning the layer texture into a gradient with three different areas with the possibility of choosing 16 bit colors. Size, softness and rotation angle of the triangular (3-sided figure) are adjustable. The gradient colors are added to the layer texture but only to dark values. The higher the "Threshold" parameter the lighter the colors get that are influenced by the gradient colors.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Red 1	First color picker / Level of red	0-65535	10000
Green 1	First color picker / Level of green	0-65535	10000
Blue 1	First color picker / Level of blue	0-65535	10000
Softness 1	Softness 1	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	32768
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-65535	10000
Green 2	Second color picker / Level of green	0-65535	10000
Blue 2	Second color picker / Level of blue	0-65535	10000
Softness 2	Softness 2	0-65535	0
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-65535	10000
Green 3	Third color picker / Level of green	0-65535	10000
Blue 3	Third color picker / Level of blue	0-65535	10000
Softness 3	Softness 3	0-65535	0
Angle	Angle	0-360	180
Threshold	Threshold	0-65535	30000

Gradient Triangle Black Lift

Allows turning the layer texture into a gradient with three different areas with the possibility of choosing 8 bit colors. Size, softness and rotation angle of the triangular (3-sided figure) are adjustable. The gradient colors are added to the layer texture but only to dark values. The higher the "Threshold" parameter the lighter the colors get that are influenced by the gradient colors.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Red 1	First color picker / Level of red	0-255	30
Green 1	First color picker / Level of green	0-255	30
Blue 1	First color picker / Level of blue	0-255	30
Softness 1	Softness 1	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	32768
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-255	30
Green 2	Second color picker / Level of green	0-255	30
Blue 2	Second color picker / Level of blue	0-255	30
Softness 2	Softness 2	0-65535	0
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-255	30
Green 3	Third color picker / Level of green	0-255	30
Blue 3	Third color picker / Level of blue	0-255	30
Softness 3	Softness 3	0-65535	0
Angle	Angle	0-360	180
Threshold	Threshold	0-255	75

Gradient Triangle

Allows turning the layer texture into a gradient with three different areas with the possibility of choosing 8 bit colors and alpha channels. Size, softness and rotation angle of the triangular (3-sided figure) are adjustable.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Red 1	First color picker / Level of red	0-255	255
Green 1	First color picker / Level of green	0-255	255
Blue 1	First color picker / Level of blue	0-255	255
Alpha 1	Alpha 1	0-255	255
Softness 1	Softness 1	0-65535	250
Pt2 X	Horizontal position of the second point	0-65535	32768
Pt2 Y	Vertical position of the second point	0-65535	65535
Red 2	Second color picker / Level of red	0-255	255
Green 2	Second color picker / Level of green	0-255	0
Blue 2	Second color picker / Level of blue	0-255	0
Alpha 2	Alpha 2	0-255	255
Softness 2	Softness 2	0-65535	250
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red 3	Third color picker / Level of red	0-255	0
Green 3	Third color picker / Level of green	0-255	255
Blue 3	Third color picker / Level of blue	0-255	0
Alpha 3	Alpha 3	0-255	255
Softness 3	Softness 3	0-65535	250
Angle	Angle	0-360	180

6.5.2.4.17 Image Adjust

BasicColorCorrection

This is a combination of the following three effects that are mostly used when your content requires a basic color correction:

First, an "RGB max [474]" effect allows setting a maximum level for the red, green and blue color channel.
Second, the "Gamma RGB [467]" effect allows shifting the RGB pixel values (individually or together) higher or lower to brighten or darken the image. Technically the red, green and blue channels are multiplied with a factor. In difference to the "Brighten [464]" effect, this gamma correction is a non-linear function.
Third, the "ColorContrastBrightness [466]" (CCB) effect allows adjusting Color, Contrast and Brightness. The Chroma parameter multiplies each RGB value with a factor. The Contrast parameter multiplies bright colors with a factor to make them brighter and divides dark colors with a factor to darken them further. The Brightness parameter adds a value to each RGB value equally.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
R max	Max. level of Red	0-255	255
G max	Max. level of Green	0-255	255
B max	Max. level of Blue	0-255	255
Gamma R	Gamma Red	0-65535	16384
Gamma G	Gamma Green	0-65535	16384
Gamma B	Gamma Blue	0-65535	16384
Chroma	Level of Chroma	0-255	128
Contrast	Level of Contrast	0-255	128
Bright	Level of Brightness	0-255	128

Brighten

Brighten simply shifts all pixel values higher or lower when adjusting the Brighten Factor. Technically, each RGB value is multiplied with a factor. In difference to the "Gamma RGB [467]" effect, this is a linear function.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Factor	Factor of Brightness	0-255	32

Burn

The Burn Effect is used to lighten areas of the image. The Threshold parameter allows defining the intensity value of the pixels from which on Burn should take effect. Use the Factor parameter to set the amount of burning.

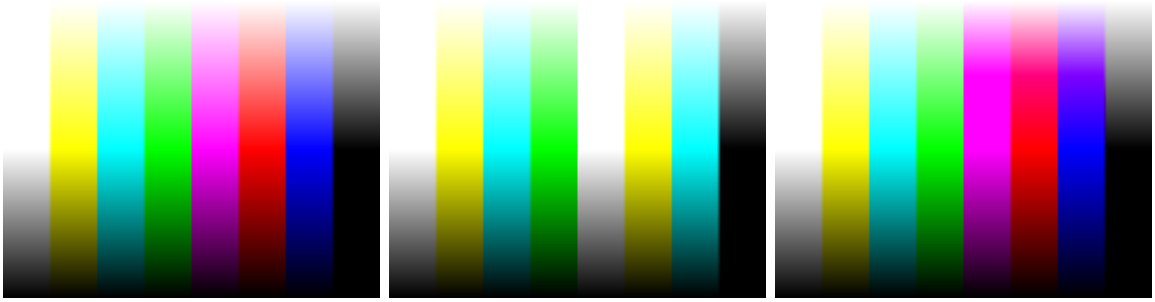
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Factor	Factor of Burning	0-255	32
Threshold	Threshold of intensity value	0-255	64

CCB Multiply

This is an alternative to the "ColorContrastBrightness" 466 effect. It allows adjusting Color, Contrast and Brightness.
The Chroma parameter itself works like the "Brighten" effect, it multiplies each RGB value with a factor. The Contrast parameter multiplies bright colors with a factor to make them brighter and divides dark colors with a factor to darken them further. The Brightness parameter does exactly the same as the Chroma parameter (in difference to the normal CCB FX).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Chroma	Chroma	0-255	128
Contrast	Contrast	0-255	128
Bright	Bright	0-255	128

CMYK Adjust



This increases or decreases the original color values based on the CMYK values. In contrast to the additive RGB (red,green,blue) color model, CMYK (cyan, magenta, yellow and black) is a subtractive color model. The second image shows the result when the "Magenta" fader is set to 0. All magenta parts of the image become pure white, all red (=M+Y) parts become yellow and blue (=M+C) become cyan. For the third image, the same fader was set to maximum and the same colors get now more saturated.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
C	Cyan	0-255	128
M	Magenta	0-255	128
Y	Yellow	0-255	128
K	Black	0-255	128

ColorContrastBrightness

The CCB Effect allows adjusting Color, Contrast and Brightness. The Chroma parameter itself works like the "Brighten" effect, it multiplies each RGB value with a factor. The Contrast parameter multiplies bright colors with a factor to make them brighter and divides dark colors with a factor to darken them further. The Brightness parameter adds a value to each RGB value equally.
Note, there is an alternative to this effect, the "CCB Multiply" 465. There, the Brightness parameter works in the same way as the Chroma one.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Chroma	Level of Chroma	0-255	128
Contrast	Level of Contrast	0-255	128
Brightness	Level of Brightness	0-255	128

Gamma RGB

This shifts the RGB pixel values (individually or together) higher or lower to brighten or darken the image. Technically each RGB value is multiplied with a factor. In difference to the "Brighten [464]" effect, this gamma correction is a non-linear function.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Gamma R	Gamma R	0-65535	16384
Gamma G	Gamma G	0-65535	16384
Gamma B	Gamma B	0-65535	16384

HighMidLowLights

Creates a lighting effect on the image, the colors for High, Mid and Low Lights can be adjusted separately.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Level of Red for High Lights	0-255	128
Green H	Level of Green for High Lights		
Blue H	Level of Blue for High Lights		
Red M	Level of Red for Mid Lights	0-255	128
Green M	Level of Green for Mid Lights		
Blue M	Level of Blue for Mid Lights		
Red L	Level of Red for Low Lights		
Green L	Level of Green for Low Lights		
Blue L	Level of Blue for Low Lights	0-255	128

HSV Adjust

Changes the color of the image based on the HSV (hue, saturation, value) RGB color model, also called HSB (hue, saturation, brightness).

Changing the Hue in- or decreases all color hues in the image accordingly.
Changing the Saturation, in- or decreases the saturation (white to full color) of all colors in the image.
Changing the Value, in- or decreases the brightness value (black to full color) of all colors in the image.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Hue	Hue	0-255	128
Saturation	Saturation	0-255	128
Value	Value	0-255	128

HSV by Hue

Changes the color of the image based on the HSV (hue, saturation, value) RGB color model, also called HSB (hue, saturation, brightness).
Pick a color / hue with color picker and set a threshold to influence neighboring colors.
The A Set of the following faders influences the chosen colors, the B set all other ones.

Changing the Hue in- or decreases the color hue of the color set.
Changing the Saturation, in- or decreases the saturation (white to full color) of the color set.
Changing the Value, in- or decreases the brightness value (black to full color) of the color set.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Threshold	Threshold	0-255	32
Hue A	Hue A	0-255	128
Saturation A	Saturation A	0-255	128
Value A	Value A	0-255	128
Hue B	Hue B	0-255	128
Saturation B	Saturation B	0-255	128
Value B	Value B	0-255	128

HSV-HS for V range

Changes the color of the image based on the HSV (hue, saturation, value) RGB color model, also called HSB (hue, saturation, brightness).
Pick a value (black to full color) and set the value range using the "Width" parameter. Note that a high width influences a small range and vice versa.

Changing the Hue in- or decreases the color hue for the value range.
Changing the Saturation, in- or decreases the saturation (white to full color) for the value range.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Value	Value	0-65535	32768
Width	Width	0-65535	8192
Hue	Hue	0-65535	32768
Saturation	Saturation	0-65535	32768

Levels RGB

Allows setting the gradation levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Dark Red	Adjusting the dark limit for Red	0-255	0
Light Red	Adjusting the light limit for Red	0-255	255
Dark Green	Adjusting the dark limit for Green	0-255	0
Light Green	Adjusting the light limit for Green	0-255	255
Dark Blue	Adjusting the dark limit for Blue	0-255	0
Light Blue	Adjusting the light limit for Blue	0-255	255

Levels

Allows setting the overall level gradation.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Dark	Adjusting the dark limit	0-255	0
Light	Adjusting the light limit	0-255	255

Posterize

Posterize allows reducing the colour range by adjusting the amount of steps.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Steps	Amount of steps	0-65535	4096

RGB Add

Adds RGB colors by the RGB color mix value, allows inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128
Invert	Inverts the RGB Levels	0-255	0

RGB Clipper

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red Min	Min. level of Red	0-255	0
Red Max	Max. level of Red	0-255	255
Green Min	Min. level of Green	0-255	0
Green Max	Max. level of Green	0-255	255
Blue Min	Min. level of Blue	0-255	0
Blue Max	Max. level of Blue	0-255	255

RGB CTB

RGB CTB (color temperature blue) emulates the CTB filter that converts tungsten light of 3200K to 'daylight' color.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amt	Amount of conversion into ‘daylight' color	0-255	0

RGB CTO

RGB CTO (color temperature orange) emulates the CTO filter that converts 'daylight' color to tungsten light of 3200K.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amt	Amount of conversion into tungsten light	0-255	0

RGB Gamma

Allows adjusting the gamma for all three channels, RGB. The default value of 255 for RGB equates to a gamma value of 1.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
R	Gamma Red,	0-255	255
G	Gamma Green	0-255	255
B	Gamma Blue	0-255	255

RGB Min Max

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red Min	Min. level of Red	0-255	128
Red Max	Max. level of Red	0-255	255
Green Min	Min. level of Green	0-255	128
Green Max	Max. level of Green	0-255	255
Blue Min	Min. level of Blue	0-255	128
Blue Max	Max. level of Blue	0-255	255

RGB Multiply

All RGB colors are multiplied by RGB color mix values. Multiply can be factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Factor	Color Multiply Factor	0-255	64
Invert	Inverts the multiplied RGB levels	0-255	0

RGB Squisher

Squishes the RGB levels. For lower parameters this effect can be compared to the "Levels RGB" 470 effect.

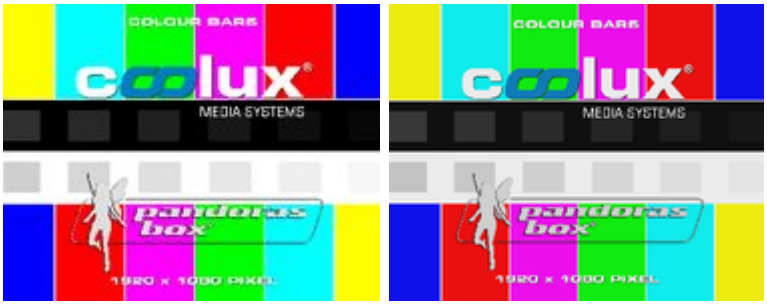
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red Min	Min. level of Red	0-255	128
Red Max	Max. level of Red	0-255	255
Green Min	Min. level of Green	0-255	128
Green Max	Max. level of Green	0-255	255
Blue Min	Min. level of Blue	0-255	128
Blue Max	Max. level of Blue	0-255	255

RGB to HSL

Transforms the image from the RGB color space into the HSL color space.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

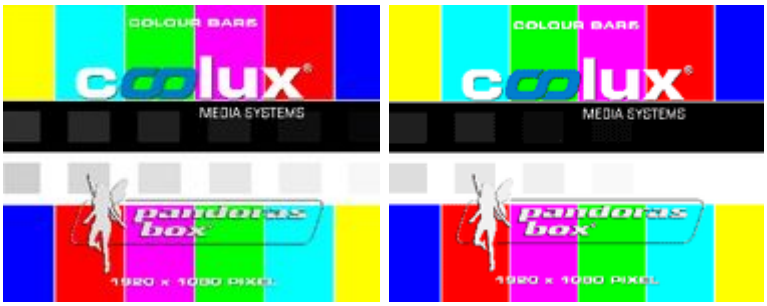
RGB to Video Colors 16-235



Transforms the image colors from RGB (value range: 0 – 255) to Video Colors (range: 16-235).

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

Video Colors 16-235 to RGB



Transforms the image from Video Colors (range: 16-235) to RGB (value range: 0 – 255).

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

Video RGB CC

Changes the RGB values of the image. There is a "Lower" and a "Higher" fader for each Red, Green and Blue. The "Lower" fader subtracts the according color from all image colors. For the second image red was subtracted. The "Higher" fader increases the according color by multiplying it with a factor. For the last image red was enhanced.

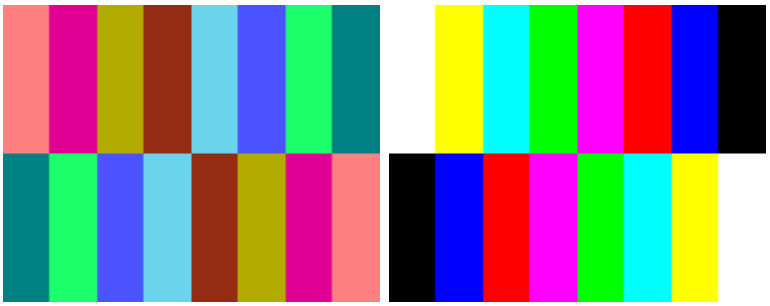
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
R Lo	R Lo	0-255	18
R Hi	R Hi	0-255	43
G Lo	G Lo	0-255	18
G Hi	G Hi	0-255	43
B Lo	B Lo	0-255	18
B Hi	B Hi	0-255	43

Video YUV to RGB

Converts the YUV colorspace (PAL / NTSC) to RGB color space.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

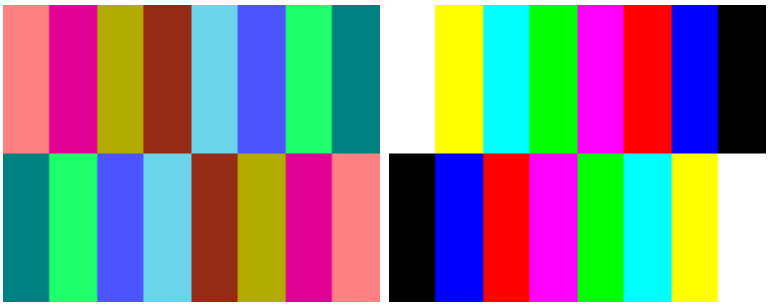
YUV(bt709)2RGB



This effect is of interest when your media file (e.g. a camera) transmits YUV color values which are wrongly interpreted as RGB values which therefore results in wrong colors. The effect leads to a proper result as the values are now interpreted as values based on the YUV BT.709 color space and converted correctly to RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

YUV2RGB



This effect is of interest when your media file (e.g. a camera) transmits YUV color values which are wrongly interpreted as RGB values which therefore results in wrong colors. The effect leads to a proper result as the values are now interpreted as values based on the YUV BT.601 color space and converted correctly to RGB values.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

6.5.2.4.18 Image Adjust - Crop

CCB Crop Edges (px)

Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

The CCB Effect allows adjusting Color, Contrast and Brightness. The Chroma parameter multiplies each RGB value with a factor. The Contrast parameter multiplies bright colors with a factor to make them brighter and divides dark colors with a factor to darken them further. The Brightness parameter adds a value to each RGB value equally.

Note, there is an alternative to this effect, the "CCB Multiply". There, the Brightness parameter works in the same way as the Chroma one.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Chroma	Chroma	0-255	128
Contrast	Contrast	0-255	128
Bright	Bright	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

High Mid Low Crop Pixel

Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Creates a lighting effect on the image, the colors for High, Mid and Low Lights can be adjusted separately.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red M	Red M	0-255	128
Green M	Green M	0-255	128
Blue M	Blue M	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

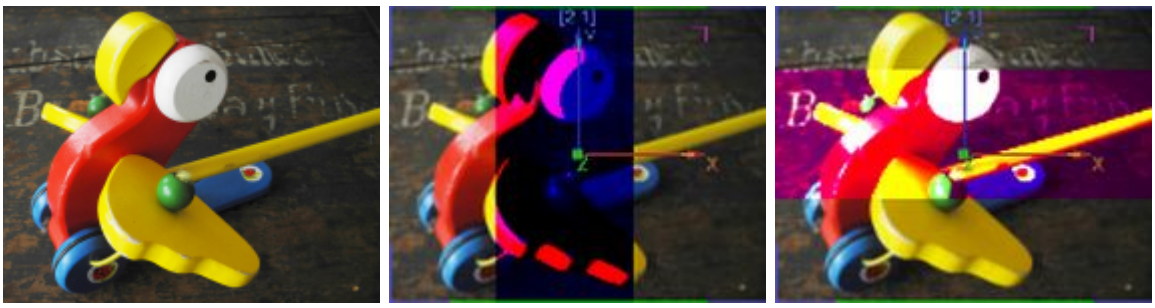
Levels Crop Edges (px)

Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Allows setting the overall level gradation.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark	Dark	0-255	0
Light	Light	0-255	255
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

Levels RGB Crop Edges (px)

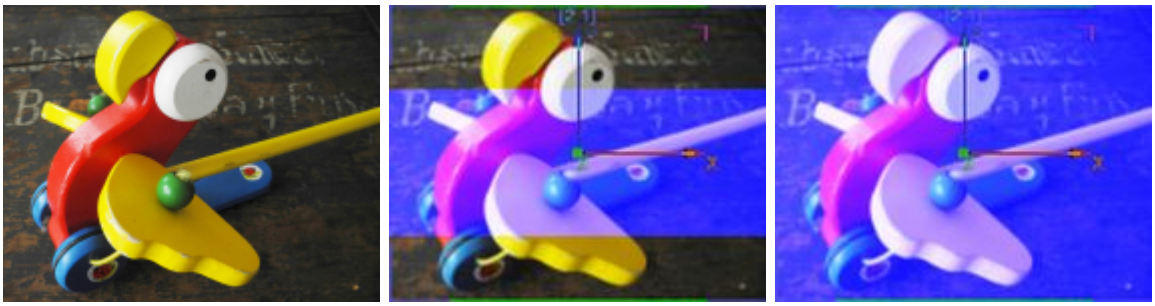


Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Allows setting the gradation levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark Red	Dark Red	0-255	0
Light Red	Light Red	0-255	255
Dark Green	Dark Green	0-255	0
Light Green	Light Green	0-255	255
Dark Blue	Dark Blue	0-255	0
Light Blue	Light Blue	0-255	255
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB Add Crop Edges (px)

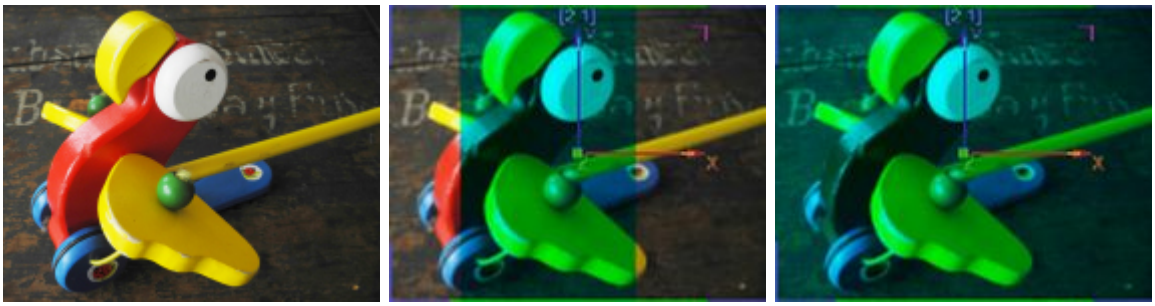


Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Adds RGB colors by the RGB color mix value, allows inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Invert	Inverts the RGB levels or the effect	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB Clipper Crop Edges (px)

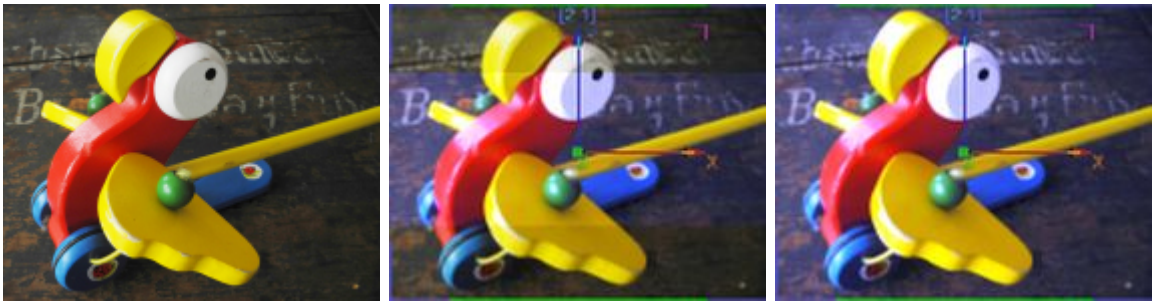


Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB CTB Crop Edges (px)

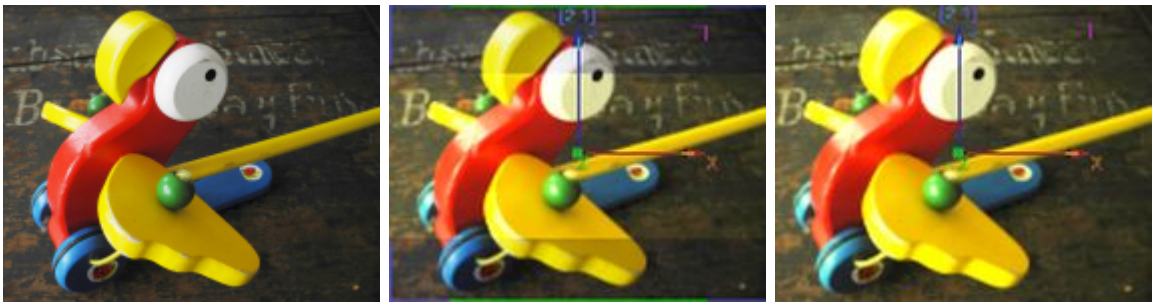


Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

RGB CTB (color temperature blue) emulates the CTB filter that converts tungsten light of 3200K to 'daylight' color.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB CTO Crop Edges (px)

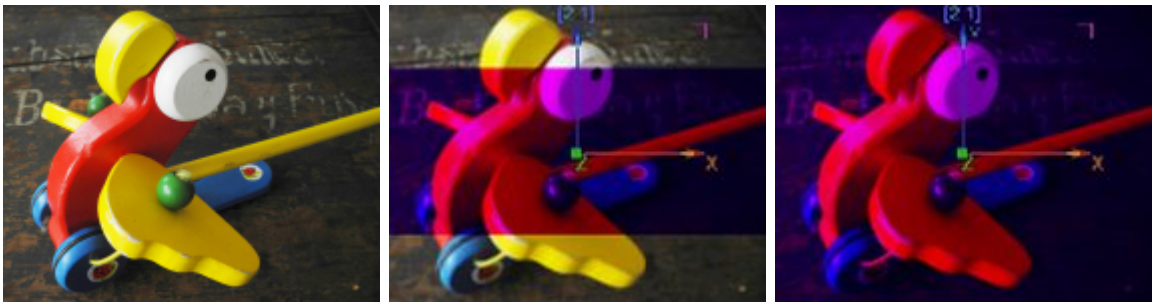


Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

RGB CTO (color temperature orange) emulates the CTO filter that converts 'daylight' color to tungsten light of 3200K.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB Gamma Crop Edges (px)

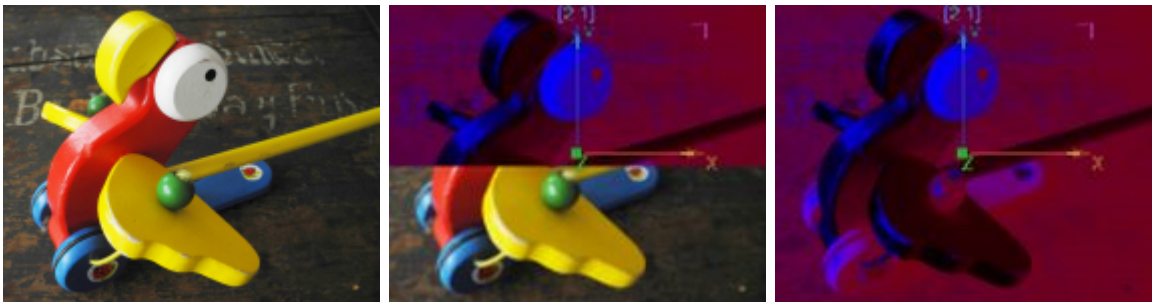


Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Allows adjusting the gamma for all three channels, RGB. The default value of 255 for RGB equates to a gamma value of 1.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
R	Color picker / Level of red	0-255	255
G	Color picker / Level of green	0-255	255
B	Color picker / Level of blue	0-255	255
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB Min Max Crop Edges (px)



Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	128
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	128
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	128
Blue Max	Blue Max	0-255	255
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

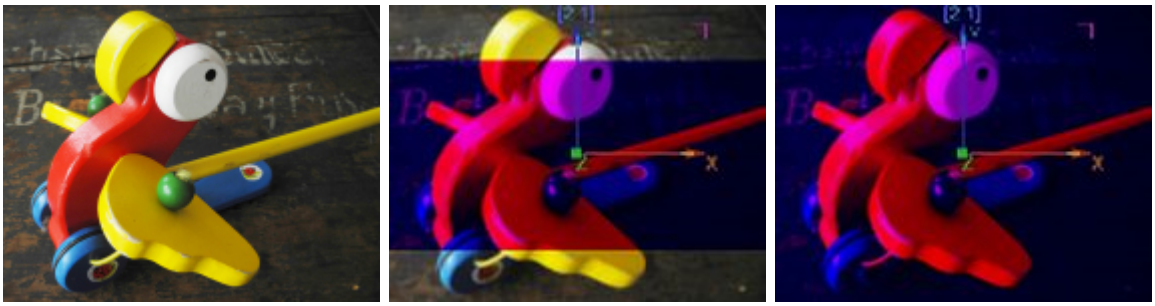
RGB Multiply Crop Edges (px)

Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

All RGB colors are multiplied by RGB color mix values. Multiply can be factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Factor	Factor	0-255	32
Invert	Inverts the RGB levels or the effect	0-255	0
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

RGB Squisher Crop Edges (px)



Applies the effect within a rectangular area with hard borders; the rectangle's edges can be shifted in- or outwards independently.

Squishes the RGB levels. For lower parameters this effect can be compared to the "Levels RGB" [480] effect.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Left (px)	Left (px)	0-8192	0
Right (px)	Right (px)	0-8192	0
Top (px)	Top (px)	0-8192	0
Bottom (px)	Bottom (px)	0-8192	0

6.5.2.4.19 Image Adjust - Iris

CCB Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

The CCB Effect allows adjusting Color, Contrast and Brightness. The Chroma parameter multiplies each RGB value with a factor. The Contrast parameter multiplies bright colors with a factor to make them brighter and divides dark colors with a factor to darken them further. The Brightness parameter adds a value to each RGB value equally.

Note, there is an alternative to this effect, the "CCB Multiply". There, the Brightness parameter works in the same way as the Chroma one.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Chroma	Chroma	0-255	128
Contrast	Contrast	0-255	128
Bright	Bright	0-255	128
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

High Mid Low Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Creates a lighting effect on the image, the colors for High, Mid and Low Lights can be adjusted separately.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red M	Red M	0-255	128
Green M	Green M	0-255	128
Blue M	Blue M	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

Levels Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Allows setting the overall level gradation.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark	Dark	0-255	0
Light	Light	0-255	255
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

Levels RGB Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Allows setting the gradation levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark Red	Dark Red	0-255	0
Light Red	Light Red	0-255	255
Dark Green	Dark Green	0-255	0
Light Green	Light Green	0-255	255
Dark Blue	Dark Blue	0-255	0
Light Blue	Light Blue	0-255	255
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

RGB Add Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Adds RGB colors by the RGB color mix value, allows inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Invert	Inverts the RGB levels or the effect	0-255	0
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

RGB Clipper Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

RGB CTB Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

RGB CTB (color temperature blue) emulates the CTB filter that converts tungsten light of 3200K to 'daylight' color.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

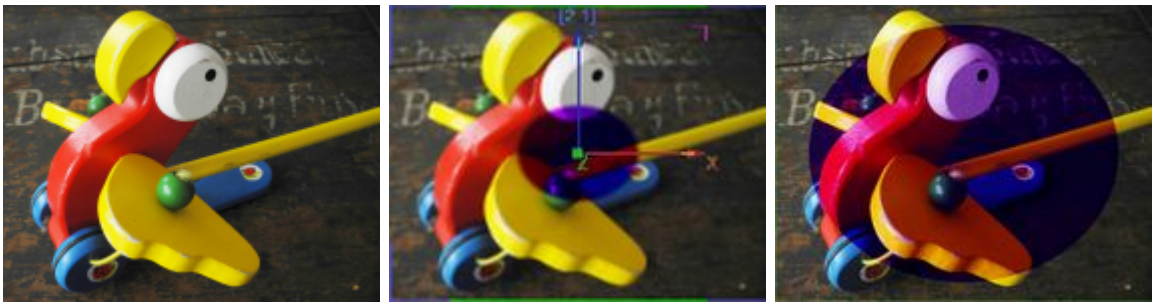
RGB CTO Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

RGB CTO (color temperature orange) emulates the CTO filter that converts 'daylight' color to tungsten light of 3200K.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

RGB Gamma Iris

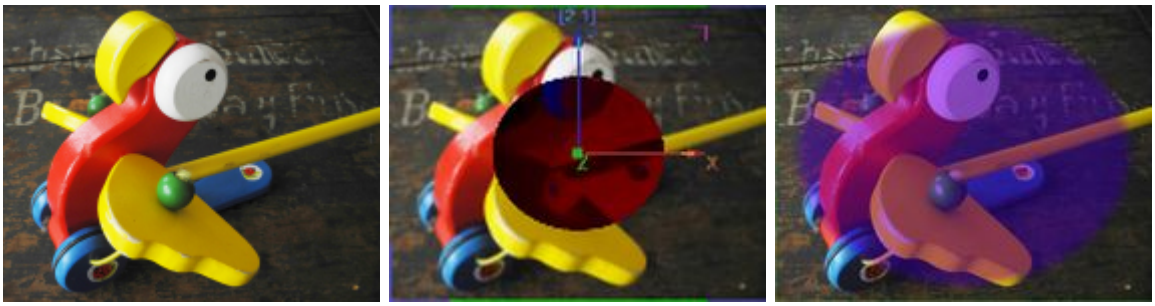


Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Allows adjusting the gamma for all three channels, RGB. The default value of 255 for RGB equates to a gamma value of 1.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
R	Color picker / Level of red	0-255	255
G	Color picker / Level of green	0-255	255
B	Color picker / Level of blue	0-255	255
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

RGB Min Max Iris



Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	128
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	128
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	128
Blue Max	Blue Max	0-255	255
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

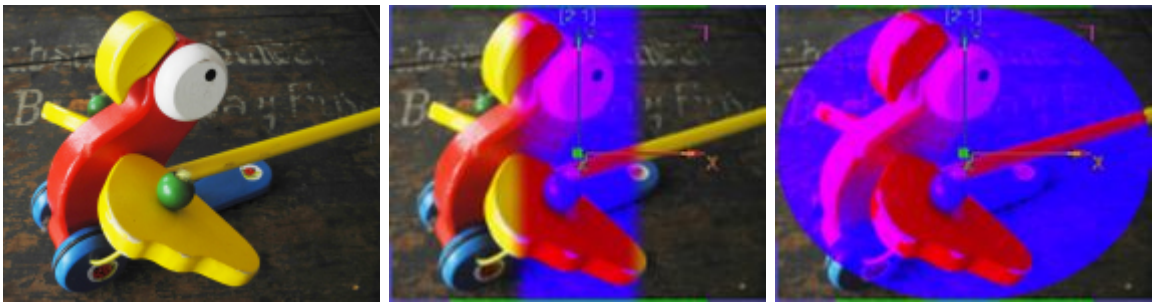
RGB Multiply Iris

Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

All RGB colors are multiplied by RGB color mix values. Multiply can be factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Factor	Factor	0-255	32
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

RGB Squisher Iris



Applies the effect within an elliptical area with a hard or soft border. Size, position and aspect of the ellipse are adjustable too, and the effect can be turned inside out.

Squishes the RGB levels. For lower parameters this effect can be compared to the "Levels RGB" 490 effect.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Size	Size	0-255	128
Softness	Softness	0-255	64
U	U	0-65535	32768
V	V	0-65535	32768
AspectIris	AspectIris	0-255	128
Invert	Inverts the RGB levels or the effect	0-255	0

6.5.2.4.20 Image Adjust - Mediamask

CCB MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

The CCB Effect allows adjusting Color, Contrast and Brightness. The Chroma parameter multiplies each RGB value with a factor. The Contrast parameter multiplies bright colors with a factor to make them brighter and divides dark colors with a factor to darken them further. The Brightness parameter adds a value to each RGB value equally.

Note, there is an alternative to this effect, the "CCB Multiply". There, the Brightness parameter works in the same way as the Chroma one.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Chroma	Chroma	0-255	128
Contrast	Contrast	0-255	128
Bright	Bright	0-255	128
Media	Media file as source for overlay or mask	-	-

High Mid Low MediaMask

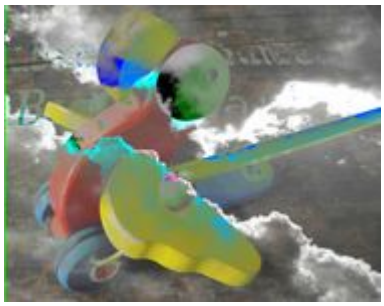

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Creates a lighting effect on the image, the colors for High, Mid and Low Lights can be adjusted separately.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red M	Red M	0-255	128
Green M	Green M	0-255	128
Blue M	Blue M	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Media	Media file as source for overlay or mask	-	-

Levels MediaMask





Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Allows setting the overall level gradation.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark	Dark	0-255	0
Light	Light	0-255	255
Media	Media file as source for overlay or mask	-	-

Levels RGB MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Allows setting the gradation levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark Red	Dark Red	0-255	0
Light Red	Light Red	0-255	255
Dark Green	Dark Green	0-255	0
Light Green	Light Green	0-255	255
Dark Blue	Dark Blue	0-255	0
Light Blue	Light Blue	0-255	255
Media	Media file as source for overlay or mask	-	-

RGB Add MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Adds RGB colors by the RGB color mix value, allows inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Invert	Inverts the RGB levels or the effect	0-255	0
Media	Media file as source for overlay or mask	-	-

RGB Clipper MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

RGB CTB (color temperature blue) emulates the CTB filter that converts tungsten light of 3200K to 'daylight' color.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Media	Media file as source for overlay or mask	-	-

RGB CTB MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Allows adjusting the gamma for all three channels, RGB. The default value of 255 for RGB equates to a gamma value of 1.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Media	Media file as source for overlay or mask	-	-

RGB CTO MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

RGB CTO (color temperature orange) emulates the CTO filter that converts 'daylight' color to tungsten light of 3200K.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Media	Media file as source for overlay or mask	-	-

RGB Gamma MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Allows adjusting the gamma for all three channels, RGB. The default value of 255 for RGB equates to a gamma value of 1.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
R	Color picker / Level of red	0-255	255
G	Color picker / Level of green	0-255	255
B	Color picker / Level of blue	0-255	255
Media	Media file as source for overlay or mask	-	-

RGB Min Max MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	128
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	128
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	128
Blue Max	Blue Max	0-255	255
Media	Media file as source for overlay or mask	-	-

RGB Multiply MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

All RGB colors are multiplied by RGB color mix values. Multiply can be factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Factor	Factor	0-255	32
Invert	Inverts the RGB levels or the effect	0-255	0
Media	Media file as source for overlay or mask	-	-

RGB Squisher MediaMask

Applies the effect according to a mask. White areas in the mask apply the effect fully. Less brighter areas reduce the effect accordingly.

Squishes the RGB levels. For lower parameters this effect can be compared to the "Levels RGB" [500] effect.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Media	Media file as source for overlay or mask	-	-

6.5.2.4.21 Image Adjust - Softborder

CCB Softborder

Applies the effect within an adjustable rectangular area with soft edges.

The CCB Effect allows adjusting Color, Contrast and Brightness. The Chroma parameter multiplies each RGB value with a factor. The Contrast parameter multiplies bright colors with a factor to make them brighter and divides dark colors with a factor to darken them further. The Brightness parameter adds a value to each RGB value equally.

Note, there is an alternative to this effect, the "CCB Multiply". There, the Brightness parameter works in the same way as the Chroma one.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Chroma	Chroma	0-255	128
Contrast	Contrast	0-255	128
Bright	Bright	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

High Mid Low Softborder

Applies the effect within an adjustable rectangular area with soft edges.

Creates a lighting effect on the image, the colors for High, Mid and Low Lights can be adjusted separately.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red H	Color picker / Level of red	0-255	128
Green H	Color picker / Level of green	0-255	128
Blue H	Color picker / Level of blue	0-255	128
Red M	Red M	0-255	128
Green M	Green M	0-255	128
Blue M	Blue M	0-255	128
Red L	Color picker / Level of red	0-255	128
Green L	Color picker / Level of green	0-255	128
Blue L	Color picker / Level of blue	0-255	128
Size	Size	0-255	16
Corner	Corner	0-255	0

Levels RGB Softborder

Applies the effect within an adjustable rectangular area with soft edges.

Allows setting the overall level gradation.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark Red	Dark Red	0-255	0
Light Red	Light Red	0-255	255
Dark Green	Dark Green	0-255	0
Light Green	Light Green	0-255	255
Dark Blue	Dark Blue	0-255	0
Light Blue	Light Blue	0-255	255
Size	Size	0-255	16
Corner	Corner	0-255	0

Levels Softborder

Applies the effect within an adjustable rectangular area with soft edges.

Allows setting the gradation levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Dark	Dark	0-255	0
Light	Light	0-255	255
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB Add Softborder

Applies the effect within an adjustable rectangular area with soft edges.

Adds RGB colors by the RGB color mix value, allows inverting the image.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Invert	Inverts the RGB levels or the effect	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB Clipper Softborder

Applies the effect within an adjustable rectangular area with soft edges.

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB CTB Softborder

Applies the effect within an adjustable rectangular area with soft edges.

RGB CTB (color temperature blue) emulates the CTB filter that converts tungsten light of 3200K to 'daylight' color.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

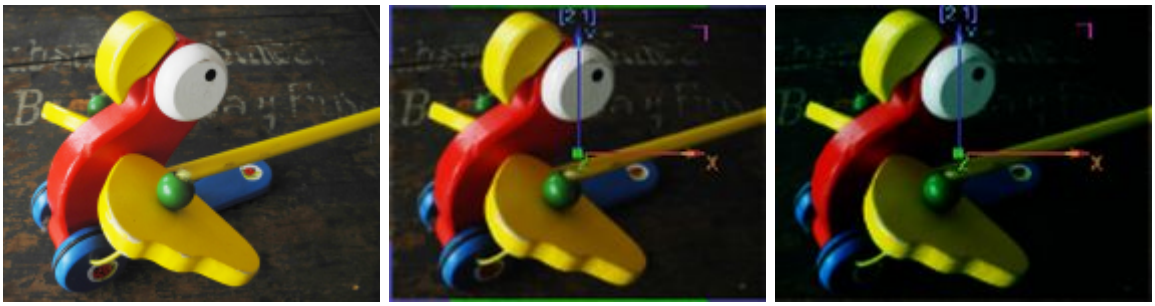
RGB CTO Softborder

Applies the effect within an adjustable rectangular area with soft edges.

RGB CTO (color temperature orange) emulates the CTO filter that converts 'daylight' color to tungsten light of 3200K.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB Gamma Softborder



Applies the effect within an adjustable rectangular area with soft edges.

Allows adjusting the gamma for all three channels, RGB. The default value of 255 for RGB equates to a gamma value of 1.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
R	Color picker / Level of red	0-255	255
G	Color picker / Level of green	0-255	255
B	Color picker / Level of blue	0-255	255
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB Min Max Softborder

Applies the effect within an adjustable rectangular area with soft edges.

Allows setting minimum and maximum levels for all three channels, RGB.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	128
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	128
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	128
Blue Max	Blue Max	0-255	255
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB Multiply Softborder

Applies the effect within an adjustable rectangular area with soft edges.

All RGB colors are multiplied by RGB color mix values. Multiply can be factorized and inverted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Factor	Factor	0-255	32
Invert	Inverts the RGB levels or the effect	0-255	0
Size	Size	0-255	16
Corner	Corner	0-255	0

RGB Squisher Softborder

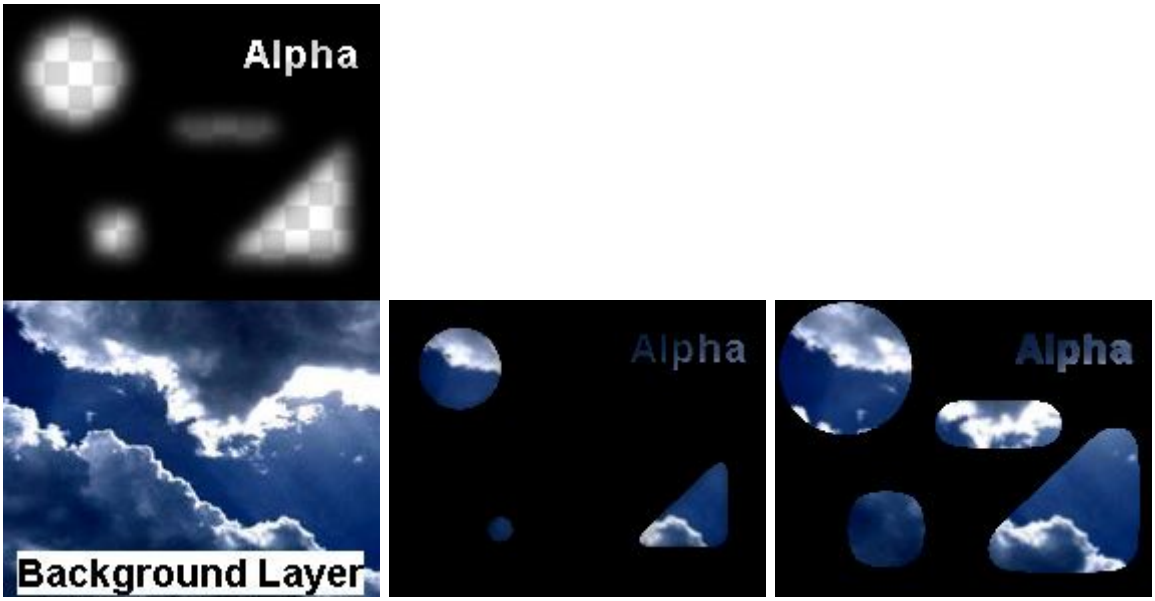
Applies the effect within an adjustable rectangular area with soft edges.

Squishes the RGB levels. For lower parameters this effect can be compared to the "Levels RGB" 507 effect.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Min	Red Min	0-255	0
Red Max	Red Max	0-255	255
Green Min	Green Min	0-255	0
Green Max	Green Max	0-255	255
Blue Min	Blue Min	0-255	0
Blue Max	Blue Max	0-255	255
Size	Size	0-255	16
Corner	Corner	0-255	0

6.5.2.4.22 Keying

Alpha Adjust



This effect influences the threshold between transparent (see-through) and opaque pixels. You can either enlarge or reduce the transparent area when it has a soft border with partly transparent pixels.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Low	Low	0-255	0
High	High	0-255	255

Black Key Mask

Allows keying out black parts of the image based on the luminance values, all other parts will be masked black. This result can be inverted: transparent areas get black.
In order to key out only black with the possibility not to get any transparency for other dark parts, use RGB key instead.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range of color tolerance	0-255	255
Invert	Inverts the Black Key Mask	0-255	0

Black Key

Allows keying out black parts of the image based on the luminance values. This result can be inverted: transparent areas get black.
In order to key out only black with the possibility not to get any transparency for other light parts, use RGB key instead.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range of color tolerance	0-255	255
Invert	Inverts the black key	0-255	0

Difference Key Mask

This effect looks for pixels that match in the textures from the layer media and effect media. Corresponding areas are keyed out, meaning that the background layer is visible. When there are differences in areas, they are not keyed out but overlap the background as a black mask. You can adjust a threshold to accept small color differences. The third image shows the result of the "Invert" parameter: differences are keyed out and matching areas overlay the background in their original color (and white if transparent).

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Threshold	Threshold	0-255	0
Media	Media File as Mask	0-255	255
Invert	Inverts the difference key	0-255	0

Difference Key

This effect looks for pixels that match in the textures from the layer media and effect media. Corresponding areas are keyed out, meaning that the background layer is visible. When there are differences in areas, they are not keyed out but overlap the background in the color from the layer media. You can adjust a threshold to accept small color differences. The third image shows the result of the "Invert" parameter: differences are keyed out and matching areas overlay the background in there original color (and white if transparent).

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Threshold	Threshold	0-255	0
Media	Media File as Mask	0-255	255
Invert	Inverts the difference key	0-255	0

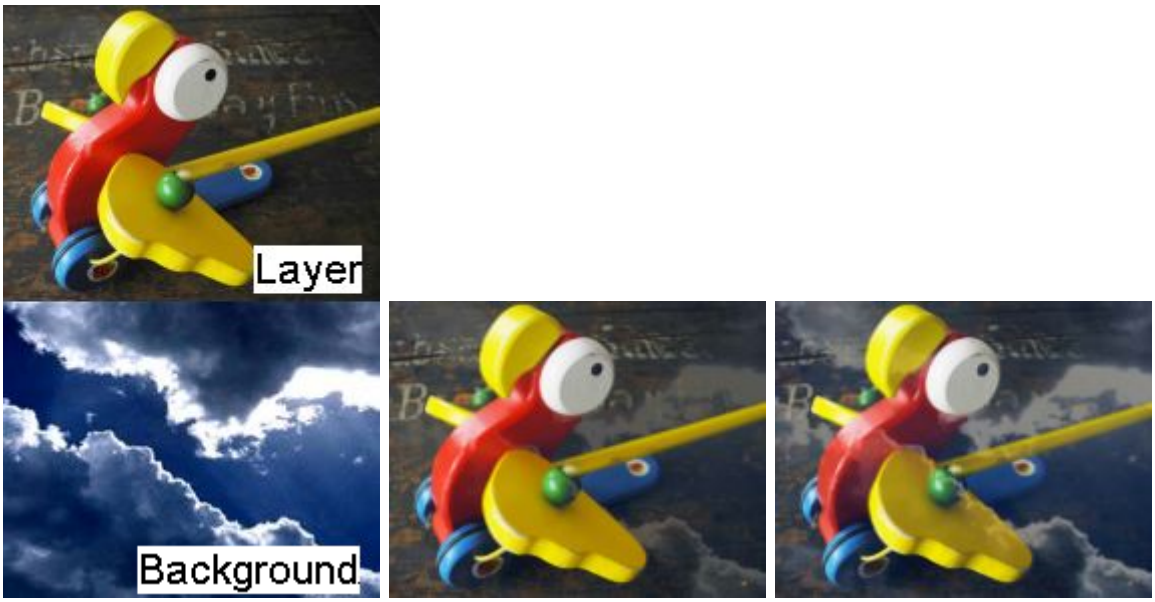
Difference Matte


This effect creates a mask from a captured frame and keys out all matching pixels from following frames in the video applied to the layer. As seen in the second image, all black pixels from the layer texture are keyed out in general. To capture a frame toggle the "Capture" parameter above 0 and back to 0. Now this frame is memorized and compared to following frames. All matching pixels are keyed out, different pixels overlap the background. The third image shows a train, which was not visible in the captured frame, hence it can be seen on top the background layer.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Capture	Capture	0-255	0
Threshold	Threshold	0-255	64

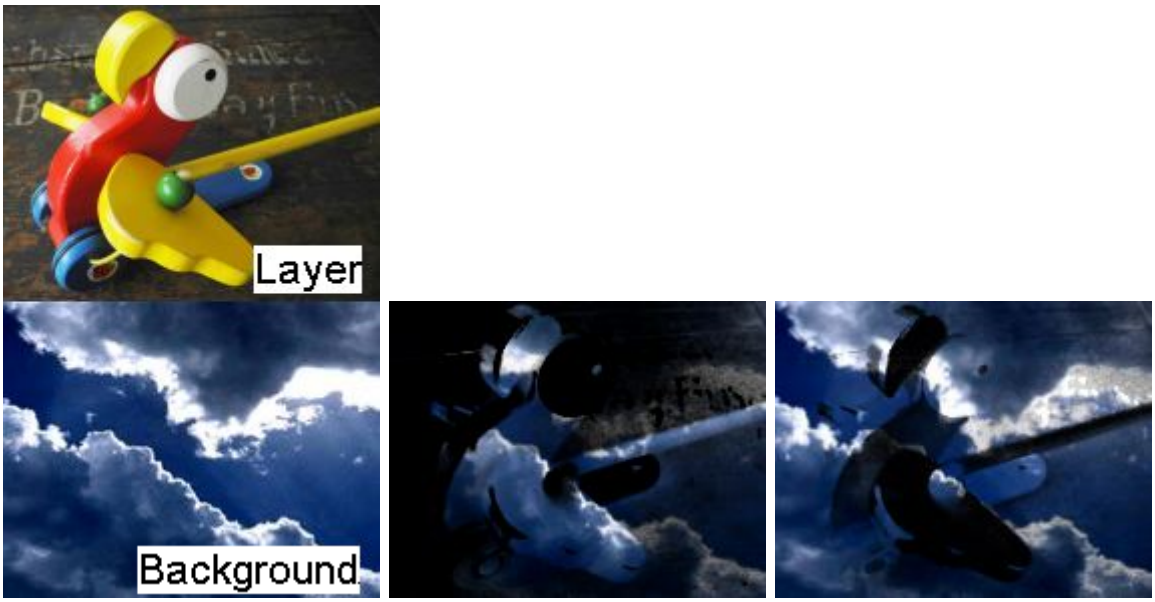
Luma Key



Allows adjusting a key based on luminance (brightness) values. A high range starts keying out, only dark colors (see second image) whilst a smaller level enlarges the value range of affected pixel brightness (see third image).

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range	0-255	255
Level	Level	0-255	255
Invert	Inverts the luma key	0-255	0

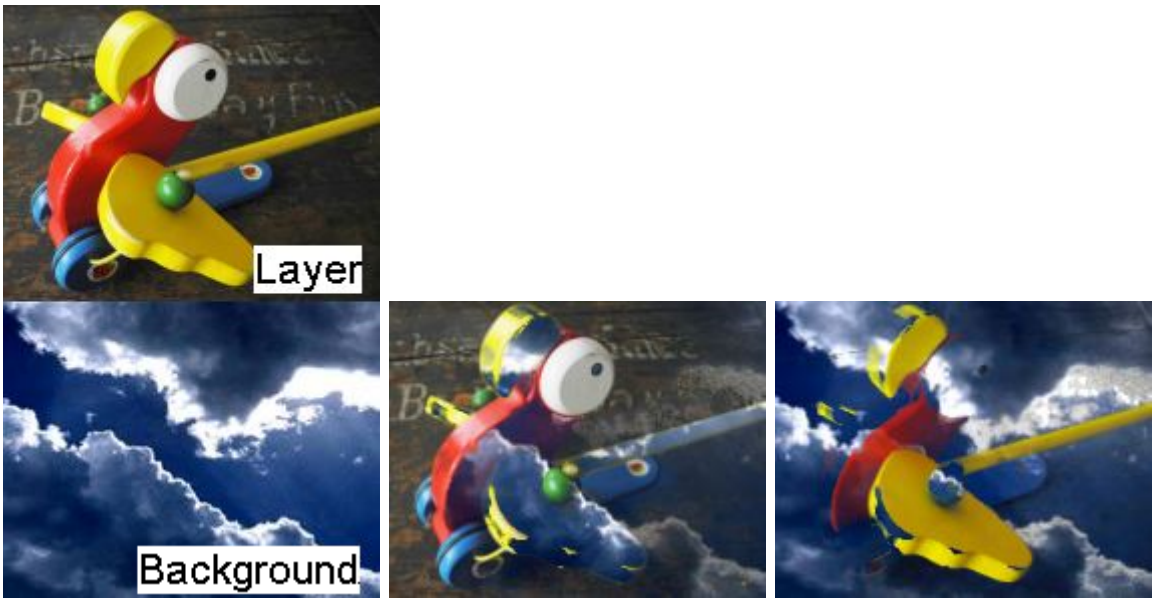
RGB Key Factor Mask



Allows keying out any RGB color included in the picked Color Range. In the second image a bright yellow color (RGB 255,255,0) and a small range was chosen which means that most dark colors, dark red, dark green and yellow tones are keyed out and the background can be seen there. All other pixels are rendered as a black mask. In the third image, the result was inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range	0-255	128
Red	Red level, Color picker	0-255	0
Green	Green level, Color picker	0-255	0
Blue	Blue level, Color picker	0-255	0
Invert	Inverts the RGB key factor mask	0-255	0

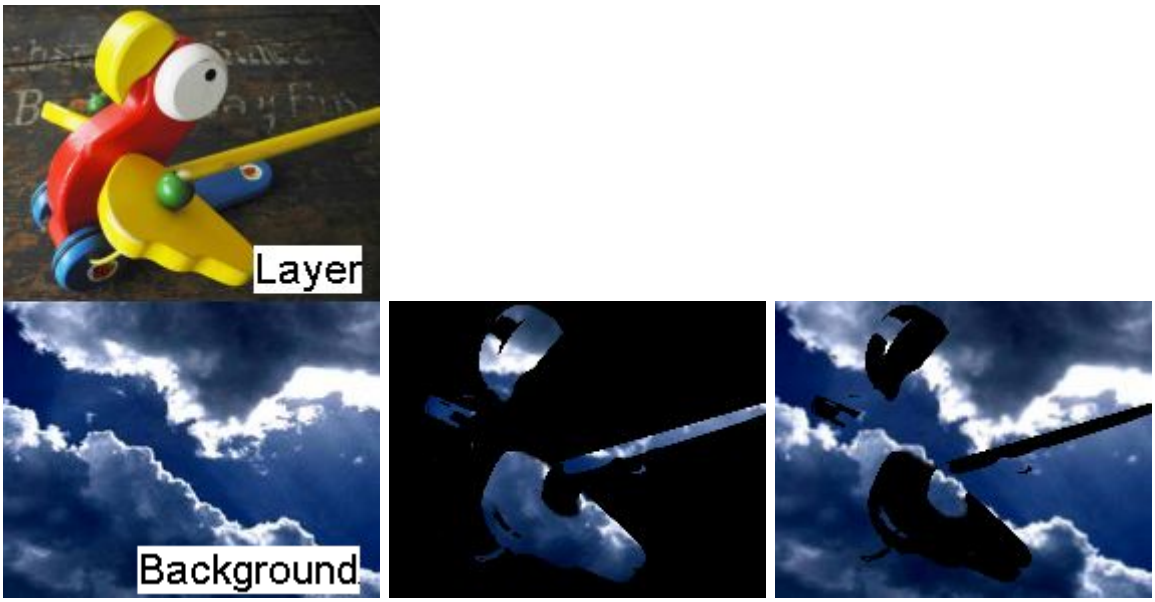
RGB Key Factor



Allows keying out any RGB color included in the picked Color Range. In the second image a bright yellow color (RGB 255,255,0) and a small range was chosen which means that most dark colors, dark red, dark green and yellow tones are keyed out and the background can be seen there. In the third image, the result was inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range	0-255	128
Red	Red level, Color picker	0-255	0
Green	Green level, Color picker	0-255	0
Blue	Blue level, Color picker	0-255	0
Invert	Inverts the RGB key factor mask	0-255	0

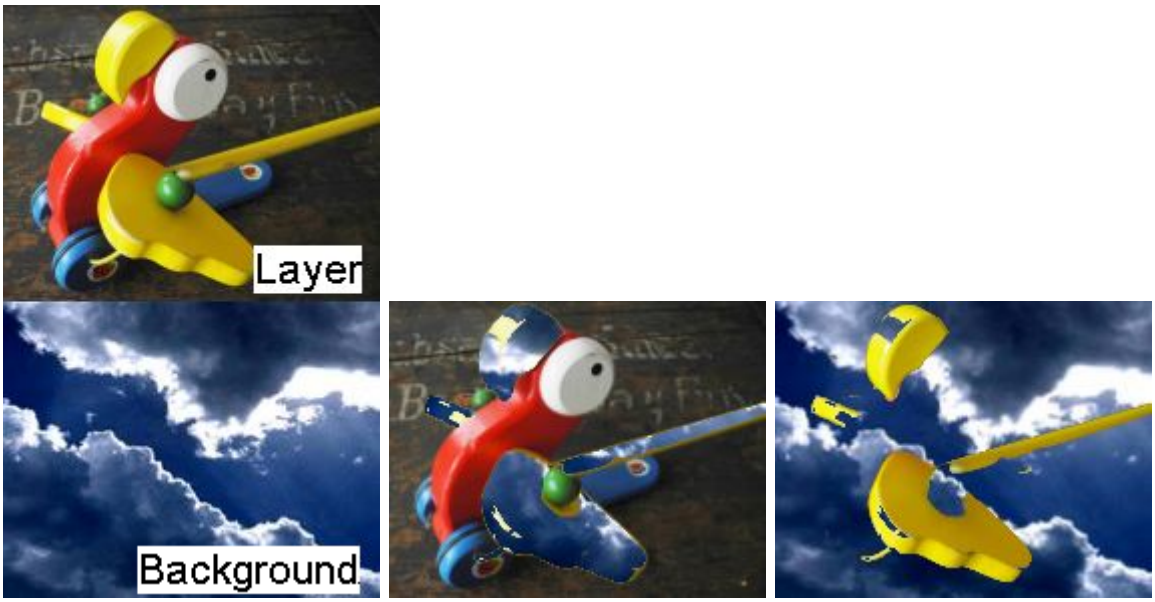
RGB Key Mask



Allows keying out any RGB color by adjusting the Color Range, Threshold, Radius. In the second image a yellow color was chosen which means that all yellow parts from the layer texture are keyed out and the background can be seen there whilst all other pixels are rendered as a black mask. In the third image, the result was inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range	0-255	128
Red	Red level, Color picker	0-255	0
Green	Green level, Color picker	0-255	0
Blue	Blue level, Color picker	0-255	0
Invert	Inverts the RGB key factor mask	0-255	0

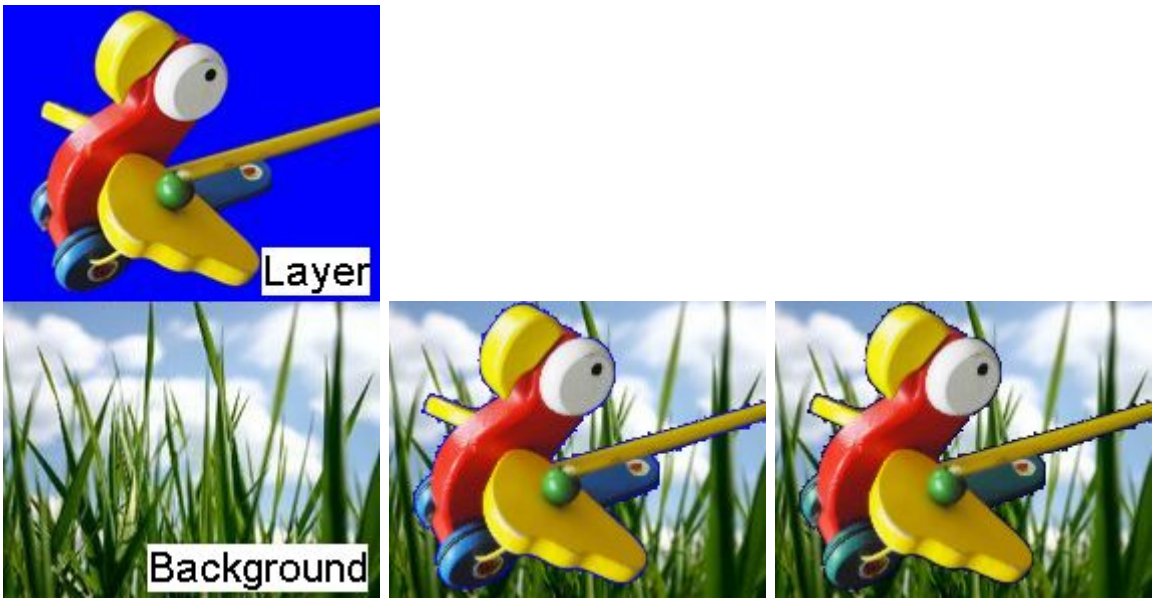
RGB Key



Allows keying out any RGB color by adjusting the Color Range, Threshold, Radius. In the second image a yellow color was chosen which means that all yellow parts from the layer texture are keyed out and the background can be seen there. In the third image, the result was inverted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range	0-255	128
Red	Red level, Color picker	0-255	0
Green	Green level, Color picker	0-255	0
Blue	Blue level, Color picker	0-255	0
Threshold	Threshold	0-255	64
Radius	Radius	0-255	32
Invert	Inverts the RGB key factor mask	0-255	0

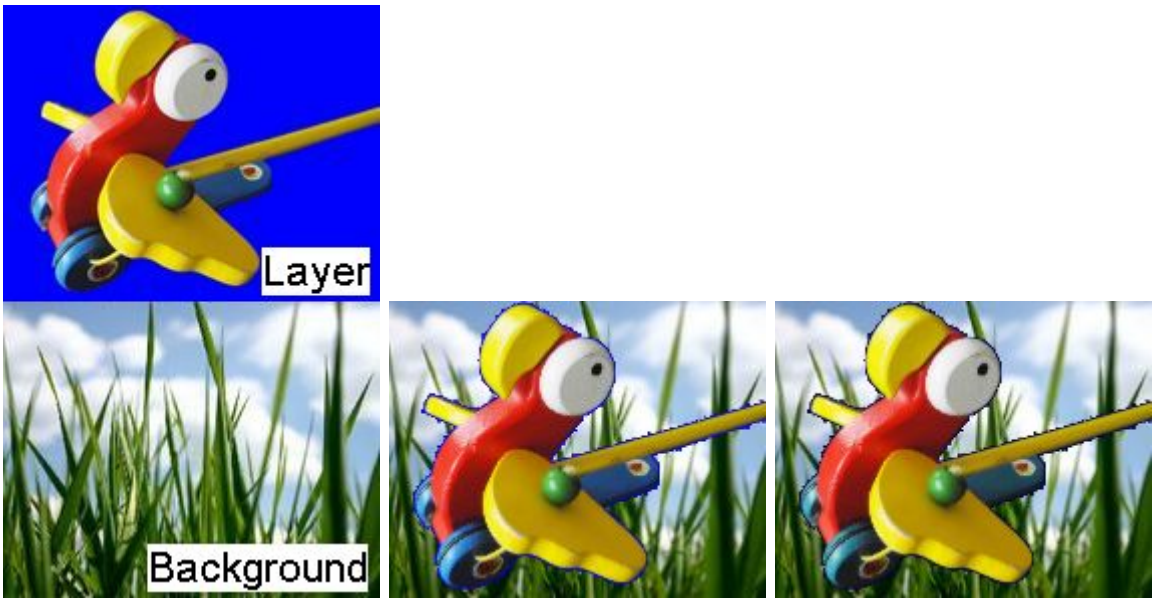
Spill Reduce Blue AVG



This effect is useful in blue screen applications. After keying out the blue color, e.g. with an "RGB key [520]" effect, sometimes there is a blue spill left (very light or dark blue pixels), which is depicted in the second image. This Spill Reduce effect reduces the blue value in all colors but affects colors with a red or green value less. In other words, pure blue colors are darkened fully (RGB 0,0,255 > 0,0,0) whilst yellow, cyan and purple change less (RGB r,g,255 > r,g,128) and red and green stay unchanged. The intensity of the reducing accords to the value of the Mix parameter (Mix 0>255). Note that depending on the colors of the cut out object, another blue spill effect might give better results, or more RGB keys set to light and dark blue.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

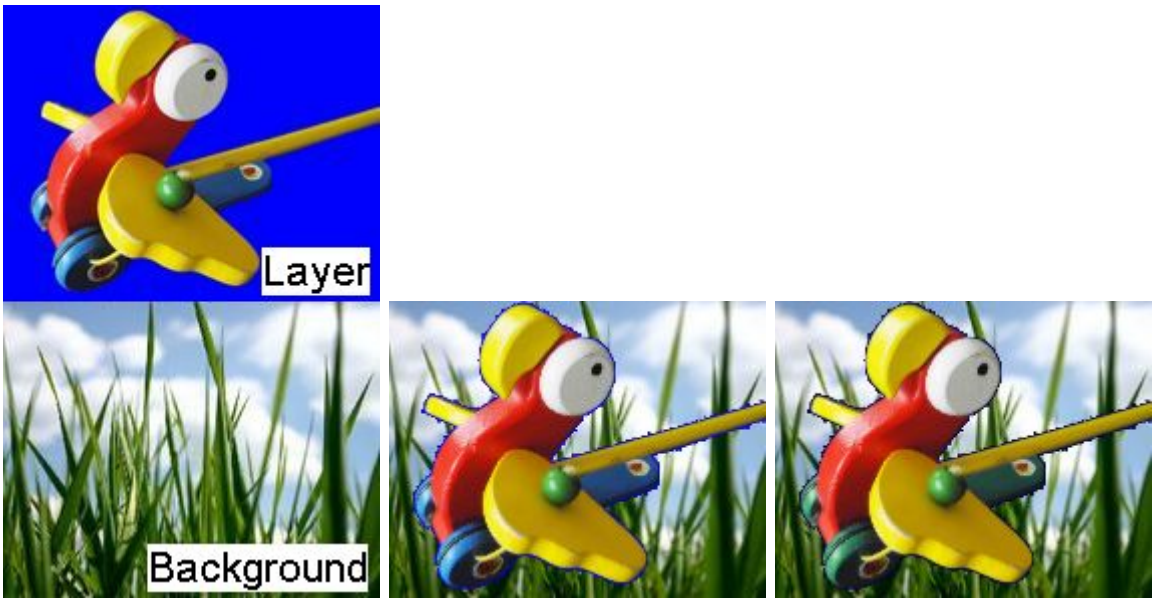
Spill Reduce Blue by Green



This effect is useful in blue screen applications. After keying out the blue color, e.g. with an "RGB key [520]" effect, sometimes there is a blue spill left (very light or dark blue pixels), which is depicted in the second image. This Spill Reduce effect reduces the blue value in all colors without a green value. In other words, blue colors are darkened (RGB 0,255,0 > 0,0,0) and purple colors turn red (RGB 255,0,255 > 255,0,0) whilst red, green, yellow and cyan stay unchanged. The intensity of the reducing accords to the value of the Mix parameter (Mix 0>255). Note that depending on the colors of the cut out object, another blue spill effect might give better results, or more RGB keys set to light and dark blue.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

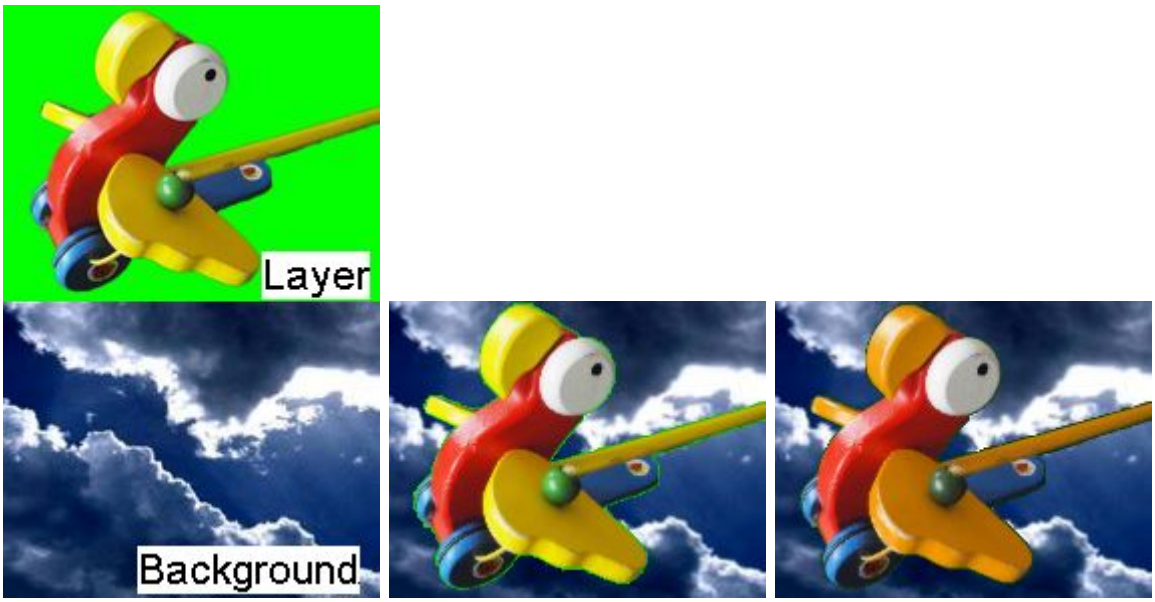
Spill Reduce Blue



This effect is useful in blue screen applications. After keying out the blue color, e.g. with an "RGB key [520]" effect, sometimes there is a blue spill left (very light or dark blue pixels), which is depicted in the second image. This Spill Reduce effect reduces the blue value in all colors without a red value. In other words, blue colors are darkened (RGB 0,0,255 > 0,0,0) and cyan colors turn green (RGB 0,255,255 > 0,255,0) whilst red, green, yellow and purple stay unchanged. The intensity of the reducing accords to the value of the Mix parameter (Mix 0>255). Note that depending on the colors of the cut out object, another blue spill effect might give better results, or alternatively more RGB keys set to light and dark blue.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

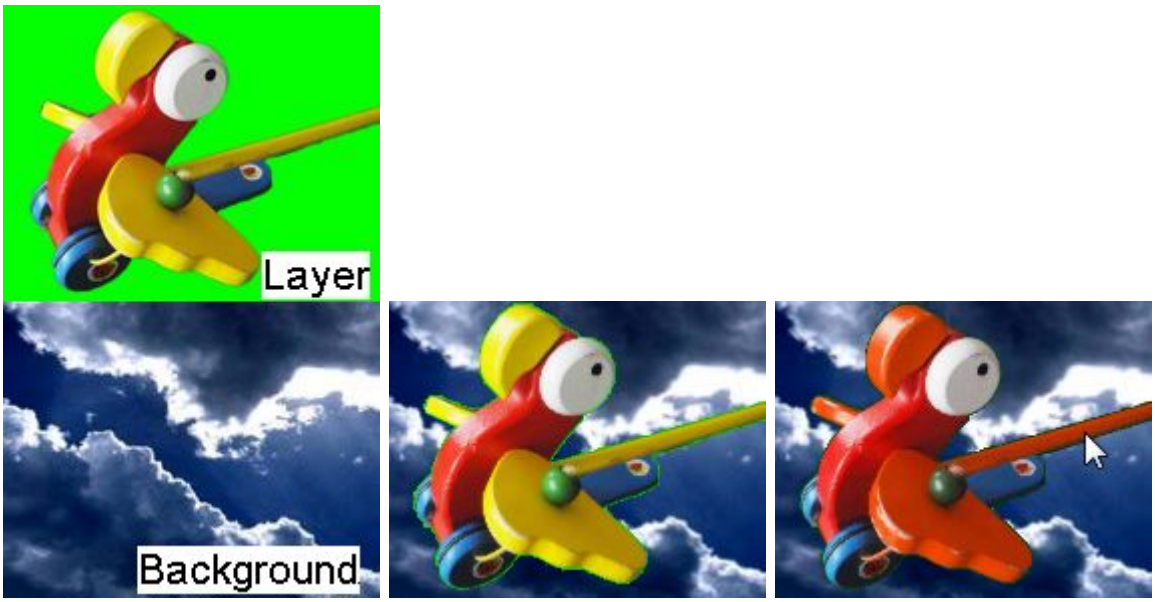
Spill Reduce Green AVG



This effect is useful in green screen applications. After keying out the green color, e.g. with an "RGB key [520]" effect, sometimes there is a green spill left (very light or dark green pixels), which is depicted in the second image. This Spill Reduce effect reduces the green value in all colors but affects colors with a blue or red value less. In other words, pure green colors are darkened fully (RGB 0,255,0 > 0,0,0) whilst yellow, cyan and purple change less (RGB r,255,b > r,128,b) and red and blue stay unchanged. The intensity of the reducing accords to the value of the Mix parameter (Mix 0>255). Note that depending on the colors of the cut out object, another green spill effect might give better results, or more RGB keys set to light and dark green.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

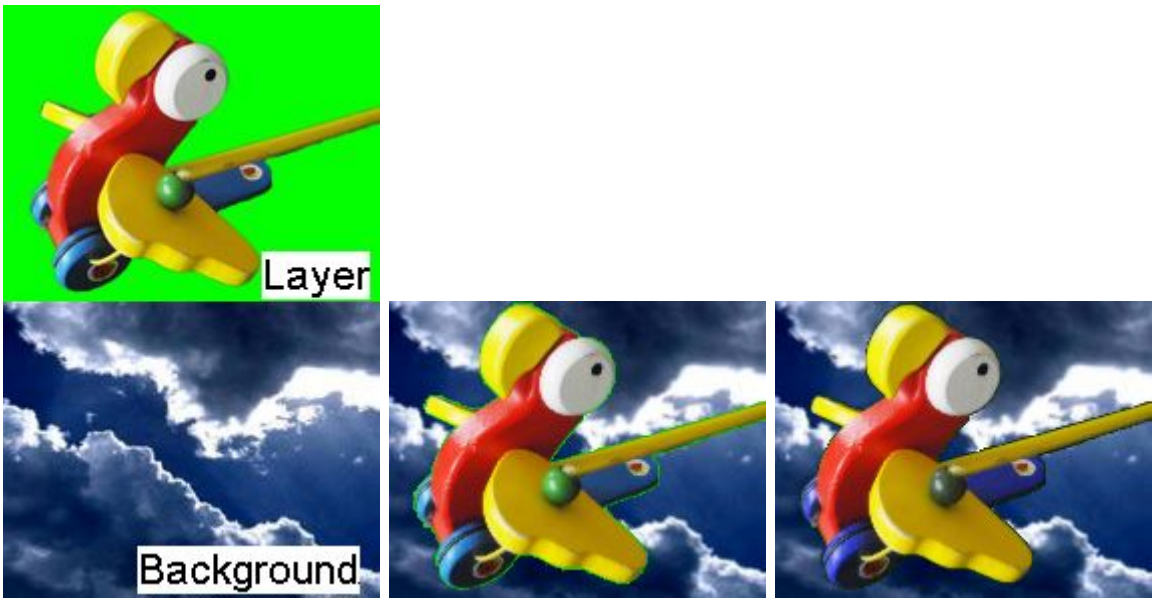
Spill Reduce Green by Blue



This effect is useful in green screen applications. After keying out the green color, e.g. with an "RGB key [520]" effect, sometimes there is a green spill left (very light or dark green pixels), which is depicted in the second image. This Spill Reduce effect reduces the green value in all colors without a blue value. In other words, green colors are darkened (RGB 0,255,0 > 0,0,0) and yellow colors turn red (RGB 255,255,0 > 255,0,0) whilst red, blue, cyan and purple stay unchanged. The intensity of the reducing accords to the value of the Mix parameter (Mix 0>255). Note that depending on the colors of the cut out object, another green spill effect might give better results, or more RGB keys set to light and dark green.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Spill Reduce Green



This effect is useful in green screen applications. After keying out the green color, e.g. with an "RGB key [520]" effect, sometimes there is a green spill left (very light or dark green pixels), which is depicted in the second image. This Spill Reduce effect reduces the green value in all colors without a red value. In other words, green colors are darkened (RGB 0,255,0 > 0,0,0) and cyan colors turn blue (RGB 0,255,255 > 0,0,255) whilst red, blue, yellow and purple stay unchanged. The intensity of the reducing accords to the value of the Mix parameter (Mix 0>255). Note that depending on the colors of the cut out object, another green spill effect might give better results, or alternatively more RGB keys set to light and dark green.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

White Key Mask

Allows keying out white parts of the image based on the luminance values, all other parts will be masked black. This result can be inverted: transparent areas get black.
In order to key out only white with the possibility not to get any transparency for other light parts, use RGB key instead.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range of color tolerance	0-255	255
Invert	Inverts the white key	0-255	0

White Key

Allows keying out white parts of the image based on the luminance values. This result can be inverted: transparent areas get white.
In order to key out only white with the possibility not to get any transparency for other light parts, use RGB key instead.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Range	Range of color tolerance	0-255	255
Invert	Inverts the white key	0-255	0

6.5.2.4.23 Lighting

++ General information for Light Effects ++

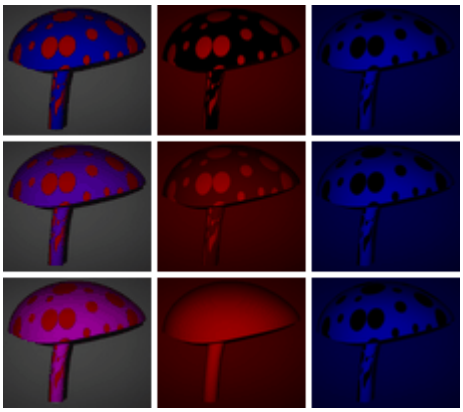
The light feature in Pandoras Box can be best understood and used when the following fundamental course of events is kept in mind:

1. Firstly, light is emitted by a light source. Here the light rays are determined in their starting position, direction, intensity and color.
A light source can be either the inherent light from the Lighting effect [536], or in all other cases, an external light from a Light Layer [666].

2. The light then shines on an object's (i.e. layer's) surface. The object's properties, also called its material, determine how the light interacts with it and in which direction and what amount and color it bounces off.
Note that in Pandoras Box light cannot pass through objects or bend around them, it can only be reflected.

3. Lastly, the reflected light enters the viewpoint. In the real world it would be the human eye whilst here, it is the virtual camera. No adjustments can be made at this point of the sequence. It is now, when the object becomes truly visible, its image is rendered.
Same as incident light, reflected light has no attenuation, which means that their intensity is not a function of the distance but is always equally intense.

Light falling on an object holds a source color; in physical terms it has a specific wavelength, in rendering terms it has an rgb value. The rgb value depends additionally on the light intensity. The default color is white - consisting of red, green and blue components in equal amounts.
When interacting with the object's material the source color changes according to the target color which is determined by the layer texture's color. The color of the reflected light is the common intersection of the source and target color.



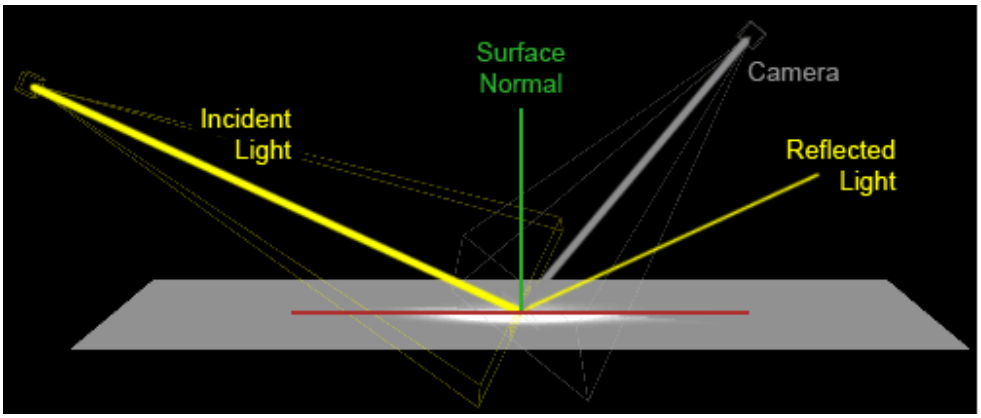
In all images seen left, the background is pure white, but the opacity is lowered to 33% thus the reflected light's intensity is reduced.

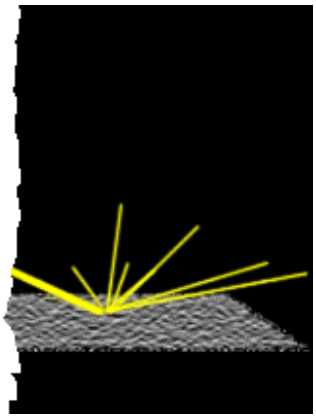
The mushrooms texture is always covered with red spots;
the main color is
- blue with rgb = 0,0,255 in image 1-3,
- purple with rgb = 125,0,255 in image 4-6 and
- magenta with rgb = 255,0,255 in image 7-9.

The light changes from left to right:
- white with rgb = 255,255,255
- red with rgb = 255,0,0
- blue with rgb = 0,0,255.

But materials in the real world do not only reflect in one color, rather with many shades and nuances. In an effort to mimic this, more than one light is used to describe an virtual object's reflection. In Pandoras Box (PB), three varieties of light have an effect: ambient, diffuse and specular light. Unlike the first two, reflected specular color does not necessarily need to be modulated by the object's color. In particular, materials like plastic often reflect only the source's color whilst metallic materials might reflect with a shifted hue.

Another difference between the three lights is in which direction they reflect from a surface. The left image below shows the first law of reflection theory, considering a perfectly smooth surface. The perpendicular surface normal divides the angle between incident light rays and their reflection equally. In other words, the angle of incidence equals the angle of reflection.
The surface normal is an imaginary line, perpendicular to the surface. If a layer has no mesh or a flat one, the entire object has parallel surface normals.
Apart from that, the polygons forming the mesh determine the direction of each line and consequently the direction of reflected light.



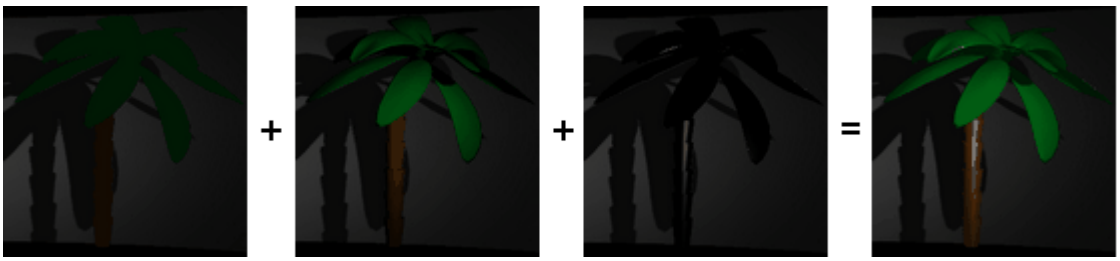


To imitate a rough surface one could apply a mesh with disarrayed polygons. However it is far easier to achieve it with a material's attribute. This is what diffuse reflection does. Diffuse light is reflected in all direction (see right upper principle depiction). The larger the angle between light source and surface normal gets, the less intense is the reflection. As a result diffuse light points the dimension of the object out and enhances surface details. Typically, an object with an high proportion of diffusion appears to have a rough surface such as wood or rock.
For the left images a light source was first positioned to the right, then to the left.

In opposite, the specular level influences how many light rays bounce off coherently / parallel. This is what creates highlights on an object's surface and makes it appear to have a smooth surface such as plastic or metal. The smoother a surface is, the smaller and sharper the highlights are. This last adjustment is done in PB with the smoothness parameter. Just as diffuse reflection, specular reflection depends on the relative position and orientation of light source and target. But as the rays are not reflected in all directions any more, the position of the camera plays an important role too. It influences how intense the highlight shines.

The third light variation that is added to the illumination term is ambient light. Other than the above mentioned, ambient light is absolutely directionless. It illuminates all surfaces equally regardless their orientation. Strictly speaking, it has no true counterpart in real-word but still it gives very good results when it comes to simulate global light e.g. sun light and all the light reflected from the surrounding environment.

The image sequence below illustrates how the ambient component with a level of 40, the diffuse component with a level of 200 and the specular component with a level of 180 (with its softness set to 60) add up and result in a realistic shaded plastic palm tree.



Please note:
The effects described below are NOT designed to be applied more than once to a layer. If you want a layer to be lit by two light source please use the according effect.

1 Light Mat-Spec-Color

Allows receiving light emitted by a Light Layer [666] by right-clicking on the Media parameter and choosing "Share Layer Texture". Adjustments can be made influencing the reflection of the light in order to imitate a specific material e.g. wood or plastic. In addition highlights caused by specular light can be tinted. In the images, the light itself was not changed! The left image has no light turned on.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Mode	None (0), Cast & Receive (1) Cast (2)	0-2	0
Light 1	Light source (via "Share layer texture")	-	-
Ambient	Global light intensity (which has no direction of propagation and to which the spot light is added)	0-255	25
-	- no global light, only the area hit by the spot light is to be seen	0	-
-	- everything is lit up, there are no dark areas anywhere	255	-
Diffuse	Intensity of diffuse light (surrounding the hot spot)	0-255	255
-	- no diffuse light surrounds the hot spot (which is set with the Softness fader)	0	-
-	- intense diffuse light; largest area possible is illuminated by diffuse light	255	-
Specular	Intensity of hot spot / highlights	0-255	50
-	- no hot spot, only diffuse light is shed (the softness fader shows no effect)	0	-
-	- intense hot spot	255	-
Softness	Softness of light cone / Size of hot spot	0-255	10
-	- large hot spot; largest area possible is illuminated by a hard spot light	0	-
-	- small hot spot; only the area within immediate range is illuminated by soft spot light	255	-
Spec Red	Red level of reflection of Specular Light	0-255	255
Spec Green	Green level of reflection of Specular Light	0-255	255
Spec Blue	Blue level of reflection of Specular Light	0-255	255

1 Light Shadow Catcher

Allows using a layer as a shadow layer. This is of interest for Virtual Reality applications as this only renders shadows from (multiple, virtual) 3D objects on to one layer, which then can be used as an overlay layer for a Studio set.

The left image depicts the scene. A layer with a color bar image lies between a background cloud image and a 3D globe which is illuminated by a light using the "1 Light" [532] effect.
For the second image, the "1 Light Shadow Catcher" FX on the color bars is switched on and assigned with the same light as the globe is. The "Blend" parameter is turned to 255.
For the right image the parameter "Blend" is turned to 0 and a purple color is chosen to overwrite the color bar texture with a solid color.

Note, for this effect, the light texture should be a solid white or transparent one!

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Mode	Mode	-	-
Light 1	Light source (via "Share layer texture")	-	-
Blend	Blend	0-255	0
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0

1 Light

Allows receiving light emitted by a Light Layer [666] by right-clicking on the Media parameter and choosing "Share Layer Texture". Adjustments can be made that influence the reflection of the light in order to imitate a specific material e.g wood or plastic. In the images, the light itself was not changed! The left image has no light turned on.

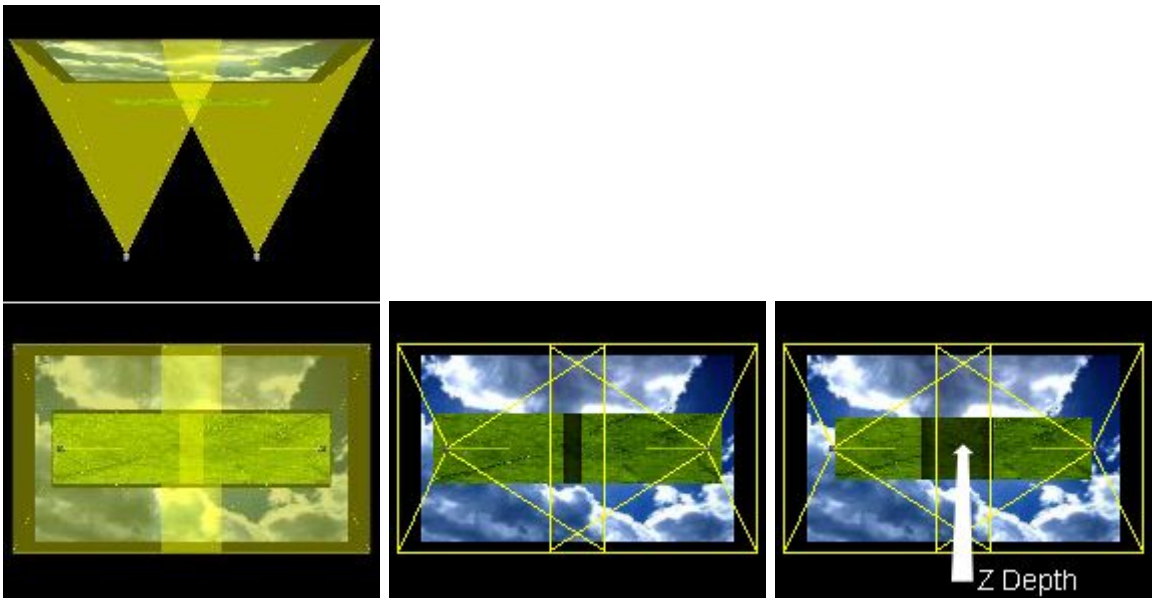
Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Mode	None (0), Cast & Receive (1) Cast (2)	0-2	0
Light 1	Light source (via "Share layer texture")	-	-
Ambient	Global light intensity (which has no direction of propagation and to which the spot light is added)	0-255	25
-	- no global light, only the area hit by the spot light is to be seen	0	-
-	- everything is lit up, there are no dark areas anywhere	255	-
Diffuse	Intensity of diffuse light (surrounding the hot spot)	0-255	255
-	- no diffuse light surrounds the hot spot (which is set with the Softness fader)	0	-
-	- intense diffuse light; largest area possible is illuminated by diffuse light	255	-
Specular	Intensity of hot spot / highlights	0-255	50
-	- no hot spot, only diffuse light is shed (the softness fader shows no effect)	0	-
-	- intense hot spot	255	-
Softness	Softness of light cone / Size of hot spot	0-255	10
-	- large hot spot; largest area possible is illuminated by a hard spot light	0	-
-	- small hot spot; only the area within immediate range is illuminated by soft spot light	255	-

2 Lights Mat-Spec-Color

see FX 1 Light Mat-Spec-Color [530]. Here, the effect layer may be influenced by two different light sources.

2 Lights Texture Projection Adaptive



Allows adaptive softedge applications. When two projectors project onto a screen that moves towards and away from them, the size of the overlapping area changes. It increases with the distance between projectors and screen. Hence, the softedge parameters must adopt to the new screen position. An adaptive softedge can be programmed using this effect (on the screen layer). The FX subtracts the light amount from 2 Light Layers [666], it darkens the overlap area which compensates for the higher light amount from the real projectors. For this setup a 3D setup is mandatory, i.e. the PB light sources (as well as PB cameras) have the same parameters (position, FOV etc.) as the real projectors and the PB screen layer corresponds with the real screen.

The left image depicts the scene. A green layer moves in front of a cloud image. The light overlap on the background image is larger than on the green image as its nearer to the lights.
For the right image the green layer was moved further away from the lights / projectors, enlarging the overlap, the area that PB now renders automatically darker.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amt	Amt	0-255	255
Light 1	Light source (via "Share layer texture")	-	-
Light 2	Light source (via "Share layer texture")	-	-

2 Lights

see FX 1 Light [532]. Here, the effect layer may be influenced by two different light sources.

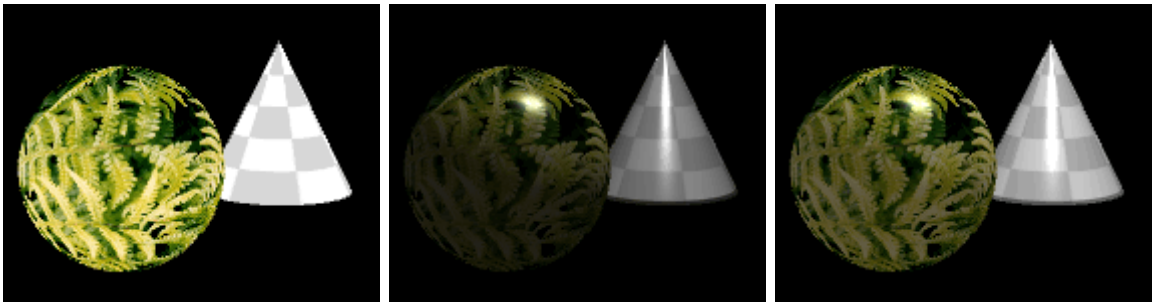
3 Lights Mat-Spec-Color

see FX 1 Light Mat-Spec-Color [530]. Here, the effect layer may be influenced by three different light sources.

3 Lights

see FX 1 Light [532]. Here, the effect layer may be influenced by three different light sources.

Global Light

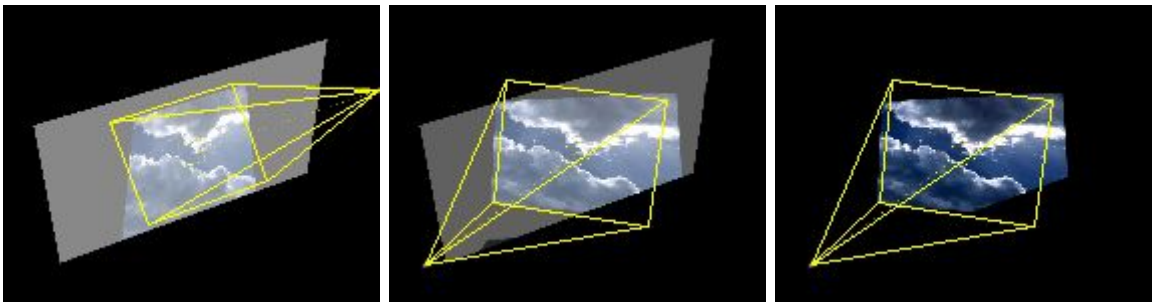


This effect is very useful when working with many 3D objects. By applying it to all layers, they will instantly share and respond to the same light source. In other words, when using the "1 Light" [532] effect an additional Light Layer needs to be created and shared with each Layer that should receive its light. The "Global Light" effect creates a global light source automatically and each Layer with this effect receives it automatically. Whilst you cannot edit the global light, you can edit how the Layer responds to it. In addition you can influence the visibility of the wireframe edges [146] of the object similar to the effect "Wireframe Edges [621]".

The left image depicts the scene with two 3D objects without the Global Light effect. The second image shows the scene with the Mix parameter at 255 but low "Ambient" parameter. In the right image, the ambient light amount is higher.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Softness	Softness of light cone / Size of hot spot	0-255	100
-	- large hot spot; largest area possible is illuminated by a hard spot light	0	-
-	- small hot spot; only the area within immediate range is illuminated by soft spot light	255	-
Ambient	Global light intensity (which has no direction of propagation and to which the spot light is added)	0-255	50
-	- no global light, only the area hit by the spot light is to be seen	0	-
-	- everything is lit up, there are no dark areas anywhere	255	-
Diffuse	Intensity of diffuse light (surrounding the hot spot)	0-255	60
-	- no diffuse light surrounds the hot spot (which is set with the Softness fader)	0	-
-	- intense diffuse light; largest area possible is illuminated by diffuse light	255	-
Specular	Intensity of hot spot / highlights	0-255	110
-	- no hot spot, only diffuse light is shed (the softness fader shows no effect)	0	-
-	- intense hot spot	255	-
Edges	Visibility of wireframe edges	0-255	128

Light Texture Projection



This is a "1 Light" [532] effect that is reduced to the parameters "Mix" and "Light 1". Right-click on the Media field and choose "Share Layer Texture" to receiving its light. This effect is ideal if you simply like to render a light texture on another layer to simulate a projected image.

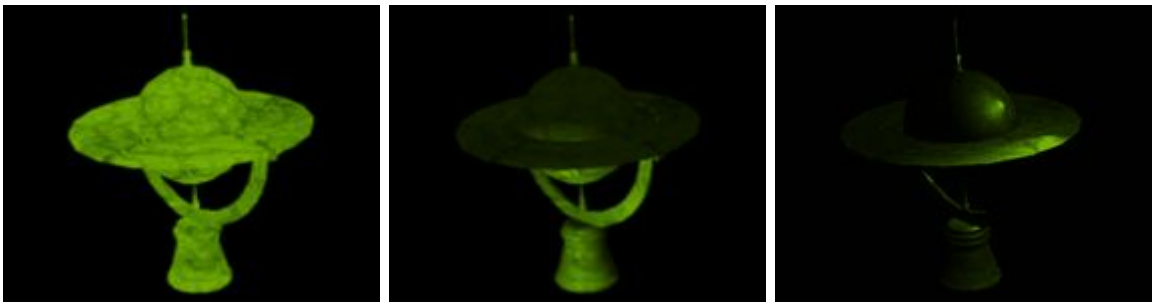
The left image depicts the scene. The Light Layer [666] projects a cloud image onto a white layer with the FX's Mix parameter at 125. For the second image the Light Layer changed the position resulting in a different angle for the projection. The right image depicts the same position but with a Mix parameter at 255, making the white texture invisible when no light is received.

Note that the effect "Texture Projection" [537] might also be of interest.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Light 1	Light source (via "Share layer texture")	-	-

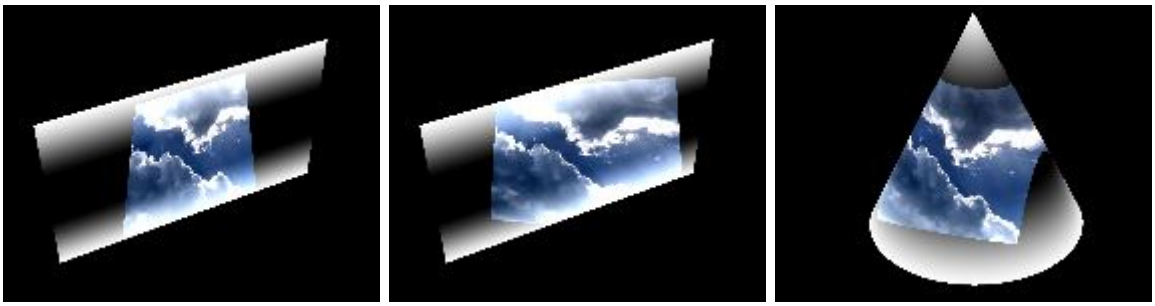
Lighting



Allows setting an own light source that can influence only the effect layer itself. The layer's reflection behavior can be adapted to match a specific material. The inherent light source can be adjusted in its position. Other than the Light Layer this light source has no direction, it is rather a point light. In the images, the light's position has changed as well as the material properties. The left image has no light turned on.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Softness	Softness of light cone / Size of hot spot	0-255	16
-	- large hot spot; largest area possible is illuminated by a hard spot light	0	-
-	- small hot spot; only the area within immediate range is illuminated by soft spot light	255	-
Ambient	Global light intensity (which has no direction of propagation and to which the spot light is added)	0-255	128
-	- no global light, only the area hit by the spot light is to be seen	0	-
-	- everything is lit up, there are no dark areas anywhere	255	-
Diffuse	Intensity of diffuse light (surrounding the hot spot)	0-255	16
-	- no diffuse light surrounds the hot spot (which is set with the Softness fader)	0	-
-	- intense diffuse light; largest area possible is illuminated by diffuse light	255	-
Specular	Intensity of hot spot / highlights	0-255	255
-	- no hot spot, only diffuse light is shed (the softness fader shows no effect)	0	-
-	- intense hot spot	255	-
Light X Pos	X Position of light source; equals a layer X position of + / - 50.000	0-65535	32768
Light Y Pos	Y Position of light source; equals a layer Y position of + / - 50.000	0-65535	32768
Light Z Pos	Z Position of light source; equals a layer Z position of + / - 50.000	0-65535	32768

Texture Projection



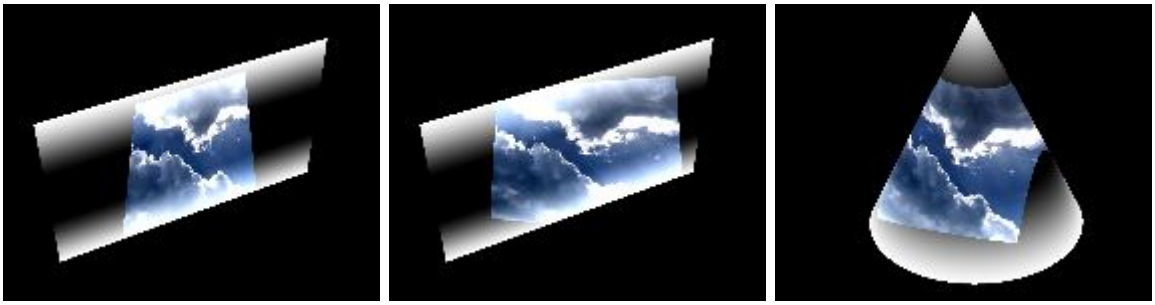
This is the same effect as "Texture Projection Add" [538] but without the parameters XY Offset.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Light source (via "Share layer texture")	-	-
Light Source X Pos	Light Source X Pos	-999.999 - +999.999	0
Light Source Y Pos	Light Source Y Pos	-999.999 - +999.999	-1200
Light Source Z Pos	Light Source Z Pos	-999.999 - +999.999	-3000
Light Target X Pos	Light Target X Pos	-999.999 - +999.999	0
Light Target Y Pos	Light Target Y Pos	-999.999 - +999.999	0
Light Target Z Pos	Light Target Z Pos	-999.999 - +999.999	0
Light Angle	Light Angle	0-180	20
Light Aspect	Light Aspect	0-20	1
Light Z Roll	Light Z Roll	-9999.99 - +9999.99	0

6.5.2.4.24 Lighting - Texture Projection

Texture Projection Add



This allows to simulate a projected image on a layer. There are similarities to the effect "Light Texture Projection" [535], but as it combines the Light Layer parameters into the effect itself, no additional Light Layer [666] is needed.

The left image depicts the scene. The Layer texture itself is a white-black-white gradient, the projected texture (FX media) is a cloud image. For the second image, the position for the effect's light source has changed, resulting in a different angle for the projection. In the right image, a 3D cone was used as a Layer object.

Note that (in difference to "Texture Projection Mix" [543]) the RGB values from the FX texture are added to the RGB values from the layer texture. This explains why the top left and bottom right corner of the cloud image cannot be seen on top the gradient texture as these areas are already full white.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Light source (via "Share layer texture")	-	-
Light Source X Pos	Light Source X Pos	-999.999 - +999.999	0
Light Source Y Pos	Light Source Y Pos	-999.999 - +999.999	10
Light Source Z Pos	Light Source Z Pos	-999.999 - +999.999	-25
Light Target X Pos	Light Target X Pos	-999.999 - +999.999	0
Light Target Y Pos	Light Target Y Pos	-999.999 - +999.999	0
Light Target Z Pos	Light Target Z Pos	-999.999 - +999.999	0
Light Angle	Light Angle	0-180	20
Light Aspect	Light Aspect	0-20	1
Light Z Roll	Light Z Roll	-9999.99 - +9999.99	0
X Offset	X Offset	-999.999 - +999.999	0
Y Offset	Y Offset	-999.999 - +999.999	0

Texture Projection Alpha Mix AB

This allows to simulate a projected image on a layer, and mixes it with the transparency from the FX media. The FX can blend between two textures. There are similarities to the effect "Light Texture Projection" [535], but as it combines the Light Layer parameters into the effect itself, no additional Light Layer [666] is needed.

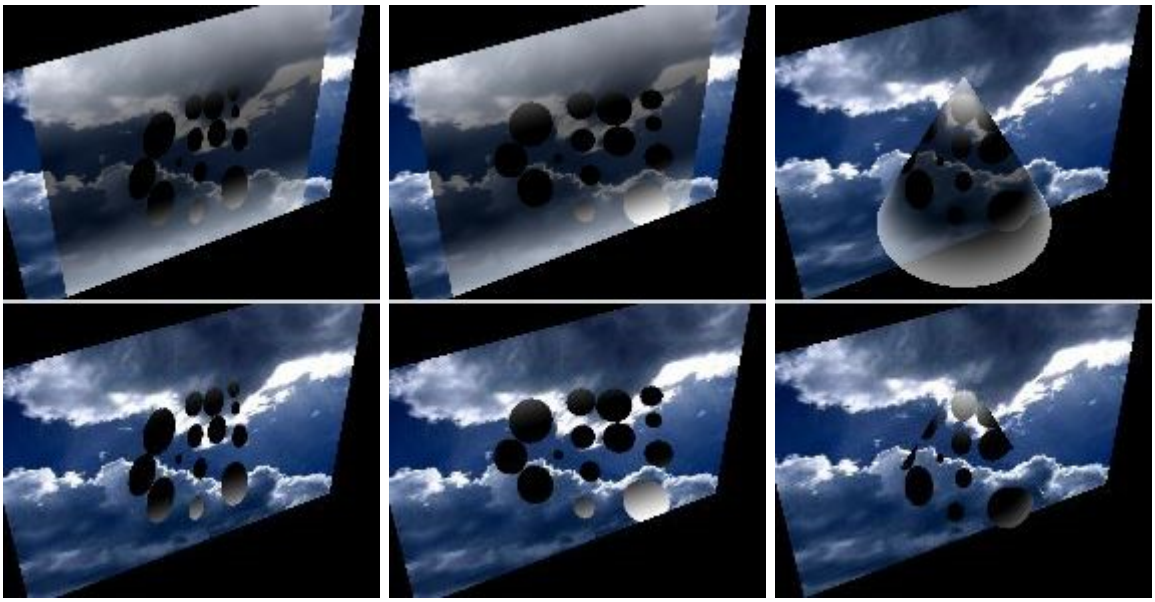
The left image depicts the scene. In front of a background cloud image, you can see the layer texture, a white-black-white gradient. The projected texture (FX media) is transparent with colorful circles. The transparency channel is projected onto the gradient layer and creates transparent areas there. The Mix parameter is at 100. For the second image, the Mix parameter was risen to 255.
In the right image, the second FX media (stars on a transparent background) is taking place as the "Mix AB" parameter is toggled.

Note that (in difference to "Texture Projection Mix" [543] and ...Add [538]) the RGB values from the FX textures are discarded. It is only the alpha information that overlays the RGB values from the layer texture.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	First Light source (via "Share-layer texture")		-
Mix AB	Mix AB	0-255	0
Media2	Second Light source (via "Share layer texture")	-	-
Light Source X Pos	Light Source X Pos	-999.999 - +999.999	0
Light Source Y Pos	Light Source Y Pos	-999.999 - +999.999	10
Light Source Z Pos	Light Source Z Pos	-999.999 - +999.999	-25
Light Target X Pos	Light Target X Pos	-999.999 - +999.999	0
Light Target Y Pos	Light Target Y Pos	-999.999 - +999.999	0
Light Target Z Pos	Light Target Z Pos	-999.999 - +999.999	0
Light Angle	Light Angle	0-180	20
Light Aspect	Light Aspect	0-20	1
Light Z Roll	Light Z Roll	-9999.99 - +9999.99	0
X Offset	X Offset	-999.999 - +999.999	0
Y Offset	Y Offset	-999.999 - +999.999	0

Texture Projection Alpha Mix



This allows to simulate a projected image on a layer, and mixes it with the transparency from the FX media. There are similarities to the effect "Light Texture Projection" [535], but as it combines the Light Layer parameters into the effect itself, no additional Light Layer [666] is needed.

The left image depicts the scene. In front of a background cloud image, you can see the layer texture, a white-black-white gradient. The projected texture (FX media) is transparent with colorful circles. The transparency channel is projected onto the gradient layer and creates transparent areas there.
For the second image, the position for the effect's light source has changed, resulting in a different angle for the projection. In the right image, a 3D cone was used as a Layer object.

Note that (in difference to "Texture Projection Mix" [543] and ...Add [538]) the RGB values from the FX texture are discarded. It is only the alpha information that overlays the RGB values from the layer texture.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Light source (via "Share layer texture")	-	-
Light Source X Pos	Light Source X Pos	-999.999 - +999.999	0
Light Source Y Pos	Light Source Y Pos	-999.999 - +999.999	10
Light Source Z Pos	Light Source Z Pos	-999.999 - +999.999	-25
Light Target X Pos	Light Target X Pos	-999.999 - +999.999	0
Light Target Y Pos	Light Target Y Pos	-999.999 - +999.999	0
Light Target Z Pos	Light Target Z Pos	-999.999 - +999.999	0
Light Angle	Light Angle	0-180	20
Light Aspect	Light Aspect	0-20	1
Light Z Roll	Light Z Roll	-9999.99 - +9999.99	0
X Offset	X Offset	-999.999 - +999.999	0
Y Offset	Y Offset	-999.999 - +999.999	0

Texture Projection Blend AB

This allows to simulate a projected image on a layer. Two FX textures can be used and blended. There are similarities to the effect "Light Texture Projection" [535], but as it combines the Light Layer parameters into the effect itself, no additional Light Layer [666] is needed.

The left image depicts the scene. The Layer texture itself is a white-black-white gradient, the projected texture (FX media) is a cloud image. The Mix parameter is at 100.
For the second image, the Mix parameter was risen to 255.
In the right image, the second FX media (grass image) is taking place as the "Mix AB" parameter is toggled.

Note that (in difference to "Texture Projection Add" [538]) the RGB values from the FX texture overlay the RGB values from the layer texture, but not if there is no projection.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	First Light source (via "Share-layer texture")		-
Mix AB	Mix AB	0-255	0
Media2	Second Light source (via "Share layer texture")	-	-
Light Source X Pos	Light Source X Pos	-999.999 - +999.999	0
Light Source Y Pos	Light Source Y Pos	-999.999 - +999.999	10
Light Source Z Pos	Light Source Z Pos	-999.999 - +999.999	-25
Light Target X Pos	Light Target X Pos	-999.999 - +999.999	0
Light Target Y Pos	Light Target Y Pos	-999.999 - +999.999	0
Light Target Z Pos	Light Target Z Pos	-999.999 - +999.999	0
Light Angle	Light Angle	0-180	20
Light Aspect	Light Aspect	0-20	1
Light Z Roll	Light Z Roll	-9999.99 - +9999.99	0
X Offset	X Offset	-999.999 - +999.999	0
Y Offset	Y Offset	-999.999 - +999.999	0

Texture Projection Mix AB

This allows to simulate a projected image on a layer. Two FX textures can be used and blended. There are similarities to the effect "Light Texture Projection" [535], but as it combines the Light Layer parameters into the effect itself, no additional Light Layer [666] is needed.

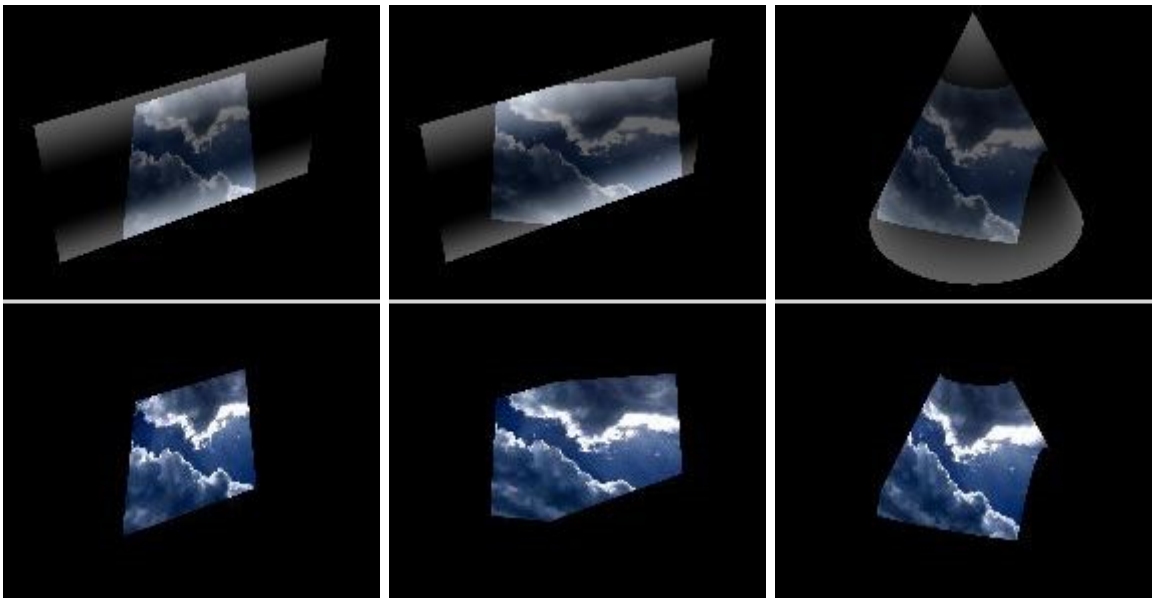
The left image depicts the scene. The Layer texture itself is a white-black-white gradient, the projected texture (FX media) is a cloud image. The Mix parameter is at 100.
For the second image, the Mix parameter was risen to 255.
In the right image, the second FX media (grass image) is taking place as the "Mix AB" parameter is toggled.

Note that (in difference to "Texture Projection Add" [538]) the RGB values from the FX texture overlay the RGB values from the layer texture. This explains why the layer texture disappears when the Mix parameter equals 255.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	First Light source (via "Share-layer texture")		-
Mix AB	Mix AB	0-255	0
Media2	Second Light source (via "Share layer texture")	-	-
Light Source X Pos	Light Source X Pos	-999.999 - +999.999	0
Light Source Y Pos	Light Source Y Pos	-999.999 - +999.999	10
Light Source Z Pos	Light Source Z Pos	-999.999 - +999.999	-25
Light Target X Pos	Light Target X Pos	-999.999 - +999.999	0
Light Target Y Pos	Light Target Y Pos	-999.999 - +999.999	0
Light Target Z Pos	Light Target Z Pos	-999.999 - +999.999	0
Light Angle	Light Angle	0-180	20
Light Aspect	Light Aspect	0-20	1
Light Z Roll	Light Z Roll	-9999.99 - +9999.99	0
X Offset	X Offset	-999.999 - +999.999	0
Y Offset	Y Offset	-999.999 - +999.999	0

Texture Projection Mix



This allows to simulate a projected image on a layer. There are similarities to the effect "Light Texture Projection" [535], but as it combines the Light Layer parameters into the effect itself, no additional Light Layer [666] is needed.

The left image depicts the scene. The Layer texture itself is a white-black-white gradient, the projected texture (FX media) is a cloud image. In the upper image, the Mix parameter is at 100; in the bottom one at 255. For the second image, the position for the effect's light source has changed, resulting in a different angle for the projection. In the right image, a 3D cone was used as a Layer object.

Note that (in difference to "Texture Projection Add" [538]) the RGB values from the FX texture overlay the RGB values from the layer texture. This explains why the layer texture disappears when the Mix parameter equals 255.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Light source (via "Share layer texture")	-	-
Light Source X Pos	Light Source X Pos	-999.999 - +999.999	0
Light Source Y Pos	Light Source Y Pos	-999.999 - +999.999	10
Light Source Z Pos	Light Source Z Pos	-999.999 - +999.999	-25
Light Target X Pos	Light Target X Pos	-999.999 - +999.999	0
Light Target Y Pos	Light Target Y Pos	-999.999 - +999.999	0
Light Target Z Pos	Light Target Z Pos	-999.999 - +999.999	0
Light Angle	Light Angle	0-180	20
Light Aspect	Light Aspect	0-20	1
Light Z Roll	Light Z Roll	-9999.99 - +9999.99	0
X Offset	X Offset	-999.999 - +999.999	0
Y Offset	Y Offset	-999.999 - +999.999	0

6.5.2.4.25 LUT

LUT Camera Arri

These LUT shaders (effects which are based on the principle of so called Look Up Tables, short: LUT) are Arri 3D LUTs for monitor grading.

Simply set the "Mix" to 255 to see the full effect and choose one of the 16 profiles from the drop-down list. More details on:
https://www.arri.com/en/learn-help/learn-help-camera-system/tools/lut-generator

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
LUT Profile	Profile for look up table	0-16	0

LUT Creative

These LUT shaders (effects which are based on the principle of so called Look Up Tables, short: LUT) are intended for color correction.

Simply set the "Mix" to 255 to see the full effect and choose one of the 50 creative LUT profiles from the drop-down list.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
LUT Profile	Profile for look up table	0-50	0

Custom 3D LUT Filter

In difference to all other predefined LUT shaders, this one let's you assign a look up table in terms of an image file (16bit PNG) which allows for a custom color correction or modification.

Examples can be found in the Stock Assets folder under Textures > Custom 3D LUTs. The "neutral_LUT" file is of interest if you like to create your own LUT or convert another LUT format, like a CUBE or 3DL file using your graphics software.

In Adobe Photoshop for example, you would load our neutral_LUT PNG file and then either apply your own color correction or choose to convert an existing LUT file. For that, first add an "Adjustment Layer" > "Color Lookup", select it and then choose in the "Properties" tab which LUT file you like to convert. At the end, please save your file as a 16bit PNG file. For a more detailed description see: https://streamshark.io/obs-guide/converting-cube-3dl-lut-to-image

In PB, simply set the "Mix" to 255 to see the full effect and assign the PNG image as the "LUT Media" to the layer, e.g. Video or Output Layer.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
LUT Media	Image file with look up table	-	0

LUT Filter 1

These LUT shaders (effects which are based on the principle of so called Look Up Tables, short: LUT) are intended for color correction.

Simply set the "Mix" to 255 to see the full effect and choose one of the 55 predefined LUT profiles from the drop-down list.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
LUT Profile	Profile for look up table	0-55	0

LUT Filter 2

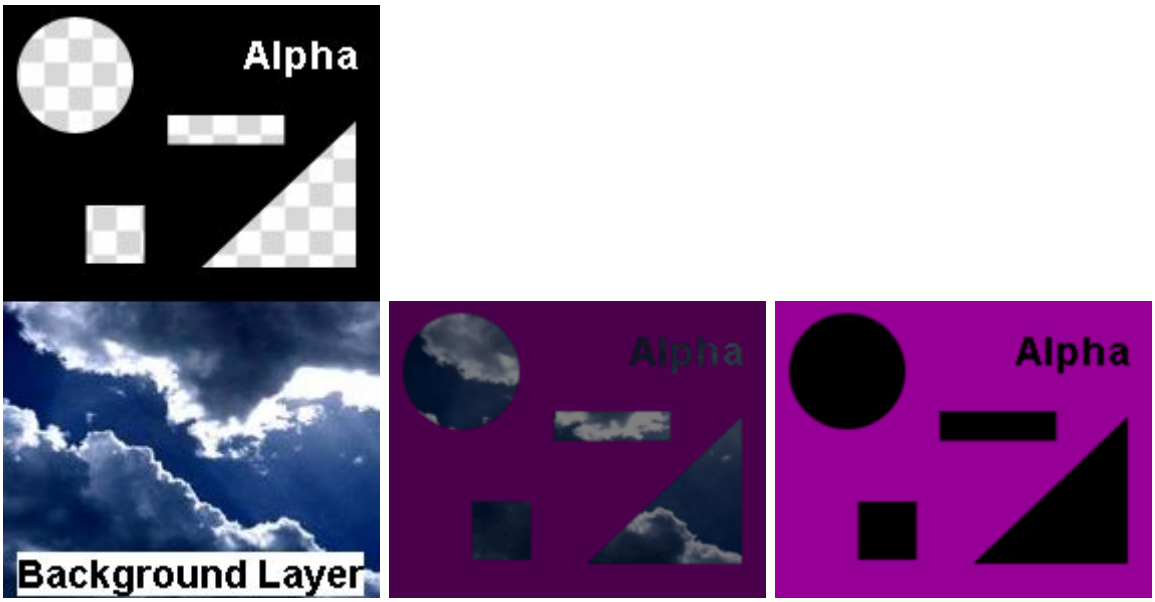
These LUT shaders (effects which are based on the principle of so called Look Up Tables, short: LUT) are intended for color correction.

Simply set the "Mix" to 255 to see the full effect and choose one of the 62 predefined LUT profiles from the drop-down list.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
LUT Profile	Profile for look up table	0-62	0

6.5.2.4.26 Masking

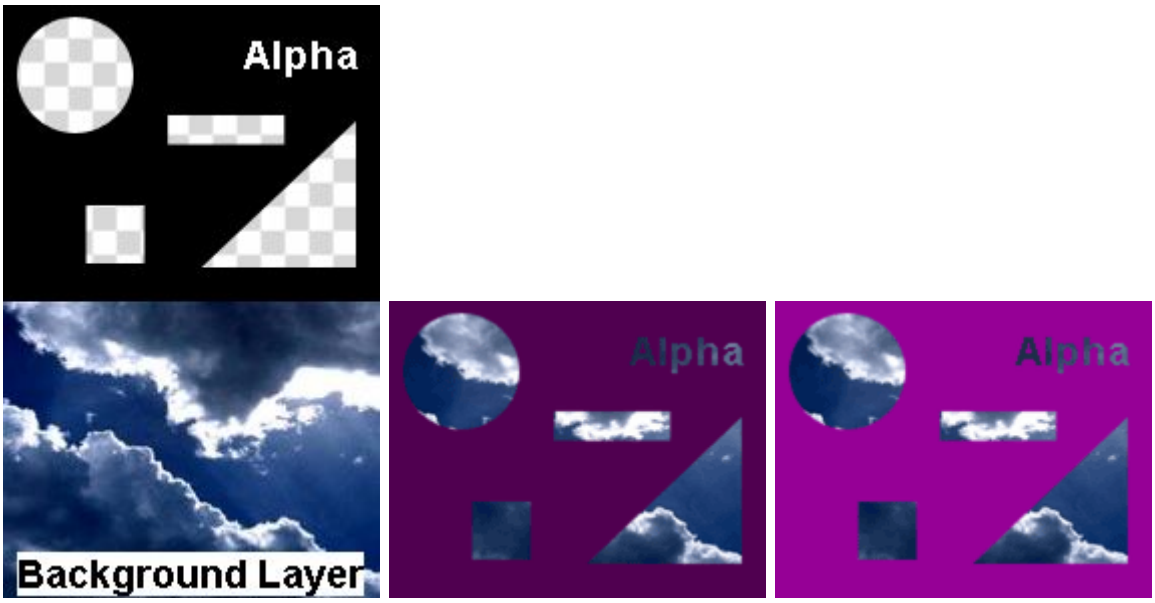
Alpha Black Fill RGB



Turns all parts containing alpha into black and allows choosing any RGB color for the non-transparent parts of the image.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	0
Green	Green level, Color Picker	0-255	0
Blue	Blue level, Color Picker	0-255	0

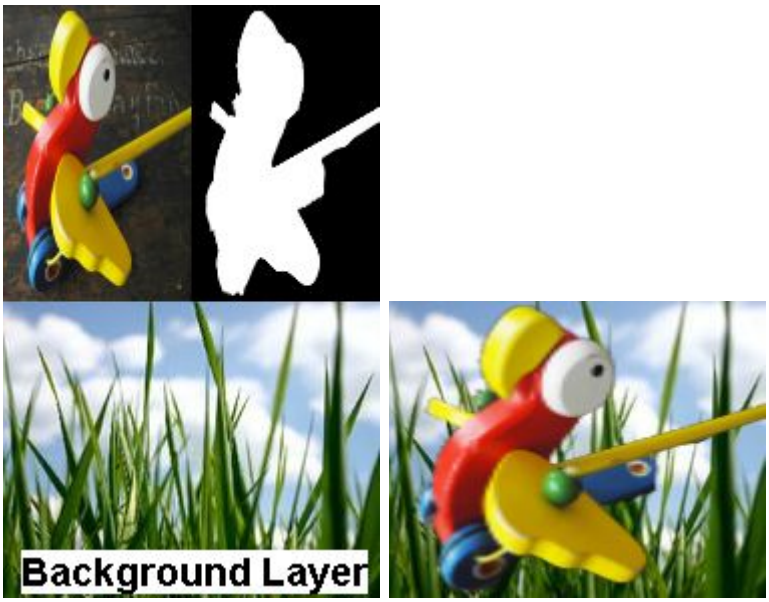
Alpha Fill RGB



All parts of the image containing alpha stay transparent, allows choosing any RGB color for the non-transparent parts of the image.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	0
Green	Green level, Color Picker	0-255	0
Blue	Blue level, Color Picker	0-255	0

Alpha Left Right



The effect expects one media file on the layer that contains two contents side-by-side. The left half represents the video itself whilst the masking information is in the right half. You may prepare such content using the Image Converter [2073].
In difference to the effect "Alpha Left Right - Retain Resolution" the original side-by-side content consists of scaled halfs and the resulting image is stretched (back) by 100%. This saves file size, resolution and playback performance but results in worse quality.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Alpha Left Right - Retain Resolution

The effect expects one media file on the layer that contains two contents side-by-side. The left half represents the video itself whilst the masking information is in the right half. You may prepare such content using the Image Converter [2073].
In difference to the effect "Alpha Left Right" the original side-by-side content consists of halfs that are not scaled and hence, the resulting image does not need to be stretched (back). This results in a better quality but a higher file size, resolution and playback performance.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

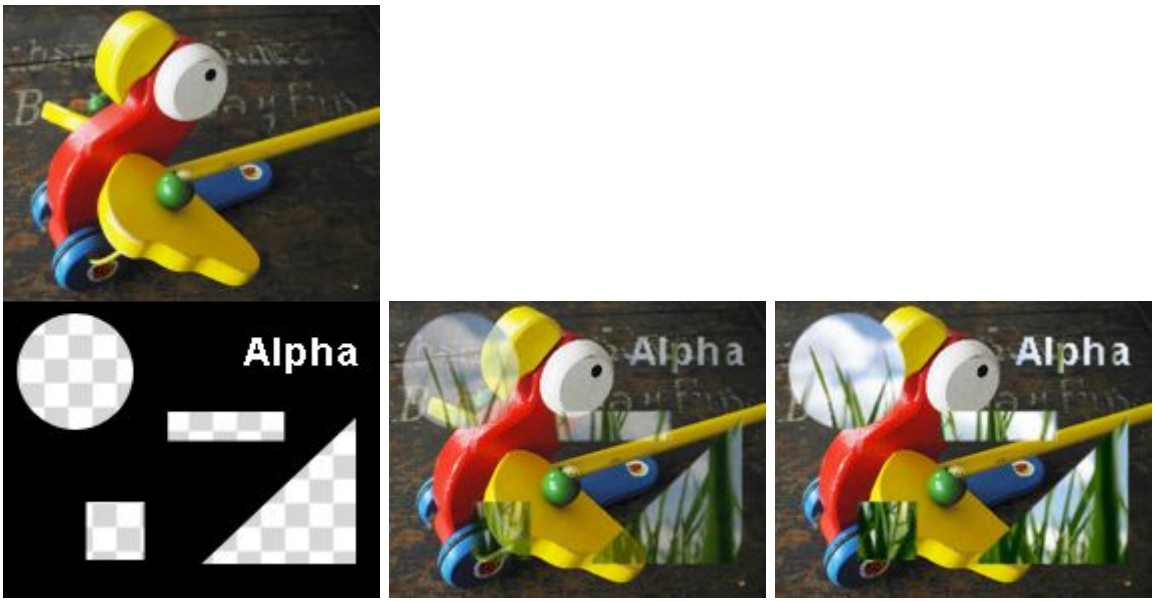
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Alpha Media As Mask Invert

All parts of the layer texture containing alpha will stay transparent, all non-transparent parts of an additional media file will turn the corresponding parts of the layer texture transparent as well.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Media	Alpha Media as Mask Invert		

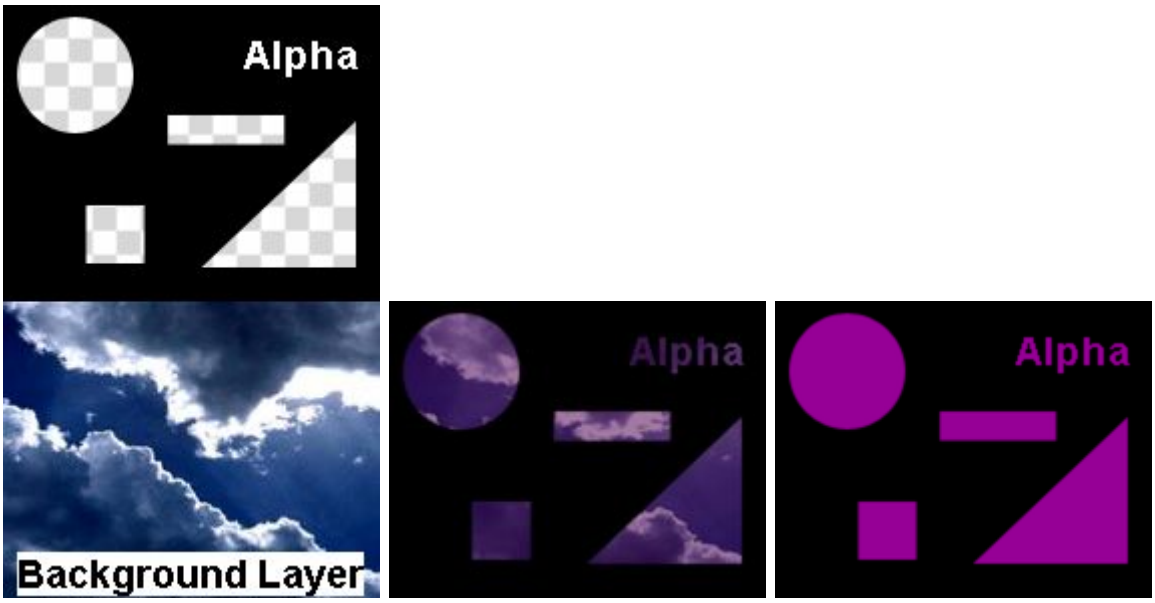
Alpha Media As Mask



All parts of the layer texture containing alpha will stay transparent, all transparent parts of an additional media file will turn the corresponding parts of the layer texture transparent as well.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Media	Alpha Media as Mask		

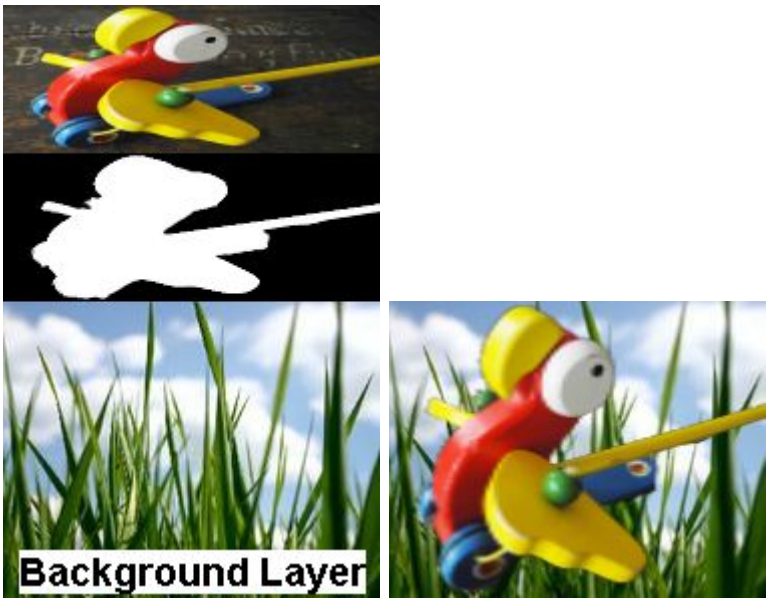
Alpha Replace



All parts of the layer texture containing alpha will be replaced by any RGB color.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	0
Green	Green level, Color Picker	0-255	0
Blue	Blue level, Color Picker	0-255	0

Alpha Top Bottom

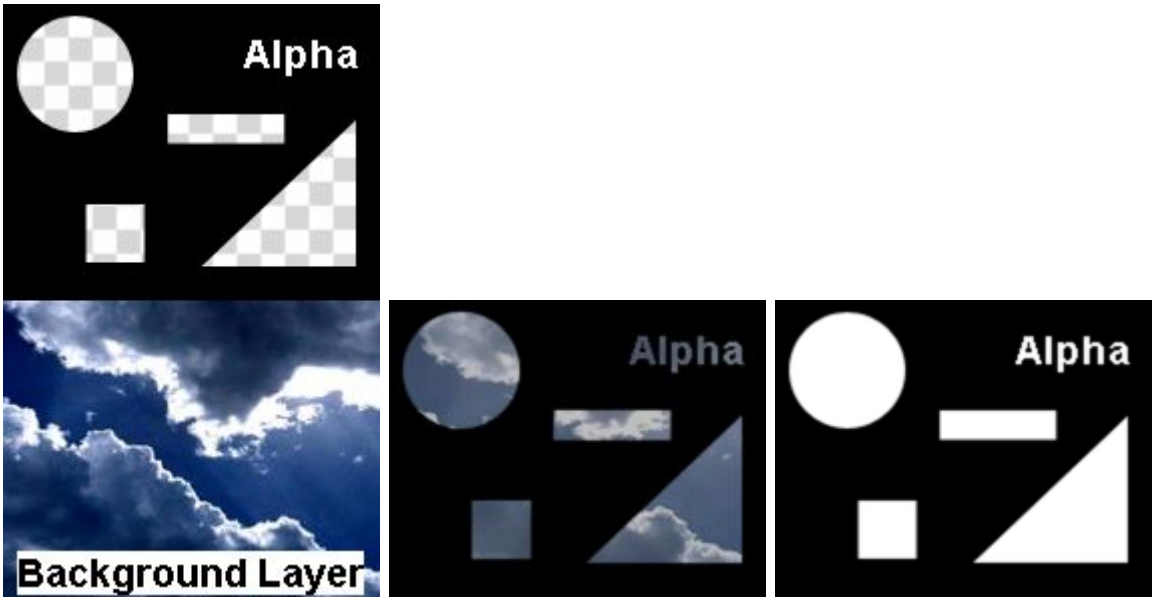


The effect expects one media file on the layer that contains two information on top of each other. The upper half represents the video itself whilst the masking information is in the lower half. You may prepare such content using the Image Converter [2073].

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

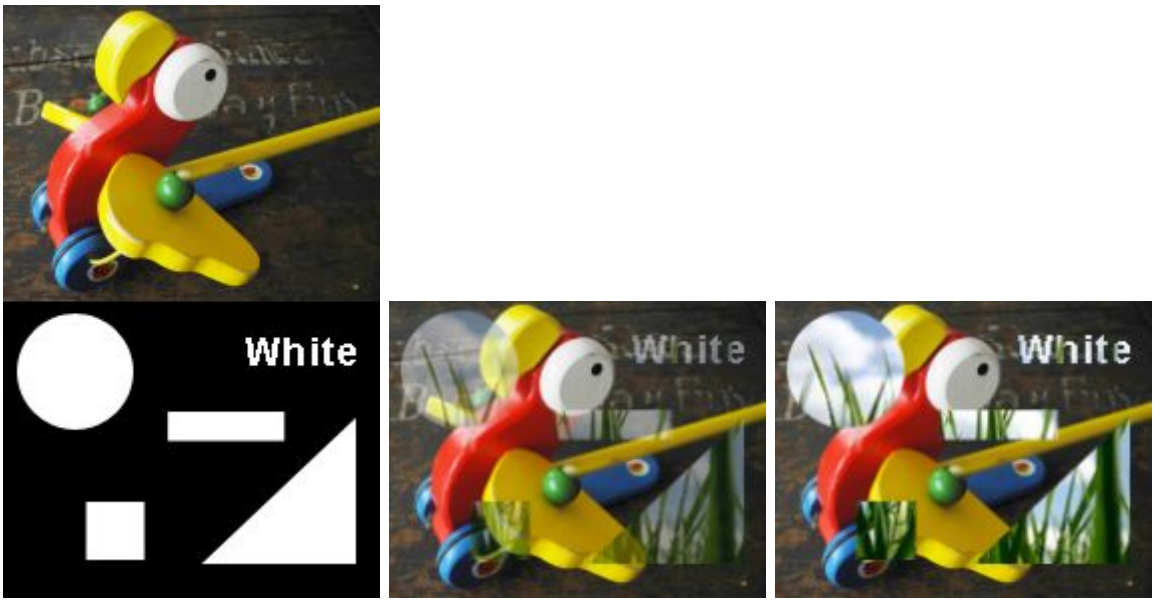
BW Mask



All parts of the layer texture containing alpha will be turned into white, all non-transparent parts will be turned into black.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

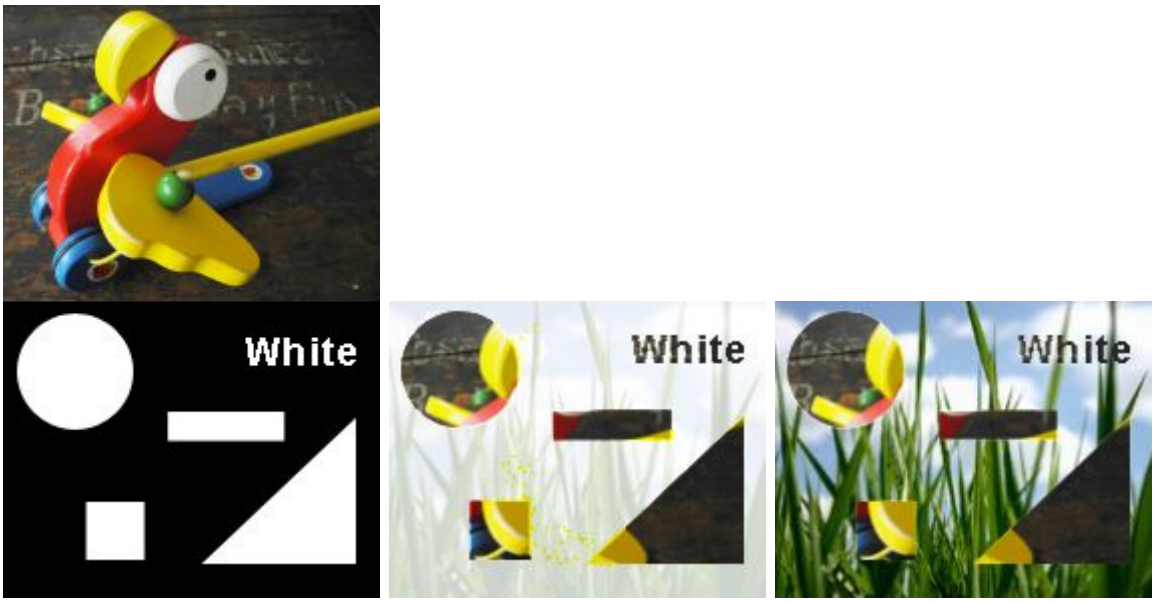
BW Media As Mask Invert



All white parts of an additional media file will turn the corresponding parts of the layer texture transparent, black parts do not influence the layer texture.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Media	BW Media as Mask Invert		

BW Media as Mask PreMul



All black parts of an additional media file will turn the corresponding parts of the layer texture transparent, white parts do not influence the layer texture. In addition to the "BW Media As Mask [554]" effect the RGB values of the keyed out area are pre-multiplied before being turned transparent. This means that all RGB values between 1-255 are set to 255 turning most pixels into white ones. Only RGB values of 0 stay unchanged. In the second image you can see that some pixels from the wing are rendered in yellow because originally their blue value was 0.

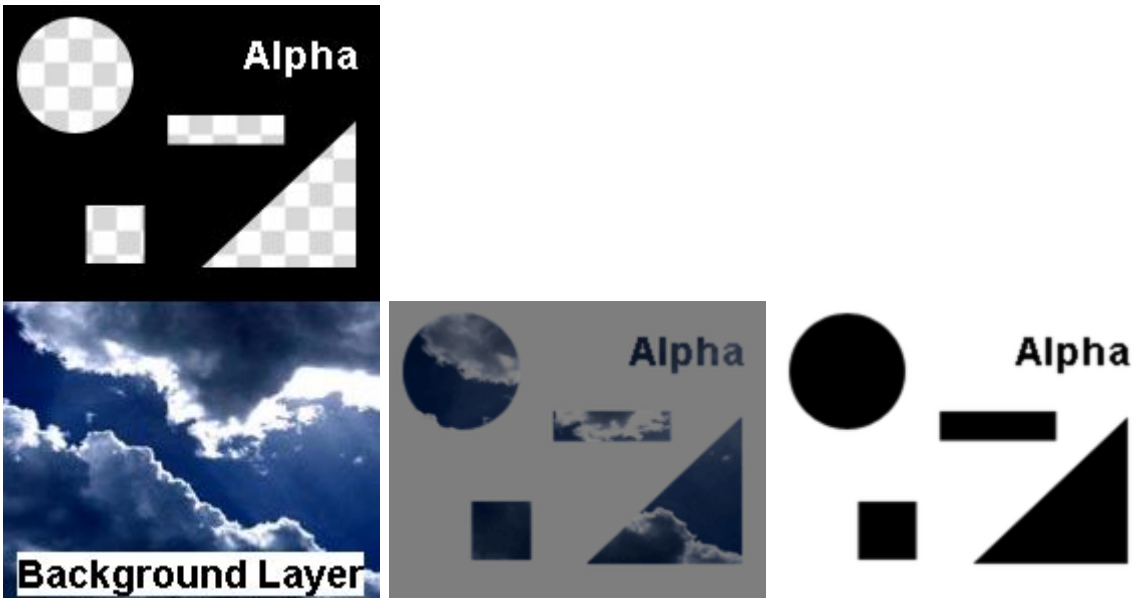
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Media file as source for overlay or mask	-	-

BW Media As Mask

All black parts of an additional media file will turn the corresponding parts of the layer texture transparent, white parts do not influence the layer texture.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Media	BW Media as Mask		

WB Mask



All parts of the layer texture containing alpha will be turned into black, all non-transparent parts will be turned into white.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0

6.5.2.4.27 Mirror

Horizontal Flip

Mirrors the texture layer horizontally, what was on the left site will be on the right.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

HV Flip

Mirrors the texture layer vertically and horizontally; in other words, it rotates the layer by 180°.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

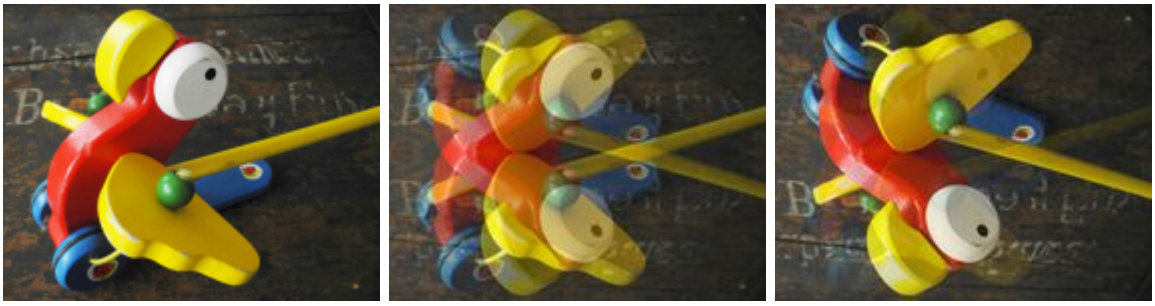
Mirror 2D

Mirrors the layer texture vertically with a slight transparent fade-out. The vertical position can be adjusted, the mirrored part may be colored by any RGB color as well.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Pos V	Vertical Position of the Mirroring	0-65535	32768
Alpha	Alpha level of the mirrored image	0-255	128
Red	Red level, Color Picker	0-255	128
Green	Green level, Color Picker	0-255	128
Blue	Blue level, Color Picker	0-255	128

Vertical Flip



Mirrors the texture layer vertically, what was on the top side, will be on the bottom.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

Horizontal Dual Media Overlay

Mixes the current texture (media file) from the layer with the FX textures. Both FX textures are displayed next to each other, taking up the full height but only 1/2 width of the layer texture.
The layer texture holds the resolution information. In the Stock Assets content you may find transparent images with different resolutions (`C:\Christie\content\Stock Assets\Textures\Alpha Dummy or C:\coolux\...`).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	Media1	-	-
Media2	Media2	-	-

Horizontal Quad Media Overlay

Mixes the current texture (media file) from the layer with the FX textures. All four FX textures are displayed next to each other, taking up the full height but only 1/4 width of the layer texture.
The layer texture holds the resolution information. In the Stock Assets content you may find transparent images with different resolutions (`C:\Christie\content\Stock Assets\Textures\Alpha Dummy or C:\coolux\...`).

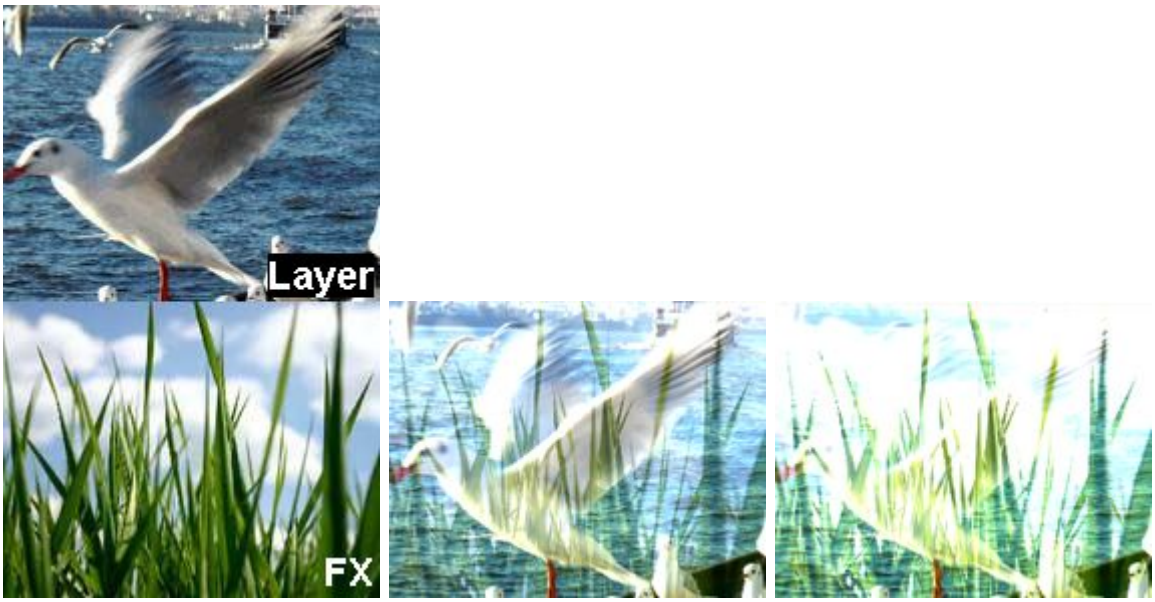
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	Media1	-	-
Media2	Media2	-	-
Media3	Media3	-	-
Media4	Media4	-	-

Horizontal Triple Media Overlay

Mixes the current texture (media file) from the layer with the FX textures. All three FX textures are displayed next to each other, taking up the full height but only 1/3 of the width of the layer texture.
The layer texture holds the resolution information. In the Stock Assets content you may find transparent images with different resolutions (`C:\Christie\content\Stock Assets\Textures\Alpha Dummy or C:\coolux\...`).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	Media1	-	-
Media2	Media2	-	-
Media3	Media3	-	-

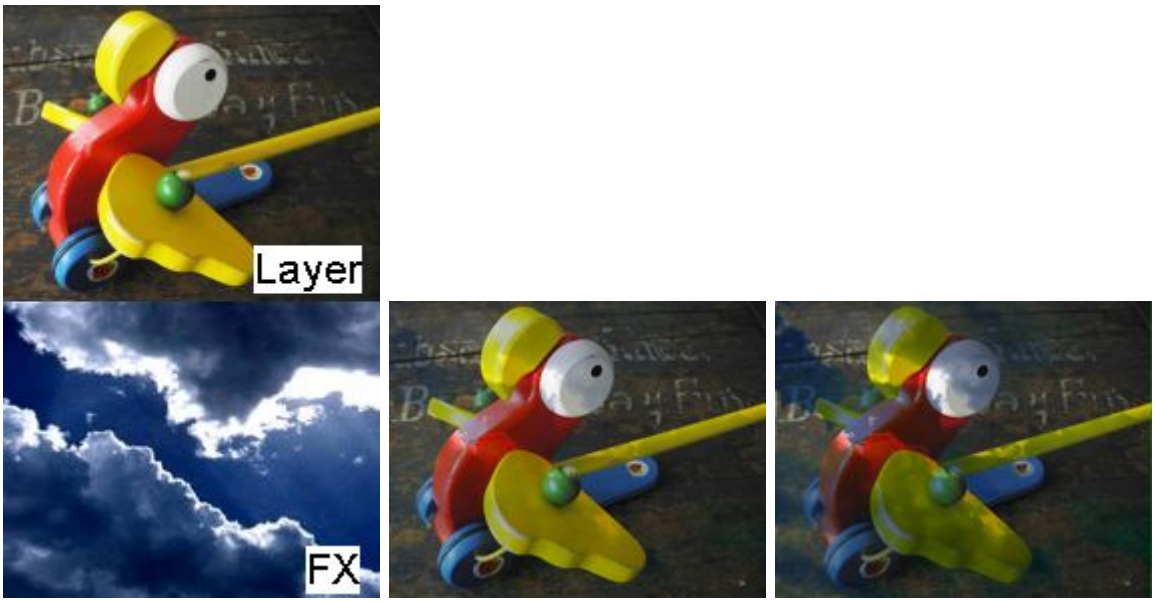
Media Overlay Add



Allows overlaying the layer texture with a media file by adding the pixel RGB values of the media file with the layer texture. Position and scaling of the overlaying media file can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
X	X Position of the overlaying media file	0-65535	32768
Y	Y Position of the overlaying media file	0-65535	32768
Width	Red level, Color Picker	0-65535	65535
Height	Green level, Color Picker	0-65535	65535
Media	Overlaying Media File		

Media Overlay Darken



Allows overlaying the layer texture with a media file by darkening the layer texture with the RGB values of the media file. Position an scaling of the overlaying media can be adjusted.

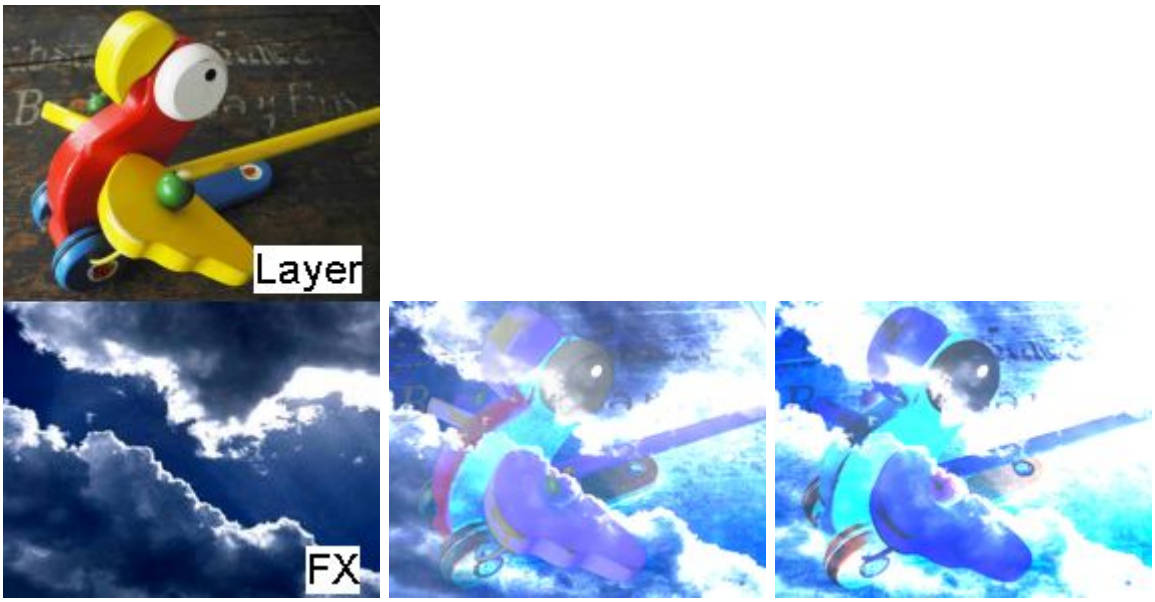
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
X	X	0-65535	32768
Y	Y	0-65535	32768
Width	Width	0-65535	65535
Height	Height	0-65535	65535
Media	Media file as source for overlay or mask	-	-

Media Overlay Divide Inv

Allows overlaying the layer texture with a media file by inverting the division of layer texture and RGB values of the media file. Position an scaling of the overlaying media can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
X	X	0-65535	32768
Y	Y	0-65535	32768
Width	Width	0-65535	65535
Height	Height	0-65535	65535
Media	Media file as source for overlay or mask	-	-

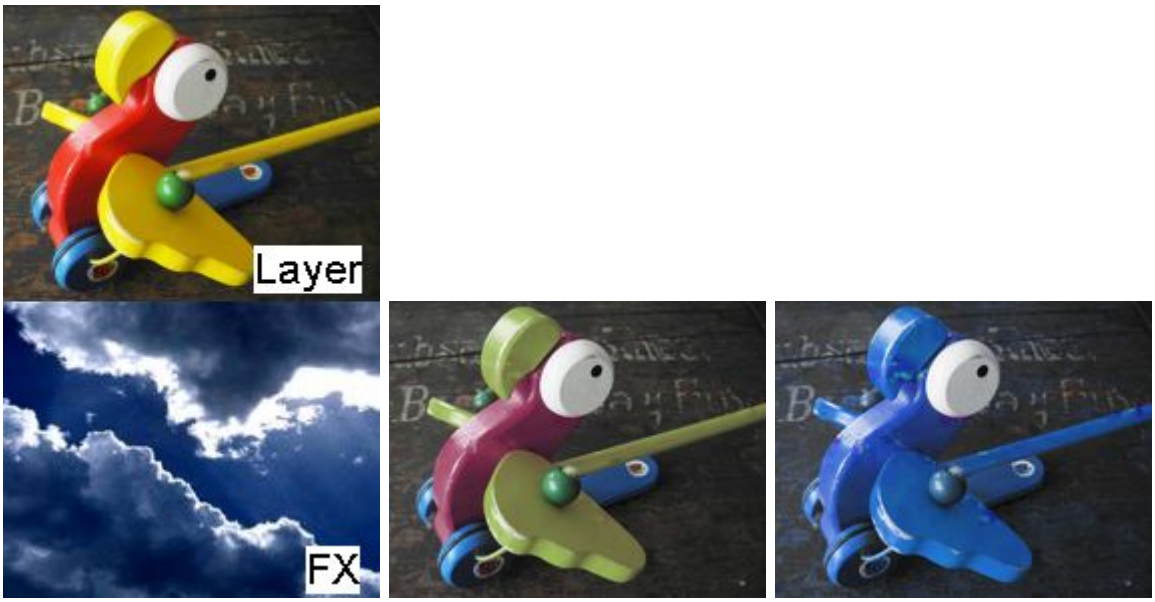
Media Overlay Divide



Allows overlaying the layer texture with a media file by dividing the layer texture and the RGB values of the media file. Position an scaling of the overlaying media can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
X	X	0-65535	32768
Y	Y	0-65535	32768
Width	Width	0-65535	65535
Height	Height	0-65535	65535
Media	Media file as source for overlay or mask	-	-

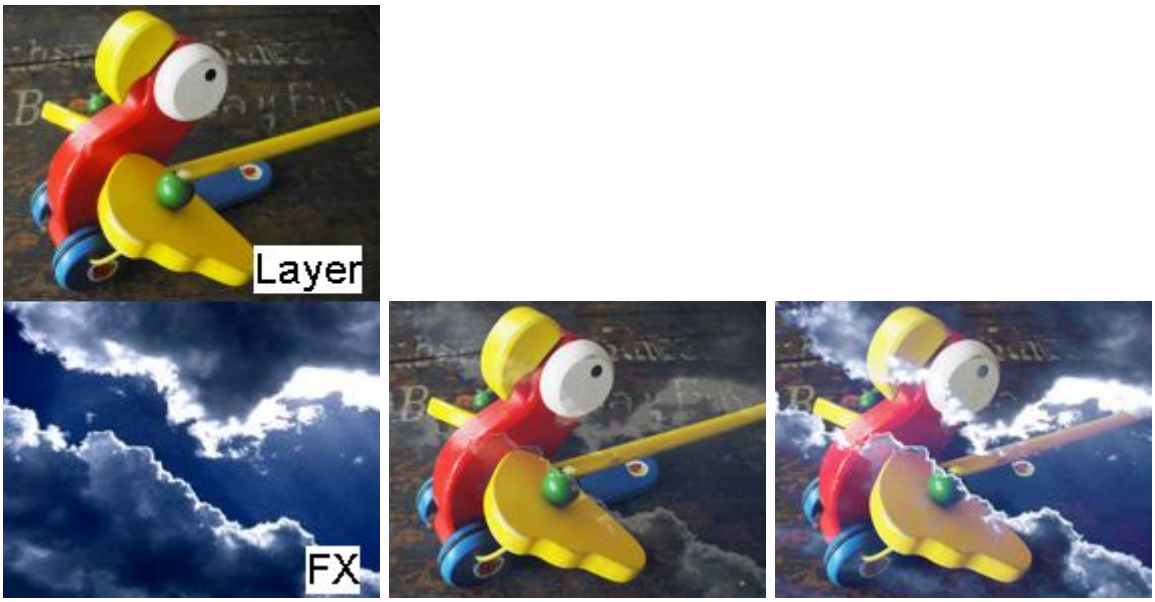
Media Overlay Hue



Allows overlaying the layer texture with a media file by mixing the layer texture with the RGB hue values of the media file. Position an scaling of the overlaying media can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
X	X	0-65535	32768
Y	Y	0-65535	32768
Width	Width	0-65535	65535
Height	Height	0-65535	65535
Media	Media file as source for overlay or mask	-	-

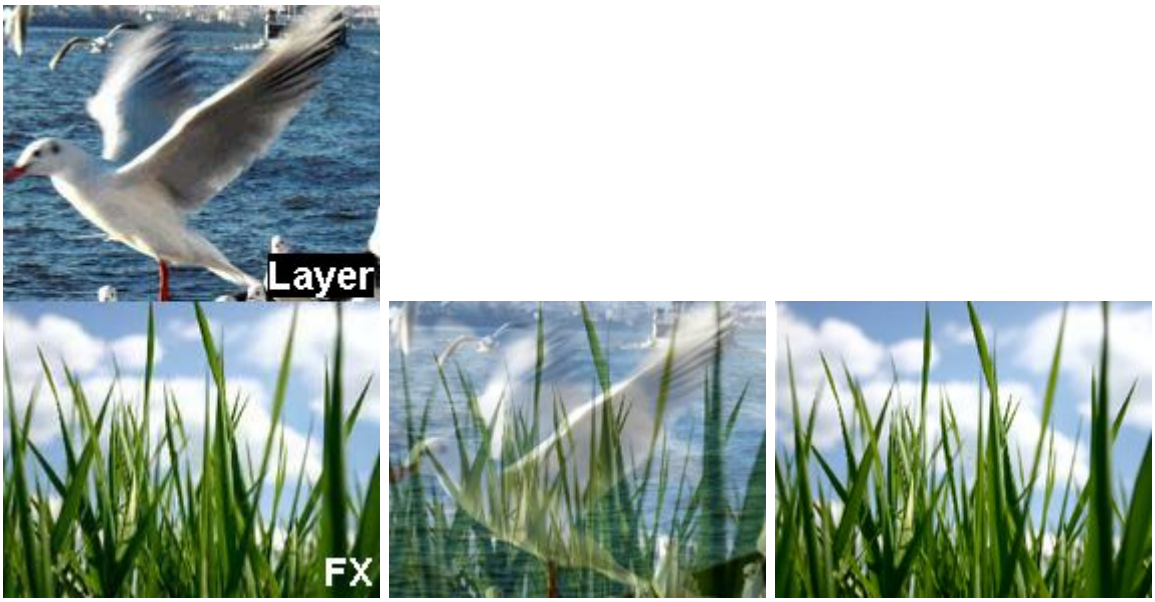
Media Overlay Lighten



Allows overlaying the layer texture with a media file by lightening up the layer texture with the RGB values of the media file. Position an scaling of the overlaying media can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
X	X	0-65535	32768
Y	Y	0-65535	32768
Width	Width	0-65535	65535
Height	Height	0-65535	65535
Media	Media file as source for overlay or mask	-	-

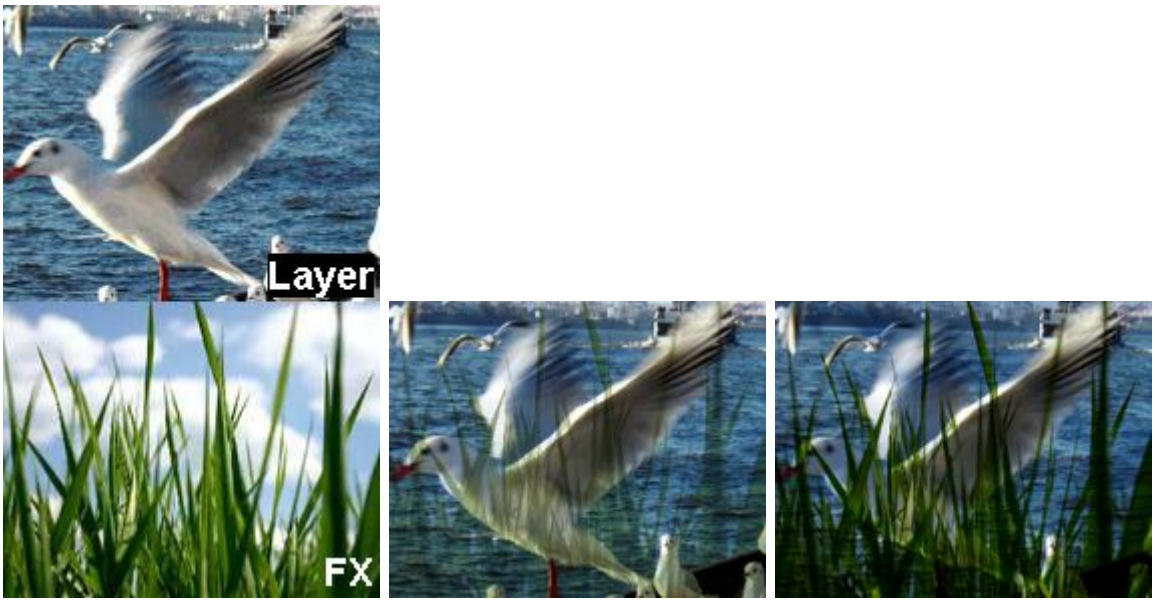
Media Overlay Mix



Allows overlaying the layer texture with a media file by mixing the pixel RGB values of the layer texture with the media file. Position and scaling of the overlaying media file can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
X	X Position of the overlaying media file	0-65535	32768
Y	Y Position of the overlaying media file	0-65535	32768
Width	Red level, Color Picker	0-65535	65535
Height	Green level, Color Picker	0-65535	65535
Media	Overlaying Media File		

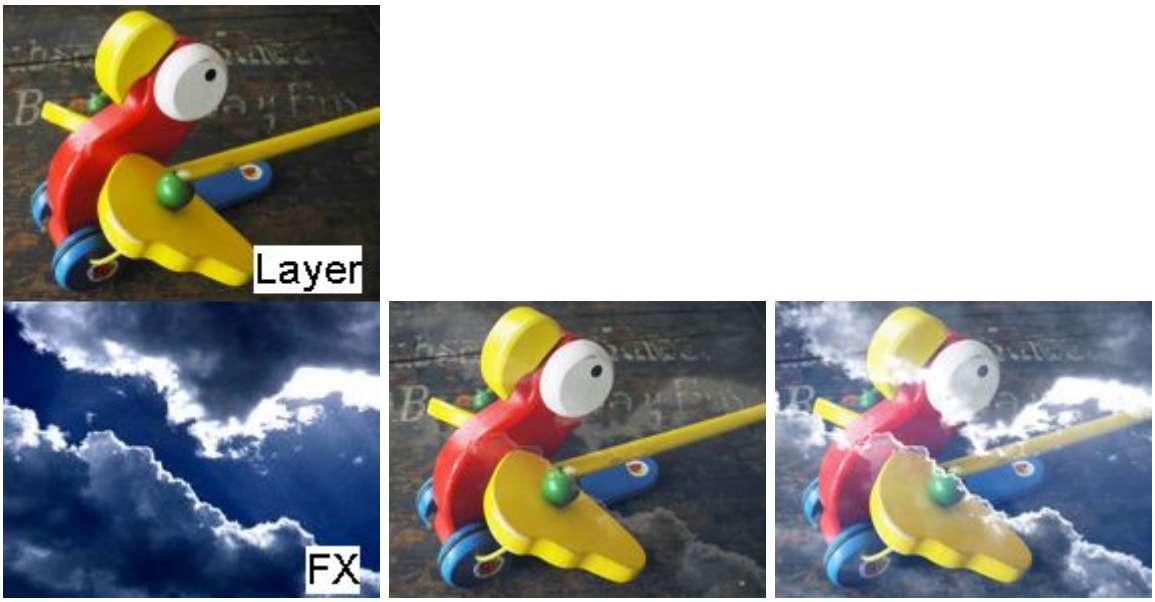
Media Overlay Multiply



Allows overlaying the layer texture with a media file by multiplying the pixel RGB values of the layer texture with the media file. Position and scaling of the overlaying media file can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
X	X Position of the overlaying media file	0-65535	32768
Y	Y Position of the overlaying media file	0-65535	32768
Width	Red level, Color Picker	0-65535	65535
Height	Green level, Color Picker	0-65535	65535
Media	Overlaying Media File		

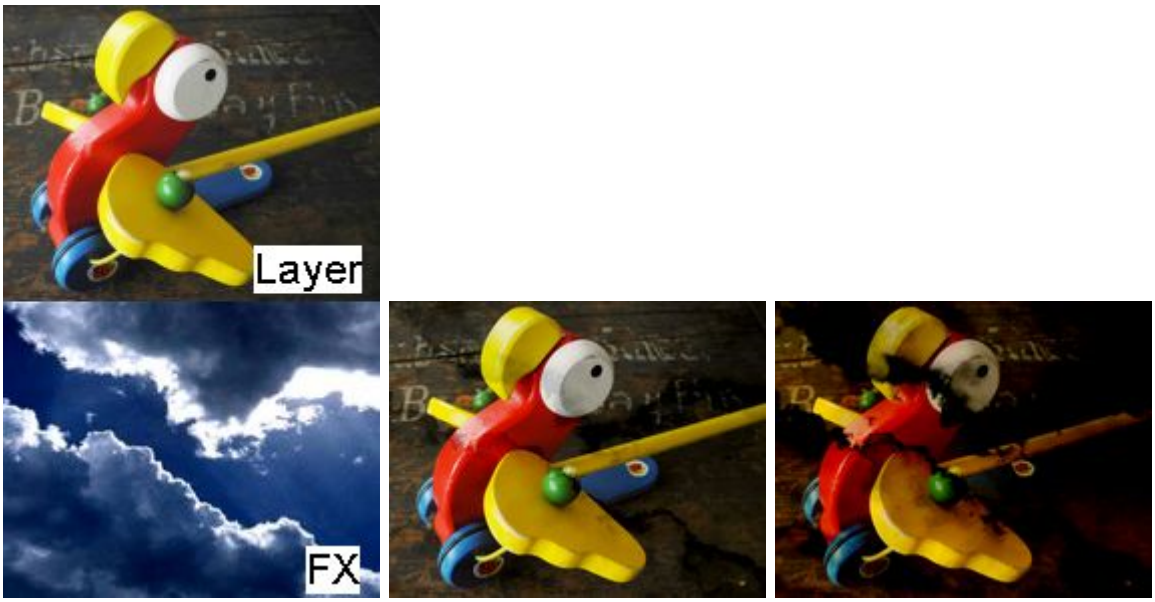
Media Overlay Screen



Allows overlaying the layer texture with a media file by using the screen blend mode, i.e. negating the RGB values of both textures, then multiplying them and negating them again. The result is a brighter image. Position an scaling of the overlaying media can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
X	X	0-65535	32768
Y	Y	0-65535	32768
Width	Width	0-65535	65535
Height	Height	0-65535	65535
Media	Media file as source for overlay or mask	-	-

Media Overlay Subtract



Allows overlaying the layer texture with a media file by subtracting the RGB values of the media file from the layer texture. Position an scaling of the overlaying media can be adjusted.

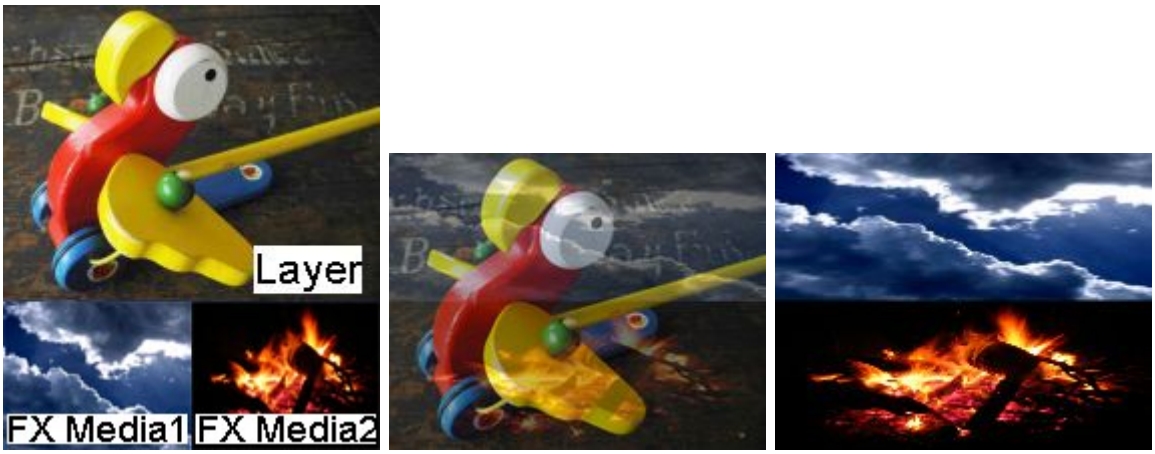
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
X	X	0-65535	32768
Y	Y	0-65535	32768
Width	Width	0-65535	65535
Height	Height	0-65535	65535
Media	Media file as source for overlay or mask	-	-

Quad Media Overlay

Mixes the current texture (media file) from the layer with the FX textures. All four FX textures are displayed in a 2x2 grid, each taking up 1/2 height and 1/2 width of the layer texture.
The layer texture holds the resolution information. In the Stock Assets content you may find transparent images with different resolutions (`C:\Christie\content\Stock Assets\Textures\Alpha Dummy or C:\coolux\...`).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	Media1	-	-
Media2	Media2	-	-
Media3	Media3	-	-
Media4	Media4	-	-

Vertical Dual Media Overlay



Mixes the current texture (media file) from the layer with the FX textures. Both FX textures are displayed one upon the other, taking up the full width but only 1/2 height of the layer texture.
The layer texture holds the resolution information. In the Stock Assets content you may find transparent images with different resolutions (`C:\Christie\content\Stock Assets\Textures\Alpha Dummy or C:\coolux\...`).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	Media1	-	-
Media2	Media2	-	-

Vertical Quad Media Overlay

Mixes the current texture (media file) from the layer with the FX textures. All four FX textures are displayed one upon the other, taking up the full width but only 1/4 height of the layer texture.
The layer texture holds the resolution information. In the Stock Assets content you may find transparent images with different resolutions (`C:\Christie\content\Stock Assets\Textures\Alpha Dummy or C:\coolux\...`).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	Media1	-	-
Media2	Media2	-	-
Media3	Media3	-	-
Media4	Media4	-	-

Vertical Triple Media Overlay

Mixes the current texture (media file) from the layer with the FX textures. All three FX textures are displayed one upon the other, taking up the full width but only 1/3 height of the layer texture.
The layer texture holds the resolution information. In the Stock Assets content you may find transparent images with different resolutions (`C:\Christie\content\Stock Assets\Textures\Alpha Dummy or C:\coolux\...`).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media1	Media1	-	-
Media2	Media2	-	-
Media3	Media3	-	-

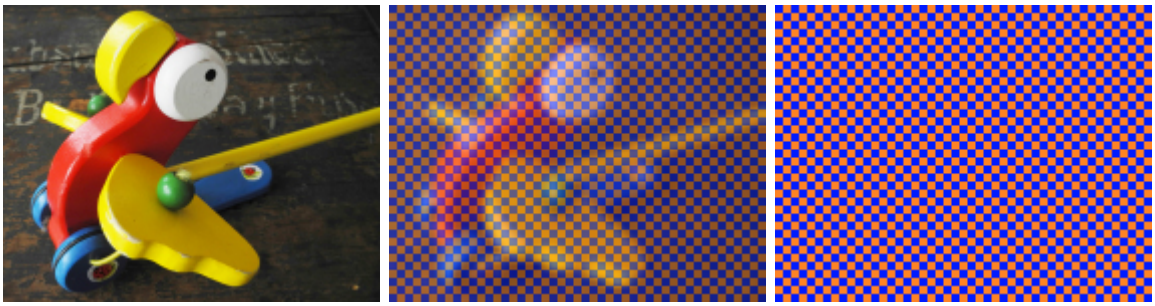
6.5.2.4.29 Pattern

BinaryPattern

Turns the layer texture into a binary pattern, the number of horizontal and vertical lines is adjustable.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Horizontal	Horizontal	0-16	0
Vertical	Vertical	0-16	0
Invert	Inverts the RGB levels or the effect	0-1	0

Checkerboard - Odd Even



This effect controls the RGBA value of all odd pixels and that from all even pixels, whereas adjacent pixels are never in the same "group".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red1	Red level for odd pixels	0-255	0
Green1	Green level for odd pixels	0-255	0
Blue1	Blue level for odd pixels	0-255	0
Alpha1	Alpha level for odd pixels	0-255	255
Red2	Red level for even pixels	0-255	255
Green2	Green level for even pixels	0-255	255
Blue2	Blue level for even pixels	0-255	255
Alpha2	Alpha level for even pixels	0-255	255

Checkerboard - Transparency

This effect controls the transparency of all odd pixels and that from all even pixels, whereas adjacent pixels are never in the same "group".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Alpha1	Alpha level for all odd pixels	0-255	0
Alpha2	Alpha level for all even pixels	0-255	0

Checkerboard

Adds a checkerboard of any RGB color and alpha level to the layer texture. The following parameters can be adjusted as well: Amount, Softness, Size, Offset X&Y.

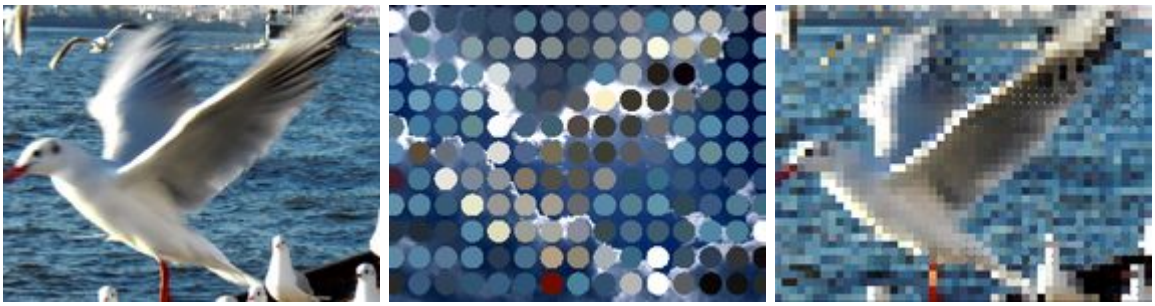
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of checkers	0-65535	1024
Softness	Softness of checker edges	0-65535	32
Size	Size of checkers	0-65535	32768
Offset X	Horizontal Offset	0-65535	32768
Offset Y	Vertical Offset	0-65535	32768
Red	Red level, Color Picker	0-255	192
Green	Green level, Color Picker	0-255	192
Blue	Blue level, Color Picker	0-255	192
Alpha	Alpha level	0-255	255

Grid

Adds a grid of any RGB color and alpha level to the layer texture. The following parameters can be adjusted as well: Amount, Softness, Size, Offset X&Y.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of grid segments	0-65535	1024
Softness	Softness of grid edges	0-65535	32
Size	Size of grid	0-65535	32768
Offset X	Horizontal Offset	0-65535	32768
Offset Y	Vertical Offset	0-65535	32768
Red	Red level, Color Picker	0-255	192
Green	Green level, Color Picker	0-255	192
Blue	Blue level, Color Picker	0-255	192
Alpha	Alpha level	0-255	255

LED



Turns the layer texture into single LED spots.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of LED spots	0-65535	8192
Circle	Size of LED spots	0-65535	28000
Alpha	Alpha level of the gaps between the LED spots	0-255	0

Lines Horizontal

Adds horizontal lines of any RGB color and alpha level to the layer texture. The following parameters can be adjusted as well: Amount, Softness, Size and the Offset.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of lines	0-65535	1024
Softness	Softness of line edges	0-65535	32
Size	Size of lines	0-65535	32768
Offset	Vertical Offset	0-65535	32768
Red	Red level, Color Picker	0-255	192
Green	Green level, Color Picker	0-255	192
Blue	Blue level, Color Picker	0-255	192
Alpha	Alpha level	0-255	255

Lines Vertical

Adds vertical lines of any RGB color and alpha level to the layer texture. The following parameters can be adjusted as well: Amount, Softness, Size and the Offset.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of lines	0-65535	1024
Softness	Softness of line edges	0-65535	32
Size	Size of lines	0-65535	32768
Offset	Horizontal Offset	0-65535	32768
Red	Red level, Color Picker	0-255	0
Green	Green level, Color Picker	0-255	0
Blue	Blue level, Color Picker	0-255	0
Alpha	Alpha level	0-255	255

Pixels

Turns the layer texture into a pattern of pixel dots. Threshold, amount and circularity can be adjusted, as well as color, brightness and alpha value.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Threshold	Threshold	0-255	128
Amount	Amount	0-65535	6000
Circle	Circle	0-65535	32768
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Level	Level	0-255	128

Simple Outline

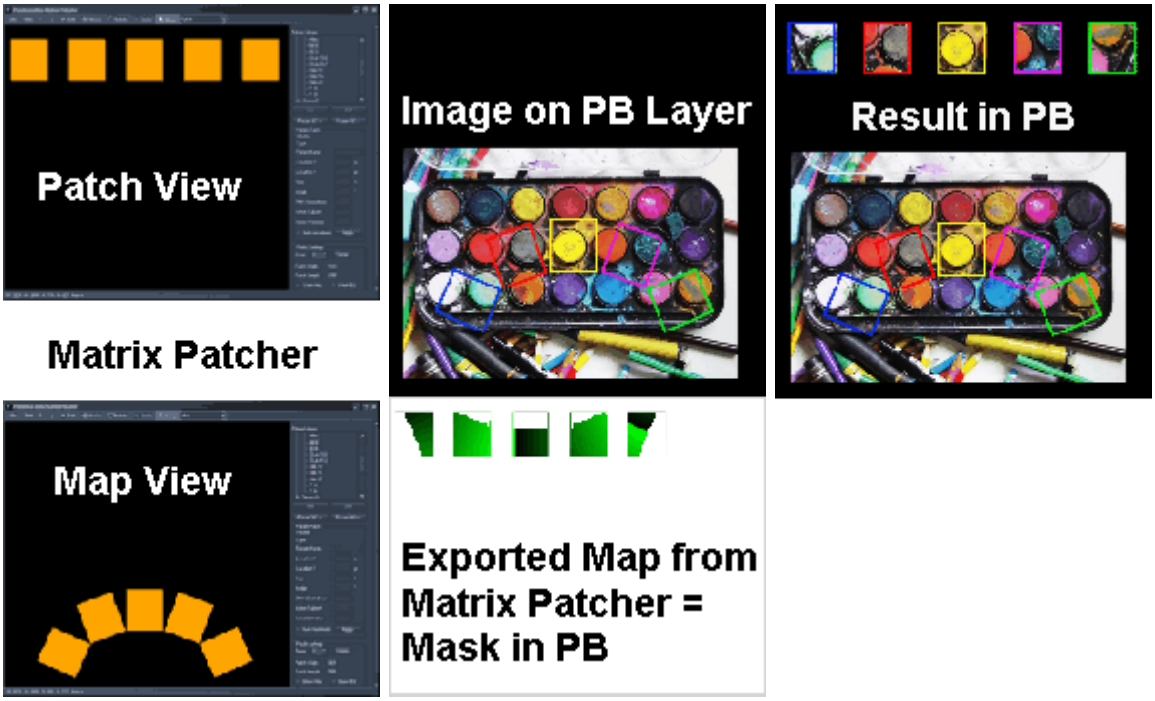
Adds a simple outline of any RGB color between transparent / non-transparent parts of the layer texture and removes the non-transparent parts of the file.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255

6.5.2.4.30 Remap

ReMap Overlay



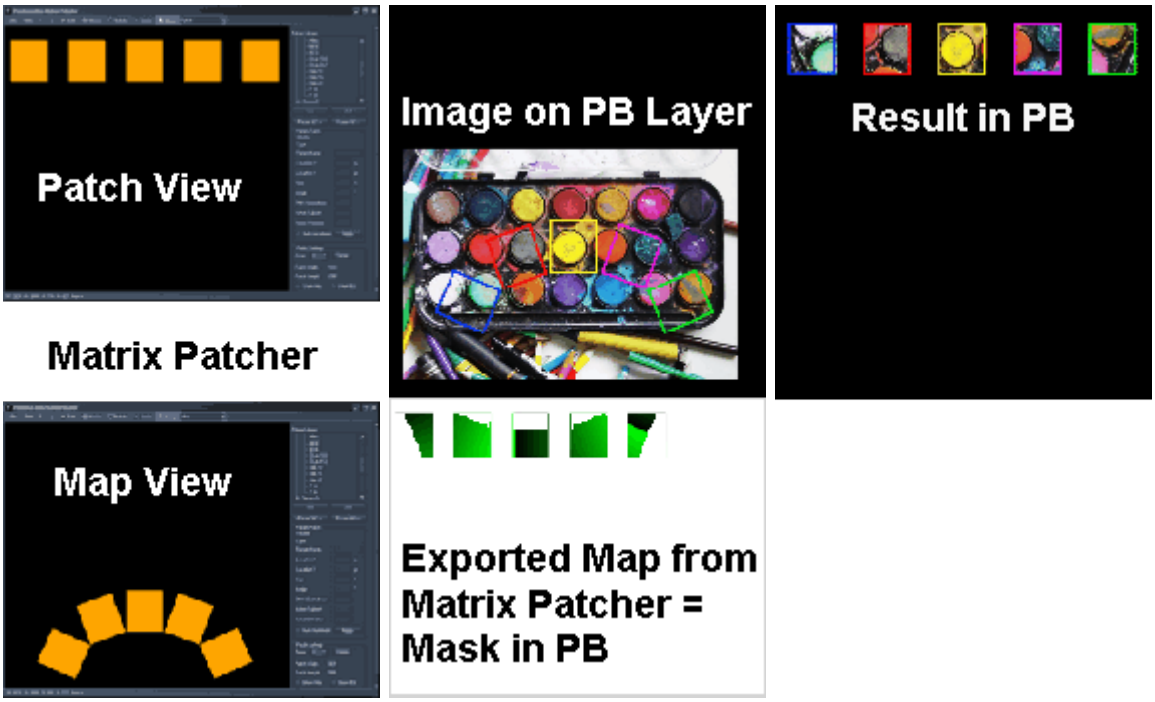
Reads the map exported from the Matrix Patcher [2077] and routes automatically any source pixel to the stored output location. Thus it routes the pixels according to the Patch / Map View setup in the Matrix Patcher. The routed pixels overlay the original image.

Note that the effect has to be assigned to an output layer and is only visible in the Preview if it is set to an output view, not the global camera.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Media	Media File as Mask	0-255	255

ReMap



Reads the map exported from the Matrix Patcher [2077] and routes automatically any source pixel to the stored output location. Thus it routes the pixels according to the Patch / Map View setup in the Matrix Patcher. Only the routed pixels are visible.

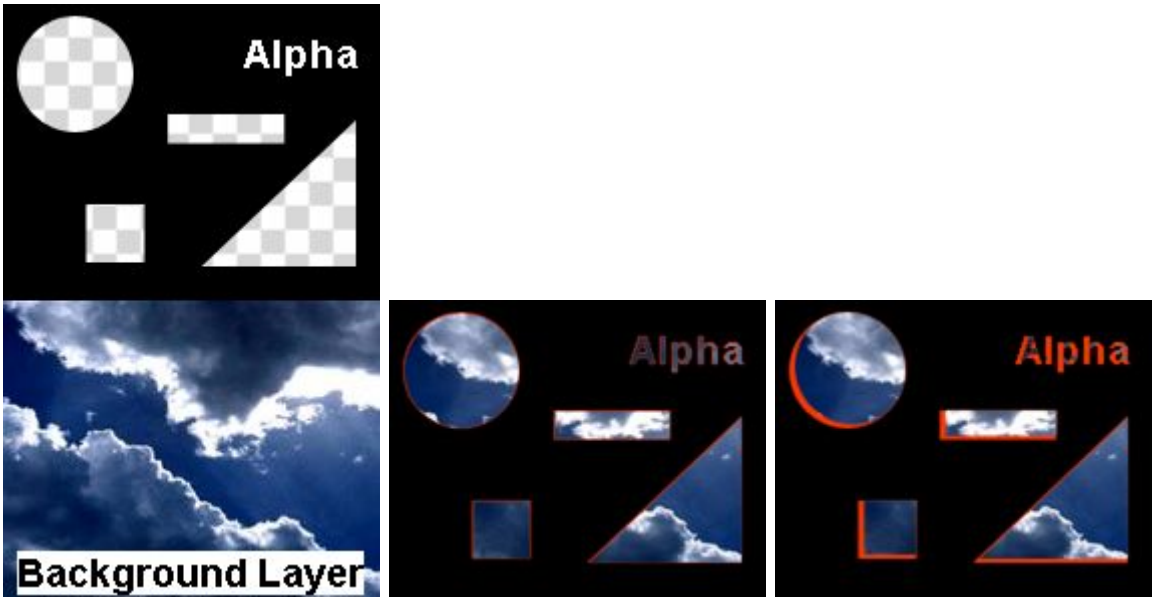
Note that the effect has to be assigned to an output layer and is only visible in the Preview if it is set to an output view, not the global camera.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Media	Media File as Mask	0-255	255

6.5.2.4.31 Shadows

Drop Shadow Alpha



Adds a drop shadow at the transitions between transparent and non-transparent parts of the layer texture. The shadow's position, color and alpha value can be adjusted.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
X	X Position	0-65535	33280
Y	Y Position	0-65535	32000
Red	Red level, Color Picker	0-255	0
Green	Green level, Color Picker	0-255	0
Blue	Blue level, Color Picker	0-255	0
Alpha	Alpha level	0-255	128

6.5.2.4.32 Shapes

Circle2D Mask

Adds a circular mask to the layer texture. The mask's color, alpha, size and radius value can be adjusted, as well as its position, the edge's softness and fill-behavior.

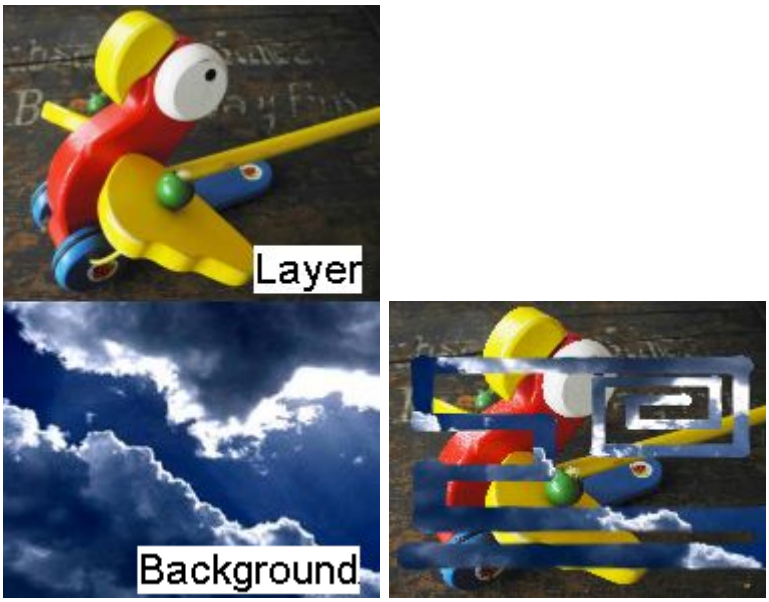
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	32768
Pt1 Y	Vertical position of the first point	0-65535	32768
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	2048
Size	Size	0-65535	1024
Radius	Radius	0-65535	16384
Fill	Fill	0-255	255

Circle2D

Adds a circle to the layer texture. The circle's color, alpha, size and radius value can be adjusted, as well as its position, the edge's softness and fill-behavior.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	32768
Pt1 Y	Vertical position of the first point	0-65535	32768
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	128
Size	Size	0-65535	128
Radius	Radius	0-65535	8192
Fill	Fill	0-255	0

Draw Point Alpha

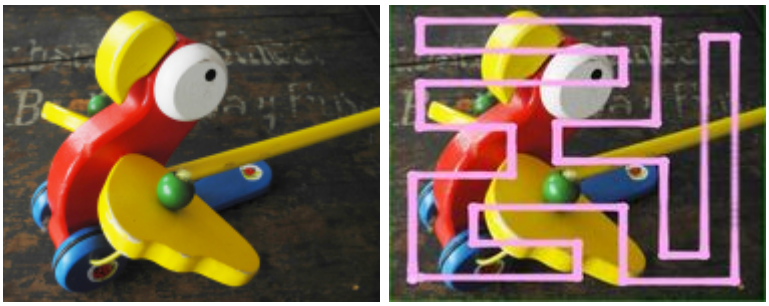


Adds a transparent line to the texture layer and reveals the underlying background. The line can be drawn manually by changing the X- and Y-value of the draw point. Size and pressure of the point can be adjusted. As long as the "Clear" parameter is set to 0, you can draw; switching to 1 clears the drawn line and covers the background again.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	255
Clear	Clear	0-1	1
Size	Size	0-65535	32768
Pressure	Pressure	0-65535	32768
X	X	0-65535	32768
Y	Y	0-65535	32768

Draw Point RGB



Overlays the layer texture with a line that can be drawn manually by changing the X- and Y-value of the draw point. Color, size and pressure of the point can be adjusted. As long as the "Clear" parameter is set to 0, you can draw; switching to 1 clears the drawn line.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	255
Clear	Clear	0-1	1
Size	Size	0-65535	32768
Pressure	Pressure	0-65535	32768
X	X	0-65535	32768
Y	Y	0-65535	32768
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255

Ellipse2D Mask

Adds an elliptic mask to the layer texture. The mask's height an width, color, alpha level and radius value can be adjusted, as well as its position, the edge's softness and fill-behavior.

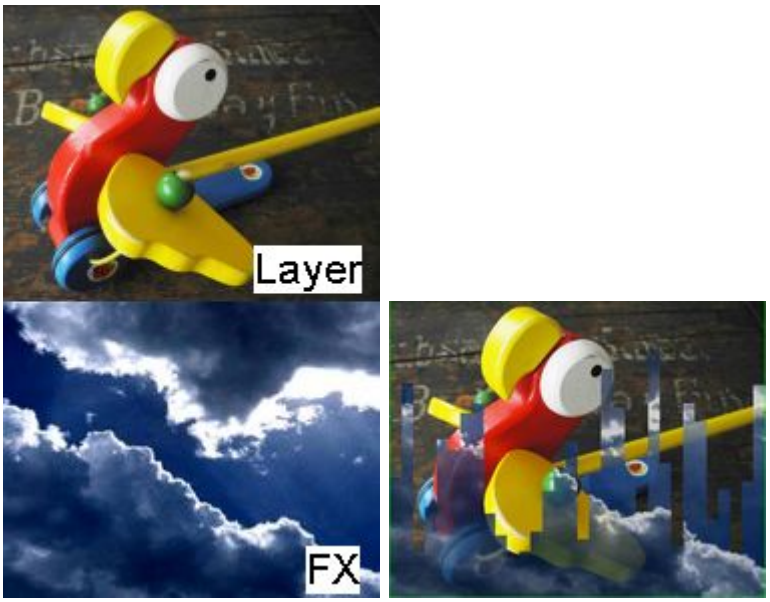
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	32768
Pt1 Y	Vertical position of the first point	0-65535	32768
Width	Width	0-65535	32768
Height	Height	0-65535	32768
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	16384
Size	Size	0-65535	2048
Fill	Fill	0-255	255

Ellipse2D

Adds an elliptic circle to the layer texture. The circle's height an width, color, alpha and radius value can be adjusted, as well as its position, the edge's softness and fill-behavior.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	32768
Pt1 Y	Vertical position of the first point	0-65535	32768
Width	Width	0-65535	32768
Height	Height	0-65535	32768
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	1024
Size	Size	0-65535	2048
Fill	Fill	0-255	0

EQ Media



Adds 32 bars to the texture of the layer, the bars' texture layer is filled with selectable media content. Each bar's level can be adjusted, e.g. by programming values into the sequence or by receiving them from other programs, e.g by the Widget Designer's Audio Processor Input node [1038].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Media	Media file as source for overlay or mask	-	-
EQ1	EQ1	0-65535	32768
EQ2	EQ2	0-65535	32768
EQ3	EQ3	0-65535	32768
EQ4	EQ4	0-65535	32768
EQ5	EQ5	0-65535	32768
EQ6	EQ6	0-65535	32768
EQ7	EQ7	0-65535	32768
EQ8	EQ8	0-65535	32768
EQ9	EQ9	0-65535	32768
EQ10	EQ10	0-65535	32768
EQ11	EQ11	0-65535	32768
EQ12	EQ12	0-65535	32768
EQ13	EQ13	0-65535	32768
EQ14	EQ14	0-65535	32768
EQ15	EQ15	0-65535	32768
EQ16	EQ16	0-65535	32768
EQ17	EQ17	0-65535	32768
EQ18	EQ18	0-65535	32768
EQ19	EQ19	0-65535	32768
EQ20	EQ20	0-65535	32768
EQ21	EQ21	0-65535	32768
EQ22	EQ22	0-65535	32768
EQ23	EQ23	0-65535	32768
EQ24	EQ24	0-65535	32768
EQ25	EQ25	0-65535	32768
EQ26	EQ26	0-65535	32768
EQ27	EQ27	0-65535	32768
EQ28	EQ28	0-65535	32768
EQ29	EQ29	0-65535	32768
EQ30	EQ30	0-65535	32768
EQ31	EQ31	0-65535	32768
EQ32	EQ32	0-65535	32768

EQ Waveform

Turns the texture of the layer into a dynamic shape with hundreds of equalizer bars. The level of each bar is automatically adjusted by the Widget Designer's Audio Processor Input node [1038].
A high "Glow" parameter adds a fading overlay to the bars (2nd img.). The parameter "Alpha" influences the background of the bars and blends between black and transparent, displaying underlying layers. The parameter "Blend" colors the bars: 0 = white, 255 = layer texture (3rd img.).

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Glow	Glow	0-255	255
Alpha	Level of transparency	0-255	0
Blend	Blend	0-255	0

EQ

Adds 32 rainbow colored equalizer bars to the texture of the layer. Each bar's level can be adjusted e.g. by programming values into the sequence or by receiving them from other programs, e.g by the Widget Designer's Audio Processor Input node [1038].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
EQ1	EQ1	0-65535	32768
EQ2	EQ2	0-65535	32768
EQ3	EQ3	0-65535	32768
EQ4	EQ4	0-65535	32768
EQ5	EQ5	0-65535	32768
EQ6	EQ6	0-65535	32768
EQ7	EQ7	0-65535	32768
EQ8	EQ8	0-65535	32768
EQ9	EQ9	0-65535	32768
EQ10	EQ10	0-65535	32768
EQ11	EQ11	0-65535	32768
EQ12	EQ12	0-65535	32768
EQ13	EQ13	0-65535	32768
EQ14	EQ14	0-65535	32768
EQ15	EQ15	0-65535	32768
EQ16	EQ16	0-65535	32768
EQ17	EQ17	0-65535	32768
EQ18	EQ18	0-65535	32768
EQ19	EQ19	0-65535	32768
EQ20	EQ20	0-65535	32768
EQ21	EQ21	0-65535	32768
EQ22	EQ22	0-65535	32768
EQ23	EQ23	0-65535	32768
EQ24	EQ24	0-65535	32768
EQ25	EQ25	0-65535	32768
EQ26	EQ26	0-65535	32768
EQ27	EQ27	0-65535	32768
EQ28	EQ28	0-65535	32768
EQ29	EQ29	0-65535	32768
EQ30	EQ30	0-65535	32768
EQ31	EQ31	0-65535	32768
EQ32	EQ32	0-65535	32768

Line Horizontal

Adds a horizontal line to the layer texture. The line's position, color, width and alpha value can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
V Pos	Vertical Position	0-65535	32768
H Start	Start of Line Horizontal	0-65535	0
H End	End of Line Horizontal	0-65535	65535
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Alpha	Alpha level	0-255	255
Width	Width of line	0-65535	2048

Line Vertical

Adds a vertical line to the layer texture. The line's position, color, width and alpha value can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
V Pos	Vertical Position	0-65535	32768
V Start	Start of Line Vertical	0-65535	0
V End	End of Line Vertical	0-65535	65535
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Alpha	Alpha level	0-255	255
Width	Width of line	0-65535	2048

Line2D Mask

Adds a line shaped mask to the layer texture. The mask's endpoints, color, alpha level and size can be adjusted, as well as the edge's softness.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	250
Size	Size	0-65535	250

Line2D

Adds a line to the layer texture. The line's endpoints, color, alpha level and size can be adjusted, as well as the edge's softness.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	250
Size	Size	0-65535	250

MetaBall2D Add

Adds two glowing balls to the layer texture and dims the rest of it. The position and color of each ball can be adjusted separately, the alpha level, blend, softness and size can be adjusted for both together.

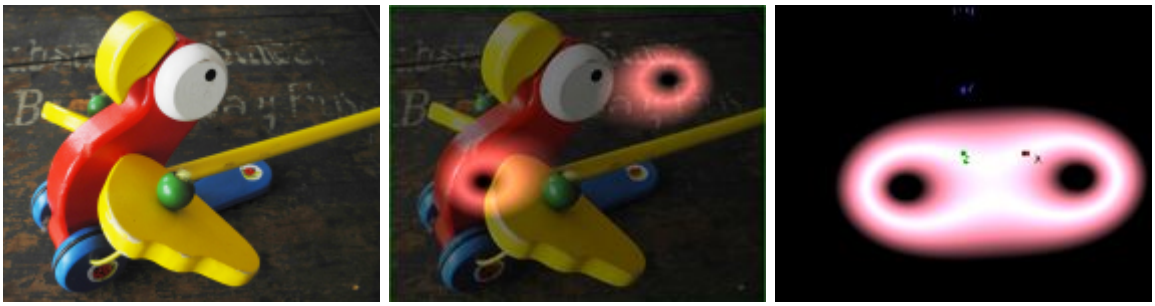
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	25000
Pt1 Y	Vertical position of the first point	0-65535	25000
Pt2 X	Horizontal position of the second point	0-65535	40000
Pt2 Y	Vertical position of the second point	0-65535	40000
Red1	First color picker / Level of red	0-255	255
Green1	First color picker / Level of green	0-255	100
Blue1	First color picker / Level of blue	0-255	0
Red2	Second color picker / Level of red	0-255	0
Green2	Second color picker / Level of green	0-255	100
Blue2	Second color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Blend	Blend	0-255	0
Softness	Softness	0-65535	16384
Size	Size	0-65535	16384

MetaBall2D Multiply

Adds two glowing balls to the layer texture whose halos are multiplied and dims the rest of it. The position and color of each ball can be adjusted separately, the alpha level, blend, softness and weight can be adjusted for both together.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	25000
Pt1 Y	Vertical position of the first point	0-65535	25000
Pt2 X	Horizontal position of the second point	0-65535	40000
Pt2 Y	Vertical position of the second point	0-65535	40000
Red1	First color picker / Level of red	0-255	255
Green1	First color picker / Level of green	0-255	128
Blue1	First color picker / Level of blue	0-255	0
Red2	Second color picker / Level of red	0-255	0
Green2	Second color picker / Level of green	0-255	128
Blue2	Second color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Blend	Blend	0-255	0
Softness1	Softness1	0-65535	16384
Weight	Weight	0-65535	32768

MetaBall2D Outline Glow



Two black balls with glowing outlines are added to the layer texture and the rest of it is dimmed. The position of each ball can be adjusted separately, the alpha level, blend, softness and size as well as color and radius can be adjusted for both together.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	25000
Pt1 Y	Vertical position of the first point	0-65535	25000
Pt2 X	Horizontal position of the second point	0-65535	40000
Pt2 Y	Vertical position of the second point	0-65535	40000
Red	Color picker / Level of red	0-255	128
Green	Color picker / Level of green	0-255	196
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Blend	Blend	0-255	0
Glow	Glow	0-65535	32768
Radius	Radius	0-65535	8192
Size	Size	0-65535	62000

Quad 16bit

Adds a quadrangle (four-sided figure) shape to the layer texture. The corner positions and the inside color and alpha level can be adjusted, as well as the softness of each edge.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	65535
Softness 1	Softness 1	0-65535	250
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Softness 2	Softness 2	0-65535	250
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Softness 3	Softness 3	0-65535	250
Pt4 X	Horizontal position of the fourth point	0-65535	0
Pt4 Y	Vertical position of the fourth point	0-65535	0
Softness 4	Softness 4	0-65535	250
Red	Color picker / Level of red	0-65535	65535
Green	Color picker / Level of green	0-65535	65535
Blue	Color picker / Level of blue	0-65535	65535
Alpha	Level of transparency	0-65535	65535

Quad Black Lift 16bit

Adds a quadrangle (four-sided figure) shape to the layer texture. The corner positions and the inside color can be adjusted, as well as the softness of each edge.
The shape color is added to the layer texture but only to dark values. The higher the "Threshold" parameter the lighter the colors get that are influenced by the shape.

This effect can be used for blacklevel adjustment [646], when it is assigned to the output or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	65535
Softness 1	Softness 1	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Softness 2	Softness 2	0-65535	0
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Softness 3	Softness 3	0-65535	0
Pt4 X	Horizontal position of the fourth point	0-65535	0
Pt4 Y	Vertical position of the fourth point	0-65535	0
Softness 4	Softness 4	0-65535	0
Red	Color picker / Level of red	0-65535	10000
Green	Color picker / Level of green	0-65535	10000
Blue	Color picker / Level of blue	0-65535	10000
Threshold	Threshold	0-65535	30000

Quad2D Mask


Adds a quadrangle (four-sided figure) shaped mask to the layer texture. The corner positions, mask color and alpha level can be adjusted, as well as the softness, size and fill-behavior of the mask.

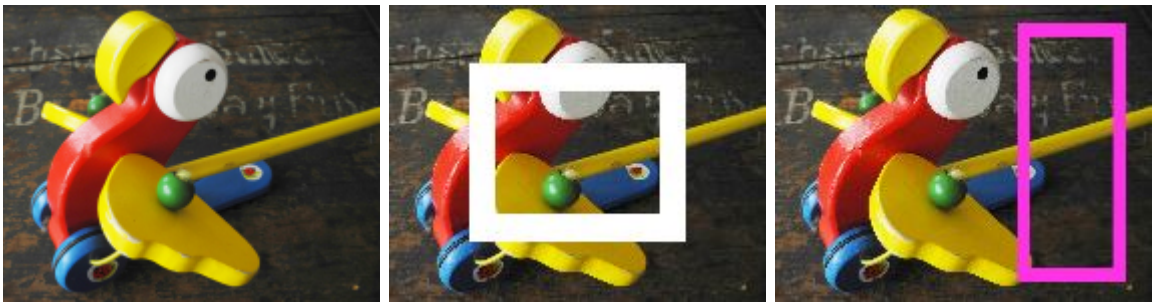
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	4096
Pt1 Y	Vertical position of the first point	0-65535	61439
Pt2 X	Horizontal position of the second point	0-65535	61439
Pt2 Y	Vertical position of the second point	0-65535	61439
Pt3 X	Horizontal position of the third point	0-65535	61439
Pt3 Y	Vertical position of the third point	0-65535	4096
Pt4 X	Horizontal position of the fourth point	0-65535	4096
Pt4 Y	Vertical position of the fourth point	0-65535	4096
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	1024
Size	Size	0-65535	0
Fill	Fill	0-255	255

Quad2D

Adds a quadrangle (four-sided figure) shape to the layer texture. The corner positions, mask color and alpha level can be adjusted, as well as the softness, size and fill-behavior of the shape.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	65535
Pt2 X	Horizontal position of the second point	0-65535	65535
Pt2 Y	Vertical position of the second point	0-65535	65535
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Pt4 X	Horizontal position of the fourth point	0-65535	0
Pt4 Y	Vertical position of the fourth point	0-65535	0
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	250
Size	Size	0-65535	250
Fill	Fill	0-255	255

Rectangle



Adds an outline of a rectangle to the layer texture. The rectangle's position, color, width and alpha value can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Pos U	Horizontal Position	0-65535	32768
Pos V	Vertical Position	0-65535	32768
Width	Width of the rectangle	0-65535	32768
Height	Height of the rectangle	0-65535	32768
Red	Red level, Color Picker	0-255	255
Green	Green level, Color Picker	0-255	255
Blue	Blue level, Color Picker	0-255	255
Alpha	Alpha level	0-255	255
Size	Thickness of the rectangle's lines	0-65535	1024

Rectangle2D Mask

Adds a rectangular mask to the layer of the texture. Its size and position as well as color, alpha and fill level, edge softness, and the angle of the mask can be adjusted.

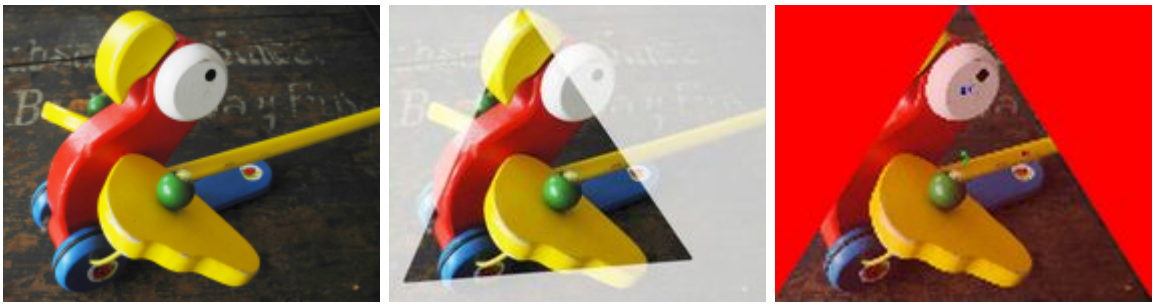
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pos U	Horizontal position	0-65535	32768
Pos V	Vertical position	0-65535	32768
Width	Width	0-65535	32768
Height	Height	0-65535	32768
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Size	Size	0-65535	1024
Softness	Softness	0-65535	1024
Fill	Fill	0-255	255
Angle	Angle	0-360	180

Rectangle2D

Adds a rectangle to the layer of the texture. Its size and position as well as color, alpha and fill level, edge softness, and the angle of the shape can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pos U	Horizontal position	0-65535	32768
Pos V	Vertical position	0-65535	32768
Width	Width	0-65535	32768
Height	Height	0-65535	32768
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Size	Size	0-65535	1024
Softness	Softness	0-65535	1024
Fill	Fill	0-255	255
Angle	Angle	0-360	180

Triangle2D Mask



Adds a triangular mask to the layer of the texture. Its size and position as well as color, alpha and fill level, edge softness, and the angle of the mask can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	32768
Pt2 Y	Vertical position of the second point	0-65535	65535
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	250
Size	Size	0-65535	250
Fill	Fill	0-255	255
Angle	Angle	0-360	180

Triangle2D

Adds a triangle to the layer of the texture. Its size and position as well as color, alpha and fill level, edge softness, and the angle of the shape can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Pt1 X	Horizontal position of the first point	0-65535	0
Pt1 Y	Vertical position of the first point	0-65535	0
Pt2 X	Horizontal position of the second point	0-65535	32768
Pt2 Y	Vertical position of the second point	0-65535	65535
Pt3 X	Horizontal position of the third point	0-65535	65535
Pt3 Y	Vertical position of the third point	0-65535	0
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Softness	Softness	0-65535	250
Size	Size	0-65535	250
Fill	Fill	0-255	255
Angle	Angle	0-360	180

6.5.2.4.33 Sharpen

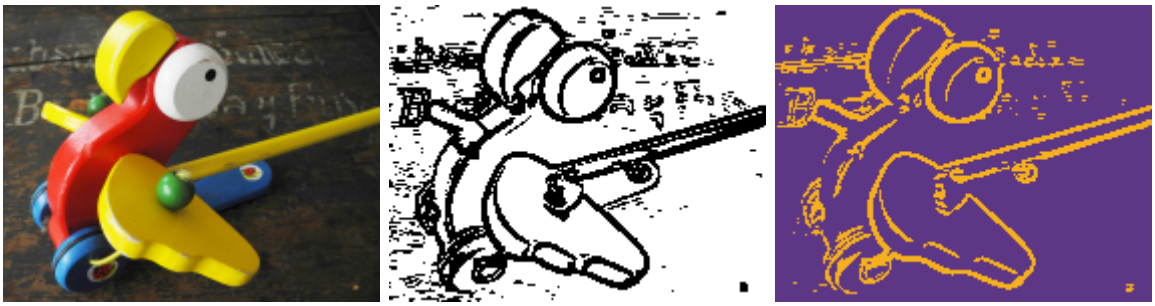
Edges

Sharpens the layer texture on all edges. The effect generates three additional textures with black and white edges. These textures then overlay the real layer. With the parameters "Width" and "Height" you can influence the horizontal and vertical offset of these textures / edges.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Width	Amount of sharpening	0-255	128
Height	Offset factor	0-255	128

Outlines



Creates area outlines of the layer texture by separating the texture into two different groups. This depends on the contrast level of adjacent pixels, which can be adjusted with the parameter "threshold".
You can setup a color and alpha level for both areas.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

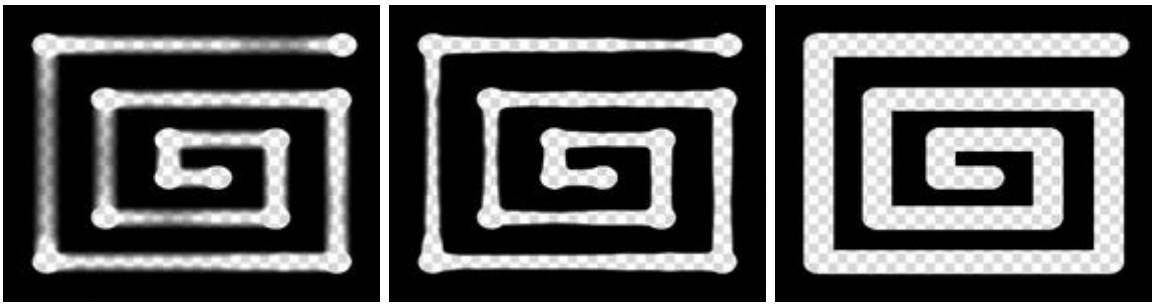
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Red H	Red level of High Color, Color Picker	0-255	255
Green H	Green level, Color Picker	0-255	255
Blue H	Blue level of High Color, Color Picker	0-255	255
Alpha H	Alpha level of High Color	0-255	255
Red L	Red level of Low Color, Color Picker	0-255	0
Green L	Green level of Low Color, Color Picker	0-255	0
Blue L	Blue level of Low Color, Color Picker	0-255	0
Alpha L	Alpha level of Low Color	0-255	255
Threshold	Threshold	0-255	32

Sharpen Alpha Edges (Fill Color)

Sharpens the edges of the alpha (transparent) channel depending on the chosen threshold and smoothing value. In addition, all non-transparent pixels, can be colored in.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Threshold	Threshold	0-255	128
Smoothing	Smoothing	0-255	128
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255

Sharpen Alpha Edges



Sharpens the edges of the alpha (transparent) channel depending on the chosen threshold and smoothing value.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Threshold	Threshold	0-255	128
Smoothing	Smoothing	0-255	128

Sharpen

Sharpens the layer texture on all edges. You can influence the amount of sharpening as well as the offset factor for an internal sharpening texture.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine 162". The chapter "FX Order 325" explains how to drag effects in the Device Tree 173 tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amount	Amount of sharpening	0-255	255
Factor	Offset factor	0-255	64

6.5.2.4.34 Snapshot

Render Feedback

The texture is replaced by the Renderhistory (see description in Composition FX) [406] creating a video feedback. This is similar to the situation when a mirror mirrors another mirror and the picture feedbacks itself again and again.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Size	Size	0-65535	32768
Off U	Texture Offset in X	0-65535	32768
Off V	Texture Offset in Y	0-65535	32768
Add Factor	Add Factor	0-65535	32768
Del Factor	Del Factor	0-65535	0

Render Freeze

Allows freezing the Renderhistory, (see description in Composition FX) [406].

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Fade	Fade	0-255	255
Freeze	Freeze	0-255	30

Video Freeze

Shows a freeze frame of the video playing on the layer as soon as "Freeze" is set to a value >0. Meanwhile the playback is going on, so this is different from having the video paused.

Technical Information
This effect can only be used with images or videos (i.e. textures) that are smaller than the "Max. Internal Texture Size" which depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]".

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Freeze	Freezes the video	0-255	0

6.5.2.4.35 Softedge

Alpha Softedge

Allows setting a softedge fading into alpha for each edge: left, right, top and bottom. The amount and alpha curve of the softedge can be adjusted, markers for each softedge can help doing the setup.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Left	Amount of Softedge left	0-65535	0
Left Alpha	Alpha curve	0-65535	32768
Left Marker	Marker for left softedge:	0-255	0
-	- no marker	0	-
-	- black marker	1	-
-	- white marker	255	-
L Width	Width of left marker	0-255	32

The parameters Left, Left Alpha, Left Marker and L Width and its values will be repeated for each corresponding edge: Right, Top and Bottom.

RGB Softedge

Allows setting a softedge fading into any RGB color for each edge: left, right, top and bottom. The amount and RGB curve of the softedge can be adjusted, markers for each softedge may help doing the setup.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Left	Amount of Softedge left	0-65535	0
Left Red	Red curve	0-65535	32768
Left Green	Green curve	0-65535	32768
Left Blue	Blue curve	0-65535	32768
Left Marker	Marker for left softedge:	0-255	0
-	- no marker	0	-
-	- green marker	255	-
L Width	Width of left marker	0-255	32

The parameters Left, Left Red/Green/Blue, Left Marker and L Width and its values will be repeated for each corresponding edge: Right, Top and Bottom.

Softedge Bottom Alpha

Allows setting a softedge fading into alpha for the bottom edge. The amount and alpha curve of the softedge can be adjusted, a marker for the softedge may help doing the setup.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amt	Amount of Softedge	0-65535	0
Curve	Red curve	0-65535	32768
Marker	Marker for softedge:	0-255	0
-	- no marker	0	-
-	- green marker	255	-
Width	Width of marker	0-255	32

Softedge Bottom RGB

Allows setting a softedge fading into any RBG for the bottom edge. The amount and RGB curve of the softedge can be adjusted, a marker for the softedge may help doing the setup.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Amt	Amount of Softedge	0-65535	0
Curve	Red curve	0-65535	32768
Marker	Marker for softedge:	0-255	0
-	- no marker	0	-
-	- green marker	255	-
Width	Width of marker	0-255	32

Softedge Left Alpha

see FX Softedge Bottom Alpha 612

Softedge Left RGB

see FX Softedge Bottom RGB 612

Softedge Right Alpha

see FX Softedge Bottom Alpha [612]

Softedge Right RGB

see FX Softedge Bottom RGB [612]

Softedge Top Alpha

see FX Softedge Bottom Alpha [612]

Softedge Top RGB

see FX Softedge Bottom RGB [612]

6.5.2.4.36 Softedge - Softedge P-Curve Alpha

Alpha Softedge P-Curve

Allows setting a p-curve softedge fading into alpha for each edge: left, right, top and bottom. Position, curve and offset of each softedge can be adjusted, markers can help doing the setup.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Amount of left softedge	0-65535	0
L Curve	Left alpha curve	0-65535	32768
L Offset	Left offset	0-65535	32768
L Marker	Left marker	0-255	0
L Width	Width of left marker	0-255	32

These parameters and its values are repeated for each corresponding edge: Right, Top and Bottom.

Softedge P-Curve Bottom Alpha

Allows setting a p-curve softedge fading into alpha for only one edge: bottom.
Amount, curve and offset of the edge can be adjusted, a marker can help doing the setup.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount of softedge	0-65535	0
Curve	Curve	0-65535	32768
Offset	Offset	0-65535	32768
Marker	Marker	0-255	0
Width	Marker width	0-255	32

Softedge P-Curve Left Alpha

see FX Softedge P-Curve Bottom Alpha 614

Softedge P-Curve Right Alpha

see FX Softedge P-Curve Bottom Alpha 614

Softedge P-Curve Top Alpha

see FX Softedge P-Curve Bottom Alpha 614

6.5.2.4.37 Softedge - Softedge P-Curve Alpha Gamma

Alpha Softedge P-Curve Gamma

Allows setting a p-curve softedge fading into alpha for each edge: left, right, top and bottom. Position, curve and offset of each softedge can be adjusted, markers can help doing the setup. A color correction can be added via an RGB-gamma.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Amount of left softedge	0-65535	0
L Curve	Left alpha curve	0-65535	32768
L Offset	Left offset	0-65535	32768
L Marker	Left marker	0-255	0
L Width	Width of left marker	0-255	32
These parameters and its values are repeated for each corresponding edge: Right, Top and Bottom.			
Gamma R	Gamma Red	0-65535	16384
Gamma G	Gamma Green	0-65535	16384
Gamma B	Gamma Blue	0-65535	16384

Softedge P-Curve Gamma Bottom Alpha

Allows setting a p-curve softedge fading into alpha for only one edge: Bottom.
Position, curve and offset of each softedge can be adjusted, markers can help doing the setup. A color correction can be added via an RGB-gamma.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Amount	Amount of softedge	0-65535	0
Curve	Softedge curve	0-65535	16384
Offset	Offset	0-65535	32768
Marker	Marker	0-255	0
Width	Marker width	0-255	32
Gamma R	Gamma Red	0-65535	16384
Gamma G	Gamma Green	0-65535	16384
Gamma B	Gamma Blue	0-65535	16384

Softedge P-Curve Gamma Left Alpha

see FX Softedge P-Curve Gamma Bottom Alpha 617

Softedge P-Curve Gamma Right Alpha

see FX Softedge P-Curve Gamma Bottom Alpha 617

Softedge P-Curve Gamma Top Alpha

see FX Softedge P-Curve Gamma Bottom Alpha 617

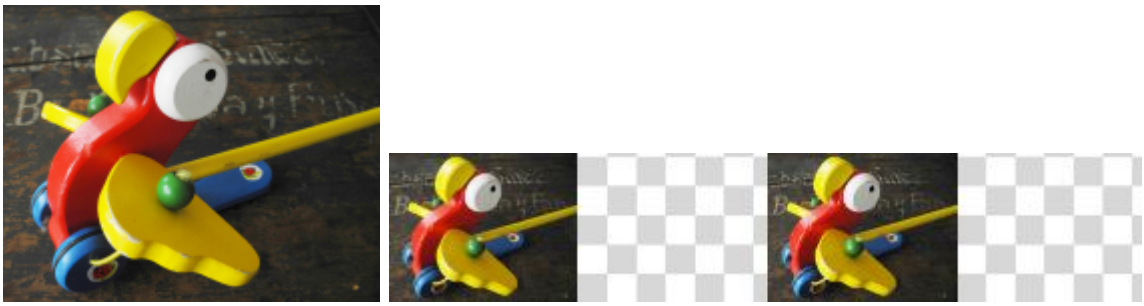
6.5.2.4.38 Strobe

Alpha Flash

Flashes a transparent alpha layer into the selected texture layer, the speed can be adjusted. The cross-fade between both texture is regular and the in-fade of the transparent texture has the same duration as the out-fade.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Flash Speed	Flash Speed	0-65535	32768

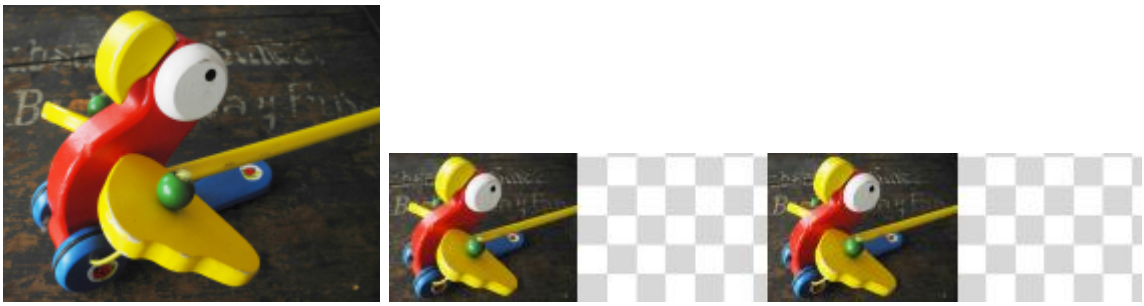
Alpha Pulse



Pulses a transparent alpha layer into the selected texture layer, the speed can be adjusted. The cross-fade between both texture is irregular as the in-fade of the transparent texture is sudden whilst the out-fade is smooth.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Flash Speed	Flash Speed	0-65535	32768

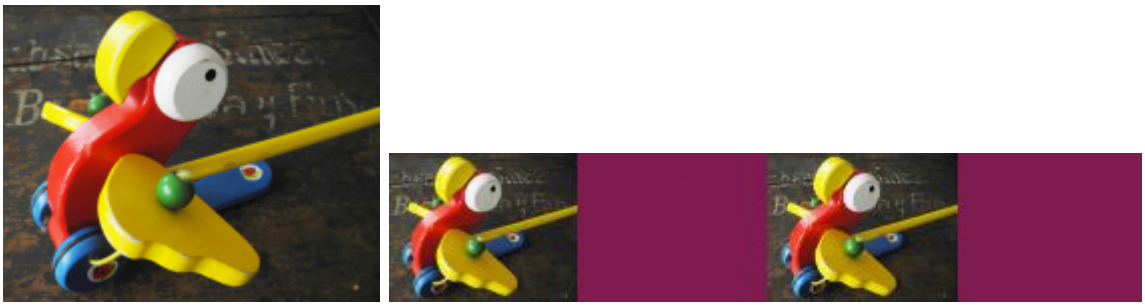
RND Alpha Flash



Flashes a transparent alpha layer into the selected texture layer, the speed can be adjusted. The cross-fade between both texture is absolutely irregular as the in-and out-fades have random durations.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Flash Speed	Flash Speed	0-65535	32768

Sync Color Strobe



Flashes a colored layer into the selected texture layer, alpha value, speed and color can be adjusted. There is no cross-fade between both textures.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red	Color picker / Level of red	0-255	255
Green	Color picker / Level of green	0-255	255
Blue	Color picker / Level of blue	0-255	255
Alpha	Level of transparency	0-255	255
Flash Speed	Flash Speed	32768-65535	32768

6.5.2.4.39 Stylize

ASCII Replace

Replaces the texture of the layer with ASCII symbols, size and color intensity of the symbols can be adjusted, as well as the type of symbols.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

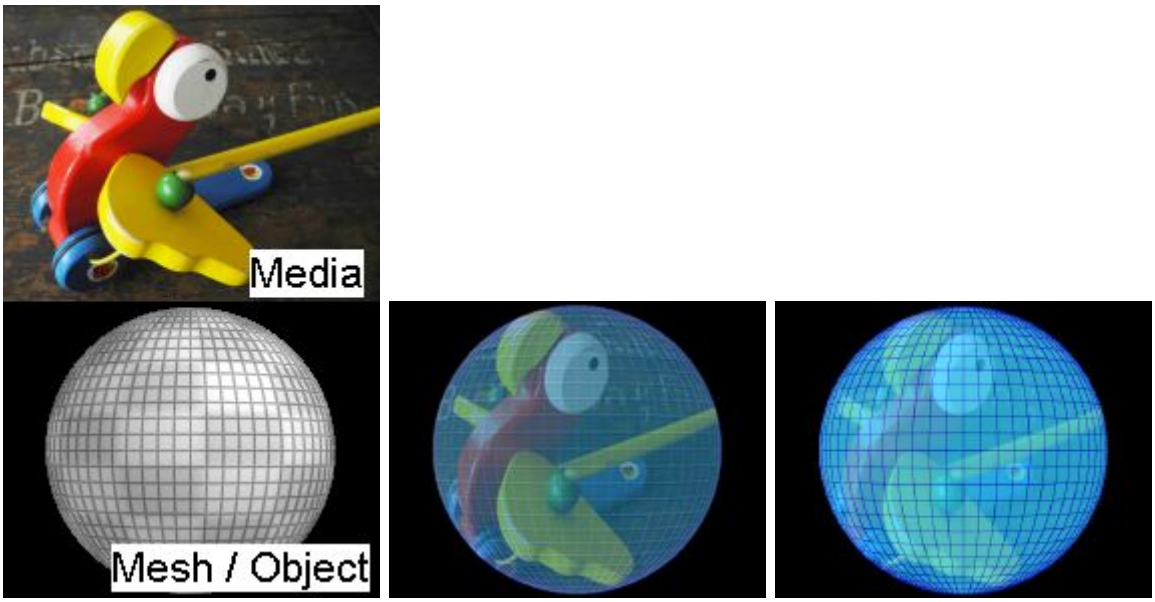
Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Tiling	Tiling	0-65535	8192
Col	Col	0-65535	0
ASCII Map	ASCII Map	-	-

Film Look

Renders the layer texture in an old-fashioned film look, different values like sepia, grain and noise can be adjusted separately.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Sepia	Sepia	0-255	225
Noise	Noise	0-255	220
Noise Size	Noise Size	0-255	12
Grain	Grain	0-255	93
Flicker	Flicker	0-255	220
Mask	Mask	0-255	72
Scroll	Scroll	0-65535	7900

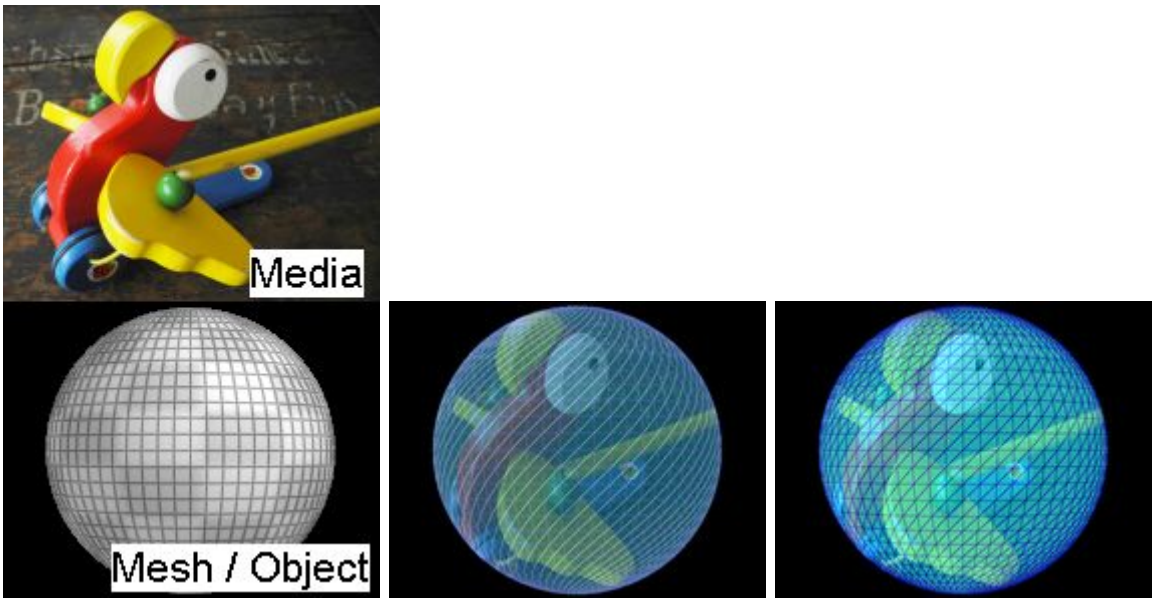
Wireframe Edges



This effect works only with an assigned 3D object. It overlays the layer texture with two adjustable colors. First, the wireframe edges [146] of the object are made visible with a color and alpha level of your choice. Second, a fill color and transparency can be assigned. The width of the wireframes can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Edge	Red Edge	0-255	255
Green Edge	Green Edge	0-255	255
Blue Edge	Blue Edge	0-255	255
Alpha Edge	Alpha Edge	0-255	255
Red Fill	Red Fill	0-255	0
Green Fill	Green Fill	0-255	0
Blue Fill	Blue Fill	0-255	0
Alpha Fill	Alpha Fill	0-255	255
Width	Width	0-65535	13500

Wireframe

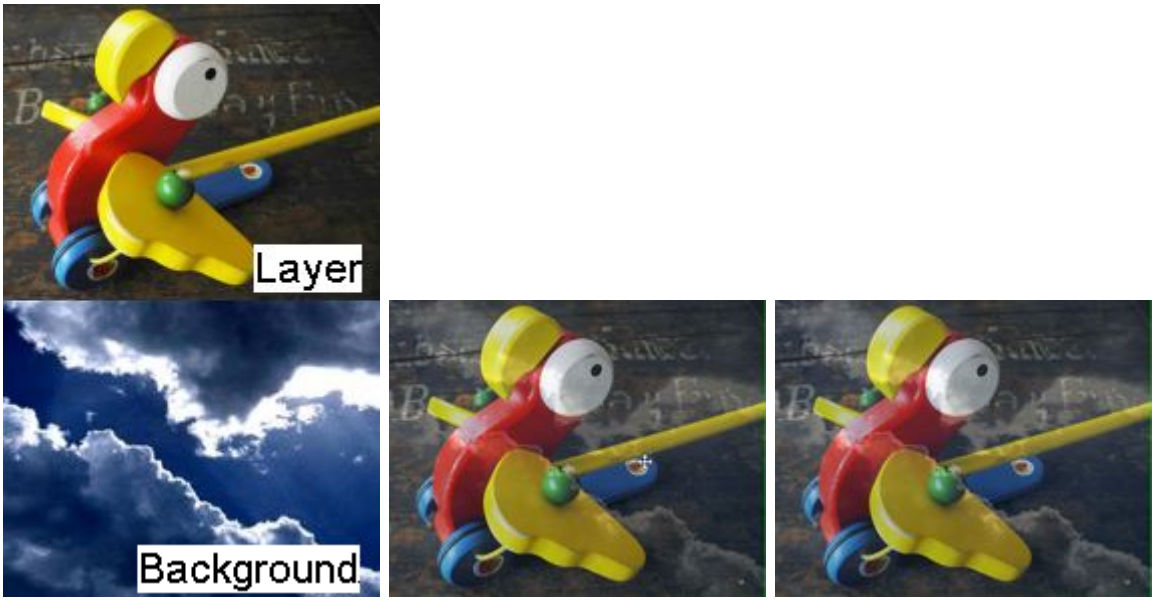


This effect works only with an assigned 3D object. It overlays the layer texture with two adjustable colors. First, the polygon wireframes of the object are made visible with a color and alpha level of your choice. Second, a fill color and transparency can be assigned. The width of the wireframes can be adjusted.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Red Edge	Red Edge	0-255	255
Green Edge	Green Edge	0-255	255
Blue Edge	Blue Edge	0-255	255
Alpha Edge	Alpha Edge	0-255	255
Red Fill	Red Fill	0-255	0
Green Fill	Green Fill	0-255	0
Blue Fill	Blue Fill	0-255	0
Alpha Fill	Alpha Fill	0-255	255
Width	Width	0-65535	13500

6.5.2.4.40 Transition

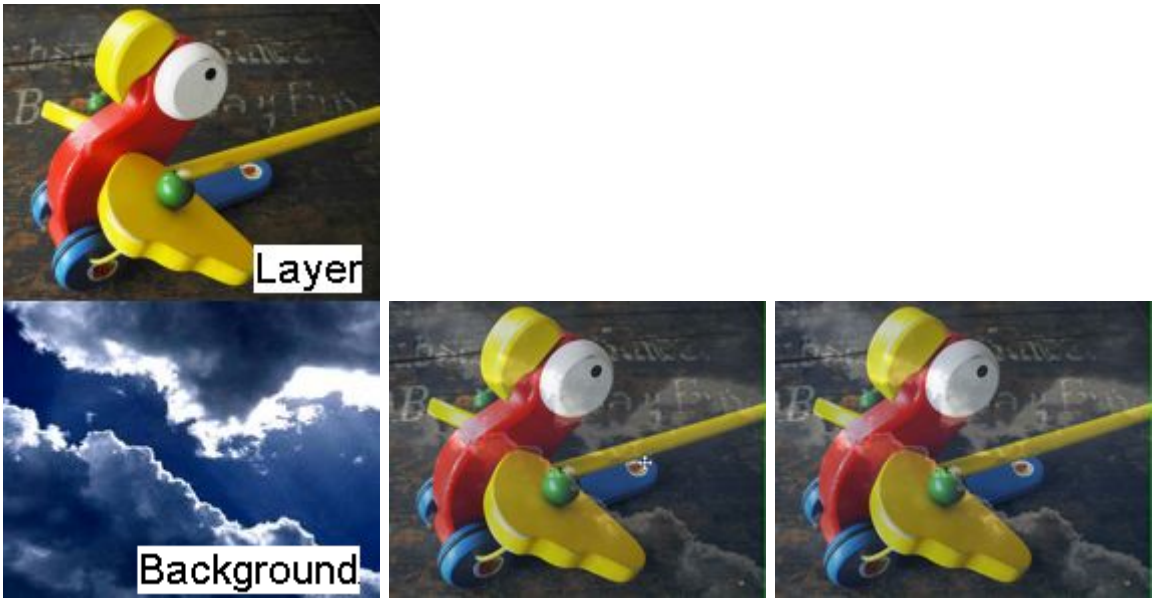
Alpha 16bit



Fades the opacity of the texture layer in 16-bit increments by multiplying the alpha value with a factor. This means that the color values are preserved.
Mix 1 = completely transparent, 65535 = completely visible.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-65535	0

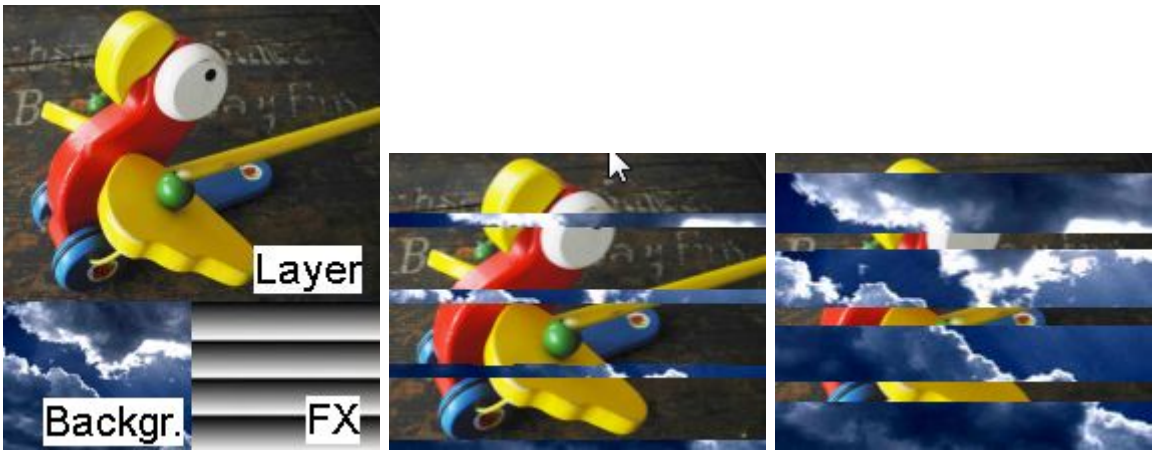
Dissolve 16bit



Dissolves the texture layer in 16-bit increments by multiplying the alpha as well as the RGB values with a factor.
Mix 1 = completely transparent, 65535 = completely visible.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-65535	0

Transition Crop Media Mask



Per default, when a layer fades out (i.e. the opacity value decreases from 255 to 0) the opacity value for each individual pixel is the same. With the TransitionFX you can set up a so-called "Opacity Map". Apply one of your own images to the effect.
The layer now fades according to the chosen map, but in contrary to other Transition FX it (dis-)appears in steps, not gradually. Dark areas in the map fade out first, white ones at last. The contrary applies for a fade in.

The TransitionFX depends only on the "XFade" parameter, not on the "Opacity" of the layer!

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
XFade	XFade	0-255	0
Media	Media file as source for overlay or mask	-	-

Transition Media Mask

Per default, when a layer fades out (i.e. the opacity value decreases from 255 to 0) the opacity value for each individual pixel is the same. With the TransitionFX you can set up a so-called "Opacity Map". Apply one of your own images to the effect.
The layer now fades according to the chosen map. Dark areas in the map fade out first, white ones at last. The contrary applies for a fade in.

The TransitionFX depends on the "Opacity" of the layer!

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Media	Media to be used for transition		

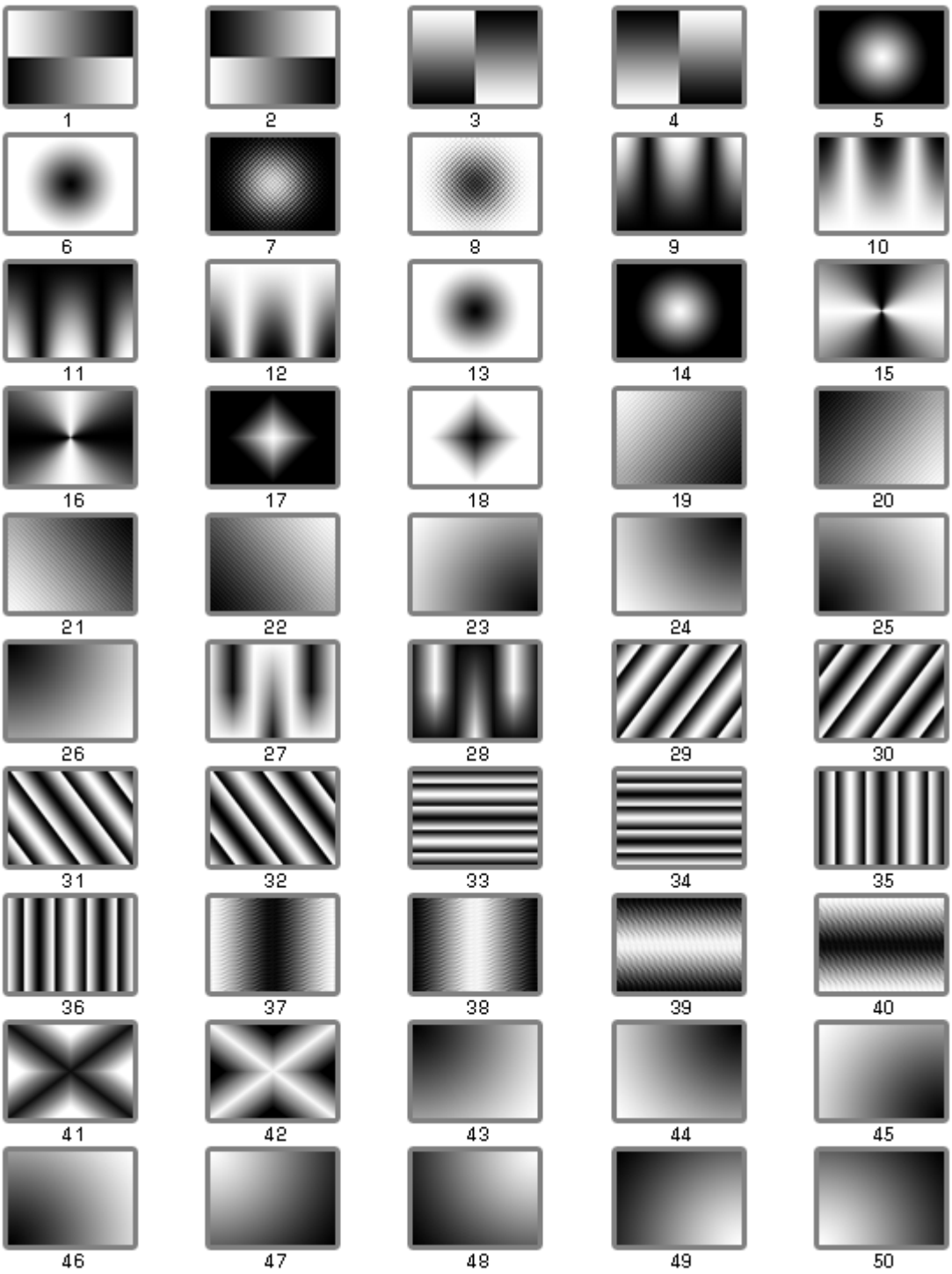
TransitionFX

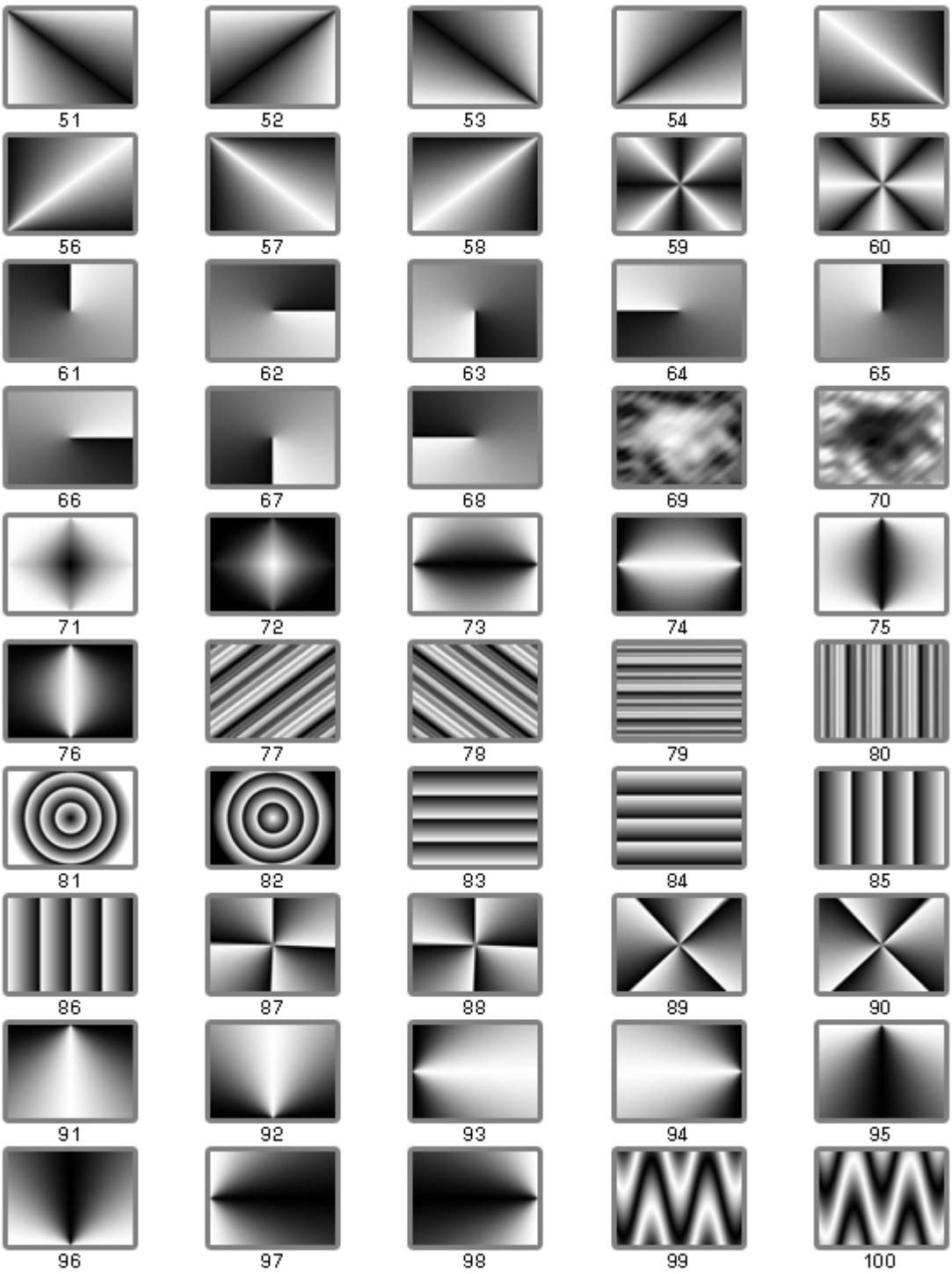
Per default, when a layer fades out (i.e. the opacity value decreases from 255 to 0) the opacity value for each individual pixel is the same. With the TransitionFX you can set up a so-called "Opacity Map". Choose one of the hundreds Pandoras Box' predefined wipes and transitions.
The layer now fades according to the chosen map. Dark areas in the map fade out first, white ones at last. The contrary applies for a fade in.

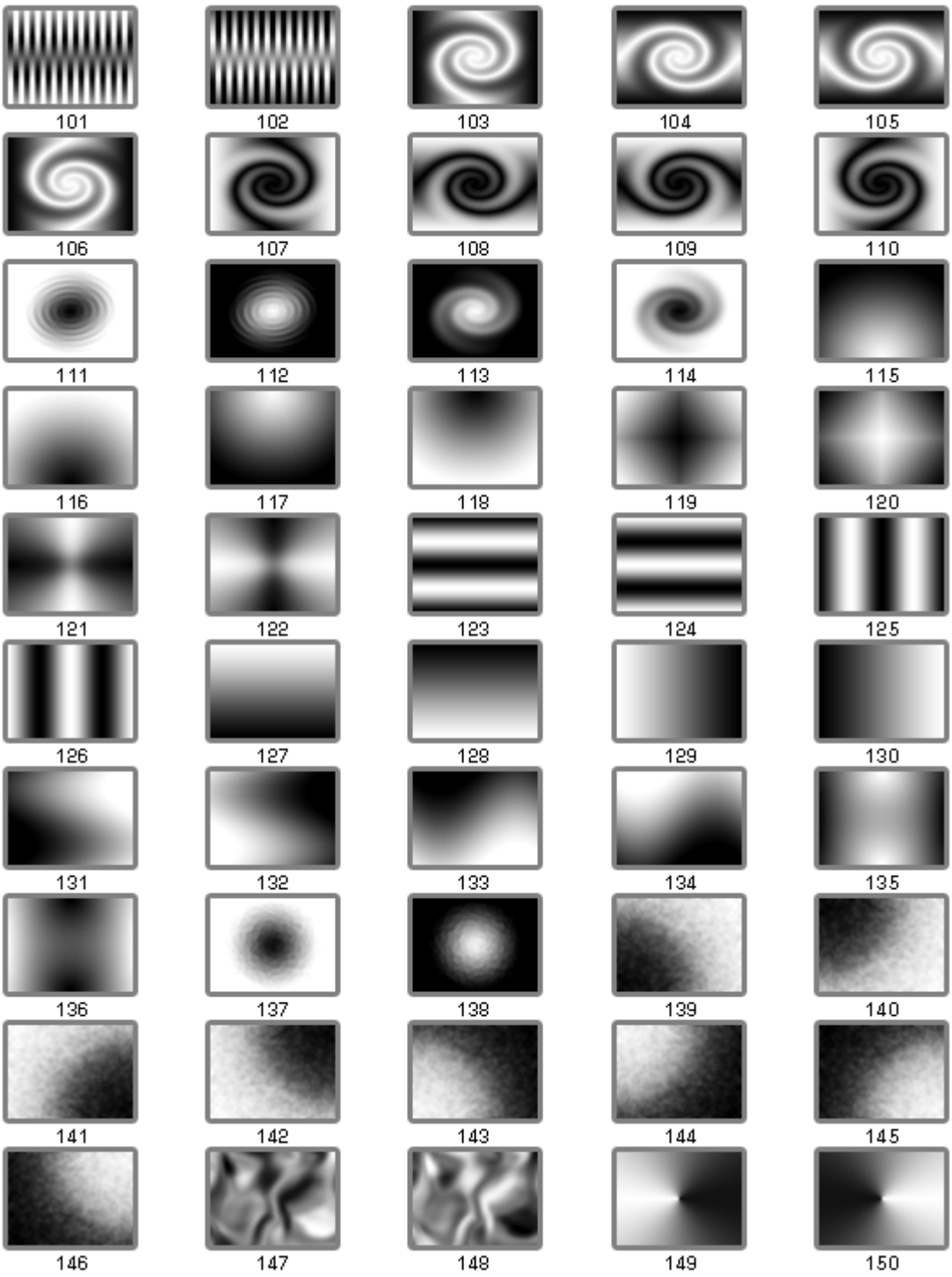
The TransitionFX depends on the "Opacity" of the layer!

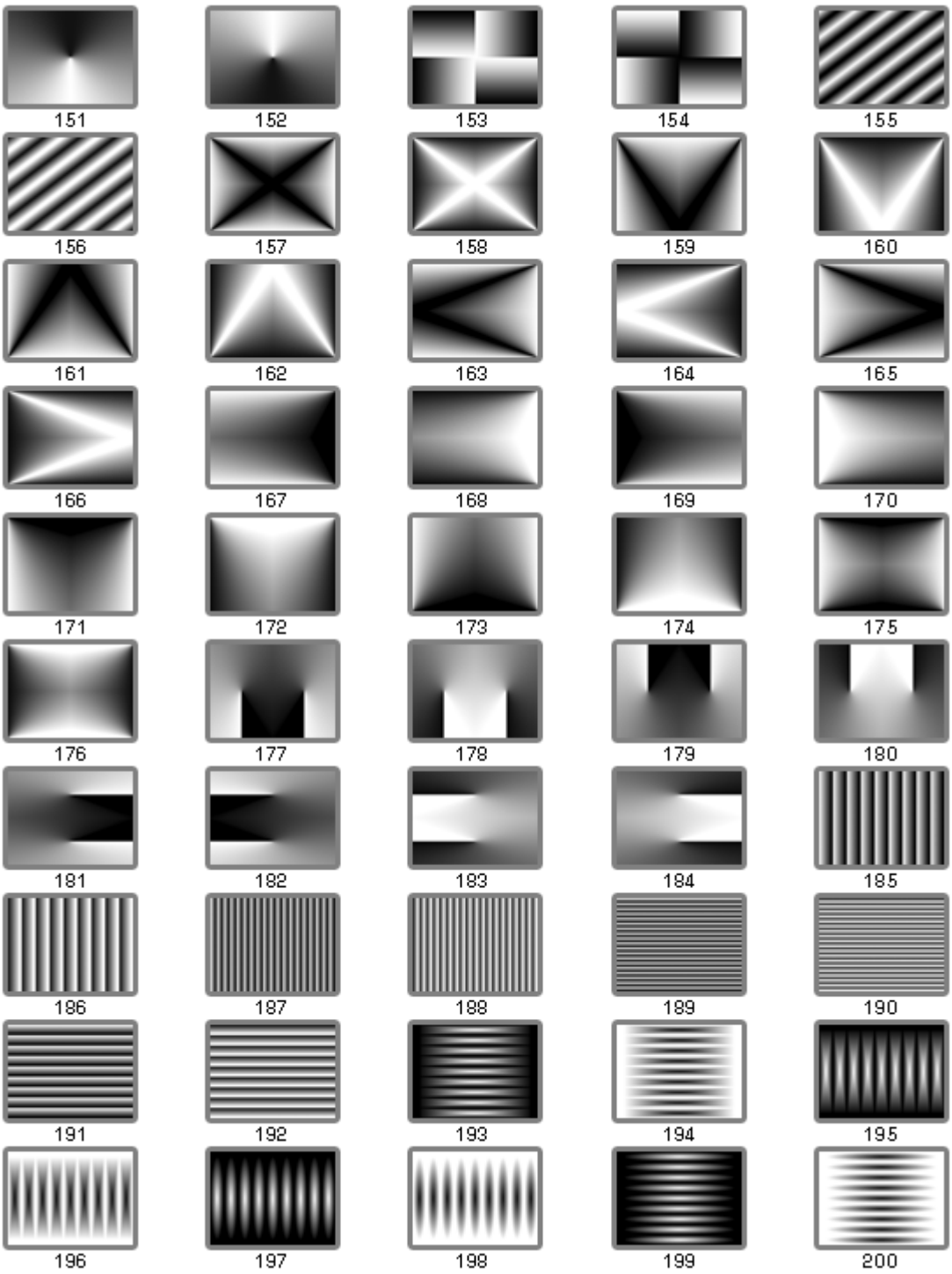
Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Opacity Map	Choose one of the Transition FX		

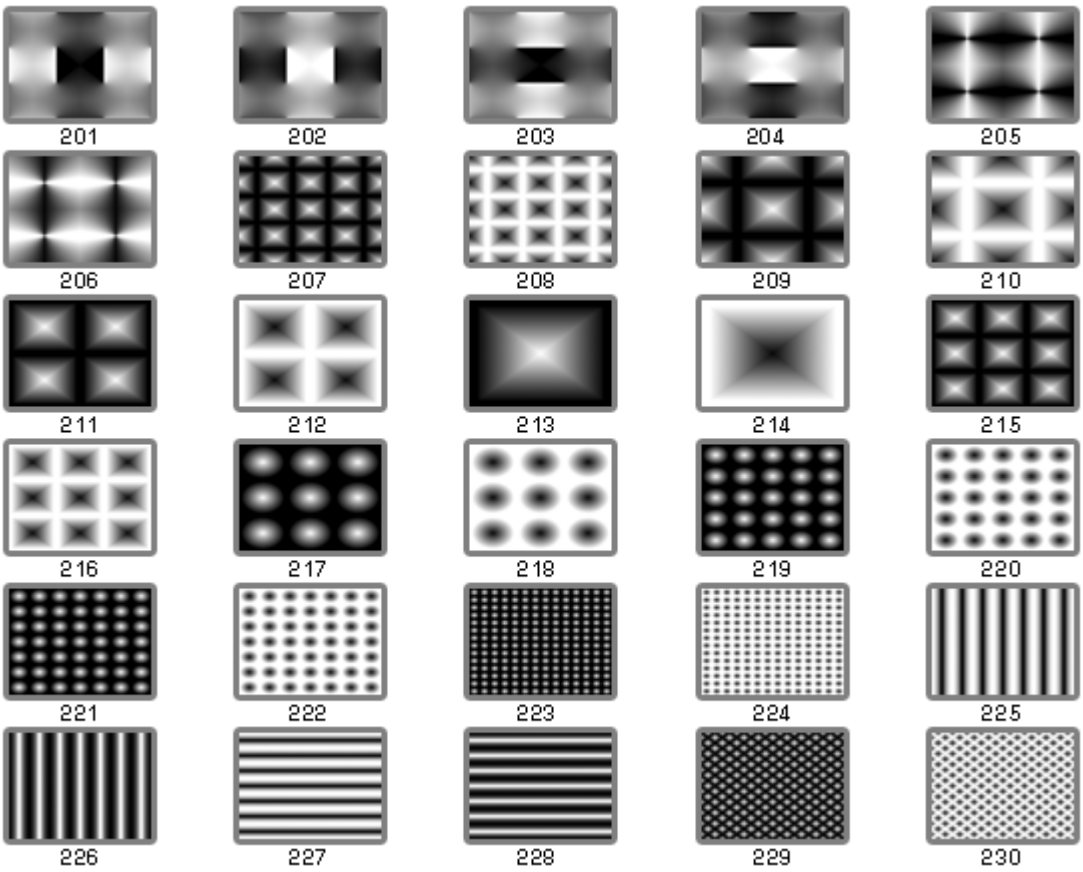
All available TransitionFX are listed below:


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50


51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100


101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150


151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200


201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230

AutoAspect Scroll Horizontal (infinite)

Scrolls the media from right to left or left to right by repeating it endlessly. The scrolling speed and horizontal aspect ratio can be adjusted. If the "Width" parameter is set to the texture's width, no scaling occurs.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Speed X	Horizontal speed	0-65535	32768
Width (px)	Width (px)	0-65535	1024

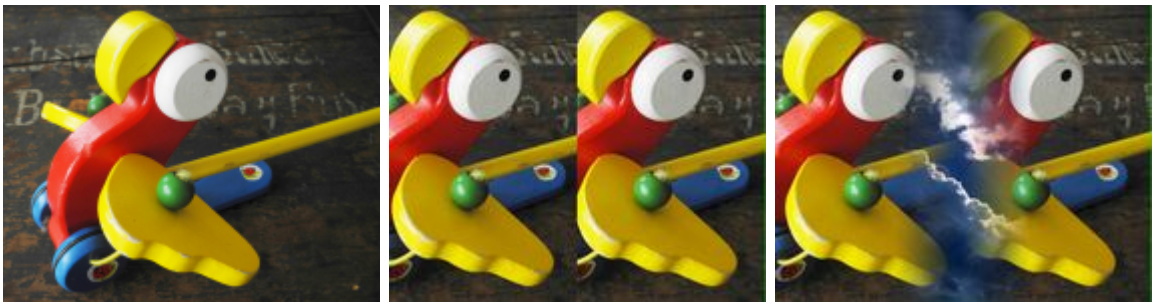
AutoAspect Scroll Horizontal (steady speed)

Please use the effect "AutoAspect Scroll Horizontal (infinite)" [631] instead. The different calculation of the scrolling allows infinite usage.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Speed X	Horizontal speed	0-65535	32768
Width (px)	Width (px)	0-65535	1024

Dual Head Split H



Splits the layer texture horizontally into two areas. The layer texture is repeated, whilst "Factor" defines the offset. The separation can be softedged manually.

This effect allows splitting the Pandoras Box output when for example using (Matrox) Dual Head Devices. Assign it to the output itself or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Factor	Factor	0-65535	0
Softedge	Softedge	0-65535	0
Curve R	Curve R	0-65535	8000
Curve L	Curve L	0-65535	8000
Center	Center	0-65535	27000

Dual Head Split V

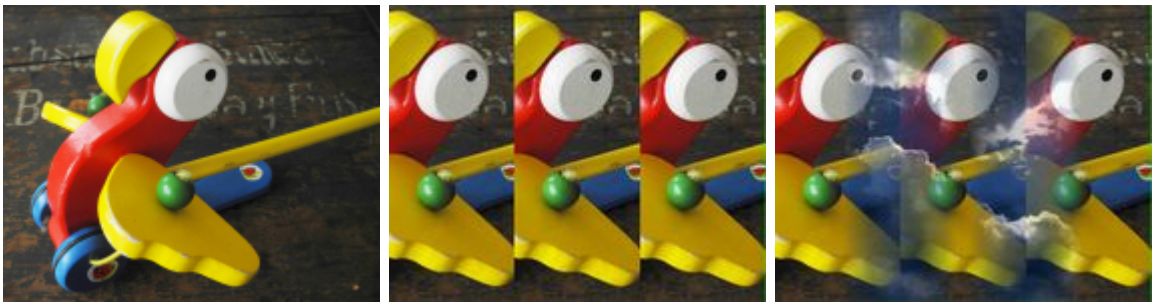
Splits the layer texture vertically into two areas. The layer texture is repeated, whilst "Factor" defines the offset. The separation can be softedged manually.

This effect allows splitting the Pandoras Box output when for example using (Matrox) Dual Head Devices. Assign it to the output itself or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Factor	Factor	0-65535	0
Softedge	Softedge	0-65535	0
Curve Top	Curve Top	0-65535	8000
Curve Bottom	Curve Bottom	0-65535	8000
Center	Center	0-65535	27000

Tripple Head Split H



Splits the layer texture horizontally into three areas. The layer texture is repeated, whilst "Factor" defines the offset. The separation can be softedged manually.

This effect allows splitting the Pandoras Box output when for example using (Matrox) Tripple Head Devices. Assign it to the output itself or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Left	Left	0-65535	0
Right	Right	0-65535	0
Softedge L	Softedge L	0-65535	0
SE L R	SE L R	0-65535	32768
SE L L	SE L L	0-65535	32768
Center1	Center1	0-65535	32768
Softedge R	Softedge R	0-65535	0
SE R R	SE R R	0-65535	32768
SE R L	SE R L	0-65535	32768
Center2	Center2	0-65535	32768

Tripple Head Split V

Splits the layer texture vertically into three areas. The layer texture is repeated, whilst "Factor" defines the offset. The separation can be softedged manually.

This effect allows splitting the Pandoras Box output when for example using (Matrox) Tripple Head Devices. Assign it to the output itself or to a layer that is toggled into the Output Rendering Pass [320] by using the Layer Inspector [211].

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Top	Top	0-65535	0
Bottom	Bottom	0-65535	0
Softedge T	Softedge T	0-65535	0
SE T T	SE T T	0-65535	32768
SE T B	SE T B	0-65535	32768
Center1	Center1	0-65535	32768
Softedge B	Softedge B	0-65535	0
SE B T	SE B T	0-65535	32768
SE B B	SE B B	0-65535	32768
Center2	Center2	0-65535	32768

Resize

Resizes and repositions the image by adjusting the UV map, i.e. without affecting the layer's scale and position values.
Allows resize factors between 0 and 1, and setting an xy offset. The texture is not repeated.

To calculate the precise size parameter: size factor*65535,
e.g for a scale of 0.25: (1/4)*65535 = 16384

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
FactorX	Size in X (100% - 0%)	0-65535	65535
FactorY	Size in Y (100% - 0%)	0-65535	65535
Pos U	Horizontal Position	0-65535	32768
Pos V	Vertical Position	0-65535	32768

Rotate Tiled

Allows rotating the UV map of the layer texture without affecting the layer's rotation values. The texture is repeated.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Rot	Rotation angle	0-65535	0

Rotate

Allows rotating the UV Mapping of the layer texture without affecting the layer's rotation values. The texture is not repeated.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Rot	Rotation of UV Mapping	0-65535	32768

Texture Zoom

Resizes and repositions the image by adjusting the UV map, i.e. without affecting the layer's scale and position values.
Allows resize factors between 1 and unlimited, and setting an xy offset. The texture is not repeated.

To calculate the precise size parameter: (1-1/2*size factor)*65535,
e.g for a scale of 4: (1-1/(2*4)*65535 = 7/8*65535 = 57343

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Factor X	Horizontal scale factor (100% -...)	0-65535	0
Factor Y	Vertical factor (100% -...)	0-65535	0
Pos U	Horizontal position	0-65535	32768
Pos V	Vertical position	0-65535	32768

Tiling

Allows tiling the UV Mapping of the layer texture.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Factor	Amount of tiling (value addition of 2048 equals one more tile or texture width)	0-65535	0
-	1 tile / texture width	0	-
-	2 tiles / texture width	2048	-
-	3 tiles / texture width	4096	-
-	11 tiles / texture width	20480	-
-	33 tiles / texture width	65536	-

Underscan

Enlarges the content of the layer texture slightly.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0

UV Offset

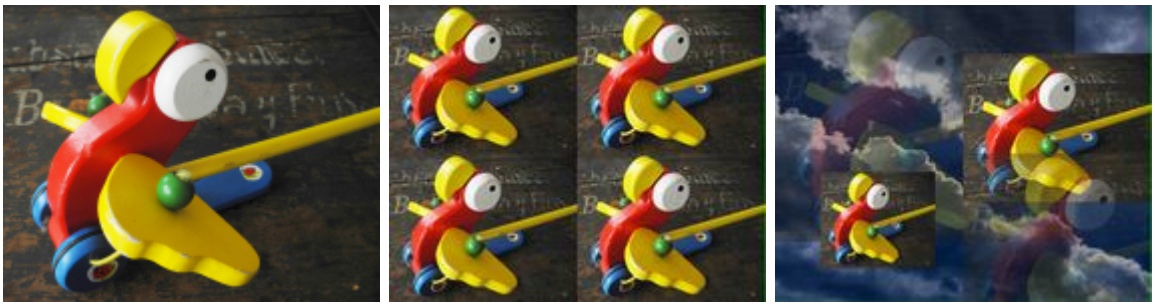
Resizes and repositions the image by adjusting the UV map, i.e. without affecting the layer's scale and position values.
Allows resize factors between 0 and unlimited, and setting an xy offset. The texture is repeated.

To calculate the precise size parameter: (1-1/2*size factor)*65535,
e.g for a scale of 4: (1-1/(2*4)*65535 = 7/8*65535 = 57343

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Offset X	Horizontal offset	0-65535	32768
Offset Y	Vertical offset	0-65535	32768
Factor X	Horizontal scale factor (0% -...)	0-65535	32768
Factor Y	Vertical scale factor (0% -...)	0-65535	32768

UV Remap 4x



Replicates the layer texture four times and adjusts the UV map.
Allows resizing and xy offsetting by mapping four individual, customizable source textures onto a target space. The source texture can not be repeated.
The color and alpha level of the target background can be adjusted. All four quadrants of the target can be edited separately similar to the "UV Remap" effect.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
T1 Opacity	T1 Opacity	0-255	255
T1 X	Horizontal t1	0-65535	0
T1 Y	Vertical t1	0-65535	0
T1 Width	T1 Width	0-65535	32768
T1 Height	T1 Height	0-65535	32768
S1 X	Horizontal s1	0-65535	32768
S1 Y	Vertical s1	0-65535	32768
S1 Width	S1 Width	0-65535	65535
S1 Height	S1 Height	0-65535	65535
T2 Opacity	T2 Opacity	0-255	255
T2 X	Horizontal t2	0-65535	32768
T2 Y	Vertical t2	0-65535	0
T2 Width	T2 Width	0-65535	32768
T2 Height	T2 Height	0-65535	32768
S2 X	Horizontal s2	0-65535	32768
S2 Y	Vertical s2	0-65535	32768
S2 Width	S2 Width	0-65535	65535
S2 Height	S2 Height	0-65535	65535
T3 Opacity	T3 Opacity	0-255	255
T3 X	Horizontal t3	0-65535	0
T3 Y	Vertical t3	0-65535	32768
T3 Width	T3 Width	0-65535	32768
T3 Height	T3 Height	0-65535	32768
S3 X	Horizontal s3	0-65535	32768
S3 Y	Vertical s3	0-65535	32768
S3 Width	S3 Width	0-65535	65535
S3 Height	S3 Height	0-65535	65535
T4 Opacity	T4 Opacity	0-255	255
T4 X	Horizontal t4	0-65535	32768
T4 Y	Vertical t4	0-65535	32768
T4 Width	T4 Width	0-65535	32768
T4 Height	T4 Height	0-65535	32768
S4 X	Horizontal s4	0-65535	32768
S4 Y	Vertical s4	0-65535	32768
S4 Width	S4 Width	0-65535	65535
S4 Height	S4 Height	0-65535	65535
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Alpha	Level of transparency	0-255	0

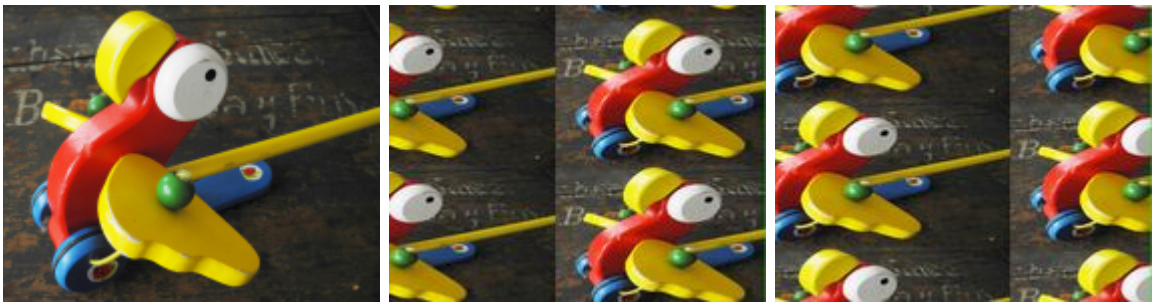
UV Remap

Resizes and repositions the image by adjusting the UV map, i.e. without affecting the layer's scale and position values.
Allows resizing and xy offsetting by mapping a customizable source texture onto a target space. The source texture can not be repeated.
The color and alpha level of the target background can be adjusted.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Target X	Horizontal target position	0-65535	0
Target Y	Vertical target position	0-65535	0
T Width	Hor. target scaling (0% -100%)	0-65535	32768
T Height	Vertical target scaling (0% -100%)	0-65535	32768
Source X	Horizontal source position	0-65535	32768
Source Y	Vertical source position	0-65535	32768
S Width	Hor. source scaling (100% -...)	0-65535	65535
S Height	Vertical source scaling (100% -...)	0-65535	65535
Red	Color picker / Level of red	0-255	0
Green	Color picker / Level of green	0-255	0
Blue	Color picker / Level of blue	0-255	0
Alpha	Level of transparency	0-255	0

UV Scroll



Scrolls the media endlessly in horizontal and / or vertical direction. The scrolling speed in x- and y-direction as well as x- and y-scale can be adjusted.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of effect itself	0-255	0
Speed X	Horizontal speed	0-65535	32768
Speed Y	Vertical speed	0-65535	32768
Factor X	Horizontal scale factor (100% -...)	0-65535	32768
Factor Y	Vertical scale factor (100% -...)	0-65535	32768

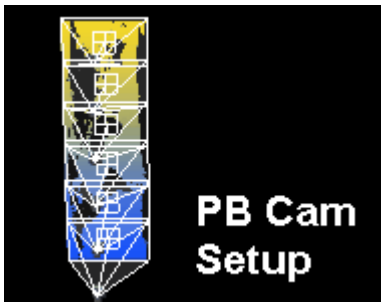
UV Wrap Blend Horizontal

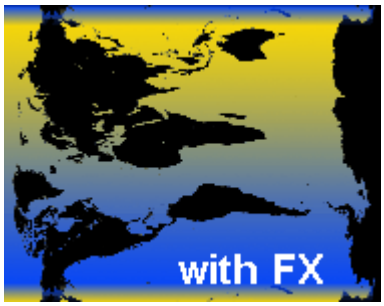
Allows to blend the left edge of a texture with a copy of the content of the right edge. At the same time the right edge blends into the left part. This is of interest when you have a 360° projection and your setup in Pandoras Box is that cameras are spread out and look onto one layer (see image). In case your content shows a hard edge because the right and left pixels do not match you can blend the area with this effect.
For other applications, you may combine this effect with the Resize, UV Offset or Texture Zoom.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Blend	Level of right / left blend	0-65535	32768

UV Wrap Blend Vertical





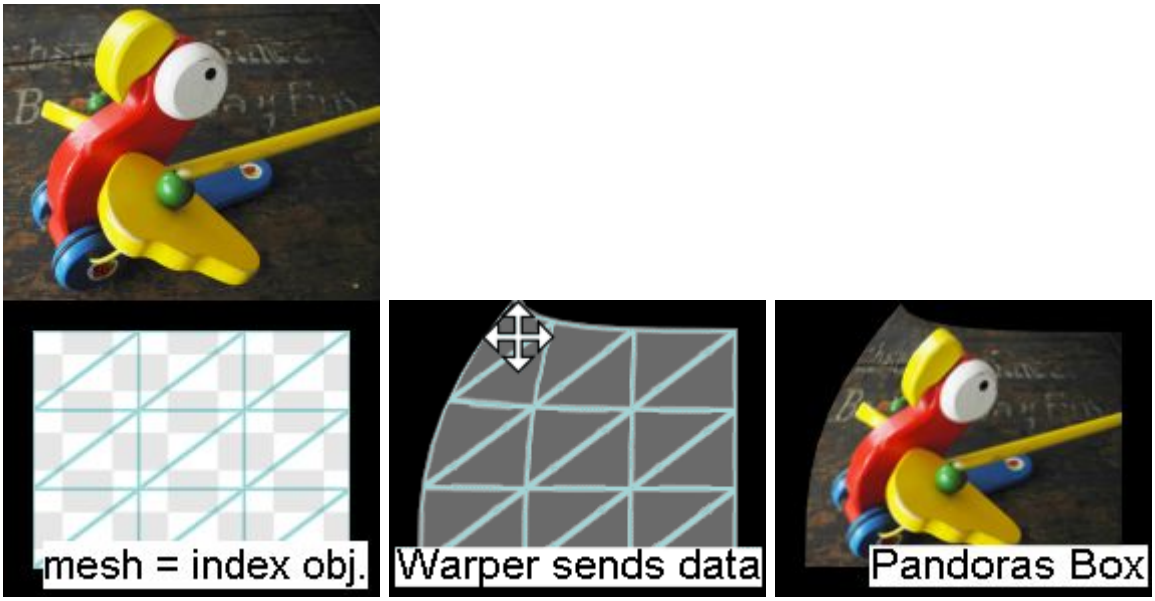
Allows to blend the upper edge of a texture with a copy of the content of the bottom edge. At the same time the bottom edge blends into the upper part. This is of interest when you have a 360° projection with rotated projectors and your setup in Pandoras Box is that cameras are spread out and look onto one layer (see image). In case your content shows a hard edge because the top and bottom pixels do not match you can blend them. For other applications, you may combine this effect with the Resize, UV Offset or Texture Zoom.

Technical Information
This effect needs to be the first effect on the layer if you are working with images or videos (i.e. textures) that are bigger than the "Max. Internal Texture Size". The Internal Texture Size depends on your PB license and is adjustable in the Configuration tab under "Render Engine [162]". The chapter "FX Order [325]" explains how to drag effects in the Device Tree [173] tab to change the order of applied effects.

Parameter	Description	Value Range	Default
Mix	Level of Effect	0-255	0
Blend	Level of top / bottom blend	0-65535	32768

6.5.2.4.42 Warp

Warp

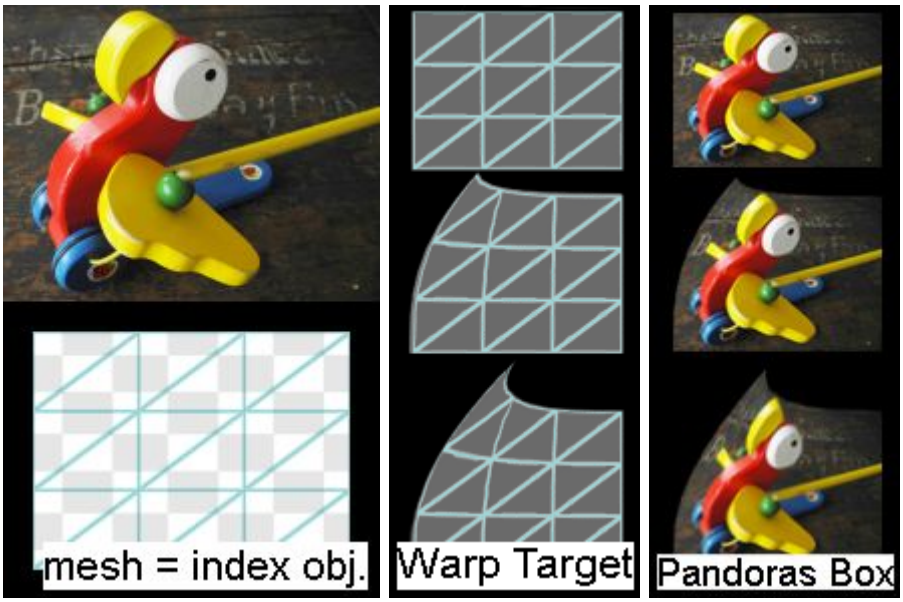


This effect allows deforming an object by moving its vertices live in the Warper tool [2129]. Please refer to the live warping tutorial [2174] in the manual for more information.

Note that this effect has no parameters, you will not see it in the Device Control tab [171] itself. Only when opening the FX parameter in the Device Tree [173] you will see the effect. The effect cannot be turned off like other effects, the only way to deactivate it, is to delete it entirely.

Parameter	Description	Value Range	Default
none!		-	-

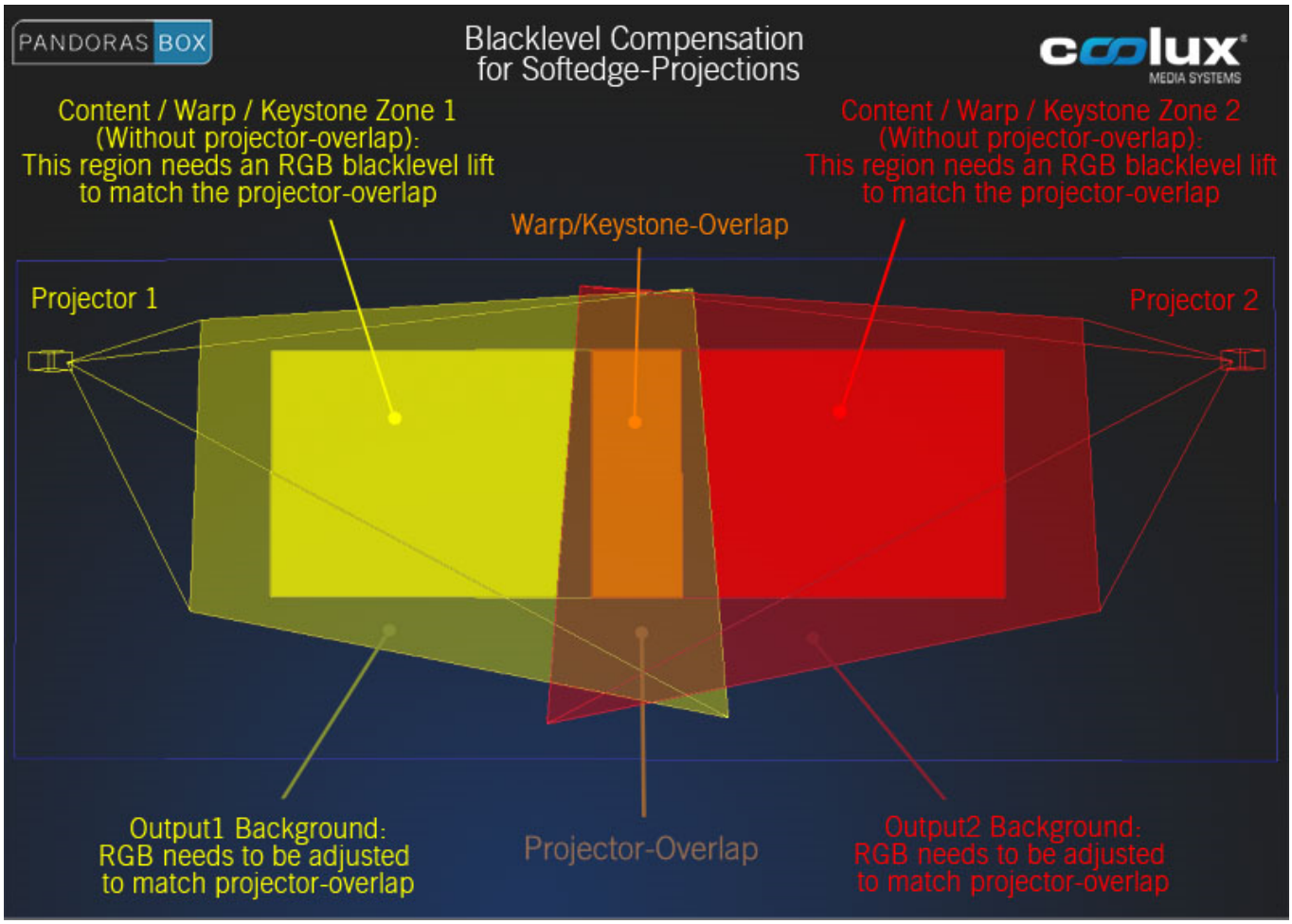
Warp Target



This effect allows deforming an object by moving its vertices according to the saved data in the Warp Target media file. The media file can be either one or several Morph Targets from the Warper tool. Per default that would be a PNG file, respectively a PNG image sequence. You may also convert your files to BMP files or a uncompressed video format, e.g. the AVI format or a lossless format from the coolux codec [102]. In any case, activate the "Video Alpha channel" in the File Inspector [191]! Note that the target media has no separate playback control, use a video file as a layer texture or work with the Share layer feature.
Please refer to the live warping tutorial [2174] in the manual for more information about the steps in the Warper.

Parameter	Description	Value Range	Default
Target	Target	-	-

6.5.2.5 Blacklevel Compensation



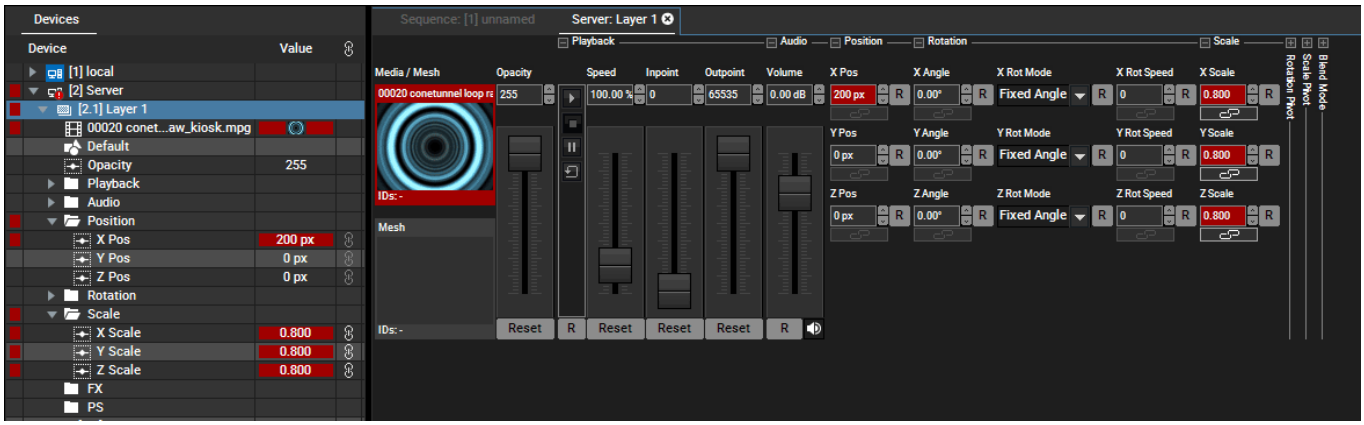
Blacklevel compensation is only available on Pandoras Box Servers.

For blacklevel compensation layers are needed that are not included in the first render pass, the composition pass (the render passes are explained in the topic Video Processing Pipeline [320]). To exclude a layer from the composition pass, check the radio button "Output" in the layer's Inspector [211]. This layer is now not affected by the output settings in regards to warp object, keystone and softedge settings.

Use the layer with effects [327] that suit your need (area-coloring or black lifting colors), e.g Gradient [452] or Shapes [583] FX. There are effects available, e.g. triangle and square gradients / shapes, to set up a geometry that matches exactly the shape needed. The different overlapping types are depicted in the above image. In addition, the shapes can be adjusted with different colors per corner. This is important when working with projectors that show a gradient due to lens errors or worn out lamps.

It is possible too, to adjust the backgrounds RGB values in the camera [675] and output [682] layer.

6.5.3 Video Layer



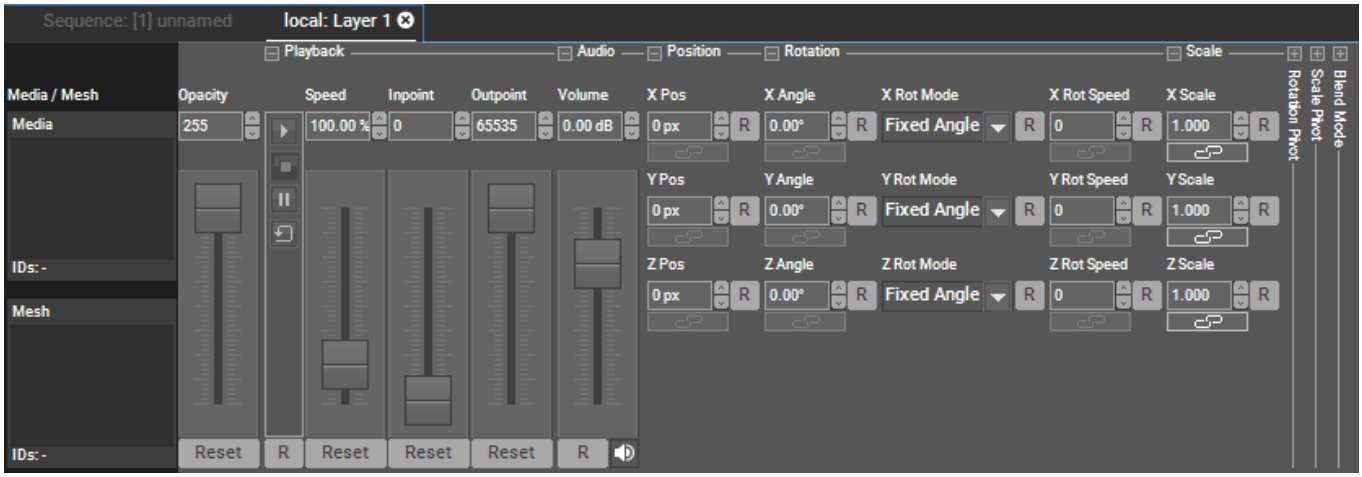
A Video Layer allows full motion video playback as well as rendering still images. The chapter "Content [90] " explains which image and video formats are supported. Further, you can assign Live Inputs [133] but also resources like a Canvas, Browser, Playlist or Text Inputs which are all explained in the chapter "Context Menu - Project" [279].

You may add an unlimited amount of Video Layers to your project. Right-click on the Site to open the context menu [178] and choose "Add (Multiple) Device(s) > Video Layer(s)". The new layer will be added below the highest Video Layer in the Device Tree. The layer can be moved to another position in the Device Tree by drag and drop [175].

The Video Layer includes the following sections. Information about DMX Control is included, addressing customers who wish to remote control a layer with a lighting desk via DMX or Art-Net. If you like to use the Widget Designer or another application instead, please refer to this parameter list [1514].

Media Selection [647]
Opacity [649]
Playback [649]
Audio [650]
Position [651]
Rotation [652]
Scale [653]
Rotation Pivot [654]
Scale Pivot [655]
Blend Mode [657]

6.5.3.1 Media / Mesh



Media Selection

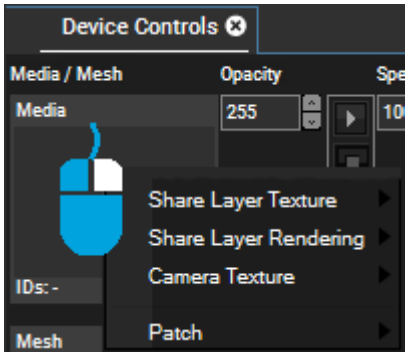
To assign a media resource (see previous chapter [647]) to a layer, drag and drop the resource from the project tab onto the designated layer. You can also highlight the target layer by left-clicking on it and use the right-click command on the resource in the project tab "assign to active device" or just double-click on the media file. Media

can also be assigned from the thumbnail browser by double clicking it for attributing it to the selected highlighted layers.

Please note: When working in a Master-Client setup, please make sure that all media and objects are spread to all Clients so that they can render the content.

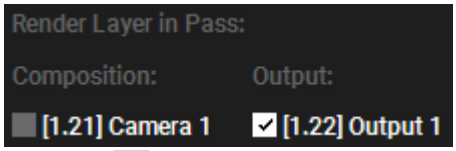
To remove the resource of the layer, right-click on the thumbnail in the Device Control Tab and choose "Reset". Alternatively you can open the media or mesh parameter in the Device Tree and after right-clicking you can choose "Reset" there.

Share Layer



Instead of assigning a media file you can also choose to automatically assign the texture of another layer. Right-click the Media field and choose one of the two "Share Layer" options.

Share Layer Texture simply loads the same texture as the chosen Layer. This way you can playback a video file (or any other content) without having to load it several times from the hard disk. This allows a better performance and saves programming time. Note that the texture only is shared. If there are effects on the linked layer, they only affect this one layer.
Share Layer Rendering on the other hand loads the rendered result of the chosen Layer. This means the assigned texture and applied effects are rendered and the result is shared.



The third option "**Camera Texture**" shares the rendered result that serves as the input for the linked Camera. This feature is only working correctly if the Layer is not part of a Camera's render pass. In the Layer's Inspector, make sure to uncheck "Camera" and check "Output". For more information about render passes, please see the chapter "Video Processing Pipeline" [320].

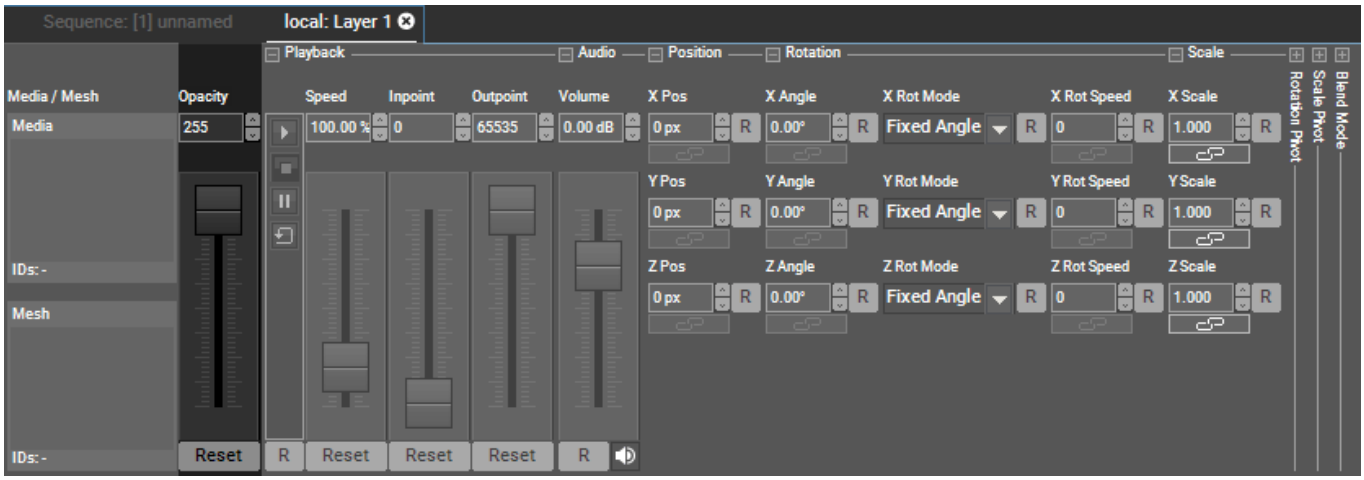
Object Selection (Mesh)

Each Layer has a default layer object which defines the mapping of the texture (your media file). The Mesh field allows loading custom object files to change this mapping behavior. The chapter Object Formats [100] explains objects in detail and what formats can be used. You can assign and reset them as explained above.

External Control via DMX, Widget Designer, etc.

You can control the Media and Mesh parameter externally. In general, media and mesh files can be assigned via their file names or by addressing them via the Folder and File ID. For DMX control, only the later is possible.
To assign a Folder and File ID, select the file (or folder) in the Project tab and look for the option in the File Inspector [191]. Both, the Folder and the File ID can be assigned with a value from 1-255.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.3.2 Opacity



The opacity sets the transparency of the image or video content.

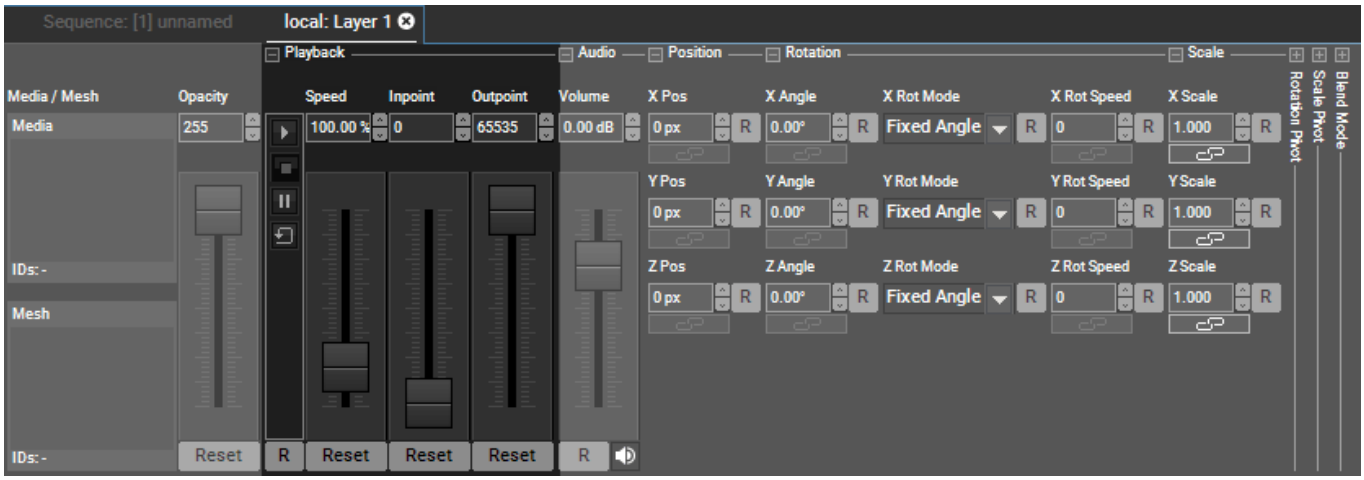
Opacity 0% (value 0) fully transparent
Opacity 100% (value 255) fully visible

If you like to fade from one layer to another, you can simply increase the Opacity value over a certain duration of time which creates a smooth transition. You can also use the Opacity control in conjunction with a Transition FX [623]. A wide variety of wipes and dissolves is available in the TransitionFX section. Together with the TransitionFX, the opacity channel can also be used to superimpose two images on top of each other or to mask specific areas. Learn how to add effects [323]...

External Control via DMX, Widget Designer, etc.

You can control the Opacity parameter externally, simply assign a value between 0-255.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.3.3 Playback



Each layer has individual playback control for video and audio files. The playback section is divided into the following parameters: Playback Transport, Speed, In- and Outpoint.

Playback Transport

The playback control channel allows setting the video mode to Play Once, Stop, Pause and Play Loop.

Playback Speed

The speed of the video playback may be varied by changing the Speed Parameter value.
The default value (1:1 Speed) is 128 (=100% in percentage view, see configuration tab [139]).

Slowing down the speed of a video file will reduce the framerate. In order to keep a smooth playback you may want to use FluidFrame™ [191] for this video file.

Please note: The speed and playback behavior varies, depending on the used codec. Some video codecs may not support different playback speeds and some codecs may not be suitable for seamless looping. Please see the Encoding [100] section and the format [97] description for further information as well.

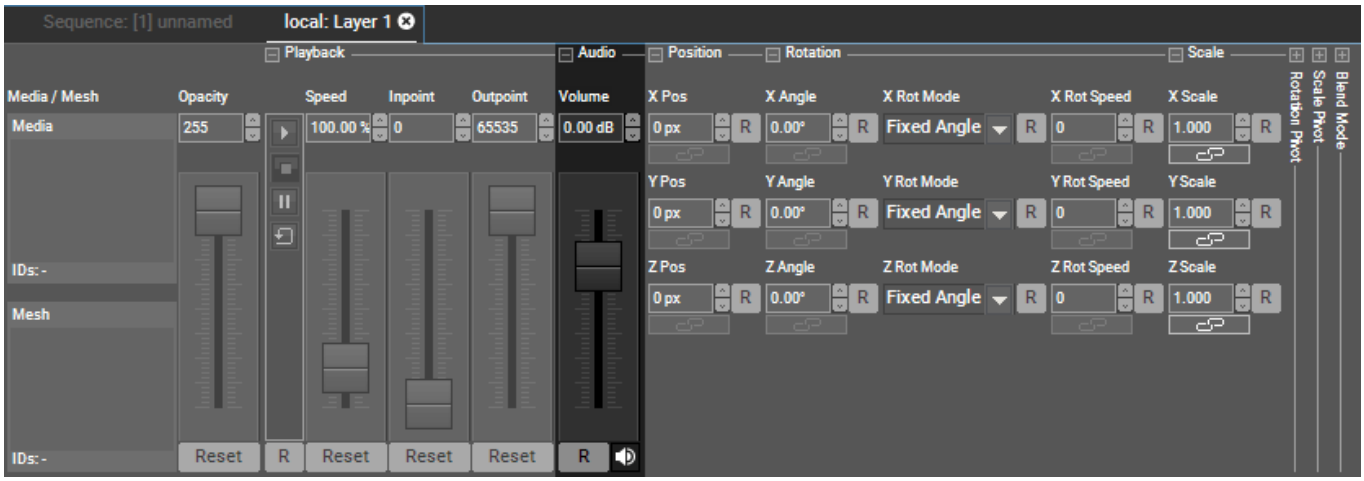
Inpoint / Outpoint

Set the in- and outpoint to define the specific start & end marks of a clip, the play once and play loop video mode will work in the resized area.
The In- & Outpoint selection works on a percentage base of the overall frame length. Use "Pause" and look up with the Inpoint the frames from where you want to start the file playback.
When programming containers in the timeline you can alternatively enter the exact frame number you like to start (or end with) by using the "In" (or "Out") value in the Clip Inspector [206]. It is also possible to enter exact timecode in frames, seconds etc.

External Control via DMX, Widget Designer, etc.

You can control all playback parameters externally. For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

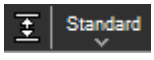
6.5.3.4 Audio



With the Volume parameter in the Audio Section you will control the Volume of both: Video Files including an audio part and plain audio files such as .wav- or .mp3-files. The volume channel ranges from -96 dB to +6 dB.

Note that playing back sound with a Video Layer eliminates the possibility to synchronize the video playback. If synchronized playback is demanded, please use Audio Tracks [661] instead.

Waveform

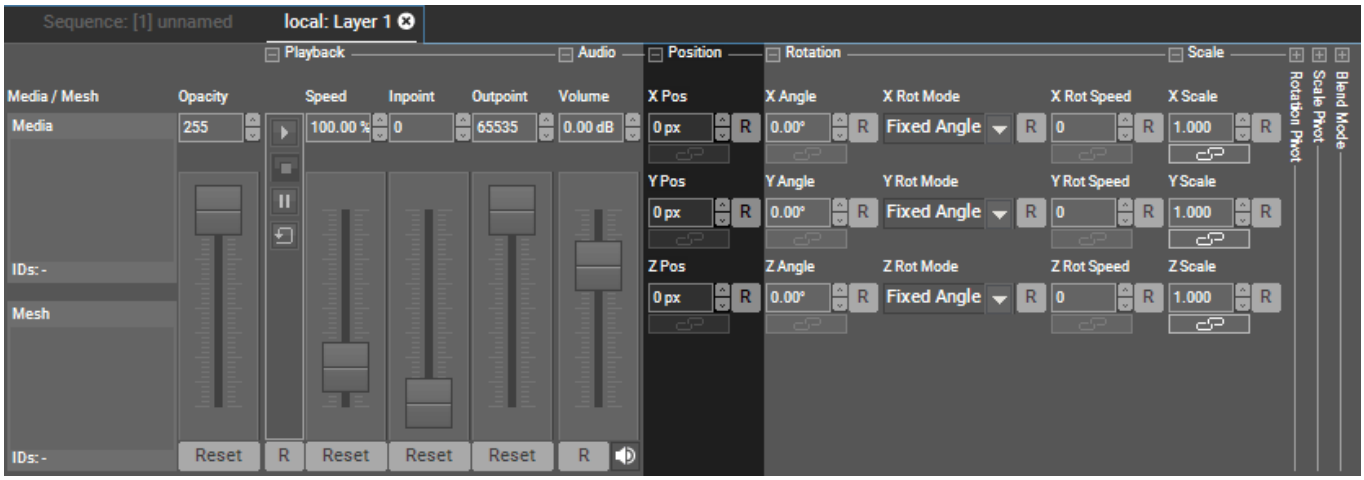
The Waveform parameter is not editable and therefore it is not included as a device parameter in this tab. It simply displays the audio peaks in the Sequence when the parameter is visible. In case you like to align other content (of Video Layers for example) as accurate as possible it is recommended to zoom in closely. Use the [+] key on the main keyboard or the Zoom-in icon  in the Sequence Button Bar to do so. There, you may also find the drop-down for changing the line height 

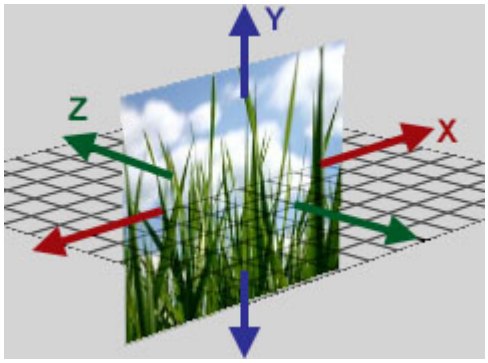
which enlarges the waveform accordingly. You can choose between two waveforms, a standard one with positive and negative peaks or a a rectified waveform with only positive peaks. You can find this setting in the Configuration tab > User > Sequence [142] > Clip settings.

External Control via DMX, Widget Designer, etc.

You can control the Volume parameter externally. For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.
More information can also be found in the chapter about the Audio Track Volume [663].

6.5.3.5 Position





Each layer can be positioned in XYZ individually. Per default, position values are displayed in pixels. A positive X value will move the layer to the right, a positive Y value to the bottom and a positive Z value backwards. Negative values move the layer in the opposite direction.

Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

To bring a layer in front of another one, you could either change the Z position or the layer order in the Device Tree [175].

When selecting multiple layers, you can use the Align function [124] in the Status Bar. Instead of applying the same value to all selected layers, "Align" follows the selected pattern and applies e.g. mirrored values.

External Control via DMX, Widget Designer, etc.

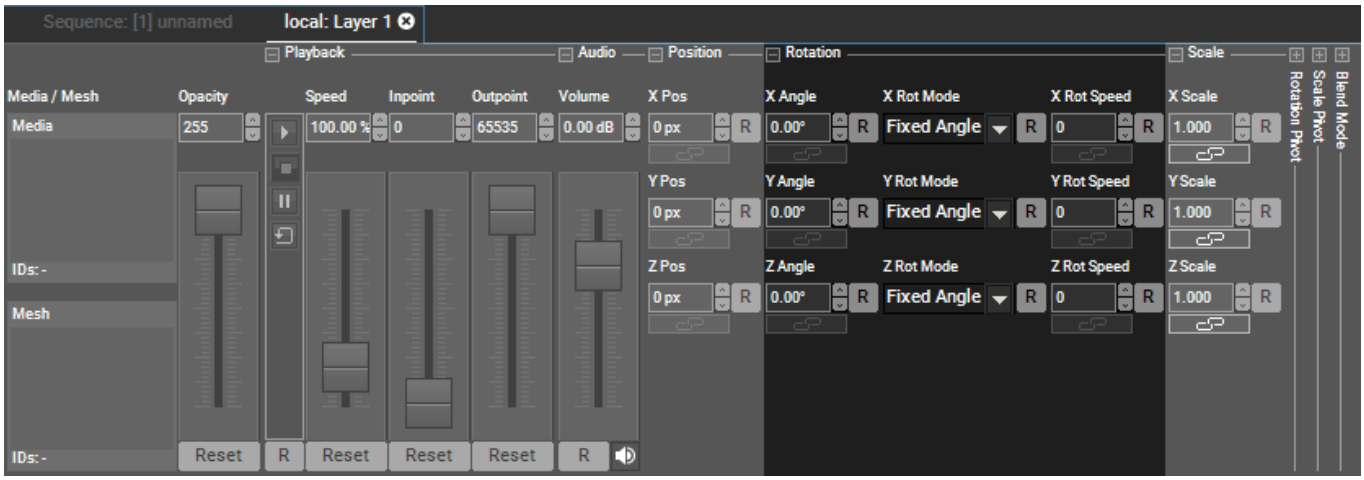
You can control the position parameters externally.

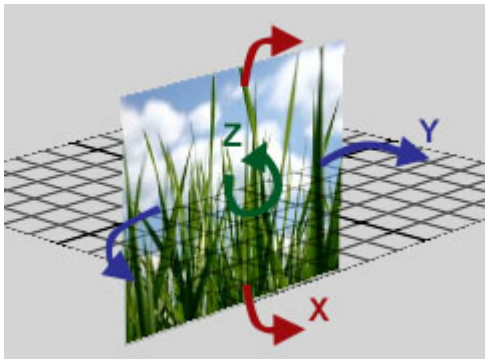
Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.3.6 Rotation





Each layer can be rotated in XYZ individually. A positive X rotation tilts the bottom forwards, a positive Y rotation tilts the right site forwards and a positive Z rotation rotates the layer anti-clockwise. Negative values rotate in the opposite direction.

By default the rotation pivot is located in the layer's X/Y/Z center. This can be changed by modifying the rotation pivot's position in the "Rotation Pivot"-Section 654.

When selecting multiple layers, you can use the Align function 124 in the Status Bar. Instead of applying the same value to all selected layers, "Align" follows the selected pattern and applies e.g. mirrored values.

Per default, the Rotation Mode parameter is set to "Fixed Angle" which means that the "Angle" parameter (in °) has control. If you change it to "Rot Speed", the "Speed" parameter (in rotations per hour) rotates the layer continuously. Any changes to the fixed Angle value won't affect the rotation any more.

Examples for Rotation Speed:
To rotate in the opposite direction, simply negate the value.
1 rotation per hour: value 1
1 rotation per minute: value 60
1 rotation per second: value 3600
2 rotations per second: value 7200 because 60 (minutes) x 60 (seconds) x 2 = 7200.

External Control via DMX, Widget Designer, etc.

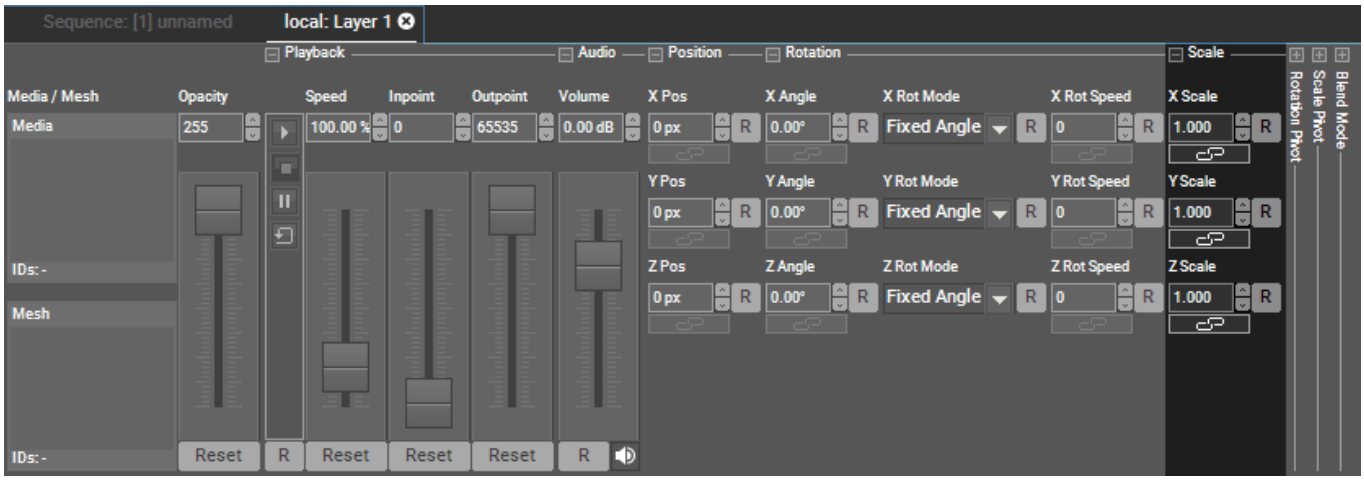
You can control all rotation parameters externally.

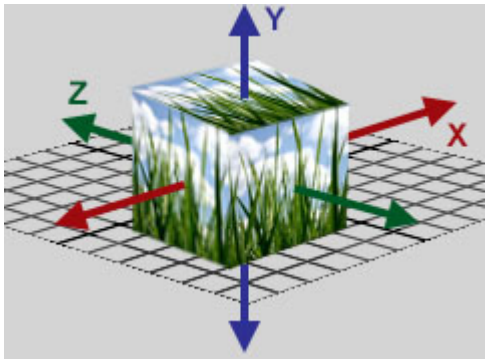
For the "Rotation Mode", simply choose 0 for "Fixed Angle" and 1 for "Rotation Speed".

If you remote control the "Fixed Angle" and "Rot Speed" via Widget Designer or the SDK, simply send the same values as you would enter them in the user interface. For exact naming etc., the Parameter List 1514 is of interest.

For DMX control, the chapter DMX Tables 708 lists all parameter information.

6.5.3.7 Scale



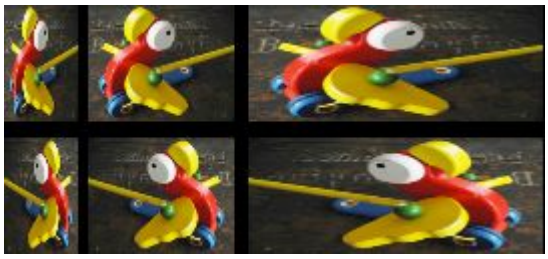


Each layer can be scaled along the XYZ axis individually or by keeping a linked relationship. Per default, the chain icon is clicked for all axes which means that you are scaling uniformly in all directions.

By default the scale pivot is located in the layer's X/Y/Z center. This can be changed by modifying the scale pivot's position in the "Scale Pivot"-Section 655.

When selecting multiple layers, you can use the Align function 124 in the Status Bar. Instead of applying the same value to all selected layers, "Align" follows the selected pattern and applies e.g. mirrored values.

The Scaling value is applied as a multiplication factor. A value of 1, means no change. A scaling of 0.5 is half the size and a scaling of 2 double the size. Negative values flip the layer as seen with the three images at the bottom. When the scale factor is 0, the layer is not visible. This image was scaled with the above mentioned values along the X axis.



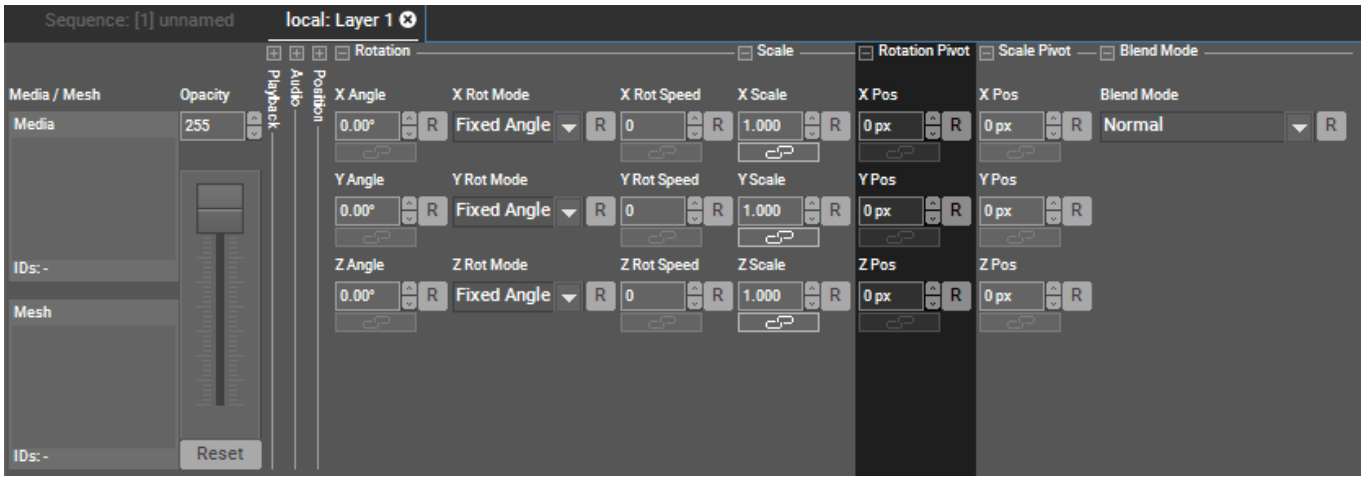
External Control via DMX, Widget Designer, etc.

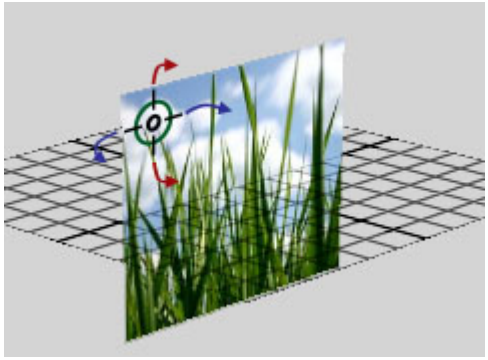
You can control all scale parameter externally.

If you remote control the parameters via Widget Designer or the SDK, simply send the same values as you would enter them in the user interface. For exact naming etc., the Parameter List 1514 is of interest.

For DMX control, the chapter DMX Tables 708 lists all parameter information, e.g. that a DMX value of 1000 refers to a scale factor of 1. Due to the reduced DMX control value range the image can not be flipped!

6.5.3.8 Rotation Pivot



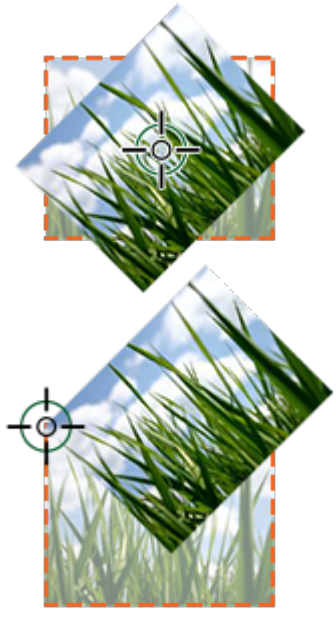


The Rotation Pivot section allows to move the position of the rotation pivot. Per default, it is in the center of the layer which means that the layer turns around its center if any X, Y and Z Rotation is applied.

The rotation pivot can be positioned in XYZ individually. Per default, position values are displayed in pixels. A positive X value will move the pivot to the right, a positive Y value to the bottom and a positive Z value backwards. Negative values move the pivot in the opposite direction.

Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" 160. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

How the Rotation Pivot Affects the Rotation



In both depicted examples, the "Z Angle" is set to 45°.

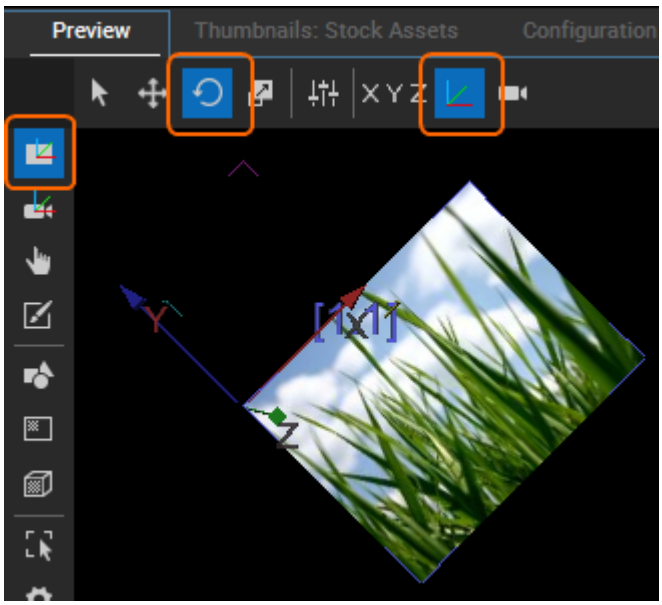
First, the rotation pivot is left at the default position (X/Y/Z Pos = 0) to turn the layer around its center.

In the second example, the rotation pivot is moved to the left upper corner so that the layer now turns around this corner.

Rotation Pivot in Preview

If you like to see the rotation pivot in the Preview window, check that the Preview is in "Layer Mode" (first button, left side) and that the Rotation Mode is activated (top row or shortcut [2]). Lastly, activate the Layer Gizmo .

For more information how to rotate the layer and move the pivot in the Preview please read the Preview chapter [250].



External Control via DMX, Widget Designer, etc.

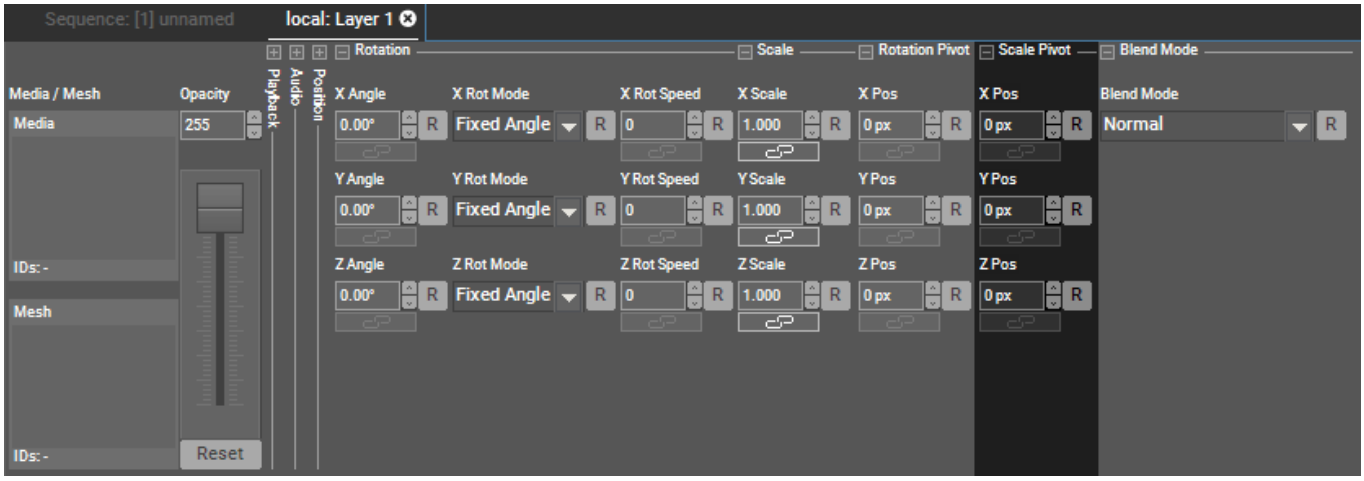
You can control the pivot parameters externally.

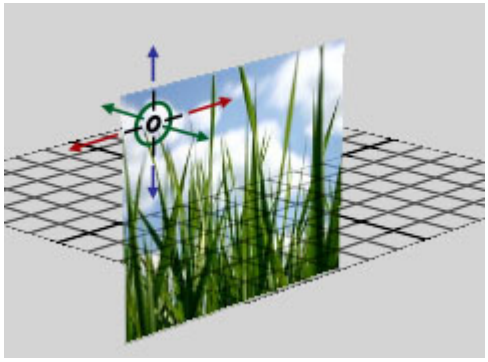
Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.3.9 Scale Pivot



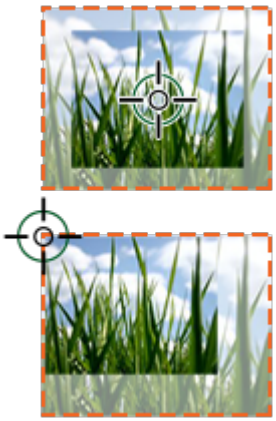


The Scale Pivot section allows to move the position of the scale pivot. Per default, it is in the center of the layer which means that the layer is scaled from its center if any X, Y and Z scaling is applied.

The scale pivot can be positioned in XYZ individually. Per default, position values are displayed in pixels. A positive X value will move the pivot to the right, a positive Y value to the bottom and a positive Z value backwards. Negative values move the pivot in the opposite direction.

Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

How the Scale Pivot Affects Scaling



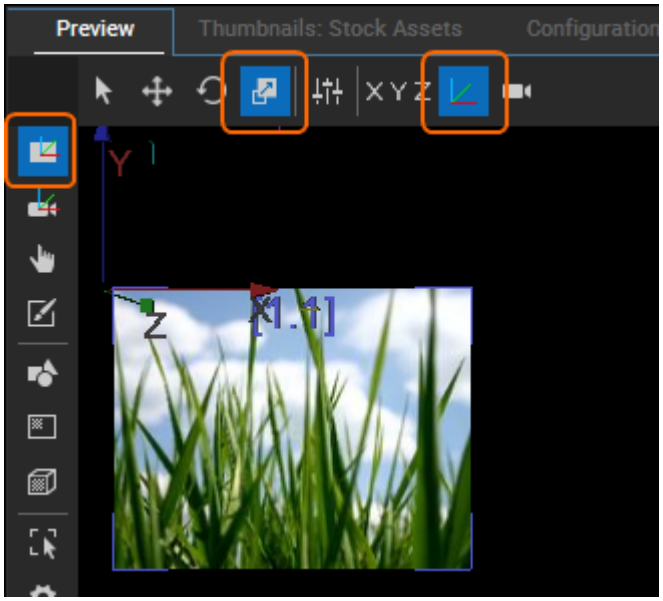
In both depicted examples, "X Scale" and "Y Scale" are set to 0.75.

First, the scale pivot is left at the default position (X/Y/Z Pos = 0) to scale the layer from its center.

In the second example, the scale pivot is moved to the left upper corner so that the layer now scales from this corner.

Scale Pivot in Preview

If you like to see the scale pivot in the Preview window, check that the Preview is in "Layer Mode" (first button, left side) and that the Scale Mode is activated (top row or shortcut [3]). Lastly, activate the Layer Gizmo  .
For more information how to scale the layer and move the pivot in the Preview please read the Preview chapter [250].



External Control via DMX, Widget Designer, etc.

You can control the pivot parameters externally.

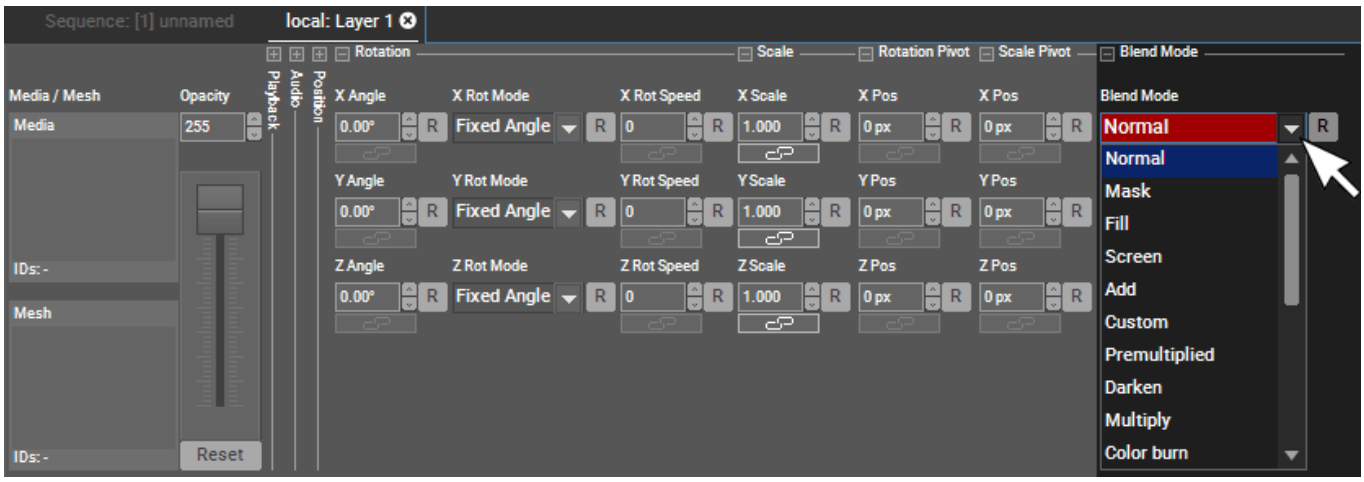
Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.3.10 Blend Mode

In digital image editing, blend modes are used to determine how a layer blends with underlying layers. In Pandoras Box, the default blend mode (called "Normal") has the effect that the top layer simply obscures the background layer(s) and its RGB values stay unchanged. Only if the top layer has (partly) transparent parts, the layer(s) below can be seen. The other blend modes mix the RGB values of the respective layers in a different way, e.g. add them up or subtract them.
To switch the blend mode, first select the top layer. Per default, that is the one with the higher Layer number or the one nearer to the Camera Layer. In the tab "Device Controls", click on the + symbol above "Blend Mode". Open the drop-down list which offers all available modes and choose one. If needed, adjust the opacity value with the "Opacity" fader.



External Control via DMX, Widget Designer, etc.

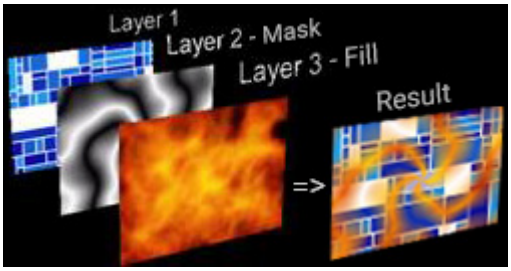
You can control the Blend Mode parameter externally, simply send a value (0 to 19) to set the drop-down to this index.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

Normal - 0

The top layer obscures the background layer(s). The RGB values of the top layer stay unchanged. Only if the top layer has transparent parts, the layer below can be seen through.

Mask & Fill - 1,2

The "Mask" and the "Fill" blend mode are meant to be used together. The "Key and Fill" effect is based on the layer stack from back to front, as the back layer is being rendered first and every other layer is rendered on top of the previous one. The below image shows an example:



The blue content should be the background, so it is assigned to Layer 1 and the blend mode is the default "Normal" one.

The fire content should be in the very front, so it is assigned to Layer 3. With the default blend mode, it simply overlays the background layer. As soon as the blend mode is changed to "Fill", it is only rendered in areas defined by the mask.

Last, select the layer in between the two, in our case Layer 2 , and change the blend mode to "Mask". If you now assign an image or video with transparency, you will see that the fire content is rendered in those transparent areas only.

Indeed, you can use any content as a mask. Simply assign a key effect [512] to define transparent areas. For this example, a black and white content should be the mask. The "White Key" effect is chosen and the "Mix" parameter is set to 255 to key out all white parts.

Please note:
When alpha blend modes are used the opacity ranges from 1-255, a zero value switches off the layer to be rendered, and therefore switches the mode back to default opacity mixing.

If the Key & Fill Layers (Layer 2 and 3 in the example above) should be faded in / faded out, please use the Mix-Parameter of the White Key of the Mask Layer to do so, instead of using the Opacity of both Layer 2 and 3.

Screen - 3

This looks at the value of the red, green and blue channel of each pixel from the inverted(!) blend layer and background layer(s) and multiplies them which always results in a lighter color. Blending with black leaves the color unchanged as the inverted vector value is 1. The result reminds on a projection of multiple images on top of each other.

Add - 4

This blend mode simply adds the RGB values of each pixel of the blend layer and background layer(s). If a red, green and blue pixel (with value 255) overlap, white is displayed.

Custom - 5

Here you may use a customized blend mode. By default the blend mode "Normal" is set.
To create your own blend mode, you have to modify the dedicated entries in the tweak config file.
This can be found here:
Pandoras Box root folder\data\config\tweak_config.txt

When opening this .txt-file scroll down to this section:

;----------;

;description:
;define custom layer blending with operation and mode for color and alpha.

;possible values for blend operations are defined in D3DBLENDOP

;find more information here: http://msdn.microsoft.com/en-us/library/bb172509(VS.85).aspx

;possible values for blend modes are defined in D3DBLEND

;find more information here: http://msdn.microsoft.com/en-us/library/bb172508(v=vs.85).aspx

;default:
;blendOpColor=1
;srcBlendColor=5
;destBlendColor=6
;blendOpAlpha=1
;srcBlendAlpha=5
;destBlendAlpha=7

[LayerBlending]
blendOpColor=1
srcBlendColor=5
destBlendColor=6
blendOpAlpha=1
srcBlendAlpha=5
destBlendAlpha=7

;--;

Modify the entries under the section [LayerBlending] to create your own blend mode, like described below in the tweak_config.

Please note:
Close the PB application before editing the tweak config file!
These edited mode settings have to be done on each PB device that should be able to use this mode!

Premultiplied Alpha - 6

With the "Premultiplied Alpha" blend mode, the alpha value specifies how much an RGB value obscures the background. Alpha and RGB are linked. To make an object transparent you must reduce both its RGB (to contribute less color) and also its alpha. The RGBA value of fully transparent objects must be zero, which means 100% transparent and black.

With the "Normal" blend mode, so called Straight or Linear Alpha is applied which means that the alpha value specifies how solid an RGB value is rendered. Alpha and RGB are independent. To make an object fade out, you would gradually reduce its alpha value while leaving RGB unchanged.

Darken - 7

This looks at the value of the red, green and blue channel of each pixel from the blend layer and background layer(s) and mixes a new color based on the lowest value. If you overlay a purely red pixel (RGB = 255,0,0) with a light cyan colored one (RGB = 127,255,255), the resulting pixel is dark red (RGB = 127,0,0) because the lowest red value is 127 and the lowest green and blue one is 0.

Multiply - 8

The Multiply blend mode simply multiplies the RGB vector values from the blend layer and background layer(s). Multiplying any color with black (vector value 0) results in black. Multiplying it with white (vector value 1) leaves the color unchanged. Hence, the result is always darker.
If you overlay a magenta pixel (RGB = 1,0,1) with a gray one (RGB = 0.5,0.5,0.5), the resulting pixel is dark magenta (RGB = 0.5,0,0.5)

Color Burn - 9

This looks at the value of the red, green and blue channel of each pixel from the blend layer and background layer(s) and mixes a new color which is always darker. Blending with white leaves the color unchanged. The contrast between the background color and blend color is increased.

Linear Burn - 10

This looks at the value of the red, green and blue channel of each pixel from the blend layer and background layer(s) and mixes a new color which is always darker. Blending with white leaves the color unchanged. The brightness of the background color is decreased.

Lighten - 11

This looks at the value of the red, green and blue channel of each pixel from the blend layer and background layer(s) and mixes a new color based on the higher value. Blending with white results in white. Blending with black leaves the color unchanged. Pixels darker than the blend color are replaced whilst lighter pixels do not change.

Color Dodge - 12

This looks at the value of the red, green and blue channel of each pixel from the blend layer and background layer(s) and mixes a new color which is always brighter. Blending with black leaves the color unchanged. The contrast between the background color and blend color is decreased.

Overlay - 13

Depending on the background colors, the Overlay blend mode works either as the Multiply or as the Screen blend mode. The blend layer overlays the background pixels while preserving their highlights and shadows. Thus, the resulting mixed color reflects the lightness or darkness of the background.

Soft Light - 14

Depending on the colors of the blend layer, the Soft Light blend mode results in darker or lighter background colors. The effect is similar to shining a diffused spotlight (light source = blend layer) on the background. If the blend color is lighter than 50% gray, the background is lightened as if it were dodged. If the blend color is darker than 50% gray, the background is darkened as if it were burned in. Blending with pure black or white produces a distinctly darker or lighter area.

Hard Light - 15

Depending on the blend colors, the Hard Light blend mode works either as the Multiply or as the Screen blend mode. The effect is similar to shining a harsh spotlight (light source = blend layer) on the background. If the blend color is lighter than 50% gray, the background is lightened, as if it were screened. This is useful for adding highlights. If the blend color is darker than 50% gray, the background is darkened, as if it were multiplied. This is useful for adding shadows. Blending with pure black or white results in pure black or white.

Difference - 16

This looks at the value of the red, green and blue channel of each pixel from the blend layer and background layer(s) and mixes a new color by subtracting the lower value from the higher one. Hence, blending with pure black leaves the color unchanged. Blending with pure white results in inverted background colors.

Exclusion - 17

This is very similar to the Difference blend mode, but the contrast is lower. Blending with pure black leaves the color unchanged. Blending with pure white results in inverted background colors.

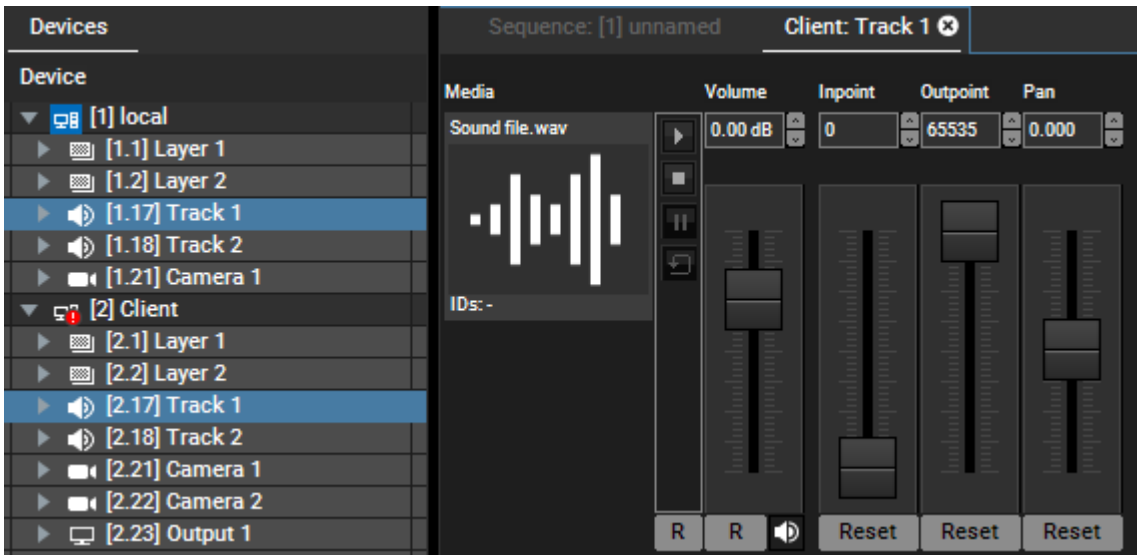
Subtract - 18

This looks at the value of the red, green and blue channel of each pixel from the blend layer and background layer(s) and mixes a new color by subtracting the blend value from the background value. The lowest value must be 0. Hence, blending i.e. subtracting with pure black leaves the background color unchanged. Blending with pure white results in pure black. The result is always darker.

Negation - 19

This blending mode is similar to the Exclusion blending mode except that it shows brighter and more vibrant colors. Blending with pure black leaves the color unchanged. Blending with pure white results in inverted background colors.

6.5.4 Audio Tracks



PB offers audio and video playback synchronization to the audio clock via its dedicated Audio Tracks. The Pandoras Box Software License offers unlimited Audio Tracks. To add one, simply right-click the Site in the Device Tree [173] and choose "Add Device".

Please note that a dedicated ASIO device or native Dante need to be setup in the "Configuration tab > ASIO Audio [166]". Pandoras Box supports all common ASIO sound cards. The chapter "Input and Output Cards" [1947] lists all boards we offer. Additionally, PB supports Dante natively since version 8. Pandoras Box Licenses are natively supporting 32 input and 32 output channels without the need of any hardware, driver configuration or licensing.

Audio tracks can play and synchronize:
- **PCM Wave files**
  16, 24 or 32bit
  mono and stereo
  sample rate should be supported by sound card

- **audio channels from input cards**
  embedded audio channels from HDMI 2.0 input cards [1978] or 12G-SDI Input Cards [1972] or dedicated audio channels from audio inputs cards [1947]
  in- and output sample rate should match

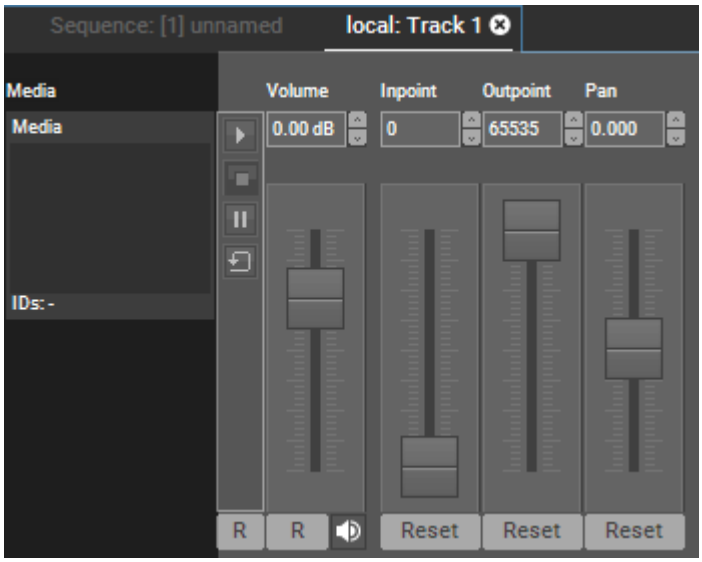
As explained in the chapter "Audio Formats [92]", another possibility to playback sound is, to use a Video Layer but in that case it cannot be synchronized. The chapter about the Configuration tab "ASIO Audio" [166] includes more information regarding ASIO and Dante support, e.g. synchronization and channel settings.

The Audio Track includes the following sections: Information about DMX Control is included, addressing customers who wish to remote control a layer with a lighting desk via DMX or Art-Net. If you like to use the Widget Designer or another application instead, please refer to this parameter list [1514].
Media [662]
Playback Transport [663]
Volume [663]
Inpoint/Outpoint [664]
Pan [665]

The Waveform parameter is not editable. It simply displays the audio peaks in the Sequence when the parameter is visible. In case you like to align other content (of Video Layers for example) as accurate as possible it is recommended to zoom in closely. Use the [+] key on the main keyboard or the Zoom-in icon  in the Sequence Button Bar to do so. There, you may also find the drop-down for changing the line height 

which enlarges the waveform accordingly. You can choose between two waveforms, a standard one with positive and negative peaks or a a rectified waveform with only positive peaks. You can find this setting in the Configuration tab > User > Sequence [142] > Clip settings.

6.5.4.1 Media



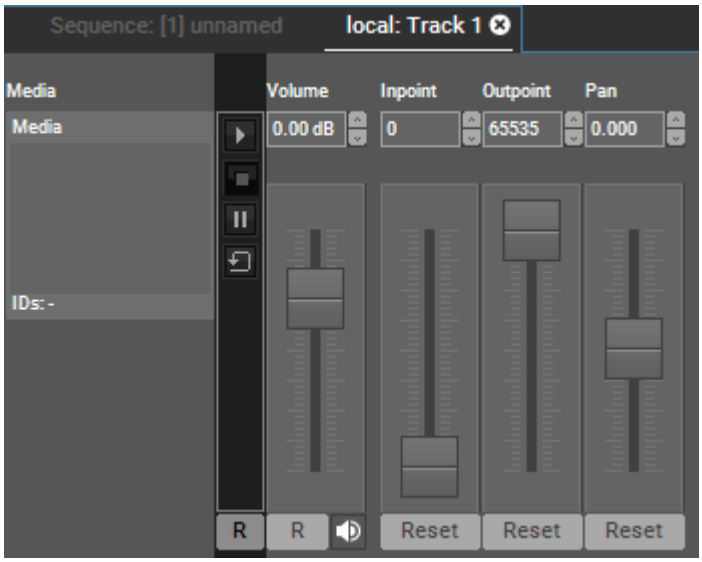
To assign an audio resource (see previous chapter [661]) to a track, drag and drop the resource from the project tab onto the designated track. You can also highlight the target track by left-clicking on it and use the right-click command on the resource in the project tab "assign to active device" or just double-click on the audio file. Media can also be assigned from the thumbnail browser by double clicking it for attributing it to the selected highlighted layers.

To remove the resource of the track, right-click in the layer tree on the media parameter and choose "reset" or right-click on the media thumbnail in the Device Control Tab.

External Control via DMX, Widget Designer, etc.

You can control the Media parameter externally. In general, media and mesh files can be assigned via their file names or by addressing them via the Folder and File ID. For DMX control, only the later is possible.
To assign a Folder and File ID, select the file (or folder) in the Project tab and look for the option in the File Inspector [191]. Both, the Folder and the File ID can be assigned with a value from 1-255.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.4.2 Playback Transport

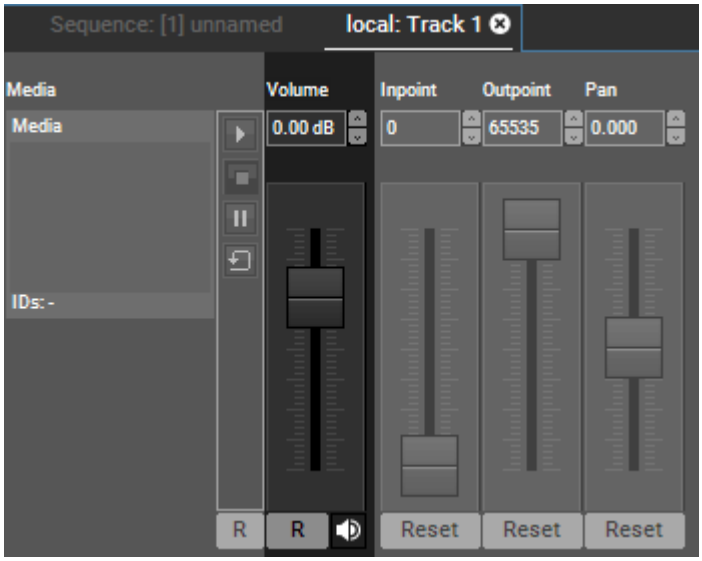


The Playback Transport channel allows setting the play mode of the audio file to Play Once, Stop, Pause and Play Loop.

External Control via DMX, Widget Designer, etc.

You can control all playback parameters externally. For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.4.3 Volume



With the Volume parameter the Volume of the audio files may be controlled. The Mute icon mutes the track. Note that the "Mute Audio" check box in the Track Inspector [217] is automatically ticked.

External Control via DMX, Widget Designer, etc.

You can control the Volume parameter externally. For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

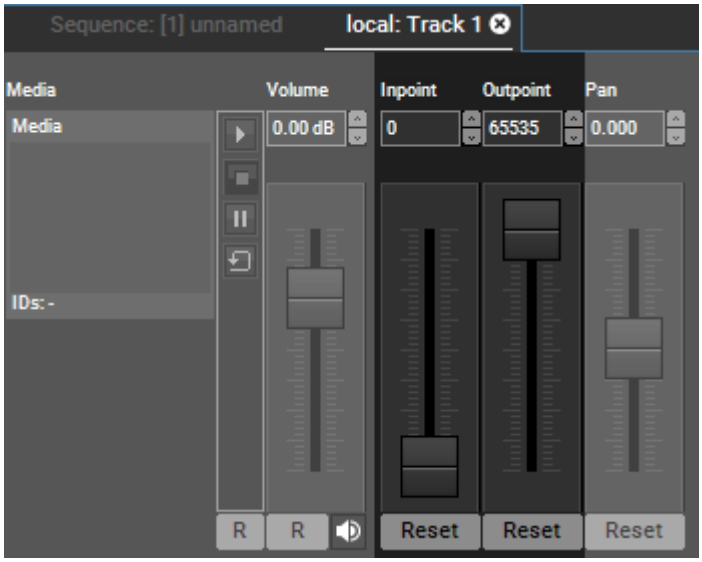
For Widget Designer or other controlling applications, the range of values is 0 - 2. The volume is measured in decibel and is based on a logarithmic function. To convert an external WD value to a decibel value in Pandoras Box or vice versa, please use this formula:

$10^{PB/20} = WD$ $\quad\quad$ $log_{10}(WD) \cdot 20 = PB$

$10^{6.02/20} = 10^{0.3} = 2.0$ $\quad\quad$ $log_{10}(2.0) \cdot 20 = 0.3 \cdot 20 = 6.02$

0	-96 dB
0.2	-13.98 dB
0.4	-7.96 dB
0.6	-4.44 dB
0.8	-1.94 dB
1.0	0 dB
1.2	1.58 dB
1.4	2.92 dB
1.6	4.08 dB
1.8	5.11 dB
2.0	6.02 dB

6.5.4.4 Inpoint / Outpoint

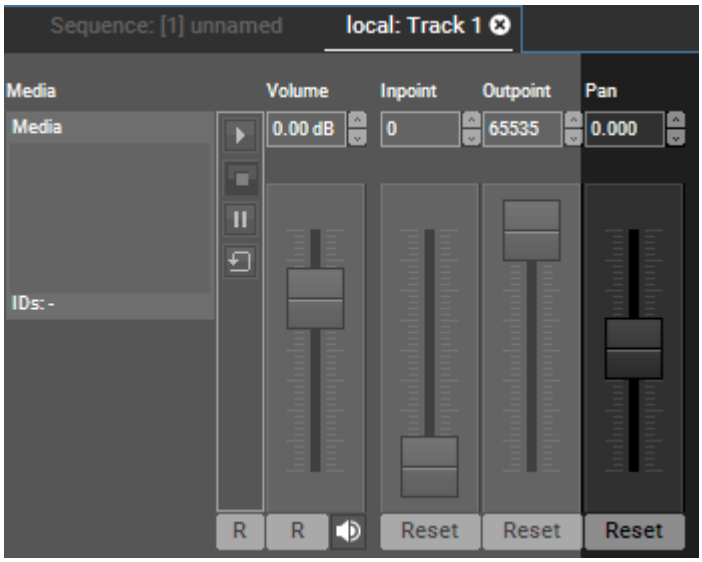


Set the in- and outpoint to define a specific start & end marks of an audio clip, the play once and play loop video mode will work in the resized area.
The In- & Outpoint selection works on a percentage base of the overall audio file length.

External Control via DMX, Widget Designer, etc.

You can control the In- and Outpoint parameters externally. For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.4.5 Pan

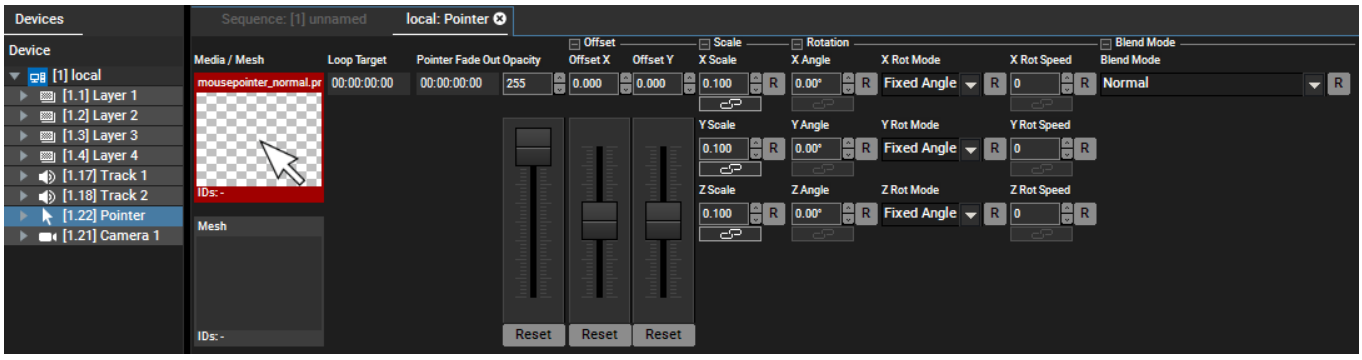


The Pan Control allows varying the relative levels of the two channels of a stereo source.
When not having the Pan Parameter modified, both channels of a stereo source will have the same level. Turning Pan to -100 will output the right channel only, turning Pan to 100 will output the left channel only. It might be of interest that you can store a parameter outside a container [299].
Note that the channels can be routed in the Audio Track Inspector [217]. If you like to use one channel only for playing mono files, you can also simply set the second channel to "0".

External Control via DMX, Widget Designer, etc.

You can control the Pan parameter externally. For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.5 Pointer Layer



A Pointer Layer displays the local mouse or touch input(s). Per default it is not shown in the Device Tree, please right-click on your Master or Client system and choose "Add Layers" > "Pointer Layer".
As the Pointer Layer has very similar functions to those of a Video Layer, this chapter focuses on the exclusive options. For information about all other sections (like position, rotation, scale…) please refer to the explanation of the Video Layer [647].

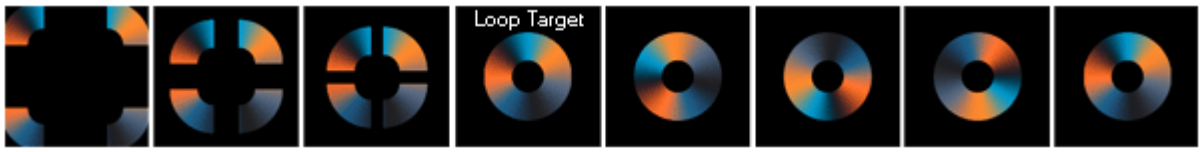
For general information how a Pointer Layer can be used, please see the chapter explaining the feature Layer Picking [252]. The number of inputs can be set up in the Pointer Inspector [213].

Loop Target and Pointer Fade Out

Both settings are more interesting when using a touch device instead of a device like a mouse. The difference between both devices is that a touch device only sends data when the device is in use, whilst a mouse device is always present.

The Loop Target can be understood as a playback command.
If you are using an always present input, you could use an image sequence or video and set the Loop Target to the starting frame.

If you are using working with touch inputs and your media file has a defined fade-in scene and then an looping scene, you can use the Loop Target Time to loop only the images AFTER the fade-in images. An example is depicted below. This means that the Pointer Layer "fades" in as soon as a touch input is received. And as long as the touch input stays present, the pointer loops starting from the Loop Target to the real end of the media file.



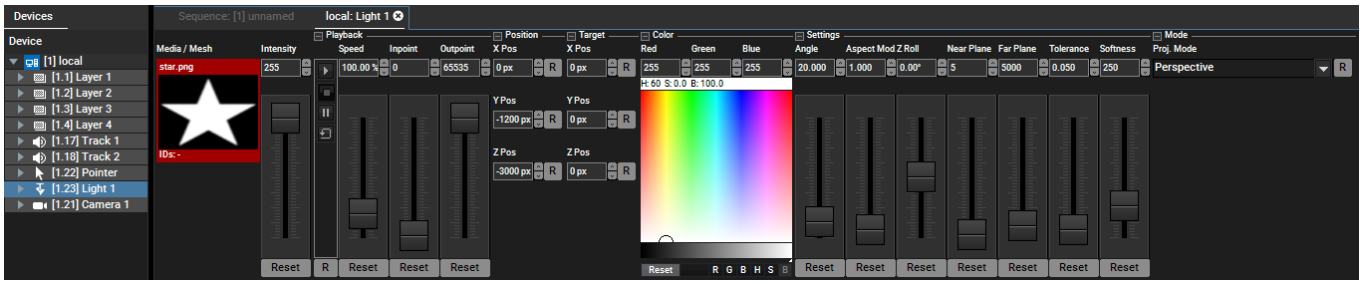
The Pointer Fade Out Time is the duration for fading the pointer out if the touch input is not present any more.

Both time can be entered in the format H:MM:SS:FF or shortened to SS:FF or even SFF. So, for example 3 seconds can be 0:00:03:00 or 03:00 or 300

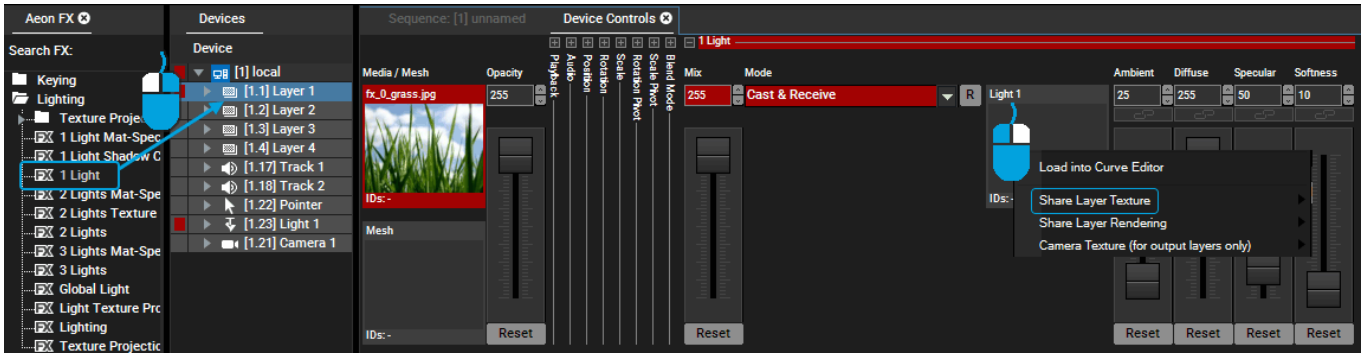
Offset X and Offset Y

Enter an offset (in generic units) for the X-axis and Y-axis that should be added to the real X and Y position of the input.

6.5.6 Light Layer



A Light Layer allows setting a light source to illuminate other Video Layers. With the help of a particular effect (Aeon FX tab [129] > Lighting [527] folder) another layer can "receive" this light and hence be illuminated. Without an lighting effect they will be rendered as always.
1a. Create a Light Layer (right-click on Site > Add Device > Add Light Layer)
1b. Assign a texture to the light, e.g. a white image from the Stock Assets.
2a. Assign a texture to a common Video Layer.
2b. Drag a Lighting FX from the Aeon FX tab onto this layer. Right-click into the effect's "Media" field, choose "Share Layer Texture" and select the respective light source. Now, the chosen light illuminates the Video Layer.

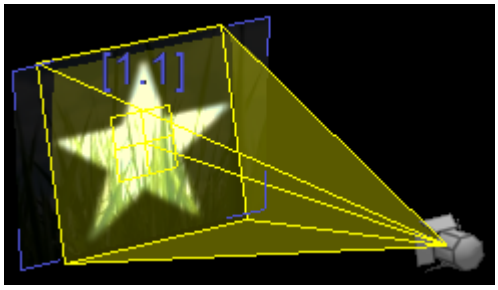


The Light Layer includes the following sections:
Media Selection [667]
Intensity [668]
Playback [649] (links to Video Layer because it is the same parameter)
Position [669]
Target [669]

Color [670]
Settings [670]
Projection Mode [676] (links to Camera Layer because it is the same parameter)

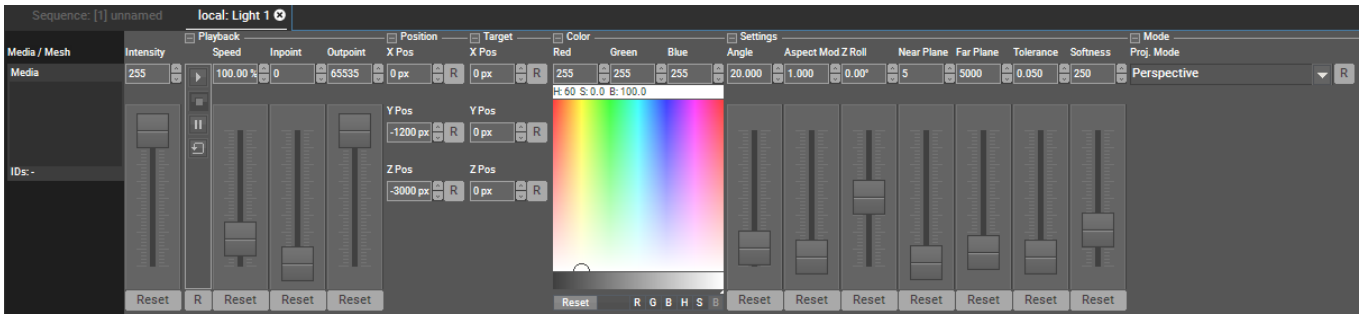
The Light Layer Inspector [214] might also be of interest.

Even though it is possible to add as many Light Layers as you wish,keep in mind that calculating light effects requires a lot of the system's performance. Depending on your settings and of course your hardware, three lights might be already the limit.



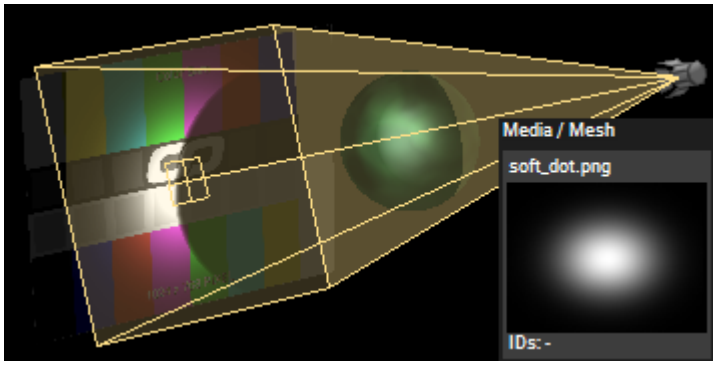
Per default, a light icon with a spanned light cone (so called frustum) is displayed in your Preview window. The style, e.g. the color of wireframes, can be changed in the Light Inspector [214]. The visualization can be toggled in the Configuration tab > Preview Display [145].

6.5.6.1 Media

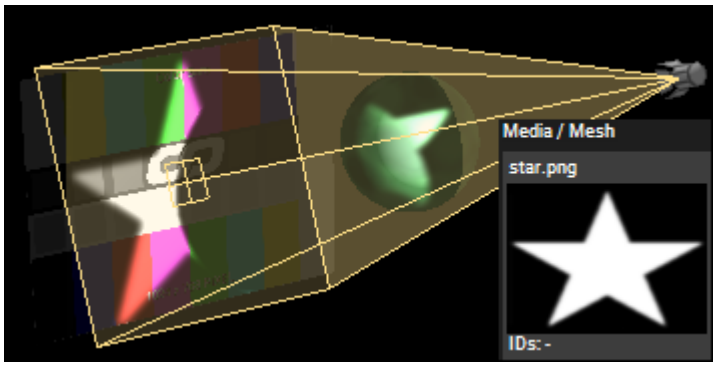


The Media parameter for a Light Layer is very similar to the one from a normal Video Layer. Please refer to this chapter [647] if you want to learn how to assign a media to a layer and what layer sharing means.
At present, it is not possible to use transparency (α channel) as see-through parts, for this simply use white color.

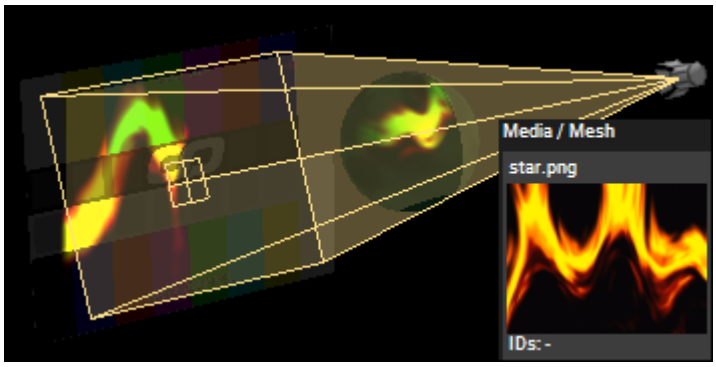
The media for a Light Layer acts as a mask or gobo or even as a transparent image in front of a light. In case of choosing a video, it turns the light into a projector.



The emitted light cone has the shape of a soft dot, the black parts are not visible.



The emitted light cone has the shape of a star.



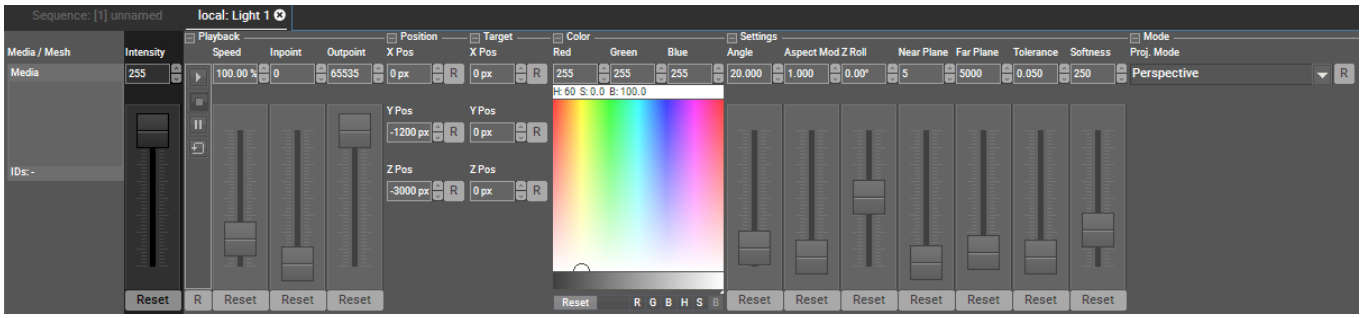
Emitted light interacts with the object's color. The reflected light depends on both colors!

Contrary to a Video Layer, you can not assign a mesh to a Light Layer.

External Control via DMX, Widget Designer, etc.

You can control the Media parameter externally. In general, media and mesh files can be assigned via their file names or by addressing them via the Folder and File ID. For DMX control, only the later is possible.
To assign a Folder and File ID, select the file (or folder) in the Project tab and look for the option in the File Inspector [191]. Both, the Folder and the File ID can be assigned with a value from 1-255.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.6.2 Intensity



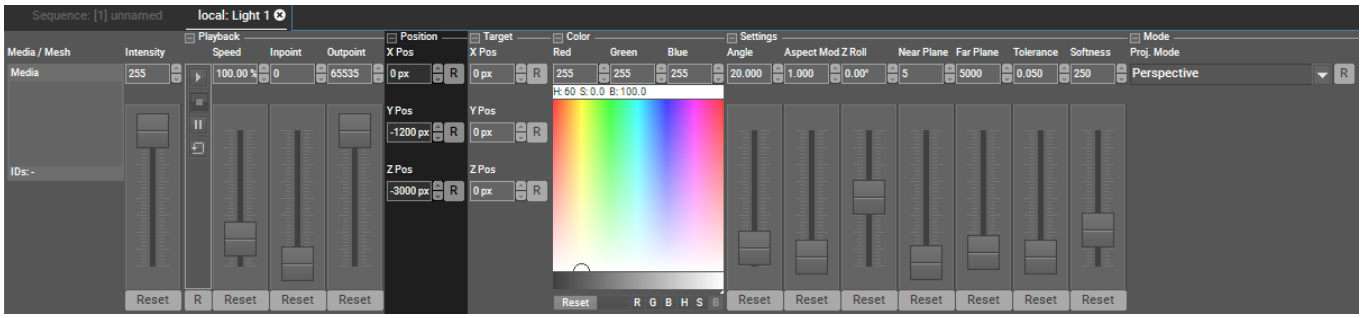
The Intensity parameter sets the intensity of a light source, i.e turning it on and off. Please keep in mind that layers receiving this light as a single light source will not be visible anymore.

Intensity 0% (value 0)	off
Intensity 100% (value 255)	on

External Control via DMX, Widget Designer, etc.

You can control the Intensity parameter externally, simply assign a value between 0-255.
For external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.6.3 Position



The Position parameters allow changing the position from where the light rays are emitted originally in the 3D space.

Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

When selecting multiple Light layers, you can use the Align function [124] in the Status Bar. Instead of applying the same value to all selected layers, "Align" follows the selected pattern and applies e.g. mirrored values.

Lights in Pandoras Box have no attenuation, which means that their intensity is not a function of the distance from a light source to an object's surface. Thus, the position does only affect the angle of incidence, i.e. how steep the light shines on an object. The second parameter influencing the angle is the target position [669]. The affect of the angle of incidence is described here [527] in detail.

External Control via DMX, Widget Designer, etc.

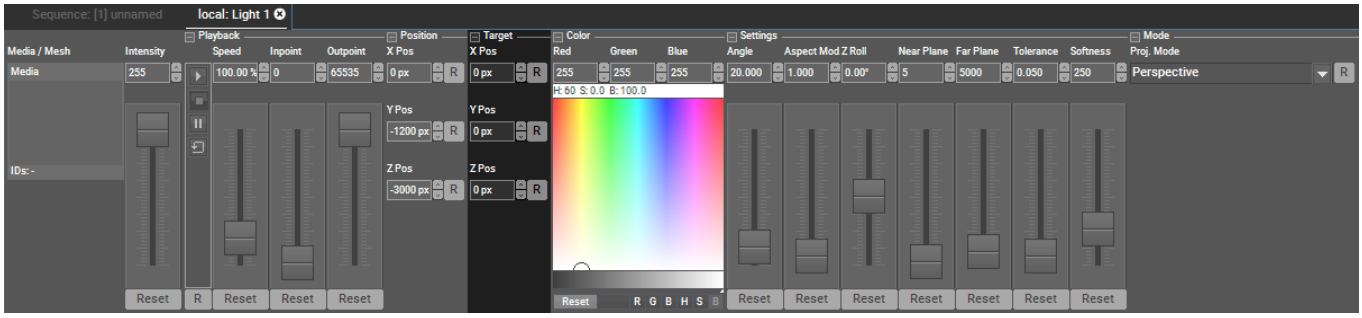
You can control the position parameters externally.

Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.6.4 Target



The Target position parameters allow changing the position to where the light rays are emitted in the 3D space without moving the light itself.

Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

Lights in Pandoras Box have no attenuation, which means that their intensity is not a function of the distance from a light source to an object's surface but is always equally intense. Thus, the target position does only affect the angle of incidence, i.e. how steep the light shines on an object. The second parameter influencing the angle is the source position [669]. The affect of the angle of incidence is described here [527] in detail.

External Control via DMX, Widget Designer, etc.

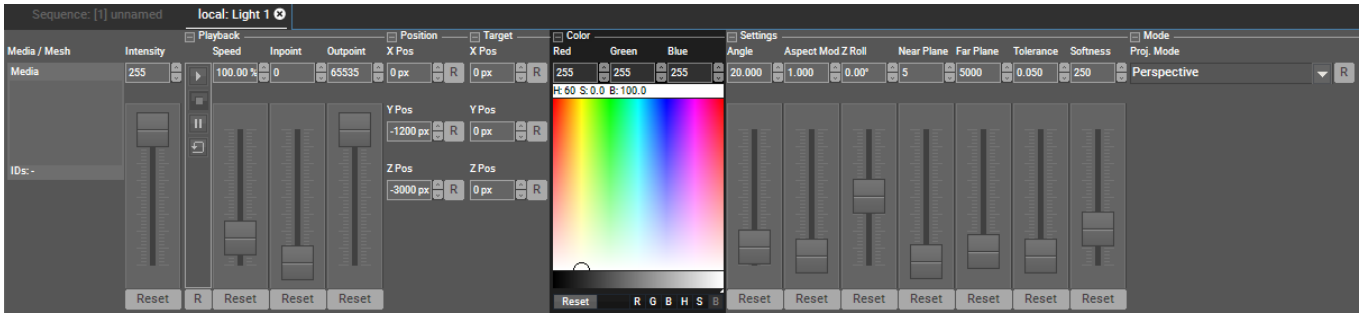
You can control the target parameters externally.

Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.6.5 Color

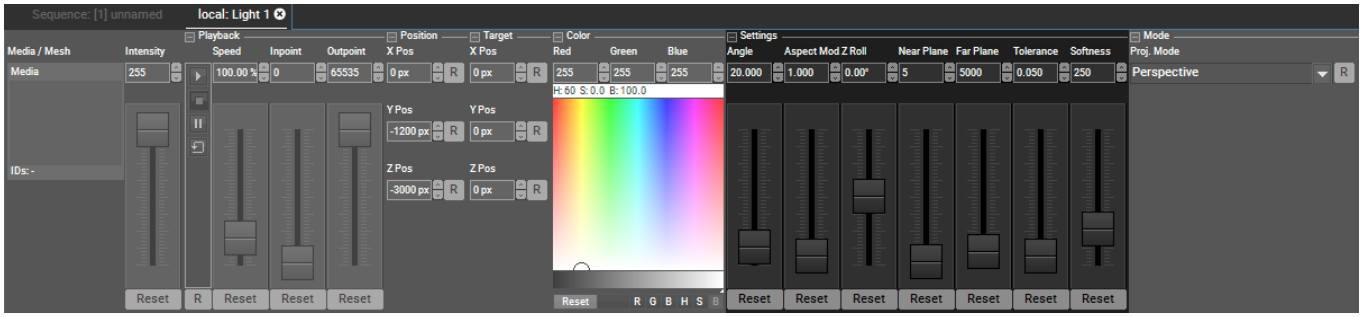


The Color parameter allow changing the color of the emitted light rays. Please refer to this chapter [323] if you want to learn how to use the color picker.

External Control via DMX, Widget Designer, etc.

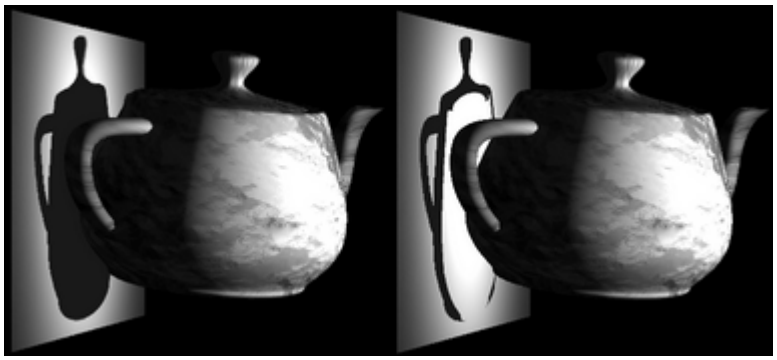
You can control the color parameters externally, simply assign a value between 0-255 to each color.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.6.6 Settings



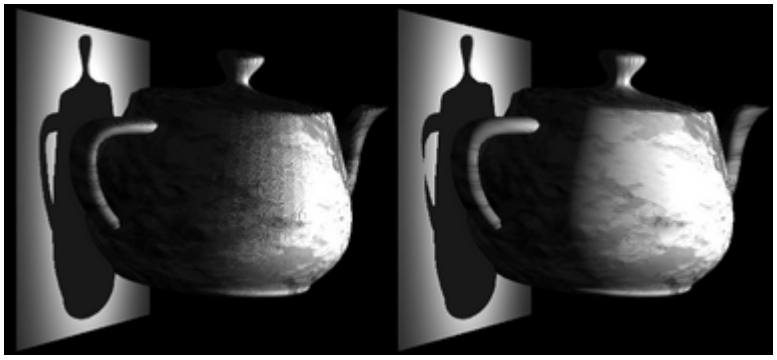
Next to the parameters Position and Target the Settings parameter section allows adjusting the area in your 3D space that will be affected by light. Layers that are not within the spanned light cone cannot be shed with light, cast shadows or be shadowed.

"Angle" resizes the dihedral angle of the light and spanns the light cone. By default it is squarish, that is the height equals the width. "Aspect Mode" influences the width-height-ration, whereas "Z Roll" rotates the entire light cone. Contrary to the next mentioned parameters all these parameters can be seen in the Preview window if the Light Wireframes [666] are toggled on.



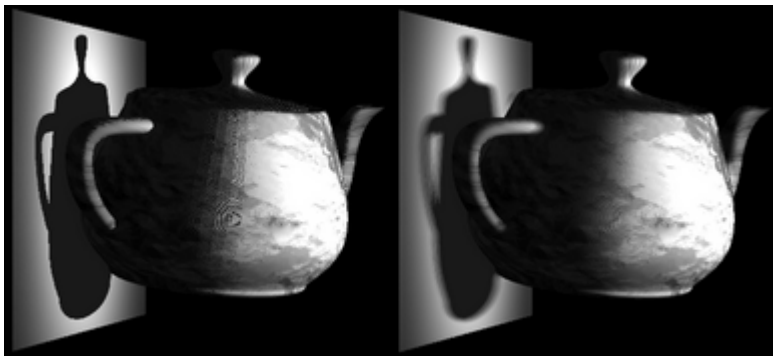
"Near Plane" and "Far Plane" have their origin in the Camera Control [675]. When looking at the distance from the light source itself, the near plane describes the minimum distance an object is allowed to have in order to cast shadows; the far plane refers to the maximum distance. In the example the near plane starts behind the front part of the jug. Even though it is dipped in light it is not casting shadow any more. Note that the light simply cuts the shadows of the front part of the jug, it is not rendering them any more. It cannot see through the first layers of the object and render the shadow of the back side instead.

Note that the planes are not designed to be an artistic tool. They are rather meant to be used if the depth resolution is too small and objects being to close to each other are rendered in a wrong way. If you have such an issue decrease the distance between both planes without cutting objects.



A higher light "Tolerance" decreases the shadowed area, either on the object itself or another object. If a particular area lies within the light-shadow-border it now tends to be on the illumined side.
The tolerance level helps to decrease the phenomenon called self-shadowing which can be especially problematic when using objects with a very rough surface.

In the example the tolerance level is set to 0.005 for the left image, and to 0.990 for the right image. Note that 0.990 is so high that even the shadow of the handle is influenced.



A higher shadow "Softness" blurs the shadow border, on the object itself and on another object. It can be especially helpful when working with a Shadow Map [214] set to a low resolution .

In the example the tolerance level mentioned above is set back to the default value of 0.10. The softness level is set to 0 for the left image, and to 900 for the right image. Note the difference to the image above. There, the light-shadow-border has moved whereas here, the border remains but the transition is softer.

External Control via DMX, Widget Designer, etc.

You can control all Settings parameters externally.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.7 Notch Layer

This chapter explains the Notch Layer in detail. The introductory chapter gives an overview on all Pandoras Box devices and layers.

General Information about Notch

Notch is a real-time content creation tool that brings a combined work flow for creatives and technologists. Notch enables artists and designers to create high-end 3D graphics, live video effects and particles, interactive experiences, virtual production and a lot more in a true real-time environment. The deep and comprehensive

integration to Pandoras Box gives you frame accurate playback and full parameter control with zero latency which allows you to edit content directly on stage and over-night renders belong to the past.

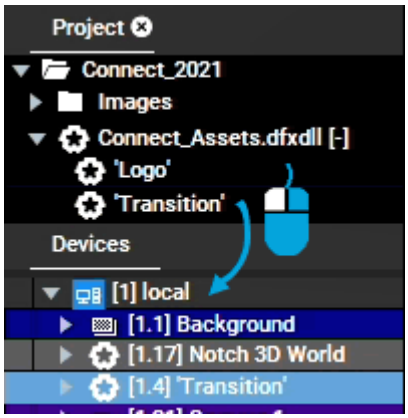
In the Notch Builder you first design your scene and then decide which parameters should be exposed in the so called Notch block that will be loaded in Pandoras Box. The exposed parameters can be everything from text, amount, color, to size or position etc. If you program items to the Notch timeline, they will be synchronized to the PB timeline when storing them as a container.

The Notch Layer in Pandoras Box can render and control Notch Layers that come with the Notch Block.

In order to work with Notch, Pandoras Box requires a Notch license. Indeed, all PB systems that should render Notch content need to be equipped with a license. As soon as the Pandoras Box Master finds a PB system with a license, it can import and analyze Notch Blocks. The Master needs its own license only, if it should render Notch content itself, either as a fullscreen output or in the Preview window.

Further, as Notch renderings can demand much GPU performance, we recommend to equip the PB system with the Notch license with an RTX graphics cards.

How to Import Notch Blocks and Create Notch Layers



In Pandoras Box, first select the Notch block, the dfxdll file, in the Assets tab and drag it into the Project tab. The first loading process takes some time. Depending on your Notch composition this can be a few seconds or even minutes when using many textures etc. When finished loading, you can open the Notch entry in the Project tab and see its sub elements.

In case you activated the Notch option "Layer as separate effects" all Notch layers are shown as sub resources in Pandoras Box and can translate again into individual layers as done in the next step. In case you leave this option deactivated, your Notch scene is exported as a baked scene and PB shows only one sub resource.

In any case, drag a sub resource, onto a Site(!) in the Device Tree and it will show a newly created Notch Layer. In the depicted example, the sub entry "Transition" is added to the Site "Local" which transformed it to a Notch Layer called "Transition". As explained above, this Site needs to be equipped with a Notch license to be able to render the content.

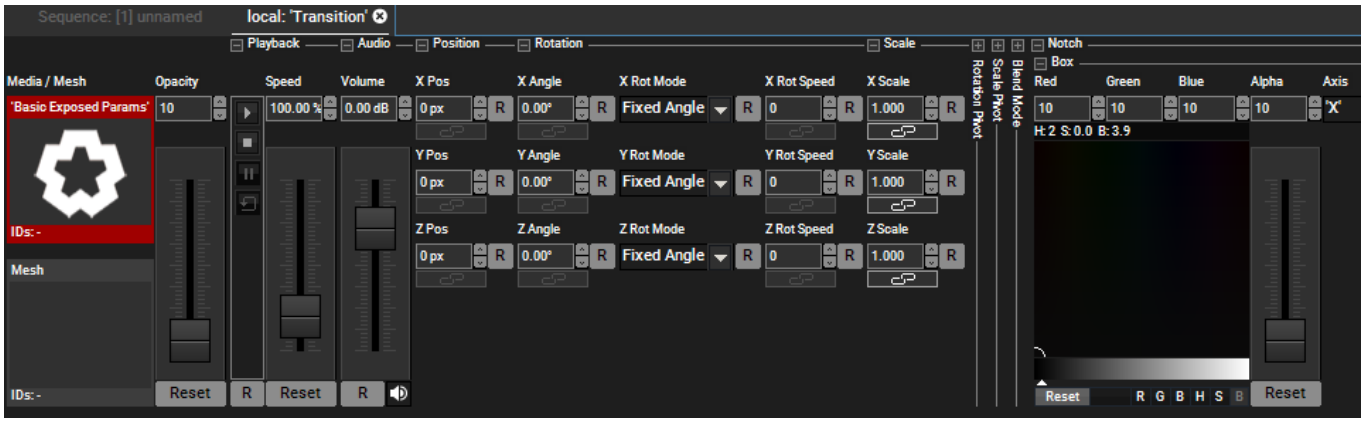
By the way, all content is spread by default. If a Client has no Notch license, it cannot render the content, hence it appears as inconsistent in the Assets tab (red exclamation mark). To avoid this, remove either some locations using the File location table [193] or deactivate automatic spreading [155] in the first place. Spread the content manually via the right-click menu in the Project tab, or use folders that spread to specific locations only (see Folder Inspector [197]).

As an alternative workflow to add Notch Layers, you could also right-click a Site and choose Add Device > Notch Layer > loaded Notch block > custom Notch layer. This is shown in the following image. In this case however, the media parameter is not assigned automatically with the Notch content as in the first workflow. This means that you need to assign the Notch content from the Project tab to this layer manually.

The Notch Layer is rendered in the same rendering order as other layers. In the Device Tree, you can drag devices further down if they should overlay other layers (see "Site, Layer and Parameter Structure") [175]. For more information about the [173] rendering passes, see the chapter "Video Processing Pipeline" [320].

Notch Parameters

Select the new Notch Layer and open the Device Control tab [171] look at its parameters.

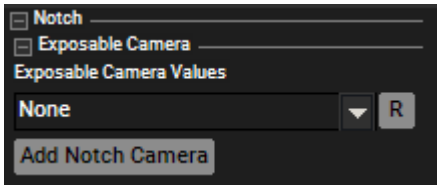


As mentioned above, you will notice that the media parameter might be already active. The following parameters, Media - Mesh - Opacity - ... - Scale - Blend Mode - are the same as the ones from a Video Layer. For more information about them, please see the chapter "Video Layer [647]". By the way, you could also add effects to a Notch Layer, just as you would do with a Video Layer. The chapter "Working with Effects [322]" explains all you need to know about this.

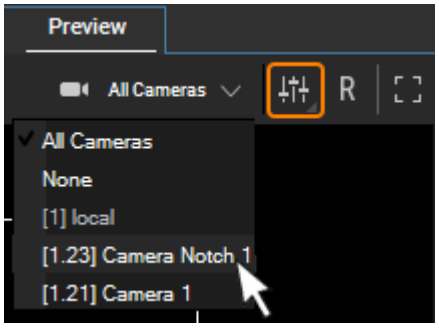
The last controls, those in the "Notch" group, are the parameters which were exposed in the Notch Builder. You can work with them as with any other PB parameter, that is, change them to an active value or store them as key frames to the timeline. If you are new to Pandoras Box, the chapter "Sequence" [292] describes how to do that.

Now, toggle back to the Sequence tab and execute the "Store Active [293]" command. On the track from the Notch Layer you will see a new Container and keys for all parameters you changed so far. When you play it back, this could look like a 2D video or transition or like an entire 3D scene with camera animations and particles reacting on content. Of course you can blend other content and live feeds with the Notch content. Or, go a step further and combine both by assigning PB content to an exposed Notch Media parameter. The new Blend Modes or LUT shaders that come with PB version 8 are very helpful for that as well as the new Share Layer Option. The "What's New" chapter gives a good overview about these features [26].

Notch Camera



In case your Notch project contains an exposed camera with exposed parameters, Pandoras Box will include them as parameters of a Notch Layer. As seen in the left image, Pandoras Box 8.3.0 shows a drop-down list and a button "Add Notch Camera". After clicking it, the drop-down becomes active and shows "Camera Notch 1" which means that the Notch scene is now linked this Camera. At the same time, the Device Tree shows a new layer named "Camera Notch 1". Select it to see it in the Device Control tab [171]. Please do not change any values yet...

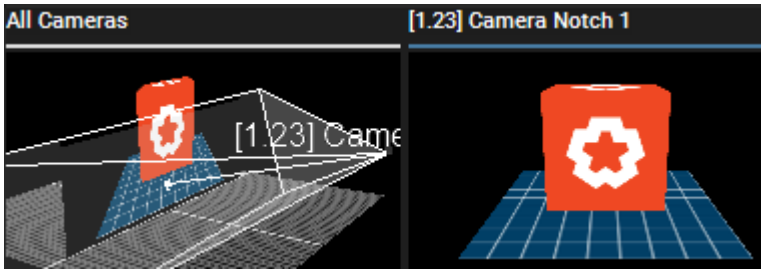


The best practice to work with a Notch Camera is to select it in the Preview [243]. Simply open the "Select Preview" drop-down and select "Camera Notch 1". Note that the "Camera Interaction Mode" switched to the Fader icon automatically in order to influence the values in the Device Control tab.

If you now zoom, pan or rotate in the Preview [248] the Device Control tab shows active Viewpoint and Target parameters for the Notch Camera. If the default "Target" parameters are far off the origin, it might be more convenient to set them first to (0,0,0) because Pandoras Box rotates around that point.

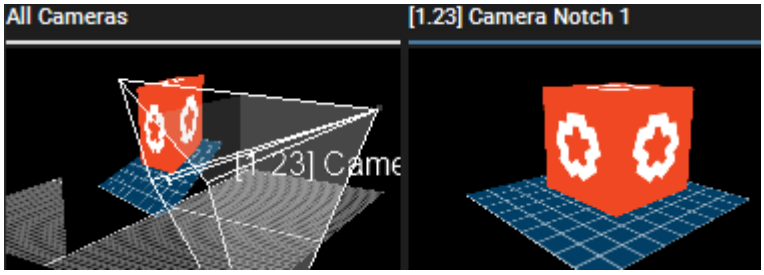
In general the parameters and parameter ranges for a Notch Camera match those of a Pandoras Box Camera. For more information about them, please see the chapter "Camera Layer [675]". However, the default values differ as they depend on the values saved in the Notch block. Further, a Notch Camera can not be assigned to a Pandoras Box Output or share its texture anywhere else.

In order to fully understand what is happening with the Notch Layer when changing values of a Notch Camera please set your Preview to show two viewports [246]: the left should show the "All Cameras" view and the right one the Notch Camera, just as depicted below.



In this example, the Notch scene shows an orange cube on top of a blue grid. The Notch Camera was set to capture the scene from a front position and some height.

As seen in the right Preview image, the cube is shown from the correct angle as expected. The same image, i.e. the rendering result of the Notch Camera, is now also shown as a 2D texture on the Notch Layer. In the "All Cameras" view it might look confusing at first because it looks like the Notch Camera looks onto the Notch Layer. But indeed, the camera wireframes depict the Notch Camera and the Notch Layer shows the resulting texture.



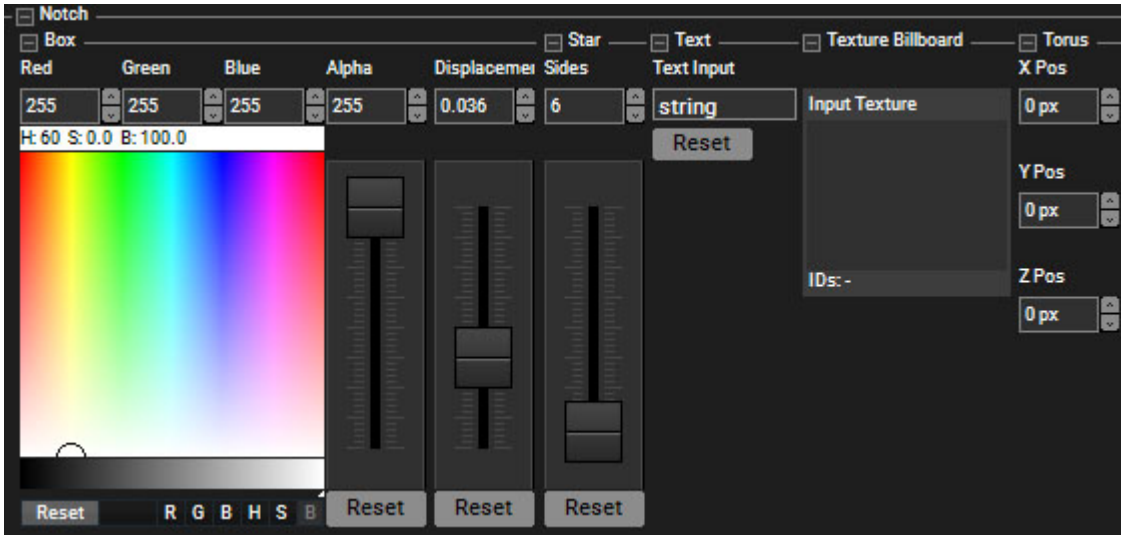
Now, the Notch Camera also moved to the side and the expected result of the movement is shown in the right side of the Preview. Again, the same result is depicted by the two-dimensional Notch Layer in the "All Cameras" view on the left side where the wireframes show the new Camera position.

By the way, you can link multiple Notch Layers to the same Notch Camera. Vice versa, you can program various Notch Cameras to the timeline and switch between them by choosing the according Camera in the Notch Layer.

As an alternative workflow to add Notch Cameras, you could also right-click a Site and choose Add Device > Notch Layer > Notch Camera. This is shown in the following image. To link this camera to a Notch scene, simply select the Camera in the drop-down list for the camera values of the Notch Layer. At first, the default values of the Notch Camera resemble those of a PB Camera. However, you can also load those values stored in a Notch block by right-clicking the Camera, choosing "Apply Defaults from Exposed Camera" and then the according Notch Layer.

Controlling Notch Layers with Widget Designer / via the SDK

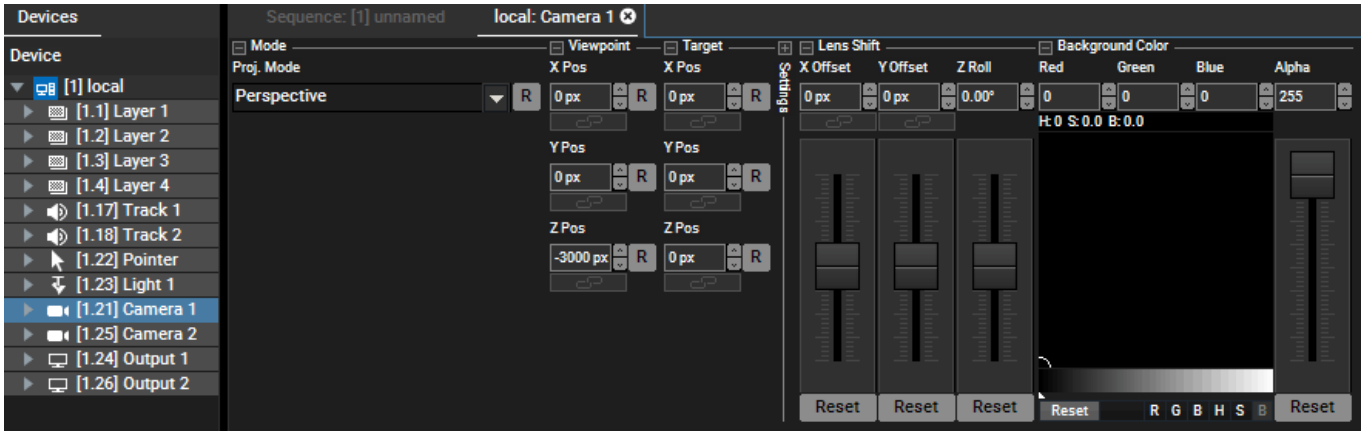
Of course, you can also control the parameters of a Notch Layer using the SDK or Widget Designer [786], e.g. through interactive devices like the AirScan [1988] or sensors like Phidgets [1383] and more. This text in the Widget Designer chapter [49] summarizes which commands or nodes are helpful when controlling a Notch Layer. Remote controlling a Notch parameter is very similar to controlling an additional effect, which means that you need to know the exact parameter name. The name is case-sensitive and all spaces count. For various reasons, Notch parameters are grouped and this hierarchy structure needs to be represented in the parameter name by using the pipe "|" character. The groups are shown in the Device Control tab but also in the Device Tree when unfolding entries. The Device Tree is helpful too, if a name is too long to be displayed in the Device Controls tab. The following list shows some parameters from the following picture.



Notch|Box|Red
Notch|Box|Alpha
Notch|Box|Displacement
Notch|Star|Sides
Notch|Text|Text Input
Notch|Texture Billboard|Input Texture
Notch|Torus|X Pos

Note that position values are displayed in Pandoras Box in pixels but the input via the SDK has a different range per default. The chapter Configuration tab > Unit Management [160] explains more about this and the check box "Interpret Automation Param Input as Pixel Values".

6.5.8 Camera Layer



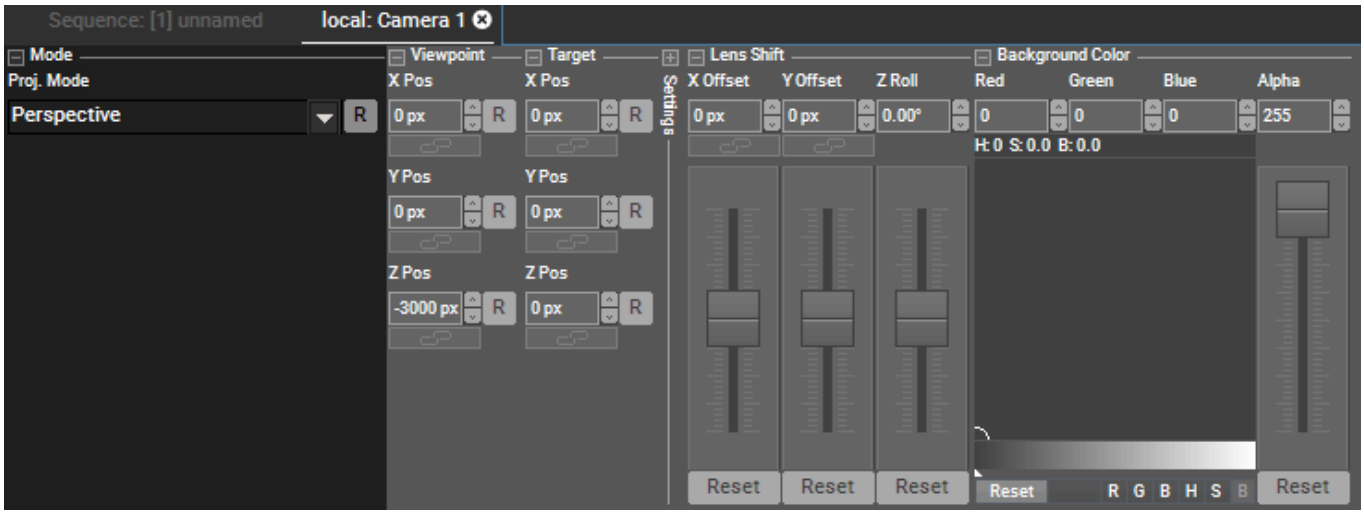
The Camera device allows setting up the 3D look-at-point onto your 3D composition of other layers. Since version 8 you can add an unlimited number of Camera Layers to a Site. To add one, simply right-click the Site in the Device Tree [173] and choose "Add Device".

With the layer controls you can determine whether each Camera should display the same or different areas in 3D space. This allows stretching layers over several Outputs or to create softedge blended setups. More information about the rendering order, video processing pipeline and how a Camera is linked to an Output Device, can be found in the chapters "Video Processing Pipeline [320]" and "Output Layer" [682]. A Camera Layer can only be removed from the Device Tree if it is not associated with an Output Layer.

The Camera chapter is divided into the following sections. Information about DMX Control is included, addressing customers who wish to remote control a layer with a lighting desk via DMX or Art-Net. If you like to use the Widget Designer or another application instead, please refer to this parameter list [1518].
Projection Mode [676]
View Point [677]
Target [678]

Settings 680
Lens Shift 680
Background Color 681

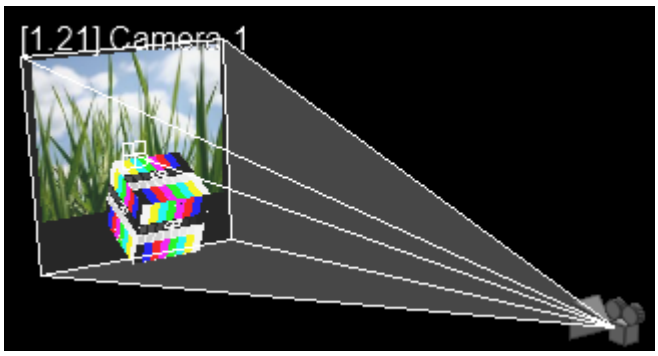
6.5.8.1 Projection Mode



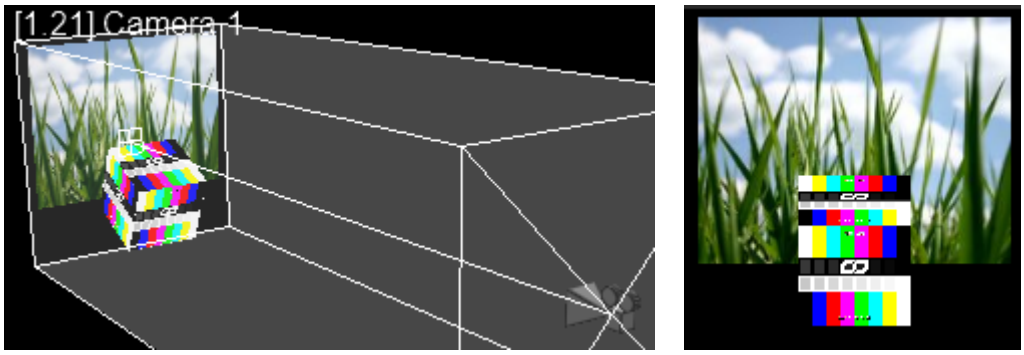
There are two projection modes available for each camera device:
- Perspective Mode (by default) and
- Orthogonal Mode.

The perspective mode allows a perspective angle to your view from different 3D scene.
The orthogonal mode switches off the depth and perspective view of the camera. The Z position of any layer will only affect the rendering order but won't make any visual change to the layer. The control channels do still access the three dimensional orientation of the 3D camera.

This is an example of how the projection mode affects the rendering. The scene pictures a background layer with a tilted cube in front of it. In the perspective mode, the edge of the cube that is the farthest from the camera position is rendered smaller then the edge nearer to the camera.



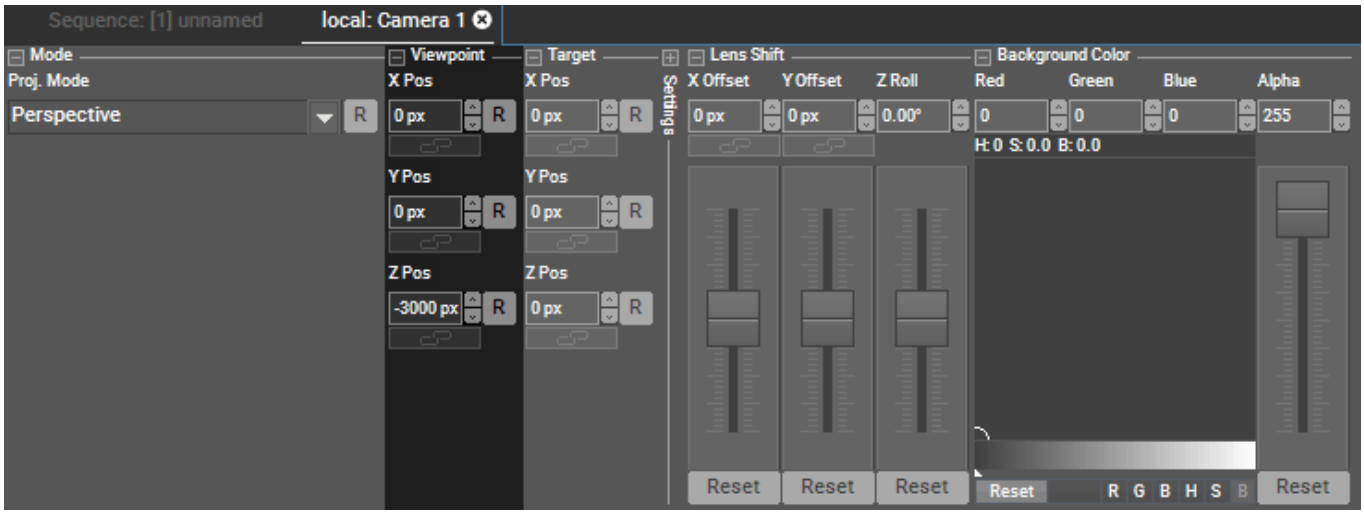
Now, the camera is switched to orthogonal mode. The 3D scene stays the same, but the camera wireframes change accordingly. In orthogonal mode there is no distortion due to perspective and all edges have the same size.



External Control via DMX, Widget Designer, etc.

You can control the Projection Mode parameter externally, simply assign the value 0 for "Perspective" and 1 for "Orthogonal".
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.8.2 Viewpoint

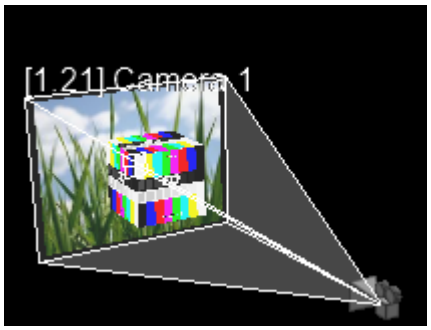


The Viewpoint position parameters allow changing the camera's position in the 3D space without affecting its target. In other words, you are changing *from where* the camera looks, not *whereto*. The camera's target position is changed with the Target [678] parameters. The output will show the 3D space from a different viewpoint and the perspective will be affected as well (as long as you do not work in Orthogonal Projection Mode [676]).

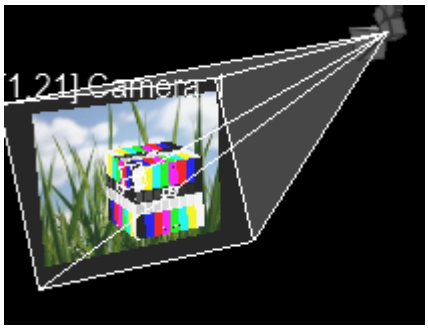
Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

When selecting multiple Camera devices, you can use the Align function [124] in the Status Bar. Instead of applying the same value to all selected layers, "Align" follows the selected pattern and applies e.g. mirrored values.

This is an example of how the camera's viewpoint affects the rendering. The scene pictures a background layer with a cube in front of it. At first, the camera's viewpoint is aligned with the center of the layer and cube.



Now, the "Y Pos" is changed. The camera renders the scene from a higher point of view.



External Control via DMX, Widget Designer, etc.

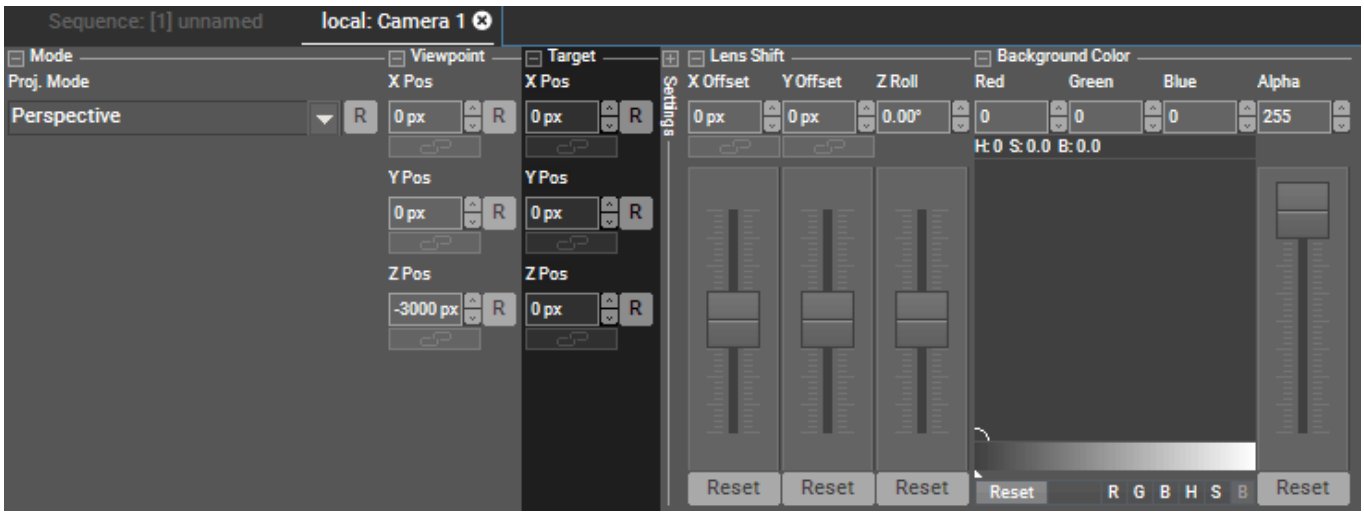
You can control the Viewpoint position parameters externally.

Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.8.3 Target



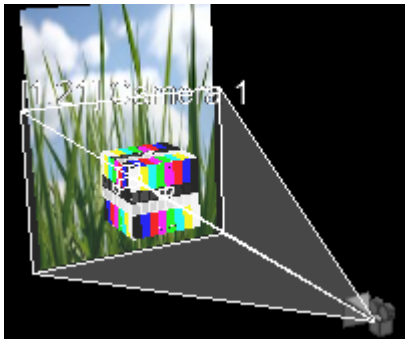
The Target parameters allow changing the camera's target position in the 3D space without moving the camera itself. In other words, you are changing *whereto* the camera looks, not *from where*. The camera's position is

changed with the Viewpoint [677] parameters. The output will show another detail of the 3D space and the perspective will be affected as well (as long as you do not work in Orthogonal Projection Mode [676]).

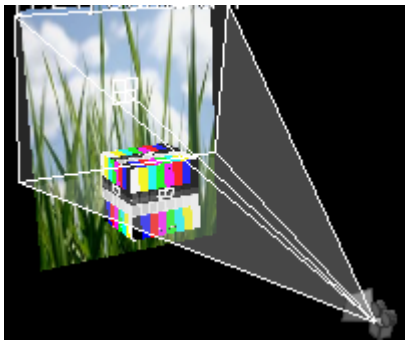
Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

When selecting multiple Camera devices, you can use the Align function [124] in the Status Bar. Instead of applying the same value to all selected layers, "Align" follows the selected pattern and applies e.g. mirrored values.

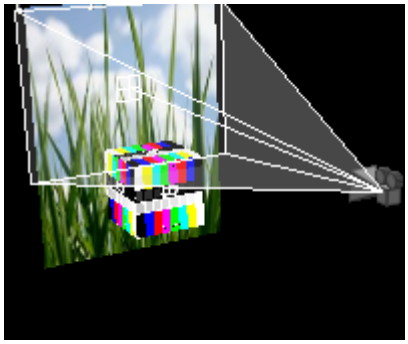
This is an example of how the camera's target position affects the rendering. The scene pictures a background layer with a cube in front of it. At first, the camera's target position is aligned with the center of the cube. The output shows the lower part of the background layer.



Now, the "Y Pos" of the target is changed. The camera renders a higher part of the scene. In the output, the edges of the background layer are not straight anymore which is a result of the different perspective...



For the last example, the camera's Viewpoint position is moved to the same position as its target. Now, In the output, the edges of the background layer are straight again.



External Control via DMX, Widget Designer, etc.

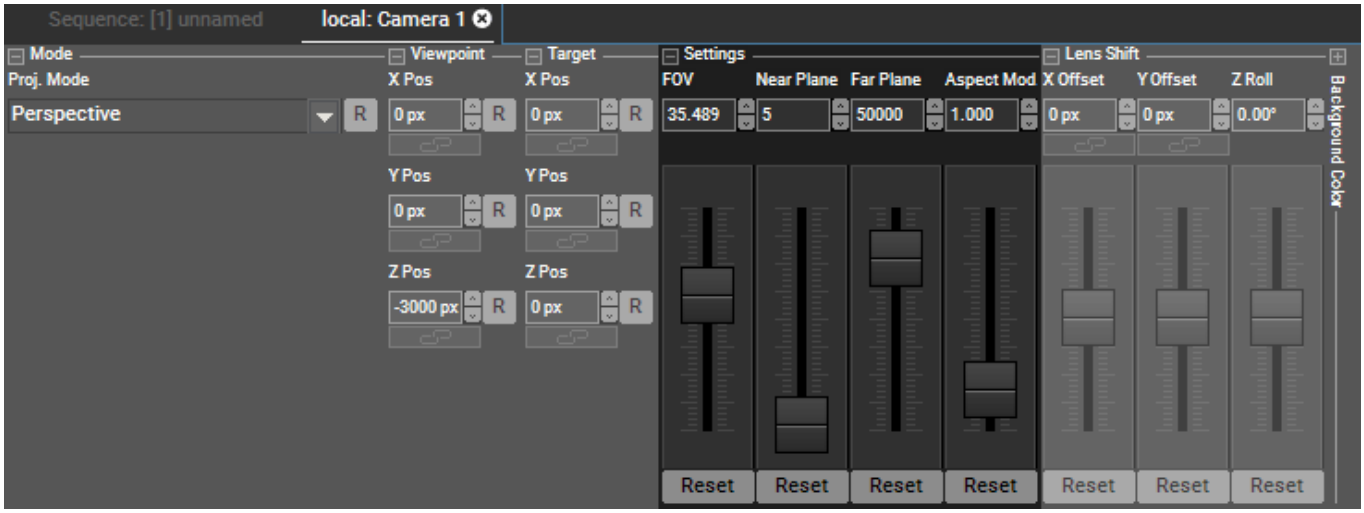
You can control the Target position parameters externally.

Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.8.4 Settings

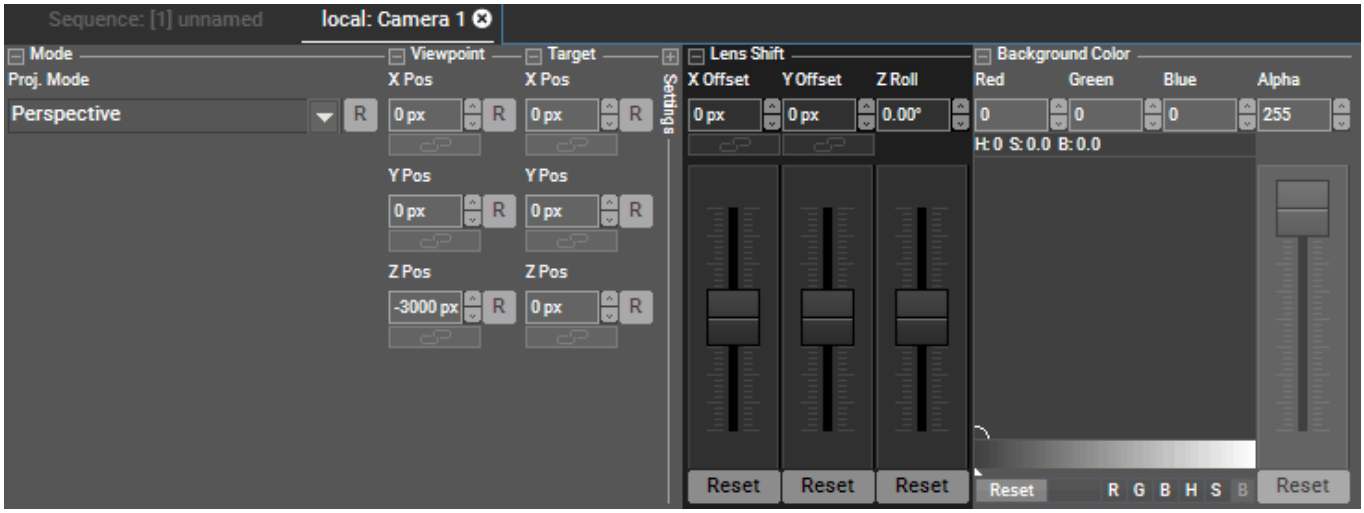


The Settings section allows adjusting the Camera's Field of View (in degrees), the area in your 3D space that will be rendered (Near and Far Plane) and the Aspect Mode of the camera.

External Control via DMX, Widget Designer, etc.

You can control all Settings parameters externally.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.8.5 Lens Shift



The Lens Shift section allows adjusting X and Y Offset as well as the Z Roll.

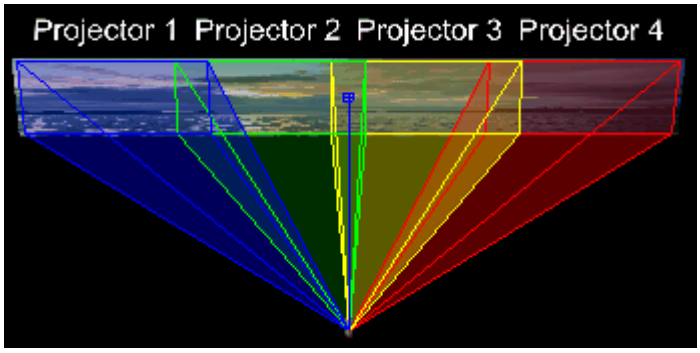
Per default, position values (incl. Offset parameters) are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how

the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

When selecting multiple Camera devices, you can use the Align function [124] in the Status Bar. Instead of applying the same value to all selected layers, "Align" follows the selected pattern and applies e.g. mirrored values.

X&Y Offset

The camera X&Y offset allows positioning the center of the cameras perspective. This is especially used when one perspective view needs to be shared across several outputs. For example, if you have 4 screens set up next to each other, you would offset each screen in the way that you can move your layers with the XYZ position smoothly across all outputs, while maintaining a correct perspective 3D view.



Z Roll

External Control via DMX, Widget Designer, etc.

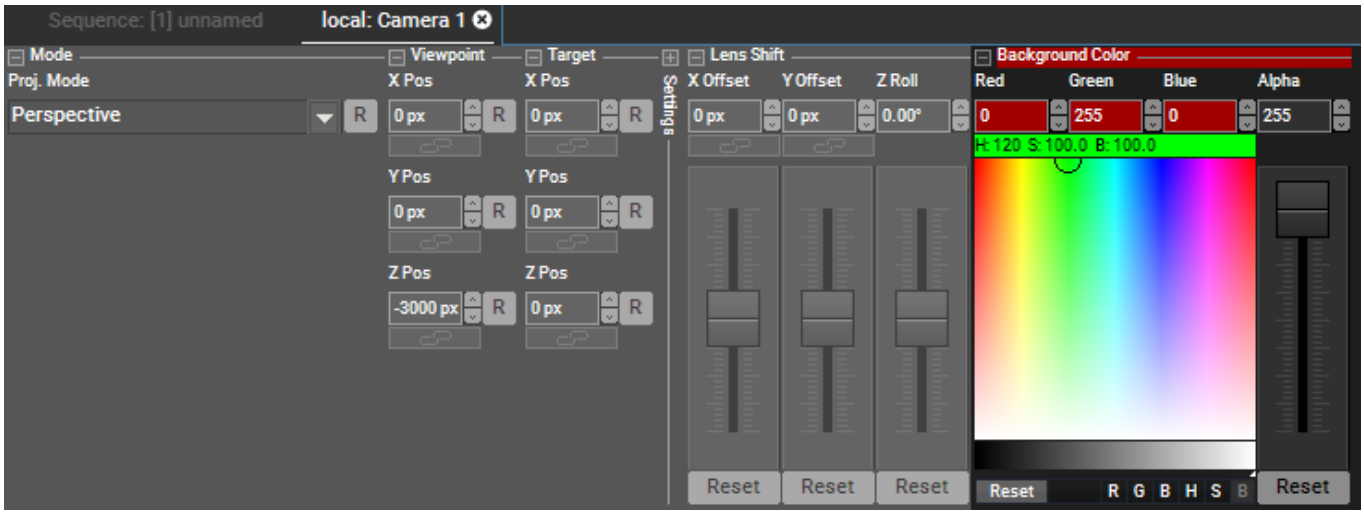
You can control all Lens Shift parameters externally.

Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.8.6 Background Color



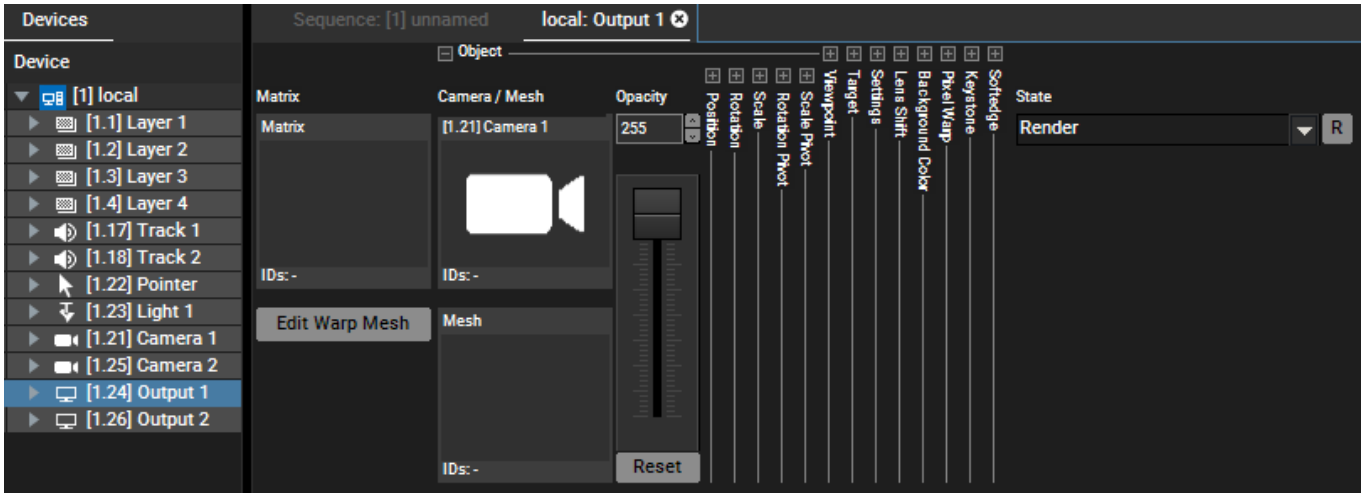
The Background Color section in the Camera Device allows adjusting the red / green / blue / alpha channels for the main background from the resulting texture rendered by the camera (the so called render target). More information about the render target can be found in the chapter Video Processing Pipeline [320], a possible application for adjusting the background color could be blacklevel compensation [646] when working with multi-softedged projection(s).

If you are not familiar how to use the color picker tool, please read the topic "Adding, Editing and Removing FX [323] ".

External Control via DMX, Widget Designer, etc.

You can control the color parameters externally, simply assign a value between 0-255 to each color or alpha parameter.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.9 Output Layer



The parameters of the Output Device controls the overall output of your Client. Here you may add a matrix file for LED walls or a 3D / Warping object if projecting on bended surfaces. Keystone and softedge settings are also included. In addition, you may add effects [322], for example if you aim for a color correction of the whole output.

As described in more detail in the chapter "Video Processing Pipeline" [320], an Output Layer is linked to a Camera Layer which means that it processes its render target, i.e. what the Camera "sees" from your 3D layer composition. The Camera is the first render path and the Output the second one. Per default, an Output is linked to the Camera with the same number, e.g. Output 1 to Camera1. If you add more Outputs than Cameras exist, a new Camera is added and linked to the new Output automatically. If you remove an Output, the linked Camera is not removed automatically.

The number of Output Layers which you can add to a Client (via right-click in the Device Tree [173]) is linked to the number of available "Pandoras Box Software Licenses". Multiple licenses can be stacked by combining dongles or they can be applied to a single dongle. If your Site in a (pre-programmed) project contains four Output Layers, it can only manifest on a PB system with four licenses. If it holds only three for example, a dialog offers to remove one Output Laye+r.

Per default, the local Site (i.e. the Master) starts with a Camera Layer but no Output. In that configuration, it can start a Multi-User session or join one. As soon as you you add Output Layers, you can use them in fullscreen mode but you cannot participate as a Multi-User.

The Output chapter is divided into the following sections. Information about DMX Control is included, addressing customers who wish to remote control a layer with a lighting desk via DMX or Art-Net. If you like to use the Widget Designer or another application instead, please refer to this parameter list [1519].

Matrix [683]
"Edit Warp Mesh" Button - either selects the Output Mesh or creates a new one for you and enters the Mesh

Editing mode [257]
Object [684]
Viewpoint [677] *
Target [678] *
Settings [680] *
Lens Shift [680] *
Background Color [686]
Pixel Warp [686]
Keystone [687]
Softedge [688]
Render State [689]

* Typically the marked parameters are not used in 99% of all cases. Advanced users only! The parameters work in the same way as described in the linked chapters about the Camera Device only that they affect the camera texture, i.e. how the Output Device which can be understood as a second camera "sees" the Camera Texture. For softedge projections etc., please use the Camera parameters to adjust which part of the 3D space each output will show!

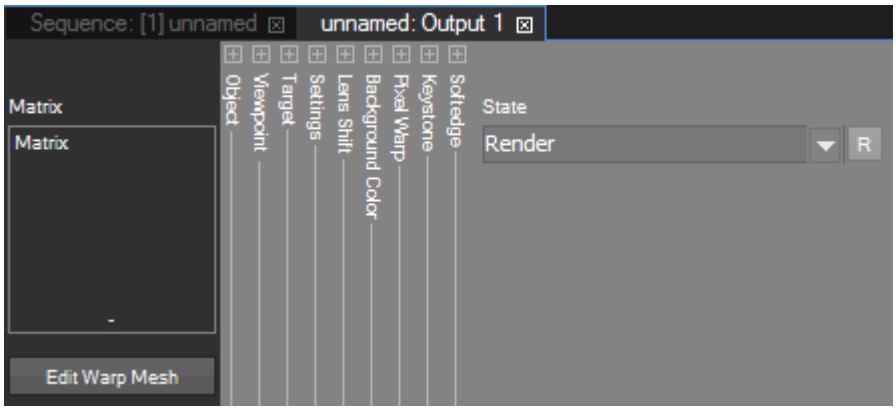
Effects on Output Devices

The new dynamic shader effects engine allows creating and combining an almost unlimited number of effects. All effects and animations are automatically synchronized across the system network.

Effects can be used on the Output Device just like on a Video Layer. The difference is that these FX will influence the overall output instead of only the layer.
Example: It could be very helpful using a color correction FX on the output to compensate color corruptions of displays or projectors.

Please refer to the topic "Working with Effects [322]".

6.5.9.1 Matrix



With the matrix section you may use Pandoras Box to output the pixels RGB(W) or CMY color information over Ethernet via the Art-Net protocol (in addition to the normal video output via the graphics card). This allows you to sent color information to specific lamps or Art-Net based LED walls. More detailed information can be found in the chapter "Matrix Patcher [2077]" as well.

Matrix

The Matrix field accepts either former CSV files or the newer PBX files that are part of the project already. Of course each Client in your PB network may be programed with different matrix files. In addition they can be saved on the timeline and may change throughout the show.
Use the right-click menu in the Preview [245] to visualize the matrix files. If changes are needed, select the matrix file in the Project tab and choose to edit it with the Matrix Patcher, alter the patch and update it in real-time.
Please note, that effects dragged onto the output layer cannot be considered by the patch file, thus you will not see output effects on your DMX panels.

Creating a Matrix File

First of all, use the Matrix Patcher to define the entire pixel workspace that you want to use and patch the fixtures to the DMX channels and subnet addresses. You can either make use of predefined fixture within the library or create individual lamp/wall types and use them as templates for your custom matrix design. You can use RGB or CMY mixing devices.
The topic Matrix Patcher [2077] has further information and a patching guide [2089].

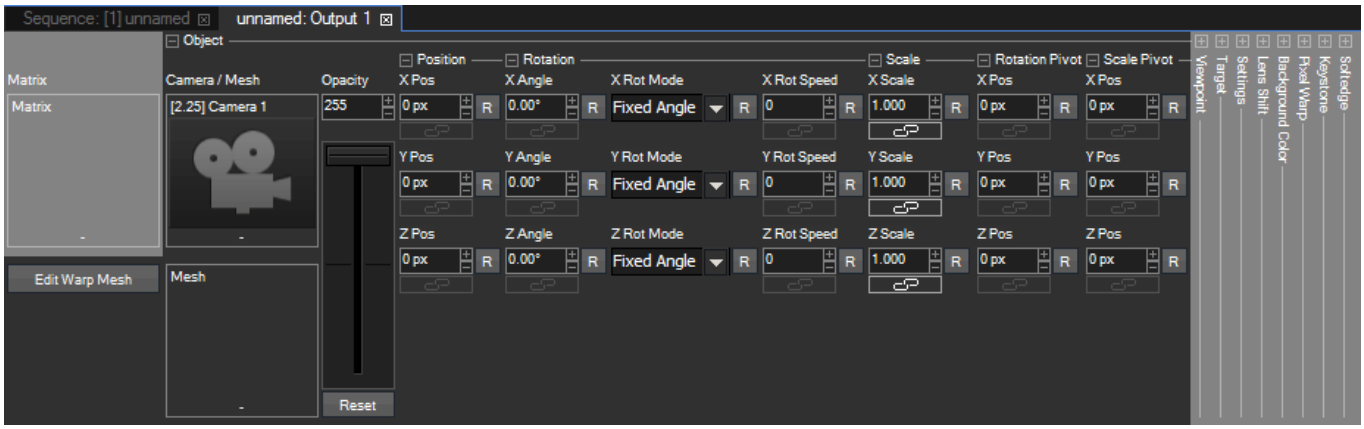
The limitation regarding how many pixel data in form of Art-Net universes can be transferred is a question of your network components and receiving devices. As a rule of thumb, we do not recommend to output more than 60 universes per network.
Per default, Art-Net is set up as a broadcast protocol; nevertheless, using the Matrix Patcher, you have the option to use the transfer rate of each transmitting and receiving device to its full capacity by sending the pixel information more efficiently with the multi- or even unicast method.

There are certain network switches like the ones from ELC that support DMX lines and dedicated Art-Net routing. If you need more information concerning these devices please contact your local ELC Dealer.

Edit Warp Mesh

This button actually belongs to the "Mesh" parameter explained in the next chapter [684].

6.5.9.2 Object



The object section allows controlling the overall opacity of the output, setting the Camera and a 3D / Warping object as well as adjusting its position, scale and rotation state.

Per default, position values are displayed in pixel units. You can change the value read-out to generic units in the Configuration tab under "Unit Management" [160]. The linked chapter also explains how the pixel values are calculated, how to change the factor, the direction of the Y axis and whether the layer's center or upper left corner is positioned at the origin of the coordinate system XYZ=0,0,0.

Object Selection

Camera
The camera control [675] is the first render path. The result of this camera filming your 3D composition is passed to the output control, which is the second render path. The passes are described in detail in the chapter "Video Processing Pipeline [320]".
Per default, each output number is linked to the according camera number. In case you added a new camera you might want to route it to an existing output. Of course, you can also save this parameter to the timeline and also change to another camera.

Right-click the camera parameter and choose "Camera Texture" to select another source for this output.

To remove the resource from the layer, right-click on the parameter field and choose "Reset".

Mesh
The mesh parameter is pretty much the same as the one from Video Layers [647]. The big difference is that the entire camera texture, and not only a single media file is displayed on the selected object.
An object can be two or three-dimensional and is used to deform or warp a texture or to define another UV map,

which is explained in the chapter General Mapping Questions [274]. But mainly, the Output object has the purpose of warping the Output which allows to project the output image on any surface without deformation which starts with a simple curved screen and goes to architectural projection.

Valid object file types are: 3DS, CLX, FBX, OBJ, and X

To assign an object to an Output Layer, drag and drop the resource from the Project tab onto the designated Output. You can also highlight the target Output layer by left-clicking on it and use the right-click command on the resource in the Project tab "assign to active device" or just double-click on the object file.

To remove the resource from the layer, right-click on the parameter field and choose "Reset".

There are different ways to create an object: you can either create an Editable Mesh directly in Pandoras Box, or import one after creating it with the Warper [2129] tool or another third party software [2177]. The chapter Mesh Editing mode [257] explains how to warp directly in PB and also covers the button **"Edit Warp Mesh"**. The chapter Planar and Perspective UV Mapping Mode [269] covers UV mapping.

Opacity

The opacity sets the overall output brightness level.

Opacity 0%	black, no output
Opacity 100%	full brightness

Position, Rotation, Scale

The 3D / Warping Object may be adjusted in its position and scale and may be rotated.
Please see the according chapters under Video Layer [647], as all outputs position, rotation and scale parameters work the same way.

External Control via DMX, Widget Designer, etc.

You can control all Object parameters externally.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

Camera: This parameter can only be controlled via Widget Designer (command ShareLayerTexture [1612]) or the SDK.

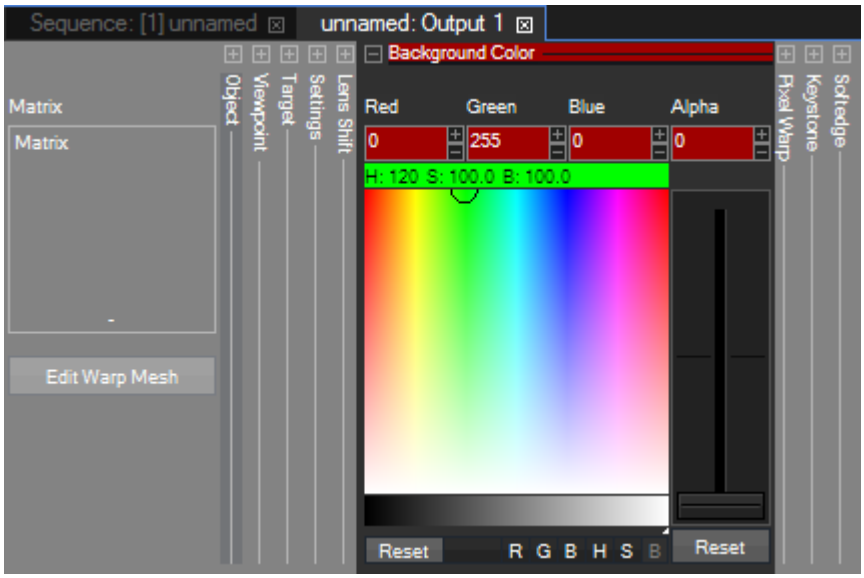
Mesh: In general, media and mesh files can be assigned via their file names or by addressing them via the Folder and File ID. For DMX control, only the later is possible.
To assign a Folder and File ID, select the file (or folder) in the Project tab and look for the option in the File Inspector [191]. Both, the Folder and the File ID can be assigned with a value from 1-255.

Opacity: Simply assign a value between 0-255.

Any position parameter: Note that the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels if you want the value to match up. See the linked chapter for more information.

When you remote control position parameters via DMX they are always interpreted as generic units. As said above, you can deactivate the "Pixel" read-out [161] to display units in Pandoras Box as well.

6.5.9.3 Background Color



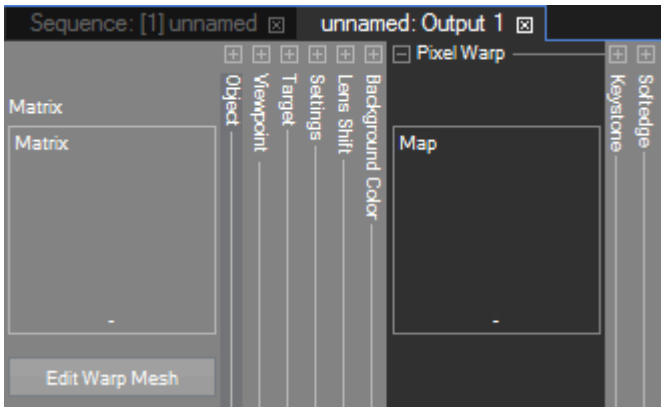
The Background Color section of the Output Device allows adjusting the red / green / blue / alpha channels for the main background from the resulting back buffer texture rendered by the graphics card. More information about the back buffer can be found in the chapter Video Processing Pipeline [320]. A possible application for adjusting the background color could be blacklevel compensation [646] when working with multi-softedged projection(s). If you are not familiar how to use the color picker tool, please read the topic "Adding, Editing and Removing FX [323]".

Please note:
- In order to see the Output's background color, the Camera's background color [681] alpha channel needs to be set to 0.
- If you like to use the alpha fader from the Output, i.e. clear the back buffer with the transparency channel, tick the option "Enable Alpha Back Buffer" in the Configuration tab > Render Engine [162].

External Control via DMX, Widget Designer, etc.

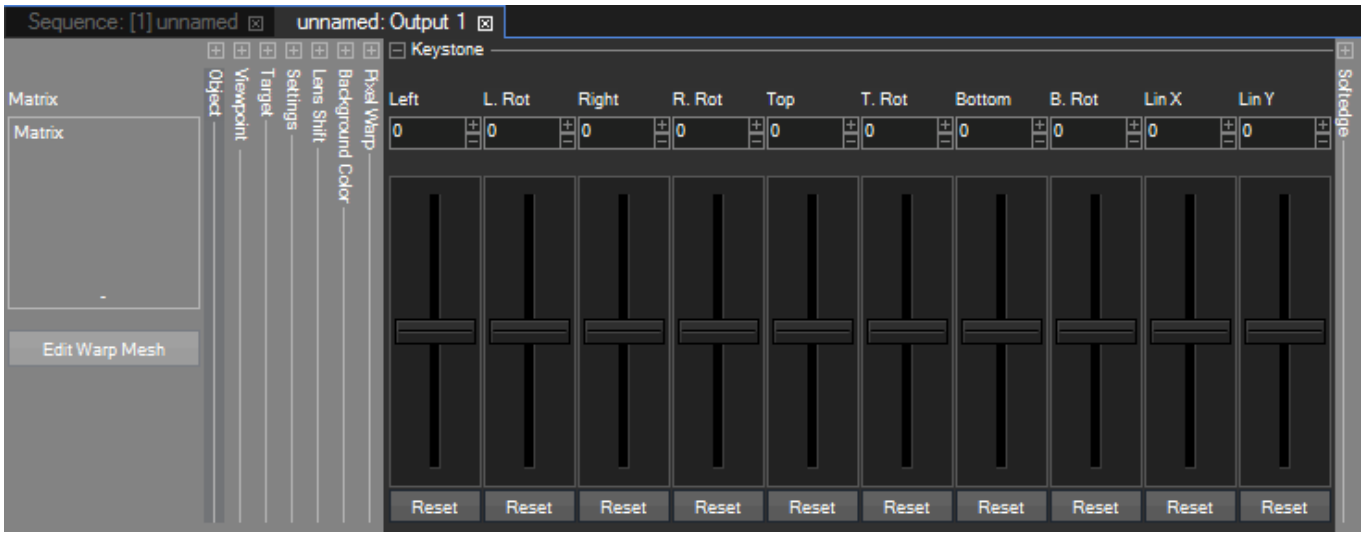
You can control the color parameters externally, simply assign a value between 0-255 to each color or alpha parameter.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

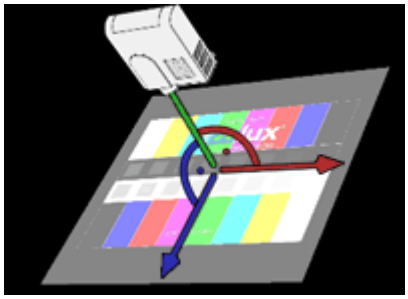
6.5.9.4 Pixel Warp



The "Pixel Warp" parameter was initially introduced to support warping with Christie Mystique which is a camera solution for automatic blending and warping calculation of complex projection areas. After setting up the hardware and software, a Canvas [279] is generated and applied to each Pandoras Box output automatically. Note that the Canvas has a different "Texture Format" setting in the Inspector [199] than a Canvas that is used to be drawn on [256].

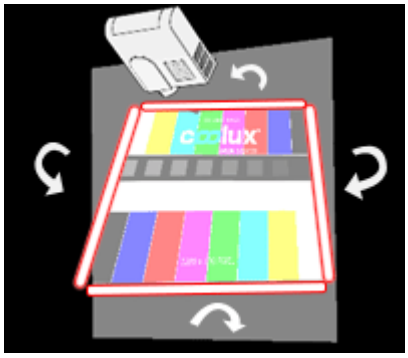
6.5.9.5 Keystone





The keystone function is a useful feature for planar projection.

If a screen is positioned perpendicularly to the projection axis of the projector, the projected image is not distorted as seen in the first image.



But as soon as one axis is not orthogonal, the projected image is not rectangular any more and image correction needs to be done. The distortion relates to the shape of the screen and to the way the projector "looks" at it which includes the orientation and distance as well as all lens settings.

If the screen is planar, as seen in the second image, the keystone parameters in Pandoras Box can deform the image in such a way, that it is rectangular again and that its linearity fits.

Each edge of the output image can be moved in or out and rotated according to the projection surface. Hence, the controls allow very individual keystone correction whilst many projectors offer only a symmetrical, horizontal correction.

Once you have set up each edge, you might compensate for the linearity distortion with the "LinX" and "LinY" parameters. To do this, use a raster grid test pattern on a layer and reset the image center with the X&Y channels.

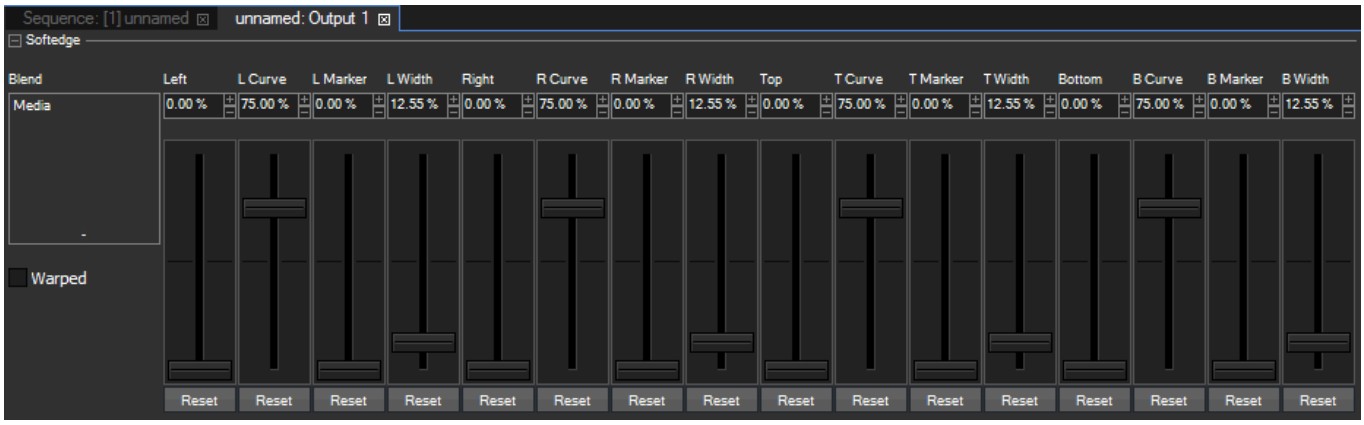
As soon as the projection surface is not a single plane anymore, e.g. it is a bended screen or a more complex object like a cube the keystoning reaches its limit and an object must be assigned to the Mesh parameter of the output [684].

Please note that it is not possible to keystone an image if an object is used on the Mesh parameter [684] as the object already deforms the camera texture.

External Control via DMX, Widget Designer, etc.

You can control all keystone parameters externally.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.9.6 Softedge



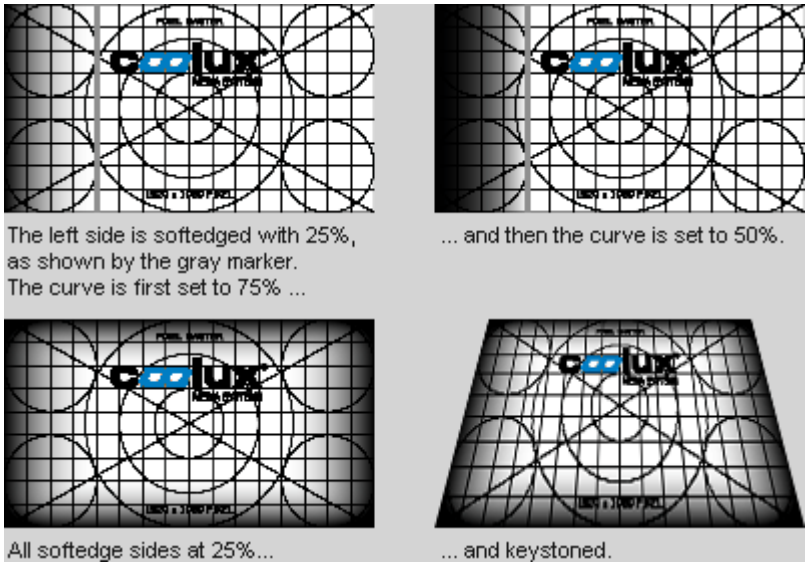
Pandoras Box features a four sided image blending for any output configuration. Softedge allows smooth image blending of multiple projectors. In addition the Blend parameters allows to blend according to an image.

Per default, Softedge settings are also influenced by the Keystone [687] and Mesh [684] parameters from the Output Layer. Uncheck the "**Warped**" check box if the softedge blending should not be warped.
Note that you can access this functionality with the "Warp Softedge" check box in the Output Inspector [220] too.

Pandoras Box Softedge Controls

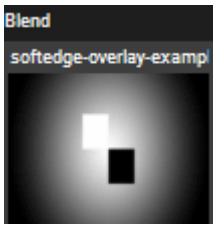
There are two possible ways to blend in Pandoras Box and both allow a variable overlap between 0% and 100% of each image edge.

The most used one is to use the softedge controls for each side which fade in a softedge gradient and allow manual softedging. The sides are labeled left, right, top and bottom. The first control, named simply like the side, e.g. "**Left**" influences how much percent of the image is faded out towards the side. The second "**Curve**" control varies the gradient curve. The third "**Marker**" and forth "**Width**" controls are just for setup and allow to see a line at the end of gradient. Whilst "**Marker**" sets up the color , "**Width**" sets the size of the line. Mostly, it is a good start to fade two sides from an overlapping area in the same way, e.g. "Left" and "Right" at 25% and the Curve at 75%.



The alternative to the above softedging technique is to use the "**Blend**" media parameter. As soon as you load an image, the other controls are without effect. The blend map simply overlays the output texture and is multiplied with it. This means that white or transparent parts of the map show the output and black ones blank it out. The "**Blend**" parameter was initially introduced to support blending with Christie Mystique which is a camera solution for automatic blending and warping calculation of complex projection areas. After setting up the hardware and software, a blending map is generated and applied to each Pandoras Box output automatically.

This shows the output without any softedge blending.



If this image is assigned to the "Blend" parameter...

...the output would look like this.

Overlap and Softedge Setup

The size of the overlapping area is not only important information for content creators but also for the technician who sets up the softedge. The overlapping area is that part of the projection, where the projected images of two projectors share the same image information and it is needed to create a smooth image-blending from one projector to the other. Due to the physical setup of the projectors - their light output and contrast - the overlapping has to be adjusted carefully to match a seamless image. The amount of the overlap depends on the image and projector quality. As a rule of thumb, 256px are a minimum to achieve a smooth blend.

After setting up the projectors properly (matching the raster, size and keystone) the overlapping edges need to be blend together. The best way to do this is by using a test pattern with a grid and positioning it precisely on top of each other. Now you may bring the soft edges in and adjust the curves until there is no more hotspot area.

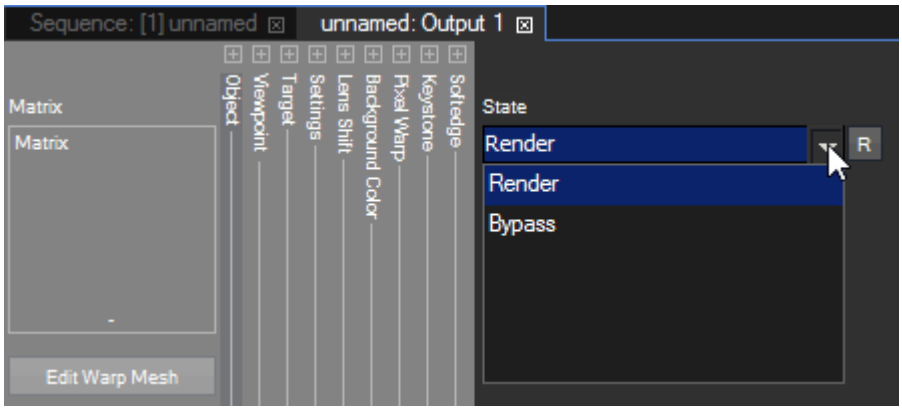
Creating Content for Softedge Projections

There are two possibilities to create content for softedge projections. You can either create split images, e.g. in case of two projectors side by side: a left part for the left projector and a right part for the right projector. Or you can create one large image and adjust the X&Y positioning of the Camera [675].
It is also possible to use both methods in the same show, i.e. having overlapping content and one large content in the same timeline but different layers.

External Control via DMX, Widget Designer, etc.

You can control all softedge parameters externally.
For DMX control, the chapter DMX Tables [708] lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List [1514] is of interest.

6.5.9.7 Render State



By default the State of the Output Control is set to "Render". That means the output is working as a second render path, all settings done in the output control will affect the camera result.

When the state is changed to "Bypass" **all settings** done in the output (like the object, keystone and softedge) will be ignored to gain the highest rendering performance. This mode might be used in setups where no keystone or screen warping is needed and highest image rendering quality is required.

External Control via DMX, Widget Designer, etc.

You can control the State parameter externally, simply send a value (0 or 1) to set the drop-down to this index. For DMX control, the chapter DMX Tables 708 lists all parameter information, for other external control e.g. via the Widget Designer the Parameter List 1514 is of interest.

6.5.10 DMX Devices

By adding DMX Devices to your timeline in the Pandoras Box master you can program for example a fog machine or the shutter or position for a moving light. It is an easy way to synchronize DMX output to video playback. Pandoras Box features both 8bit and 16 bit DMX parameters and supports the protocols DMX, Art-Net, MA-Net and sACN.

By the way, it is also possible to input DMX [706] and remote control layers by an external lighting desk.

Built-In DMX Devices

As said above, to output DMX data to a device, it needs to be part of the timeline. Open the Device Types tab [183] and open the folder "DMX fixtures". This shows a list with all built-in devices. You can also add devices as described below. Choose one and drag it to the Device Tree [173]. Now, you can control DMX devices with parameter keys in the same way as with Video Layers.

The chapter about DMX output [721] explains how to patch [228] a device, configure the output protocol and finally, send data.

Custom DMX Devices

You may add a custom DMX device to the list in the Device Types tab [183]. A custom DMX library file can be created as a text file e.g. with WordPad. Please see below an example for one fixture. Store the file with the extension ".clib" in the installation folder of the Master under: `"\data\types\DMX Fixtures"`. Note that you will need administrator rights to store it there. In Pandoras Box, you can right-click the "DMX Fixture" folder and choose "Refresh" to see your DMX device instantly. Now, please follow the chapter about DMX output [721] that explain how to patch a device, configure the output protocol and finally, send data.

Please find here an example of a DMX .clib file for a Martin MAC500:

```
<?xml encoding="yes" ?>
<descripDevice type="fixtureDmx" artNetIndexScope="16">
<descripModule type="param8bit" name="STROBE" default="34" artNetIndexOff="0" />
<descripModule type="param8bit" name="DIM" default="0" artNetIndexOff="1" />
<descripModule type="param8bit" name="COLOR1" default="0" artNetIndexOff="2" />
<descripModule type="param8bit" name="COLOR2" default="0" artNetIndexOff="3" />
<descripModule type="param8bit" name="GOBO1" default="0" artNetIndexOff="4" />
<descripModule type="param8bit" name="GOBO1 ROT" default="0" artNetIndexOff="5" />
<descripModule type="param8bit" name="GOBO2" default="0" artNetIndexOff="6" />
<descripModule type="param8bit" name="FOCUS" default="50" artNetIndexOff="7" />
<descripModule type="param8bit" name="IRIS" default="0" artNetIndexOff="8" />
<descripModule type="param8bit" name="PRISMA1" default="0" artNetIndexOff="9" />
<descripModule type="param16bit" name="PAN" default="32768" artNetIndexOff="10" />
<descripModule type="param16bit" name="TILT" default="32768" artNetIndexOff="12" />
<descripModule type="param8bit" name="SPEED1" default="0" artNetIndexOff="14" />
<descripModule type="param8bit" name="SPEED2" default="0" artNetIndexOff="15" />
</descripDevice>
```

As you can see the first line gives the general device description and the overall DMX channel count "artNetIndexScope=".
The following lines describe each DMX channel of the fixture by a zero based offset of the DMX channel count at the end of each line by "artNetIndexOff=".
You may also choose the type of parameters "param8bit" or "param16bit" as well as setting the descriptive name and reset and default value.

6.5.11 Serial Link Device

With the Serial Link device you may control most routing switchers, projectors, shutters or other device parameters that are remote controllable via the common RS 232 and 422 serial protocols. Even though, the device's name is Serial Link, you do not necessarily need the hardware device itself [2044]. The Serial Link device was often used to send TCP/IP commands; since version 6.1.1 Pandoras Box offers the dedicated "TCP Device [693]" for that.

To add a "Serial Link Device" to your project, open the Device Types [183] tab and drag the "Serial_Link" to the Device Tree [173].

Select the device in the tree to load it's properties into the Inspector tab [190]. If you like to send commands to the Serial Link hardware [2044], check the option "Use Serial Link header". If you like to send commands to another TCP device with a built-in TCP Server, untick this option (but again, using the TCP device [693] is a simpler workflow).

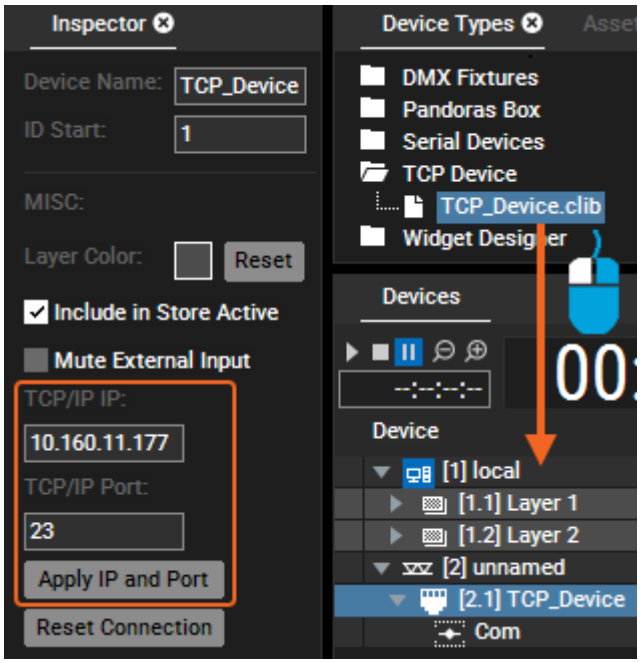
Now, type in the right IP address of the device and its connection port and click the button "Apply IP and Port". The red exclamation mark on the Serial Link device in the Device Tree disappears which means that you are connected.

As the connection is now established, the commands can be send. Simply add the serial command according to the device's syntax as a key to the timeline. In detail this means to open the parameter "COM" and insert a key frame by right-clicking on the according time whenever you want to send commands. Then select the created key frame and edit it in the Inspector. Enter the command in the "Data" field. Please refer to the user manual of the projector to get the specific commands you want to send. If your device demands to send a special character like a "carriage return" or "start of text" send your entire message "As Hex" and write everything including that special character using hexadecimal values. You may find a list with ASCII characters [944] and the according hex code in the Widget Designer chapter. The next chapter about the "TCP Device [693]" shows some more examples.

As soon as the nowpointer reaches the key frame the command is send via the TCP connection. In that way any serial ASCII or hex string can be stored and copied to any point in the timeline.

6.5.12 TCP Device

The new TCP Device (introduced in version 6.1.1) allows for simple communication to devices such as projectors or matrix switched which are remote controllable via their TCP/IP Server interface.

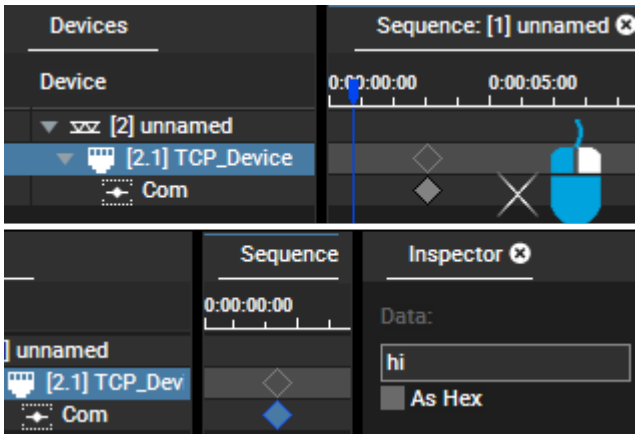


Please open the Device Types [183] tab and drag "TCP Device" into the Device Tree [173].

Select the device in the tree to load it's properties into the Inspector tab [190]. There, simply enter the IP address and port of the device you like to communicate with. The IP and port can be either set on the device itself or they are predefined. The Christie Mirage 3D projectors for example listen to port 3002. Please check the device's manual.

Click the button "Apply IP and Port" to establish the connection. The red exclamation mark on the TCP device in the Device Tree disappears which means that you connected successfully.

In case you cannot connect, check whether you can ping the IP. Another issue could be, that your device cannot open two connections at the same time. In that case please close any other remoting ports which includes interfaces for browsers too.



As the connection is now established, TCP commands can be send.

Open the TCP device folder in the Device Tree to see the track of the "Com" parameter. In the Sequence tab, right-click this track at a time where you like to send a TCP command. Select the inserted key frame and in the Inspector, edit the field "Data".

Enter the command according to the device's syntax. Please refer to the user manual of your device for that. For example, (SHU 0) or (SHU 1) are common commands to open and close the shutter. The round brackets are not requested from Pandoras Box but most devices demand that a command is enclosed with brackets.

If your device demands to send a special character like a "carriage return" or "start of text" send your entire message "**As Hex**" and write everything including that special character using hexadecimal values. You may find a list with ASCII characters [944] and the according hex code in the Widget Designer chapter. "0D 68 69" would for example send a "carriage return" and the word "hi".

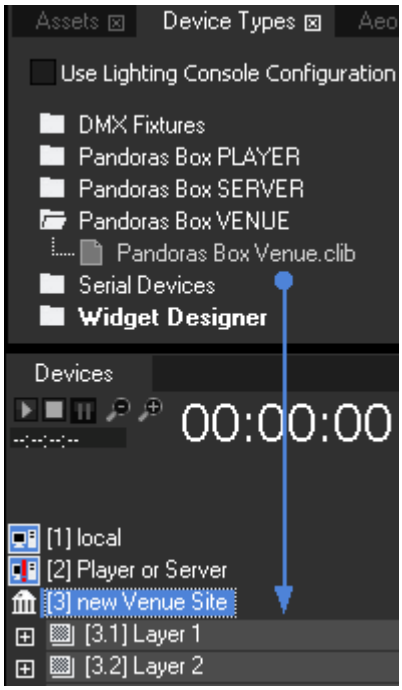
As soon as the Nowpointer reaches the key frame the command is send via the TCP connection to the external device. To repeat the command, simply copy and paste the selected key frame.

6.5.13 Venue Site

This chapter describes the Venue Site feature in Pandoras Box.

A Venue Site can be added from the Device Types tab [183] to the Devices tab [173] respectively the Sequence. In difference to normal Sites a Venue Site is never real hardware like a Server. The purpose of a Venue Site is rather pure (pre-) visualization. It renders your programming (i.e. final composition) on 3D objects that form a virtual copy of your stage. That turns the Preview tab from being an abstract image to a very real representation of what is happening on stage.

Understanding Venue Sites requires also to know about similarities and differences of them to other Clients like Servers.
Regarding Clients, you include them into the sequence by dragging them from the Assets tab [131] or from the Device Types tab [183] into the Devices tab [173]. You render content on them by spreading media files to them and assigning these resources to layers. With respective settings (opacity, scaling, effects) the resources appear in the 3D space of Pandoras Box. Whenever a Camera Device sees a resource, it renders it as part of its camera texture. Then this texture is send to the Output Device to be re-rendered with the output settings (warping, keystone, softedge). This texture is the real output of your graphics card and is connected to a display device.



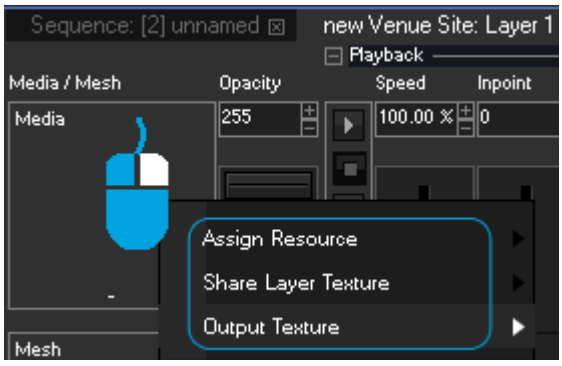
Regarding Venue Sites, you include them into the sequence by dragging them from the Device Types tab [183] into the Devices tab [173].

Unlike other clients, a Venue Site cannot connect to another system and thus the Inspector does not offer settings like "IP address" etc. But in the whole, the Venue Site Inspector equals the Device Inspector [210].

A Venue Site can be toggled to be rendered in the Preview tab. Right-click on the Venue Site in the Device Tree and choose "Toggle Preview" and note that the symbol is now shaded in blue. All Cameras and Outputs from previewed sites appear in the drop-down list in the Preview's upper right corner. Choose the Venue Site Camera.

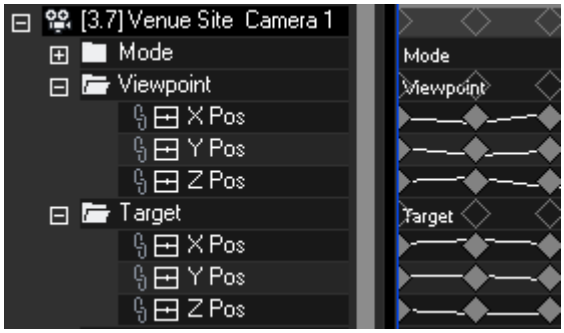
The main purpose of a Venue Site is to render content from other Clients, which means that their Camera and Output textures are routed to the Venue Site.
Include at least one other client to your project and assign some content to its layers. You can also assign an object to the Output layer to simulate output warping. Make sure that all clients, that should be visualized in the Venue Site, are toggled into the Preview. For larger projects you might want to visualize many outputs which requires the Master to render many textures. To save performance, the routed textures are scaled down. Per default this "Venue Texture Size Factor" (Configuration tab > Local Preview [150]) is set to 0.25 . Increase it for higher quality and decrease it for more performance.

Now, we will assign content to a Venue Site layer.



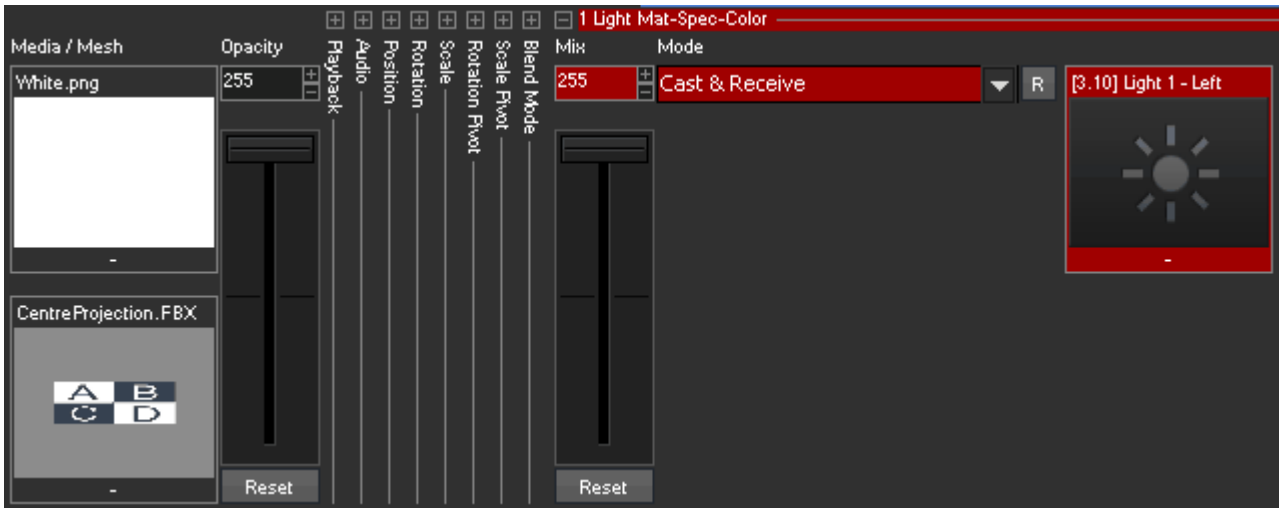
There are three options available when you right-click on the Media field in the Device Controls tab from a Venue Site layer:
Assign Resource: shows a list with all media files added to the Project tab (you can of course also drag it from there onto the layer or assign it via a double-click)
Share Layer Texture: shows a list with all layers from the Venue Site; select one to share its current texture with this layer
Output Texture: shows a list with all available camera and output textures from clients that are added to the Device Tree and toggled into the preview (!) This option is probably the one most used with Venue Sites.
Besides the texture, you would mostly apply a 3D object to a Virtual Layer too. The object represents a part of your stage. This can be for instance a flat 2D object representing a display, or it can be a curved object representing a curved projection screen. You could then share the camera

or output texture from the client that also renders the real content for the respective display or screen. You can of course also assign other objects visible on the stage like a chair object, moving head object, human object, etc. and then assign a "normal" media file to the Venue Site layer, e.g. a brown image for the chair or the texture that belongs to the 3D object and fits to its UV mapping [274].



From that point on, the render pass is the same as for real clients. The Venue Layer is captured by a Venue Camera which sends its own texture to the Venue Output, again offering the known output settings. However, the Venue Output does not send its result to a real display but to the Preview tab.
You may also preview a Venue Camera and animate it on the timeline. Remember to toggle the Venue Site and all clients into the preview and to select the Venue Site camera with the drop-down menu in the upper right corner of the Preview tab. This way you can fly through your 3D stage and look at the content from different angles. Of course you can always export this animation using the Video Export feature [135]. And it is possible to add further cameras with a right-click on the Venue Site in the Device Tree and program another fly through with other perspectives.

As with other sites, all settings for Venue Sites can be kept as active values [127] or be stored to the timeline [296] in form of containers. Or you remote control them using for instance the Widget Designer [786].



There is the possibility to (roughly) simulate projectors including the deformation when hitting screens. That is what makes keystoning and warping necessary. Softedging can be roughly simulated too.
First, add a Light Layer [666] to your Venue Site with a right-click on it in the Device Tree > Add Device > Light. The light represents the projector. The light controls like position stand for the projector's position and the field of view, represents the lens FOV. The projectors image is of course a camera texture from another site.

Now you need another Venue Site layer with a 3D object that represents the real screen. The left image depicts a layer with a white media file and a 3D object for a wide projection screen. A light effect [527] was added to the layer to "receive" the projector's light. The example uses the effect "1 Light Mat-Spec-Color", the Mix fader is set to 255, the Mode to "Cast &Receive" and the above added Light Layer is set to be the source (right-click into the media field > Share Layer).

A sample project can be found in C:\Christie\content\projects\VenueSites_Training_Stage (or C:\coolux\...). In version 6, please use the Import function to load it.

6.5.14 Widget Designer Device

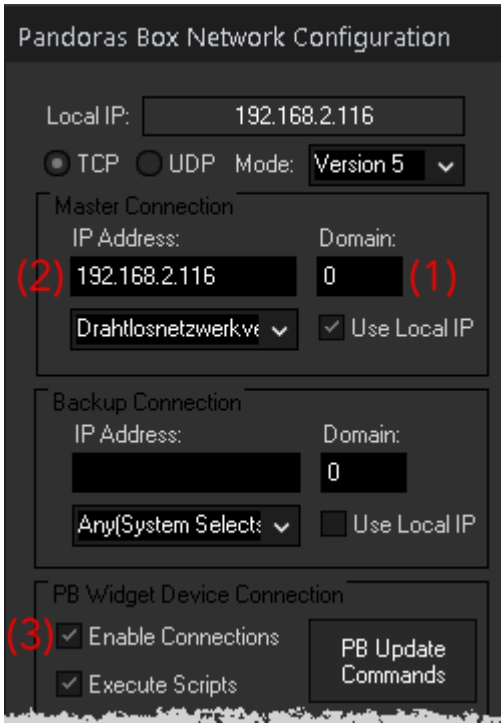
This topic explains how to add a Widget Designer Device to the Devices tab allowing to:
- program Widget Designer commands directly in timeline
- send (trigger or feedback) values from the timeline to Widget Designer
- send mouse and touch input events from an attached Client to Widget Designer

Widget Designer [786] is another software product from Christie Digital Systems Germany. One of the main purposes of Widget Designer is to enable users who do not have any programming skills, to create their own custom applications or interfaces. Widget Designer provides a graphical user interface and runtime environment. You can create user-controls such as faders, buttons, labels and many more and instantly use them. The PRO version allows to program visually with so called nodes. By simply connecting those visual control components you may create a dedicated interaction logic. In addition WD has a built-in script language allowing you to create customized routines of commands. For example, a command can be assigned to a button click. Currently there are over 1000 commands that control specific features of Pandoras Box, Widget Designer or other 3rd party products and protocols. There are topics explaining how to use commands [1511] and containing the entire command list [1520].

If you like to execute commands at a defined time within the timeline you can simply add the Widget Designer Device to the Device Tree tab and then program keys directly into the timeline, just as you would add keys for a Video Layer's position for example. If you are not familiar with the timeline and key programming, please read the topic "Sequence [292]".

The second purpose of a Widget Designer Device is to route mouse / touch input from the PB Master or even PB Client to Widget Designer itself. This is described step-by-step in the Layer picking chapter of the Preview [252]. If you like to route this data to a Canvas asset in order to draw on it, the node "Layer UV Draw to Canvas [1245]" is of interest. Its chapter includes a step-by-step description of all necesary settings in WD and PB. You can also use the data to click on Custom Script Buttons [822] as described in its chapter.

Please follow these steps to set up a Widget Designer Device:

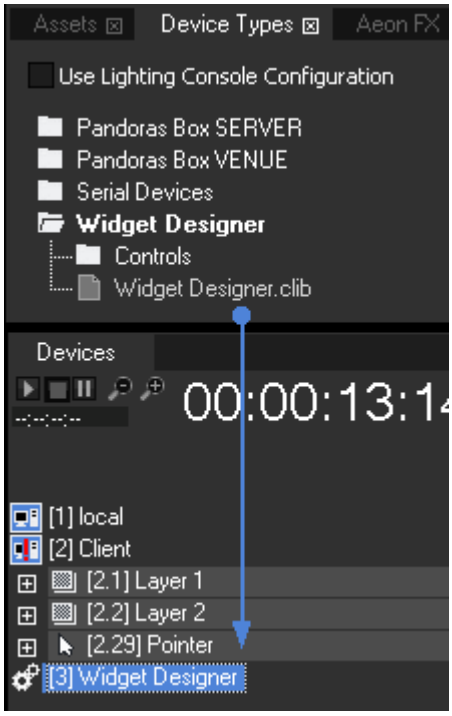


First of all, start the Widget Designer 4.0 and enable the connection to a Widget Designer Device in Pandoras which we will set up in the next step.
To do so, right-click in the main window and choose "Connection > PB Configuration [1256]". Then make sure that the Domain (1)and IP address (2) under "Master Connection" matches with the PC where the Pandoras Box Master is running on. In Pandoras Box, the IP address is shown in the Assets tab, the Domain in the Configuration tab.

Have a look in the section "Pandoras Box Widget Device Connection" and enable the check box "Enable Connections" (3). You can close the dialog.

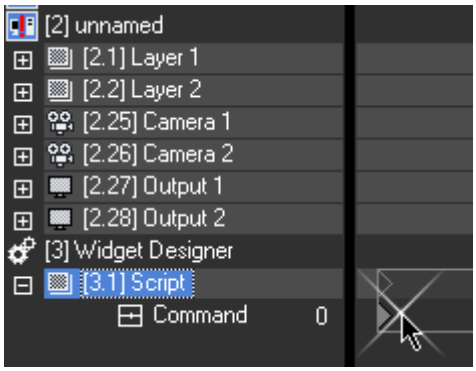
As an example, please create a fader. Again, right-click in the main window to call the main context menu. Choose Create > Fader Controls > Fader Vertical and make one left-click into the main window. You have now created a fader with the default settings: ID = 1, minimum value = 0 and maximum value (also current value) = 255 .

So far, everything is set up in Widget Designer, please switch to your Pandoras Box Master software.



After creating a project, you can add a WD devices to it. For this, go to the "Device Types" tab > "Widget Designer" > and drag the "Widget Designer.clib" into the Device Tree tab.
As we have already set up the WD in the first step, the device can connect successfully.

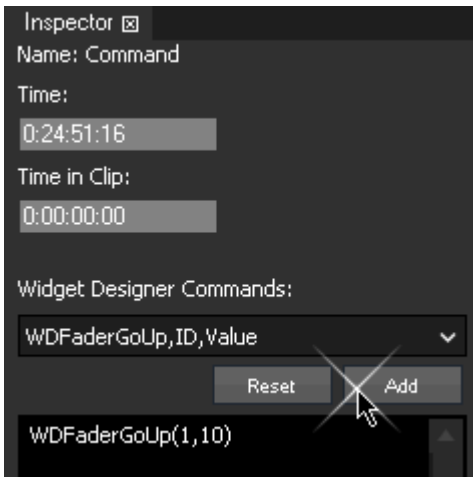
If you see a red exclamation mark in the symbol within the Device Tree (as shown in the left image for the Server device), check the IP address and domain number (in the Configuration tab). If the Widget Designer runs on a different PC, the Widget Designer device must be set to the according IP address. Select the Widget Designer device in the Device tab and edit the text field "IP" within the Inspector tab.



To send commands from the timeline double-click "Widget Designer" in the Device tab, and then "Script" as well. You should see the parameter "Command" whereto we will now program a key.

Make a right-click into the timeline at the height of the command parameter, as seen in the left image.

To assign a command to this key select it and switch to the Inspector tab.



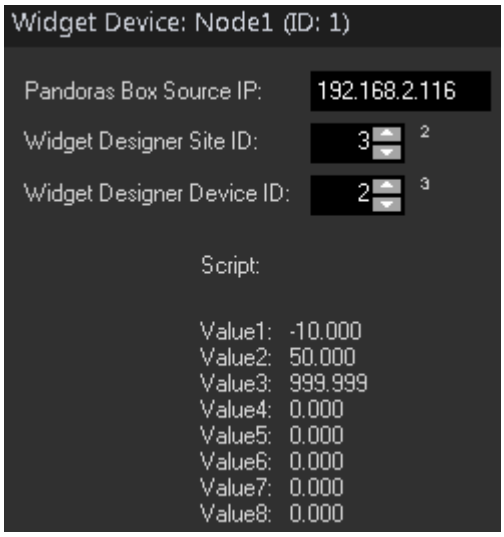
In the key's Inspector, choose a command from the drop-down list and paste it into the text field by pressing Add. Mostly, a command consists of several arguments separated by a comma - the first one always describes the command function the following ones are variables. Substitute them with according values.

Let's say, we want the fader created in Widget Designer in the very first step to lower its value by 50 steps.

The command to add is:
WDFaderGoDown,ID,Value.
As the ID is 1 and the value to deduct is 50, the customized command in the text field should be:
WDFaderGoDown,1,50

As soon as the nowpointer in the timeline passes the key, the command is executed and the fader will go down 50 steps.

If you like to execute more than one command press carriage return and add another one. There is the possibility to work with functions and macros [1897] as well.

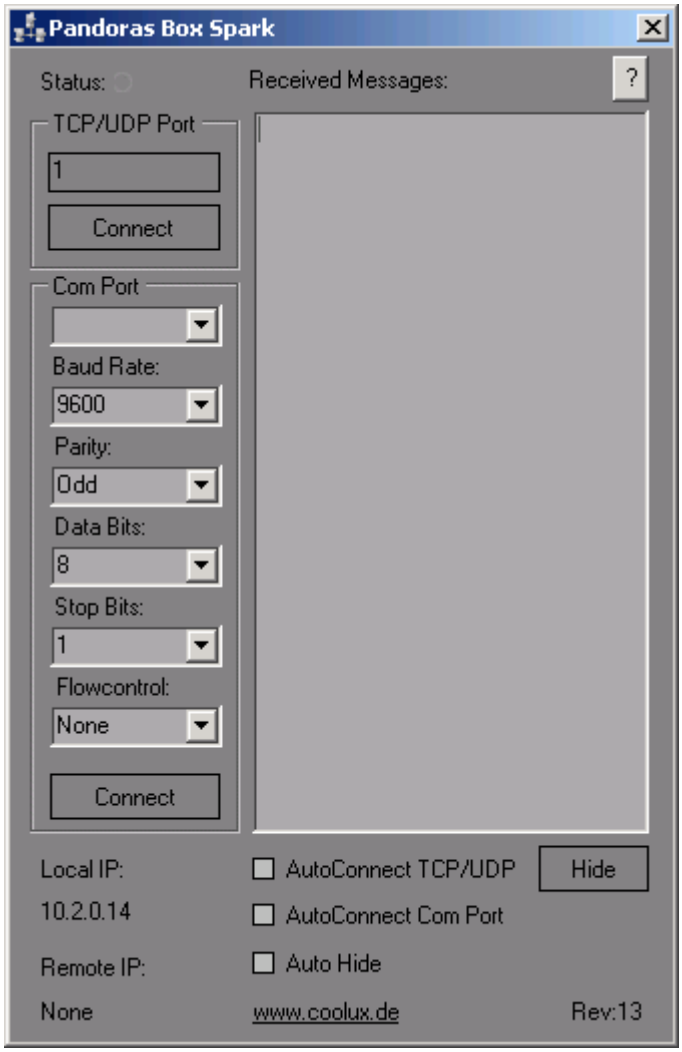


More advanced users who have a Widget Designer PRO version may use another feature that allows to send (trigger) values to WD.

- in the Device Types tab open the folder "Widget Designer > Controls" and drag "Values.clib" onto the Widget Designer site in the Device Tree tab. Now the WD site offers one script device [3.1] with the command parameter and a new values device [3.2] with eight value parameters (each with a value range from -999.999 to +999.999)
- program keys either by right-clicking in the timeline as described in the example above. Or, you switch over to the Device Control tab and then store active values
- create a node [936] in Widget Designer: Nodes > Input > Pandoras Box > Widget Device in order to retrieve the values and send them to filter nodes or directly to output nodes
- in the node's Item Properties dialog, adjust the IP address, the site and device ID. Following the example above, the site ID is 3 and the device ID refers to the newly generated value device, in this case number 2

6.5.15 Spark

for WINDOWS XP (Microsoft .Net 2 Runtime must be installed)



PB Spark is designed to be connected to any Pandoras Box timeline as a Serial Link 2044 TCP IP device.

PB Spark can be used for remote controlling applications, mouse and keyboard actions as well as displaying full-screen video playback and web page presentations among many other useful remote control features.

Pandoras Box Spark is available as a freeware download from our Download-Center.

To get started, install PB Spark on the desired PC that you want to control (Spark can be accessed also on the same machine that runs Pandoras Box)

Important Notice: Please switch off any firewall on all connected systems before using this application!

PB SPARK SETUP

Launch PB Spark.
Since PB Spark acts as a TCP IP server that will wait for a Pandoras Box connection you will need to specify the TCP/IP port that Spark should use to listen to incoming commands.
Set the TCP/IP port (any number between 1 and 9999) you want to listen to in PB Spark.
To start the TCP/IP server click "connect", PB Spark will now wait for a connection to be established.

PANDORAS BOX SETUP

Go to the Pandoras Box Master.
First add a Serial Link device from the Device Tab to the Device Control Tree.
Open the tree icon and click on the Serial Link Device to view its properties in the Inspector Tab.
Uncheck the check box "Use Serial Link Header" as this option is only for use with Serial Link devices.
Enter the IP address of the computer that PB Spark is running on as well as the Port number entered in PB Spark.
Click on "Apply IP and Port".

Tip: PB Spark displays its local IP address in the bottom left corner.

The red "!" in the device tree should disappear once the connection is established and the connection icon in PB SPARK should turn to green.
Now you may start implementing commands as keys in the timeline for the COM parameter.

Spark will give you the following TCP messages as feedback:
on success (SparkProc,"Original Message",Processed)
on fail (SparkProc,"Original Message",ProcessError)
on syntax error (SparkProc,"Original Message",SyntaxError)

COMMAND LIST

The following commands are all in ASCII format and need to be put in parentheses ()

Remote control features offer by PB Spark:

PB REMOTE SERVER CONTROL 700
COMPORT ASCII COMMUNICATION 700
SYSTEM COMMANDS 701
WEB BROWSER (Fullscreen) 701
SOUND COMMANDS 701
VIDEO PLAYER (Fullscreen) 701
APPLICATION COMMANDS 701
KEYBOARD EVENTS 701
MOUSE EVENTS 702
DESKTOP FADE TO BLACK -BETA 702
WAKE ON LAN 702
PB AUTOMATION REMOTE CONTROL 702

COMMAND DESCRIPTION

PB REMOTE SERVER CONTROLS
compatible with Pandoras Box Menu Rev7 and later

(PBShutdownAll) - Shuts down All Pandoras Box Servers on the network
(PBRebootAll) - Reboots All Pandoras Box Servers on the network
(PBShutdown,IP Address) - Shutdown Pandoras Box Server by IP Address
(PBReboot,IP Address) - Reboot Pandoras Box Server by IP Address
(PBStartMaster,IP Address) - Start Pandoras Box Server / Master by IP Address
(PBStartClient,IP Address) - Start Pandoras Box Client by IP Address
(PBClose,IP Address) - Close Pandoras Box Server / Master or Client by IP Address

UDP Sender allows sending broadcast ASCII messages
UDPSend,Port,Message

COMPORT ASCII COMMUNICATION

(Serial,Settings,"Portname","Baudrate","Parity","Databits","Stopbits","Flowcontrol")

ex: (Serial,Settings,Com12,9600,None,8,1,None)
Valid Parameters:
Portname: Com1 - Com 255
Baudrate: 110 / 300 / 1200 / 2400 / 4800 / 9600 / 19200 / 38400 / 57600 / 115200 / 230400 / 460800 / 921600
Parity: "Even"/ "Odd" / "None" / "Mark" / "Space"

Databits: 5 / 6 / 7 / 8
Stopbits: 1 / 1.5 / 2
Flowcontrol: "None" / "Xon/Xoff"

(Serial,Connect,"Portname")
(Serial,Send,"Message")

(Serial,SendHex,"Message")
example: (Serial,SendHex,41 42 43)
This hex values should send ABC as hex

(Serial,Disconnect)

SYSTEM COMMANDS

(System,Logoff) - logoff current user
(System,Shutdown) - shut down the system
(System,Reboot) - reboot the system

WEB BROWSER (Fullscreen)

(Browser,Open) - starts web browser
(Browser,Go,www.christiepandorasbox.com) - navigate in an open web browser to www.christiepandorasbox.com

(Browser,Close)

SOUND COMMANDS

(Sound,Play,C:\myaudio.wav) - plays the file myaudio.wav located in C:\
(Sound,Volume,50) - sets system volume to 50%, valid range is 0 to 100%
(Sound,Mute,On) - disables the system sound output
(Sound,Mute,Off) - enables the system sound output

VIDEO PLAYER (Fullscreen)

(Video,Load,C:\myvideo.avi) - open & loads video player with myvideo.avi located in C:\ - to play video use (Video,Play)
(Video,Playfile,C:\myvideo.avi) - open & plays video player with myvideo.avi located in C:\
(Video,Play) -plays video from current position
(Video,Pause) - pause video at current position
(Video,Stop) - stop video show
(Video,Close) - close video player

APPLICATION COMMANDS

(Application,Start,C:\notepad.exe) - starts C:\notepad.exe
(Application,StartFile,C:\notepad.exe,C:\mytext.txt) - starts notepad with mytext.txt
(Application,FileOpen,C:\myvideo.avi) - opens file with registered application, in this case windows media player
(Application,Stop,notepad.exe) - based on the process name in the windows taskmanager the specified process will be terminated

KEYBOARD EVENTS

(Keycode,A) - will generate a key stroke for the letter "A"
Access special Keys:
use "~" to send an [Enter]key stroke

The keys argument can specify any single key or any key combined with [Alt], [Ctrl], or [Shift] (or any combination of those keys). Each key is represented by one or more characters, such as a for the character "a", or {ENTER} for the [Enter] key.

To combine a key with [Shift], precede the key code with + (plus sign).
To combine a key with [Ctrl], precede the key code with ^ (caret).
To combine a key with [Alt], precede the key code with % (percent sign).

To specify repeating keys, use the form {key number}. You must put a space between key and number. For example, {LEFT 42} means "press the [left arrow] key 42 times"; {h 10} means "press 'h' 10 times."

The following table lists the codes that can be used to specify characters that are not displayed when you press the corresponding key (such as [Enter] or [Tab]).

[Backspace] {BACKSPACE} or {BS}, [Break] {BREAK}, [Caps Lock] {CAPSLOCK}, [Clear] {CLEAR}, [Del] {DELETE} or {DEL}, [down arrow] {DOWN}, [End] {END}, [Enter] (on numeric keypad) {ENTER}, [Enter] ~, [Esc] {ESCAPE} or {ESC}, [Help] {HELP}, [Home] {HOME}, [Ins] {INSERT}, [Left arrow] {LEFT}, [Num Lock] {NUMLOCK}, [Page down] {PGDN}, [Page up] {PGUP}, [Return] {RETURN}, [Right arrow] {RIGHT}, [Scroll Lock] {SCROLLLOCK}, [Tab] {TAB}, [Up arrow] {UP}, [F1] through [F15] {F1} through {F15}.

MOUSE EVENTS

Screen coordinates are handled as percentage of primary screen
x coordinate is 0 to 100.00 left to right
y coordinate is 0 to 100.00 bottom to top

(Mouse,Move,x as Percent,y as Percent) will move the mouse to xy screen coordinate
(Mouse,LeftClick, x as Percent, y as Percent) will perform a left click at specific screen coordinate
(Mouse,RightClick,x as Percent, y as Percent) will perform a right click at specific screen coordinate
(Mouse,MiddleClick, x as Percent, y as Percent) will perform a mouse middle click at specific screen coordinate
(Mouse,LeftClick) will perform a left click at current screen coordinate
(Mouse,RightClick) will perform a right click at current screen coordinate
(Mouse,MiddleClick) will perform a mouse middle click at current screen coordinate

DESKTOP FADE TO BLACK -BETA

(FadeToBlack,Out,1.5) fades out to black in 1.5 sec (timing may not be 100% accurate, based on systems performance)
(FadeToBlack,In,4.7) fades in from black in 4.7 sec (timing may not be 100% accurate, based on systems performance)

WAKE ON LAN

(WakeOnLan,Macadress) - wakes up a target computer with its matching MacAdress
Make sure the Ethernet Card accepts WakeOnLan with Magic Packet

PB AUTOMATION REMOTE CONTROL

Please note:
The following commands are only valid for PB Spark Rev. 16 or higher!
Old syntax (up to Spark Rev. 12) is still valid (see here [703]).

(PBA,Connect,IP_Adress,Domain)
(PBA,Disconnect)

(PBA,DeviceSetParam,SiteID,DeviceID,ParamName,AbsoluteValue)
(PBA,DeviceSetParamRelative,SiteID,DeviceID,ParamName,RelativeValue)
(PBA,DeviceSetMedia,SiteID,DeviceID,FolderID,FileID)
(PBA,DeviceSetMesh,SiteID,DeviceID,FolderID,FileID)

(PBA,ActivateAll)
(PBA,ActivateSite,SiteID)
(PBA,ActivateDevice,SiteID,DeviceID)
(PBA,ActivateParam,SiteID,DeviceID,ParamName)

(PBA,ClearAllActive)
(PBA,ClearActiveSite,SiteID)
(PBA,ClearActiveDevice,SiteID,DeviceID)
(PBA,ClearActiveParam,SiteID,DeviceID,ParamName)

(PBA,ResetAll)
(PBA,ResetSite,SiteID)
(PBA,ResetDevice,SiteID,DeviceID)
(PBA,ResetParam,SiteID,DeviceID,Opacity)

(PBA,SeqSetTimecode,SeqID,Hours,Minutes,Seconds,Frames)
(PBA,SeqGotoCue,SeqID,CueID)
(PBA,SeqNextCue,SeqID)
(PBA,SeqLastCue,SeqID)
(PBA,SeqNextFrame,SeqID)
(PBA,SeqLastFrame,SeqID)
(PBA,SeqSetState,SeqID,State) - States: Play, Pause or Stop

(PBA,SeqStoreActive,SeqID)
(PBA,SeqStoreActiveToTime,SeqID,Hours,Minutes,Seconds,Frames)

(PBA,ResourceSetText,FolderID,FileID,Text)

(PBA,ResourceAdd,Path,SiteID,FolderID,FileID)

(PBA,ResourceSpreadAll)
(PBA,ResourceSpreadMedia,FolderID,FileID)
(PBA,ResourceSpreadMesh,FolderID,FileID)

(PBA,ResourceSetFrameBlending,FolderID,FileID,State) - State: True or False
(PBA,ResourceSetDeinterlacing,FolderID,FileID,State) - State: 1,2,3,4 - according to De-Interlace Combobox in PB File Inspector
(PBA,ResourceSetAnisotropicFiltering,FolderID,FileID,State) - State: True or False
(PBA,ResourceSetUnderscan,FolderID,FileID,State) - State: True or False
(PBA,ResourceSetMpegColourSpace,FolderID,FileID,State) - State: True or False
(PBA,ResourceSetAlphaChannel,FolderID,FileID,State) - State: True or False

(PBA,ResourceRemoveInconsistent)
(PBA,ResourceRemoveMedia,FolderID,FileID)
(PBA,ResourceRemoveMesh,FolderID,FileID)

(PBA,ToggleFullScreen,SiteID)

OLD SYNTAX FOR PB AUTOMATION REMOTE CONTROL (up to Rev. 12):

(PBA,Connect, "IP_Adress", "Domain")

(PBA,Disconnect)

(PBA,SetParam,"NodeID","DeviceID","ParamName","Value")

(PBA,SetMedia,"NodeID","DeviceID","DMX_Folder_ID","DMX_File_ID")

(PBA,SetMesh,"NodeID","DeviceID","DMX_Folder_ID","DMX_File_ID")

(PBA,SetCue,"SeqID","CUE_ID")

(PBA,SetSeq,"SeqID","Mode") - Mode "Play","Pause", "Stop")

(PBA,SetTime,"SeqID",Hours,Minutes,Seconds,Frames)

(PBA,NextFrame,"SeqID")

(PBA,LastFrame,"SeqID")

(PBA,NextCue,"SeqID")

(PBA,LastCue,"SeqID")

(PBA,SeqLevel,"Value")

UDP COMMUNICATION -
Send messages to ports

(UDPSend,Port,Message)

PB Server V4 Control

(PBShutdownAll)
(PBRebootAll)
(PBShutdown,IP)
(PBReboot,IP)
(PBStartMaster,IP)
(PBStartClient,IP)
(PBClose,IP)

CHANGELOG

Rev 12
- HTTP Querystring listener built-in.
This means you can use a webbrowser or link to send commands as well ex:
http://192.168.0.141/?(Commandstring)
- Comport Input Support
- All Inputs TCP/UDP/HTTP/COM can send input messages asynchronously
- multiple commands can be send in one message

Rev7
- UDP Send and Receive Support
- PB Server Network Controls, for application commands and system shutdown
- General Code Cleanup Rev6 - support for broken packets, AMX related fix
- Autoconnect as Start option
- TCP connection close optimization
- PB Spark will always send a null character upon null receive
- Revision ID shown GUI
- New Feature Comport Send Hex-codes

v1.0.0.9
- PBAutomation v4 commands added

v1.0.0.8
- New Feature Application Stop to terminate running programs and processes

v1.0.0.7
- New Feature Remote PB Automation Support
- Help page display

v1.0.0.6
- New Feature Fullscreen video player

- New Feature Com Port ASCII communication
- fixed pending Network Process on closing application

6.6 External Control (DMX,Midi,...)

Pandoras Box offers many ways to use various industry standard control protocols.

You may want to remote control lighting or sound or interact with media through contact closures or sensors.

With Pandoras Box Spark, even Windows PC applications can be controlled and triggered at the right cue of the show.

Currently supported protocols are:
TCP/IP, RS232/422, SMPTE, MIDI, MSC, Art-Net, MA-NET, DMX and CITP.

Please find in the following topics the detailed information regarding the listed protocols. Our software Widget Designer [786] offers a huge number of commands [1520] explicitly written for controlling Pandoras Box and even the "WD Free" edition has some control elements [818] available to interface with PB.

Input Protocols	Output Protocols
DMX Tables [708] (Patch Sheets for all products) [708]	
DMX Input [706]	DMX Output [721]
MIDI [716]	
MSC (MIDI Show Control) [716]	
Serial Link [719]	Serial Link [722]
SMPTE Input [719]	SMPTE Output [723]
	Spark [722]
TCP / IP [720]	TCP / IP [723]
UDP - PB Automation and SDK [721]	

6.6.1 Input Protocols

Since Pandoras Box can be used in many different configurations and applications, you may choose from several available control protocols to remote control or interact with the system.
- DMX Input [706]
- Midi [716]
- MSC Midi Show Control [716]
- Serial Link' [719]
- SMPTE Input [719]
- TCP/IP [720]
- UDP - PB Automation [721]

The chapter about Output Protocols [721] explains how to control other devices with Pandoras Box.

6.6.1.1 DMX Input

DMX input can be used in many different ways with Pandoras Box. The entire system is mainly network based. DMX input is mainly designed to remote control individual sequences, layers and output settings of your Pandoras Box system via an external DMX device, e.g. a lighting desk.
If you, on the other hand, are interested in controlling DMX devices with Pandoras Box itself, please read the topic covering the DMX Output [721].
For any DMX input the following settings need to be done.

1 - Connection
Connect the Master system to the remote system. In general, only the Master needs to be connected as it controls its Clients. In other words, it is not necessary to connect the Client with the remote device.

2 - Patch
In case you like to remote control a sequence, select the according sequence in the Project tab [278] and open the Patch tab [228]. Enter the channel, subnet and universe. The patch must accord to the patch in the remote device. Please load a fixture or see the DMX tables [708] for more information.
In case of remote controlling layers from the Master itself, select the Master in the Device Tree tab [173] and patch it.

In case of remote controlling layers from Clients, drag the Client from the Asset tab [131] or Device Types tab [183] into the Device Tree tab first. Select the Client node and patch it.

3 - Configuration
Activate the DMX input in the Configuration tab [139] in the section Remote Control Protocols [148]. You can choose a dedicated network adapter.

As soon as data is sent to the Master system, all patched layers or sequence parameters are remote controlled.

If you like to work with media files, i.e. assign it to a layer, it needs to have a so called File and Folder ID. This can be assigned in the File Inspector [191] as well as Folder Inspector [197]. Pandoras Box supports a CITP based thumbnail exchange.

MA-Net

To interface with GrandMA lighting consoles, please enable the MA-Net in the Configuration tab [139] > Remote Control Protocols [148].
Please note that MA-Net supports up to 256 DMX universes in MA-Net Series 2.

Art-Net

As one of the most popular Ethernet protocols, Art-Net can be used to control multiple units with multiple DMX universes via a single Ethernet network.
To use Art-Net please connect your lighting system that is Art-Net enabled and make sure that both lighting control and Pandoras Box system are in the same IP address range.

The first Art-Net specification demands a 2.xxx.xxx.xxx IP address, the most common used subnet mask is 255.0.0.0 . Pandoras Box is still tied to this specification when working with DMX fixtures (i.e. sending Art-Net data). If you are working witch matrix patches [2077], you may address a different IP range there. Since version 5.3 the IP address does not necessarily need to be in the 2.x.x.x range to be able to receive Art-Net.

To enable Art-Net within a Pandoras Box Master system, please activate the Art-Net Mode in the Configuration tab [139] > Remote Control Protocols [148].

DMX Link In

DMX input via USB is supplied by the DMX Link USB interface cable [1999]. To enable it, press the input button "DMX via USB" in the Configuration tab [139] > Remote Control Protocols [148].
Please be aware that a single USB DMX interface will allow you to control a maximum of 512 DMX channels on subnet 0 and universe 0.

DMX Link 8

The DMX link 8 converts up to 10 DMX universes to Ethernet (Art-Net protocol). Backwards, 8 DMX universes can be converted from Art-Net to DMX.
Please see the DMX Link 8 Device Manual [2009] for further information.

Streaming ACN

In the Configuration tab > Remote Control Protocols [149], click the button "Run sACN" if you want to control Pandoras Box via Streaming ACN (sACN).
Please note that Streaming CAN does not use Universe 0, so do not patch below Universe 1.

6.6.1.1.1 DMX Tables

Pandoras Box is designed to be fully controlled via DMX or Art-Net from any DMX Lighting Controller. The general workflow how to work with DMX Input is described in the previous chapter [706].

In this section you will find all necessary information about the individual DMX fixture libraries. Since version 8, all products merged into one license. Hence the "Player" fixtures are not of interest anymore.

DMX Table - Sequence Control [708] - to control the Master's timeline
DMX Table - Audio Track [708] - to control any Audio Track [661]
DMX Table - Video Layer [709] - to control any Video Layer [647]
DMX Table - Light Layer [711] - to control any Light Layer [666]
DMX Table - Camera Device [712] - to control any Camera Layer [675]
DMX Table - Output Device [713] - to control any Output Layer [682]

6.6.1.1.1.1 DMX Table - Sequence Control

Download: DMX Tables Sequence V8.pdf or DMX Tables V8_All.pdf

5 CH	Parameter		Values	Meaning	Default
1	Opacity	8 Bit	0 1.255	Sequence not visible Opacity for all layers on sequence	255
2	Transport Control	8 Bit	0 128 255	Pause Play Stop	0
3	Cue	8 Bit	0 1.255	no Cue Cue ID	0
4	Frame	16 Bit	0 65535	Set nowpointer to frame pos	0

6.6.1.1.1.2 DMX Table - Audio Track

Download: DMX Tables Audio Track V8.pdf or DMX Tables V8_All.pdf

11 CH	Parameter		Values	Meaning	Default
1	Folder	8 Bit	0 1..255	no Folder Folder ID	0
2	Media	8 Bit	0 1..255	no Media File ID	0
3	Video Control	8 Bit	0 64 128 192	Stop Play Once Pause Play Loop	192
4	Volume	16 Bit	0 32768 46300 65535	no Audio -96db 0db 3db full Audio	0
6	Inpoint	16 Bit	0 65535	File Beginning End of File	0
8	Outpoint	16 Bit	0 65535	File Beginning End of File	65535
10	Pan	16 Bit	0 32768 65535	Left Ch only Center - L+R Ch Right Ch only	32768

6.6.1.1.1.3 DMX Table - Video Layer

Download: DMX Tables Video Layer V8.pdf or DMX Tables V8_All.pdf

	Console 97 CH	Standard 53 CH	Parameter		Values	Meaning	Default
Media	1	1	Folder	8 Bit	0 1..255	no Folder Folder ID	0
	2	2	Media	8 Bit	0 1..255	no Media File ID	0
Mesh	3	3	Object Folder	8 Bit	0 1..255	no Folder Folder ID	0
	4	4	Object Media	8 Bit	0 1..255	no Object File ID	0
Opacity	5	5	Opacity	8 Bit	0 1..255	Layer not visible Opacity	0
Playback	6	6	Video Control	8 Bit	0 64 128 192	Stop Play Once Pause Play Loop	192
	7	7	Speed	8 Bit	0..127 128 129..255	10% to 100% 1:1 Speed (100%) 100% to 500%	128
	8	8	Inpoint	16 Bit	0 65535	File Beginning End of File	0
	10	10	Outpoint	16 Bit	0 65535	File Beginning End of File	65535
Audio	12	12	Volume	16 Bit	0 32768 46300 65535	No Audio, -96dB 0dB 3dB Max. Volume	0
Position			***By default, the GUI displays position parameters as pixel values but incoming DMX values are in generic units. To display generic units, see chapter "Unit Management 160".*** ***One display width equals 16.000 units, the height is calculated by the aspect ratio.*** ***Resolution: 1 DMX step = 0.008 units***				
	14	14	X Pos	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
	16	16	Y Pos	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	32768
	18	18	Z Pos	16 Bit	0 32768 65535	-256.000 units backwards Center +256.000 units forwards	32768
Rotation			***Resolution: 1 DMX step = 0.033°***				
	20	20	X Angle	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
	22	22	Y Angle	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
	24	24	Z Angle	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
			Mode Channel switches between "Angle" and "Speed"				
	26	26	X Rot Mode	8 Bit	0 1	Fixed Angle Rotation Speed	0
	27	27	Y Rot Mode	8 Bit	0 1	Fixed Angle Rotation Speed	0
	28	28	Z Rot Mode	8 Bit	0 1	Fixed Angle Rotation Speed	0
			Resolution: 1 DMX step = 1 rotation per hour				
	29	29	X Speed	16 Bit	0 32768 65535	Max. rotation speed counterclockwise No continuous rotation Max. rotation speed clockwise	32768
	31	31	Y Speed	16 Bit	0 32768 65535	Max. rotation speed counterclockwise No continuous rotation Max. rotation speed clockwise	32768
	33	33	Z Speed	16 Bit	0 32768 65535	Max. rotation speed counterclockwise No continuous rotation Max. rotation speed clockwise	32768

Scale			***Resolution: 1 DMX step = 0.001 unit***				
	35	35	X Scale	16 Bit	0 1000 2000 65535	Not visible Original size Double size Maximum scaling 65x	1000
	37	37	Y Scale	16 Bit	0 1000 2000 65535	Not visible Original size Double size Maximum scaling 65x	1000
	39	39	Z Scale	16 Bit	0 1000 2000 65535	Not visible Original size Double size Maximum scaling 65x	1000
Rotation Pivot			***See "Position" Note***				
		41	X Rot Pivot	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
		43	Y Rot Pivot	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	32768
		45	Z Rot Pivot	16 Bit	0 32768 65535	-256.000 units backwards Center +256.000 units forwards	32768
Scale Pivot		47	X Scale Pivot	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
		49	Y Scale Pivot	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	32768
		51	Z Scale Pivot	16 Bit	0 32768 65535	-256.000 units backwards Center +256.000 units forwards	32768
Blend Mode		53	Blend Mode	8 Bit	0 1..255	Normal Blend Mode Blend Mode ID	0
Effects			***All MultiFX have nine 16Bit parameter channels. Not all parameters are used for all FX. See FX List 327 for detailed information on each FX.***				
	41		FX1 Selection	8 Bit	0 1..255	No FX FX ID	0
	42		FX 1.1	16 Bit	0..65535	Parameter 1	32768
	44		FX 1.2	16 Bit	0..65535	Parameter 2	32768
	46		FX 1.3	16 Bit	0..65535	Parameter 3	32768
	48		FX 1.4	16 Bit	0..65535	Parameter 4	32768
	50		FX 1.5	16 Bit	0..65535	Parameter 5	32768
	52		FX 1.6	16 Bit	0..65535	Parameter 6	32768
	54		FX 1.7	16 Bit	0..65535	Parameter 7	32768
	56		FX 1.8	16 Bit	0..65535	Parameter 8	32768
	58		FX 1.9	16 Bit	0..65535	Parameter 9	32768
	60		FX2 Selection	8 Bit	0 1..255	No FX FX ID	0
	61		FX 2.1	16 Bit	0..65535	Parameter 1	32768
	63		FX 2.2	16 Bit	0..65535	Parameter 2	32768
	65		FX 2.3	16 Bit	0..65535	Parameter 3	32768
	67		FX 2.4	16 Bit	0..65535	Parameter 4	32768
	69		FX 2.5	16 Bit	0..65535	Parameter 5	32768
	71		FX 2.6	16 Bit	0..65535	Parameter 6	32768
	73		FX 2.7	16 Bit	0..65535	Parameter 7	32768
	75		FX 2.8	16 Bit	0..65535	Parameter 8	32768
	77		FX 2.9	16 Bit	0..65535	Parameter 9	32768
	79		FX 3 Selection	8 Bit	0 1..255	No FX FX ID	0
	80		FX 3.1	16 Bit	0..65535	Parameter 1	32768
	82		FX 3.2	16 Bit	0..65535	Parameter 2	32768
	84		FX 3.3	16 Bit	0..65535	Parameter 3	32768
	86		FX 3.4	16 Bit	0..65535	Parameter 4	32768
	88		FX 3.5	16 Bit	0..65535	Parameter 5	32768
	90		FX 3.6	16 Bit	0..65535	Parameter 6	32768
	92		FX 3.7	16 Bit	0..65535	Parameter 7	32768
	94		FX 3.8	16 Bit	0..65535	Parameter 8	32768
	96		FX 3.9	16 Bit	0..65535	Parameter 9	32768

6.6.1.1.1.4 DMX Table - Light Layer

Download: DMX Tables Light Layer V8.pdf or DMX Tables V8_All.pdf

	Console 34 CH	Standard 39 CH	Parameter		Values	Meaning	Default
Media	1	1	Folder	8 Bit	0 1..255	no Folder Folder ID	0
	2	2	Media	8 Bit	0 1..255	no Media File ID	0
Intensity	3	3	Intensity	8 Bit	0 1..255	Light not visible Intensity	255
Playback	4	4	Video Control	8 Bit	0 64 128 192	Stop Play Once Pause Play Loop	0
	5	5	Speed	8 Bit	0..127 128 129..255	10% to 100% 1:1 Speed (100%) 100% to 500%	128
	6	6	Inpoint	16 Bit	0 65535	File Beginning End of File	0
	8	8	Outpoint	16 Bit	0 65535	File Beginning End of File	65535
Position			***By default, the GUI displays position parameters as pixel values but incoming DMX values are in generic units. To display generic units, see chapter "Unit Management 160". One display width equals 16.000 units, the height is calculated by the aspect ratio. Resolution: 1 DMX step = 0.008 units***				
	10	10	X Pos	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
	12	12	Y Pos	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	34048
	14	14	Z Pos	16 Bit	0 32768 65535	-256.000 units backwards Center +256.000 units forwards	29568
Target	16	16	X Pos	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
	18	18	Y Pos	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	32768
	20	20	Z Pos	16 Bit	0 32768 65535	-256.000 units backwards Center +256.000 units forwards	32768
Color	22	22	Red	8 Bit	0..255	Red level, Color Picker	255
	23	23	Green	8 Bit	0..255	Green level, Color Picker	255
	24	24	Blue	8 Bit	0..255	Blue level, Color Picker	255
Settings	25	25	Angle	16 Bit	0 7282 65535	0.000 20.000 180.000	7282
	27	27	Aspect Ratio	16 Bit	0 3277 65535	0 1 20	3277
	29	29	Z Roll	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
		31	Near Plane	16 Bit	0..65535		5
		33	Far Plane	16 Bit	0..65535		5000
	31	35	Tolerance	16 Bit	0 3277 65535	0.000 0.050 1.000	3277
	33	37	Softness	16 Bit	0 16384 65535	0 250 1000	16384
Mode		39	Projection Mode	8 Bit	0 1	Perspective Orthogonal	0

6.6.1.1.1.5 DMX Table - Camera Device

Download: DMX Tables Camera Device V8.pdf or DMX Tables V8_All.pdf

	Console	Standard					
	23 CH	**31 CH**	**Parameter**		**Values**	**Meaning**	**Default**
Mode	1	1	Projection Mode	8 Bit	0 1	Perspective Orthogonal	0
Viewpoint Position			***By default, the GUI displays position parameters as pixel values but incoming DMX values are in generic units. To display generic units, see chapter "Unit Management 160 ". One display width equals 16.000 units, the height is calculated by the aspect ratio. Resolution: 1 DMX step = 0.008 units***				
	2	2	X Pos	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
	4	4	Y Pos	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	32768
	6	6	Z Pos	16 Bit	0 32768 65535	-256.000 units backwards Center +256.000 units forwards	29568
Target Position	8	8	Target X Pos	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
	10	10	Target Y Pos	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	32768
	12	12	Target Z Pos	16 Bit	0 32768 65535	-256.000 units backwards Center +256.000 units forwards	32768
Settings	14	14	FOV	16 Bit	0 17745 65535	0.001 35.489 131.070	17745
		16	Near Plane	16 Bit	0..65535		5
		18	Far Plane	16 Bit	0..65535		50000
	16	20	Aspect Ratio	16 Bit	0 10924 32768 65535	0.000 1.000 3.000 6.000	10924
Lens Shift	18	22	X Offset	16 Bit	0 32768 65535	-256.000 units left Center +256.000 units right	32768
	20	24	Y Offset	16 Bit	0 32768 65535	-256.000 units down Center +256.000 units up	32768
	22	26	Z Roll	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
Background Color		28	Red	8 Bit	0 255	Red level, Color Picker	0
		29	Green	8 Bit	0 255	Green level, Color Picker	0
		30	Blue	8 Bit	0 255	Blue level, Color Picker	0
		31	Alpha	8 Bit	0 255	Alpha Level	255

6.6.1.1.1.6 DMX Table - Output Device

Download: DMX Tables Output Device V8.pdf or DMX Tables V8_All.pdf

	Console 124 CH	Standard 125 CH	Parameter		Values	Meaning	Default
Matrix		1	Matrix Folder	8 Bit	0 1..255	no Folder Folder ID	0
		2	Matrix File	8 Bit	0 1..255	no Matrix File ID	0
State	1	3	Bypass	8 Bit	0 1	Render Bypass	0
Object		4	Camera	16 Bit	-	cannot be assigned via DMX control	
	2	6	Object Folder	8 Bit	0 1..255	no Folder Folder ID	0
	3	7	Object Media	8 Bit	0 1..255	no Object File ID	0
	4	8	Opacity	8 Bit	0 1..255	Not visible Opacity	255
Position			***By default, the GUI displays position parameters as pixel values but incoming DMX values are in generic units. To display generic units, see chapter "Unit Management 160 ". One display width equals 16.000 units, the height is calulated by the aspect ratio. Resolution: 1 DMX step = 0.008 units***				
	5	9	X Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
	7	11	Y Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
	9	13	Z Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
Rotation			***Resolution: 1 DMX step = 0.033°***				
	11	15	X Angle	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
	13	17	Y Angle	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
	15	19	Z Angle	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
			Mode Channel switches between "Angle" and "Speed"				
	17	21	X Rot Mode	8 Bit	0 1	Fixed Angle Rotation Speed	0
	18	22	Y Rot Mode	8 Bit	0 1	Fixed Angle Rotation Speed	0
	19	23	Z Rot Mode	8 Bit	0 1	Fixed Angle Rotation Speed	0
			Resolution: 1 DMX step = 1 rotation per hour				
	20	24	X Speed	16 Bit	0 32768 65535	Max. rotation speed counterclockwise No continuous rotation Max. rotation speed clockwise	32768
	22	26	Y Speed	16 Bit	0 32768 65535	Max. rotation speed counterclockwise No continuous rotation Max. rotation speed clockwise	32768
	24	28	Z Speed	16 Bit	0 32768 65535	Max. rotation speed counterclockwise No continuous rotation Max. rotation speed clockwise	32768
Scale			***Resolution: 1 DMX step = 0.001 unit***				
	26	30	X Scale	16 Bit	0 1000 2000 65535	Not visible Original size Double size Maximum scaling 65x	1000
	28	32	Y Scale	16 Bit	0 1000 2000 65535	Not visible Original size Double size Maximum scaling 65x	1000
	30	34	Z Scale	16 Bit	0 1000 2000 65535	Not visible Original size Double size Maximum scaling 65x	1000

Rotation Pivot			***See "Position" Note***				
		36	X Rot Pivot	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		38	Y Rot Pivot	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		40	Z Rot Pivot	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
Scale Pivot			***See "Position" Note***				
		42	X Scale Pivot	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		44	Y Scale Pivot	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		46	Z Scale Pivot	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
Viewpoint Position			***See "Position" Note***				
		48	X Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		50	Y Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		52	Z Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	29568
Target Position			***See "Position" Note***				
		54	Traget X Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		56	Target Y Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		58	Target Z Pos	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
Settings		60	FOV	16 Bit	0 17745 65535	0.001 35.489 131.070	17745
		62	Near Plane	16 Bit	0..65535		5
		64	Far Plane	16 Bit	0..65535		50000
		66	Aspect Ratio	16 Bit	0 10924 32768 65535	0 1.000 3.000 6.000	10924
Lens Shift		68	X Offset	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		70	Y Offset	16 Bit	0 32768 65535	-256.000 units Center +256.000 units	32768
		72	Z Roll	16 Bit	0 32768 65535	-1080.00° 0° +1080.00°	32768
Background Color		74	Red	8 Bit	0..255	Red level, Color Picker	0
		75	Green	8 Bit	0..255	Green level, Color Picker	0
		76	Blue	8 Bit	0..255	Blue level, Color Picker	0
		77	Alpha	8 Bit	0..255	Alpha Level	255
Pixel Warp		78	Pixel Warp Folder	8 Bit	0 1..255	no Folder Folder ID	0
		79	Pixel Warp Media	8 Bit	0 1..255	no Pxel Warp File File ID	0
Keystone	32	80	Keystone L	16 Bit	0..65535		32768
	34	82	Keystone L Pos	16 Bit	0..65535		32768
	36	84	Keystone R	16 Bit	0..65535		32768
	38	86	Keystone R Pos	16 Bit	0..65535		32768
	40	88	Keystone T	16 Bit	0..65535		32768
	42	90	Keystone T Pos	16 Bit	0..65535		32768

	44	92	Keystone B	16 Bit	0..65535		32768
	46	94	Keystone B Pos	16 Bit	0..65535		32768
	48	96	Linearity X	16 Bit	0..65535		32768
	50	98	Linearity Y	16 Bit	0..65535		32768
Softedge		100	Blend Folder	8 Bit	0 1..255	no Folder Folder ID	0
		101	Blend Media	8 Bit	0 1..255	no Blend File File ID	0
	52	102	Softedge Left	16 Bit	0..65535		0
	54	104	Softedge Left Curve	16 Bit	0..65535		32768
		106	Softedge Left Marker	8 Bit	0 1..255	no marker black to white marker	0
		107	Softedge Left Maker Width	8 Bit	0..255		32
	56	108	Softedge Right	16 Bit	0..65535		0
	58	110	Softedge Right Curve	16 Bit	0..65535		32768
		112	Softedge Right Marker	8 Bit	0 1..255	no marker black to white marker	0
		113	Softedge Right Maker Width	8 Bit	0..255		32
	60	114	Softedge Top	16 Bit	0..65535		0
	62	116	Softedge Top Curve	16 Bit	0..65535		32768
		118	Softedge Top Marker	8 Bit	0 1..255	no marker black to white marker	0
		119	Softedge Left Top Width	8 Bit	0..255		32
	64	120	Softedge Bottom	16 Bit	0..65535		0
	66	122	Softedge Bottom Curve	16 Bit	0..65535		32768
		124	Softedge Bottom Marker	8 Bit	0 1..255	no marker black to white marker	0
		125	Softedge Bottom Maker Width	8 Bit	0..255		32
Effects			***All MultiFX have nine 16Bit parameter channels. Not all parameters are used for all FX. See FX List [327] for detailed information on each FX.***				
	68		FX1 Selection	8 Bit	0 1..255	No FX FX ID	0
	69		FX 1.1	16 Bit	0..65535	Parameter 1	32768
	71		FX 1.2	16 Bit	0..65535	Parameter 2	32768
	73		FX 1.3	16 Bit	0..65535	Parameter 3	32768
	75		FX 1.4	16 Bit	0..65535	Parameter 4	32768
	77		FX 1.5	16 Bit	0..65535	Parameter 5	32768
	79		FX 1.6	16 Bit	0..65535	Parameter 6	32768
	81		FX 1.7	16 Bit	0..65535	Parameter 7	32768
	83		FX 1.8	16 Bit	0..65535	Parameter 8	32768
	85		FX 1.9	16 Bit	0..65535	Parameter 9	32768
	87		FX2 Selection	8 Bit	0 1..255	No FX FX ID	0
	88		FX 2.1	16 Bit	0..65535	Parameter 1	32768
	90		FX 2.2	16 Bit	0..65535	Parameter 2	32768
	92		FX 2.3	16 Bit	0..65535	Parameter 3	32768
	94		FX 2.4	16 Bit	0..65535	Parameter 4	32768
	96		FX 2.5	16 Bit	0..65535	Parameter 5	32768
	98		FX 2.6	16 Bit	0..65535	Parameter 6	32768
	100		FX 2.7	16 Bit	0..65535	Parameter 7	32768
	102		FX 2.8	16 Bit	0..65535	Parameter 8	32768
	104		FX 2.9	16 Bit	0..65535	Parameter 9	32768
	106		FX 3 Selection	8 Bit	0 1..255	No FX FX ID	0
	107		FX 3.1	16 Bit	0..65535	Parameter 1	32768
	109		FX 3.2	16 Bit	0..65535	Parameter 2	32768
	111		FX 3.3	16 Bit	0..65535	Parameter 3	32768
	113		FX 3.4	16 Bit	0..65535	Parameter 4	32768
	115		FX 3.5	16 Bit	0..65535	Parameter 5	32768
	117		FX 3.6	16 Bit	0..65535	Parameter 6	32768
	119		FX 3.7	16 Bit	0..65535	Parameter 7	32768
	121		FX 3.8	16 Bit	0..65535	Parameter 8	32768
	123		FX 3.9	16 Bit	0..65535	Parameter 9	32768

6.6.1.2 Midi

The Midi Input protocol supports the following message types:
Channel Voice Messages
All Note-On messages are processed.

PRESET CONTROL

The channel is mapped to the Pandoras Box preset tree. Channel 1 = GLOBAL ... Channel 9 = CTRL

The note calls the equivalent preset. The notes are being processed with a negative offset of 35 to assign the value 1 to the lowest note of a standard midi keyboard

SEQUENCE CONTROL

Channel 10 is used for transport control. Each octave controls one sequence:

The lowest octave (Offset 35) controls sequence 1, the next octave sequence 2 etc,...

Play, (C)
Pause, (D)
Stop, (E)
Prev Cue + Play, (F)
Prev Cue + Pause (F#)
Next Cue + Play (G)
Next Cue + Pause (G#)

GUI - Configuration Tab

Midi Device: Choose an installed Midi Device.
ID: ID of MSC Message Mapping
Sequence: Choose the sequence to be controlled via MSC.
Run Midi: Launch MSC and MIDI Input.

6.6.1.3 MSC Midi Show Control

The Midi Show Control (MSC) input protocol allows remote controlling the Pandoras Box timeline and layer parameters. MSC commands are based on System Exclusive messages (SysEx) according to the MSC specification.

MSC Syntax

This shows how the format of a MSC (Midi Show Control message) is defined and what Pandoras Box accepts. Below you find some examples.

Beginning of SysEx Message	<device_ID>	Defines message as SysEx	<command_format>	<command>	<data>	End of Message
F0h 7Fh	= PB Sequence ID as set up in the Configuration tab > Remote Control Prototcols > Midi [149] 01h = Seq1 etc.	02h	01h Lighting 30h Video 40h Projection 7Fh All-Types (PB processes all formats)	01h GO 02h STOP 03h RESUME 05h LOAD 06h SET 0Ah RESET 0Bh GO_OFF	<data> is optional and described below, all others are needed	F7h
In other words, if you like to control Sequence 1 and choose the "All Types" command format, your string should look as follows.						

F0h 7Fh	01h	02h	7Fh	<command and optional data>	F7h

<data>
All data must be encoded from ASCII Format. So the decimal number 1 in ASCII is ONE and in hex 31h.
Multiple data elements must be separated (SET) with 00h.
Separators as dec.(2Eh) are accepted but the next value is not processed.

The chapter "Syntax TCP- / UDP- / Serial Messages [944]" shows a table with hexadecimal values translated to more human readable decimal values and ASCII.

In case you like to test the MSC feature with Widget Designer [786], you can send a command called MidiSendSysEx [1559]. As shown in the command's help, the syntax is slightly different: instead of the above shown "XXh" format it should be "hXX" in WD.

Examples for PB Sequence Control

As above, for the examples the Sequence ID should be 1 and the command format is "All Types", which gives us this string:

```
F0h 7Fh 01h 02h 7Fh <command> <optional data> F7h
```

GO-command = 01h
GO - without a cue:
a) if the last command was a LOAD cue, the loaded cue will be called.
b) in all other cases the timeline will continue from its actual play position.

<command> = GO = 01h
<data> = not needed

```
F0h 7Fh 01h 02h 7Fh 01h F7h
```

GO - in conjunction with a cue:
Jump to Cue + Play (the cue will overwrite a previously loaded cue)

<command> = GO = 01h
<data> = Cue, e.g. Cue 24 is TWO FOUR which is 32h 34h

```
F0h 7Fh 01h 02h 7Fh 01h 32h 34h F7h
```

It is valid to send a cue 24.001 which is : 32h 34h (TWO FOUR) 2E (decimal separator) 30 30 31 (ZERO ZERO ONE) 00 (SET separator is optional)

```
F0h 7Fh 01h 02h 7Fh 01h 32h 34h 2E 30 30 31 00 F7h
```

As decimal places for cue numbers are not supported in Pandoras Box, you can use the check box option "Use Cue Subsection" in the Configuration tab > Remote Control Prototcols > Midi [149]. Doing this will multiply the incoming cue command x1000 (for example: GO to Cue 24.001 will no be interpreted as GO to Cue 24001). With this option three decimal places are supported.

PB also accepts this string which is according to the GrandMA format with cue list and cue path – only the first cue is processed.

```
F0h 7Fh 01h 02h 7Fh 01h 32h 34h 2Eh 30h 30h 31h 00h 31h 00h 32h 00h F7h
```

Go to Cue 0 (ZERO = 30h) can have different result according to the option "Cue ID 0 Handling" in the Configuration tab > Remote Control Prototcols > Midi [149].

```
F0h 7Fh 01h 02h 7Fh 01h 30h F7h
```

Regular Cue if it exists, PB goes to Cue 0 and plays the timeline

Ignore	PB ignores the message
Stop	PB stops the timeline
Stop & Reset	PB stops the timeline and resets all active values

STOP-command = 02h
This pauses the PB timeline at the current position.

F0h 7Fh 01h 02h 7Fh 02h F7h

RESUME-command = 03h
This has the same effect as the GO-command.

F0h 7Fh 01h 02h 7Fh 03h F7h

LOAD-command = 05h
LOAD must be used with a cue. The example loads Cue 24 which is processed by the next incoming GO or RESUME command. In other words, the LOAD command by itself is not visible in Pandoras Box.

F0h 7Fh 01h 02h 7Fh 05h 32h 34h F7h F0h 7Fh 01h 02h 7Fh 01h F7h

RESET and GO_OFF-commands = 0Ah or 0Bh
Both commands stop the timeline so that the time is 0:00:00:00

F0h 7Fh 01h 02h 7Fh 0Ah F7h or F0h 7Fh 01h 02h 7Fh 0Bh F7h

Examples for PB Parameter Control

As above, the examples refer to Sequence ID 1 and use the command format "All Types", which gives us the same string to start with:

F0h 7Fh 01h 02h 7Fh <command> <optional data> F7h

SET-command = 06h
This allows you to transmit individual device parameters. The <data> message consists of 4 values which must be followed by a 00h-separator. All decimal values will be ignored.

<data1>: Device ID
<data2>: Layer ID
<data3>: Parameter ID
<data4>: Value

F0h 7Fh 01h 02h 7Fh 06h <Device ID> 00h <Layer ID> 00h <Parameter ID> 00h <Value> 00h F7h

Sample message with:
Device 1 = ONE = 31h
Layer 3 = THREE = 33h
X-Position = see table, EIGTH = 38h
Value 32768 = THREE TWO SEVEN SIX EIGTH = 33h 32h 37h 36h 38h

F0h 7Fh 01h 02h 7Fh 06h 31h 00h 32h 00h 38h 00h 33h 32h 37h 36h 38h F7h

The Value 32768 translates to 0 generic units for all position parameters. Per default, Pandoras Box version 6 displays pixel values where applicable. In the Configuration dialog > Unit Management 160 values can be set back to the old default. In that case value 0 (30h) translates to -256 units and 128 (31h 32h 38h) to -255 etc.

List of Pandoras Box Parameter IDs:

None 0
Opacity 1
Mesh 2
Media 3
Inpoint 4
Outpoint 5
Transport 6
TransFx 7
XPos 8
YPos 9
ZPos 10
XRot 11
YRot 12
ZRot 13
XScale 14
YScale 15
ZScale 16
Colour1 17
Colour2 18
Colour3 19
ColourFx 20
Fx1 21
Fx2 22
Fx3 23
VideoFx 24
XAxis 25
YAxis 26
ZAxis 27
Rot 28
XOffset 29
YOffset 30
Mode 31
KSL 32
KSLR 33
KSR 34
KSRR 35
KST 36
KSTR 37
KSB 38
KSBR 39
LinX 40
LinY 41
SEL 42
SELC 43
SER 44
SERC 45
SET 46
SETC 47
SEB 48
SEBC 49

6.6.1.4 Serial Link

Please note, that Christie Digital Systems Germany has discontinued the Serial Link. Please refer to JLCooper Electronics and their product eBOX.

The optional external SERIAL Link interface provides 4x serial RS 232/422 ports via ethernet.
A SERIAL Link port can be used to control the playback control of the timeline via RS232 or RS422. Each port can be connected to a separate sequence to allow multiple input devices to control individual sequences.

Please see the chapter TCP/IP [720] for information how to connect a Sequence in Pandoras Box to an external device. Once the serial connection is set up, the following commands are valid:

(Play)
Sets the sequence state to play

(Pause)
Sets the sequence state to pause

(Stop)
Stops and rewinds the sequence

(CueID)
ID = enter number based on cue ID to jump to that cue directly

Example:
(Cue13), sets the timeline now-pointer to the time of cue 13

See further instructions in the Serial Link Device Manual [2044].

6.6.1.5 SMPTE Input

The Pandoras Box Master can send and receive LTC SMTPE via the USB SMPTE Link interface [2000].

To connect the SMPTE Link please refer to Configuration tab [139], section SMPTE Time Code [152]. Then go to Sequence Inspector [204] to set up the "Mode", "Offset" and "Stop Action".

6.6.1.6 TCP/IP

This chapter describes how to control Pandoras Box via TCP/IP. For other input protocols, please refer to the main chapter [706].

The built-in TCP/IP interface allows to send basic commands to control the Pandoras Box sequence via Ethernet. The Sequence acts as a TCP Client that can connect to an external TCP Server and receive commands from it. In case, you like to use the (discontinued) Serial Link [2044] interface for that, please refer to the chapter Input Protocols > Serial Link [706].

To control a Sequence, please start a TCP Server on your external device first. The TCP/IP port must be set to 23 and cannot be changed in Pandoras Box.

Then go to the Pandoras Box master and open the Sequence Inspector [204]. Enter the IP address from the control device (which should now show that a Client is connected) and change then the so called "Serial port" to "1".

Before sending the command to control the Sequence, the header information is requested. The header consists of the three bytes "FFh 00h 00h" followed by a forth byte that refers to the just mentioned serial port (or virtual port) which derives from the four Serial Link ports. If PB is set to "1", the hexadecimal value to address it would be "00". If it is set to the maximum port "4", the hexadecimal value is "03". In other words, you can control up to four Sequences via TCP. To make the following examples easier, we assume that the serial port in the Sequence Inspector of PB is set to "1". This results in the following hexadecimal values for the header:
FFh 00h 00h 00h

Once the connection is set up, the following commands are valid to control the Sequence:

ASCII command	Hexadecimal command
(Play)	28 50 6C 61 79 29
Sets the sequence state to "Play".	
(Pause)	28 50 61 75 73 65 29
Sets the sequence state to "Pause".	
(Stop)	28 53 74 6F 70 29
Sets the sequence state to "Stop" which jumps to the time 00:00.	
(CueID) e.g (Cue13)	28 43 75 65 31 33 29
Sets the timeline Nowpointer to the time of cue 13.	

In this example, the header and the control command "Play" is sent. If sending several commands, you can send the header once and then your commands stand-alone or you can always include the header. The examples are based on the scripting commands TCPSend [1616] and TCPSendHex [1617] from Widget Designer [786]. The command arguments are enclosed by round brackets. The first "1" refers to an internal ID for a TCP Server connection that was set up beforehand. The string to be sent is enclosed by quotation marks.

Examples for Widget Designer
TCPSend(1,"[hFF h00 h00 h00]") or TCPSendHex(1,"FF 00 00 00")
Sends header alone as hexadecimal data
TCPSend(1,"[d255 d0 d0 d0]")
Sends header alone as decimal data
TCPSend(1,"(Play)")
Sends command alone
TCPSendHex(1,"FF 00 00 00 28 50 6C 61 79 29")
Sends header and command as hexadecimal data
TCPSend(1,"[hFF h00 h00 h00] (Play)")
Sending header and command in combination of hexadecimal and ASCII data

6.6.1.7 TCP or UDP via SDK

Another way to control Pandoras Box is offered by the SDK and uses a TCP or UDP connection. For more information about the integration and use of the PB Automation SDK please refer to the SDK description [730].

In short, the software development kit (SDK) including the PB Automation allows to integrate Pandoras Box interaction into your custom application. This can be realized with common programming languages such as Visual Basic scripts as well C# and C++. The Pandoras Box Automation allows your custom program to control almost any aspect from the PB software, e.g. parameters on Video Layers, Sequence values and content data.

6.6.2 Output Protocols

Pandoras Box can be set up to output various standard protocols allowing to control a huge amount of other devices:
- DMX Output [721]
- Serial Link [722]
- Spark [722]
- SMPTE Output [723]
- TCP/IP [723]

The chapter about Input Protocols [706] explains how to control Pandoras Box with other devices.

6.6.2.1 DMX Output

DMX output is designed to remote control external DMX device from the timeline in the Pandoras Box Master. You may for example program a synchronized light and video show by including moving lights, spots and other possible DMX devices into the timeline.
If you, on the other hand, are interested in controlling Pandoras Box with a lighting desk for example, please read the topic covering the DMX Input [706].
For any DMX output the following settings need to be done.

1 - Connection
Connect the Master system to the DMX devices.

2 - Patch
All Pandoras Box systems ship with an extensive library of DMX devices. You may access all built-in DMX devices in the Device Type tab [183]. Simply drag and drop the desired library into the Device Tree [173]. If your device is not included, you may write a custom fixture and then drag your custom device into the Device Tree. This is explained in the chapter DMX Devices [691].
Select your device and patch the channel, subnet and universe using the Patch tab [228].

3 - Configuration
Activate the DMX output in the Configuration tab [139] in the section Remote Control Protocols [148]. You can choose a dedicated network adapter.

As soon as a key is stored in the timeline [292], the DMX data will be sent constantly. Only changes to DMX values will be sent over Art-Net. Resetting a DMX node will send all values at once.

Art-Net

As one of the most popular Ethernet protocols, Art-Net can be used to control multiple units with multiple DMX universes via a single Ethernet network.
To use Art-Net please make sure that the lighting system and the Pandoras Box system are in the same IP address range.

The first Art-Net specification demands a 2.xxx.xxx.xxx IP address, the most common used subnet mask is 255.0.0.0 . Pandoras Box is still tied to this specification when working with DMX fixtures (i.e. sending Art-Net data). If you are working witch matrix patches [2077], you may address a different IP range there. Since version 5.3 the IP address does not necessarily need to be in the 2.x.x.x range to be able to receive Art-Net.

To enable Art-Net within a Pandoras Box Master system, please activate the Art-Net Mode in the Configuration tab [139].

DMX Link Out

DMX output via USB is supplied by the DMX Link USB interface cable [1999]. To enable it, press the output button "DMX via USB" in the Configuration tab [139].
Please be aware that a single USB DMX interface will allow you to control a maximum of 512 DMX channels on subnet 0 and universe 0.

DMX Link 8

The DMX link 8 converts up to 10 DMX universes to Ethernet (Art-Net protocol). Backwards, 8 DMX universes can be converted from Art-Net to DMX.
Please see the DMX Link 8 Device Manual [2009] for further information.

6.6.2.2 Serial Link

Please note, that Christie Digital Systems Germany has discontinued the Serial Link. Please refer to JLCooper Electronics and their product eBOX.

The serial control will give you access and control to most routing switchers, projectors, shutters or other device parameters that are remote controllable via the common RS 232 and 422 serial protocols.

In order to control external serial devices over RS 232 or RS 422 you need to add a "Serial Link Device" to your project. Any serial ASCII or hex string can be stored and copied to any point in the timeline. It can be recalled at any time the time cursor hits the stored serial key. This is described in detail in the chapter about the Serial Link Device in Pandoras Box [692].

You will need a serial link device to receive and output serial commands. See further instructions in the Serial Link Device Manual [2044].

6.6.2.3 Spark

for WINDOWS XP (Microsoft .Net 2 Runtime must be installed)

Spark offers you various remote control over external applications as well as standalone Windows XP PCs. Spark is designed to be connected to any Pandoras Box timeline as a Serial Link [2044] TCP IP device.

Spark can be used for remote controlling applications, mouse and keyboard actions as well as displaying full-screen video playback and web page presentations among many other useful remote control features. Pandoras Box Spark is available as a freeware download from our Download-Center.

To get started, install Spark on the desired PC that you want to control (Spark can be accessed also on the same machine that runs Pandoras Box)

COMMAND LIST

See here an overview of remote control features offered by Spark:

COMPORT ASCII COMMUNICATION

SYSTEM COMMANDS - Shutdown & Reboot

WEB BROWSER (Fullscreen)

SOUND COMMANDS

VIDEO PLAYER (Fullscreen)

APPLICATION COMMANDS

KEYBOARD EVENTS

MOUSE EVENTS

DESKTOP FADE TO BLACK -BETA

WAKE ON LAN

PB AUTOMATION REMOTE CONTROL

Please see Spark [699] for detailed information.

6.6.2.4 SMPTE Output

The Pandoras Box Master can send and receive LTC SMTPE via the USB SMPTE Link interface [2000].

To connect the SMPTE Link please refer to Configuration tab [139], section SMPTE Time Code [152]. Then go to Sequence Inspector [204] to set up the "Mode", "Offset" and "Stop Action".

6.6.2.5 TCP/IP

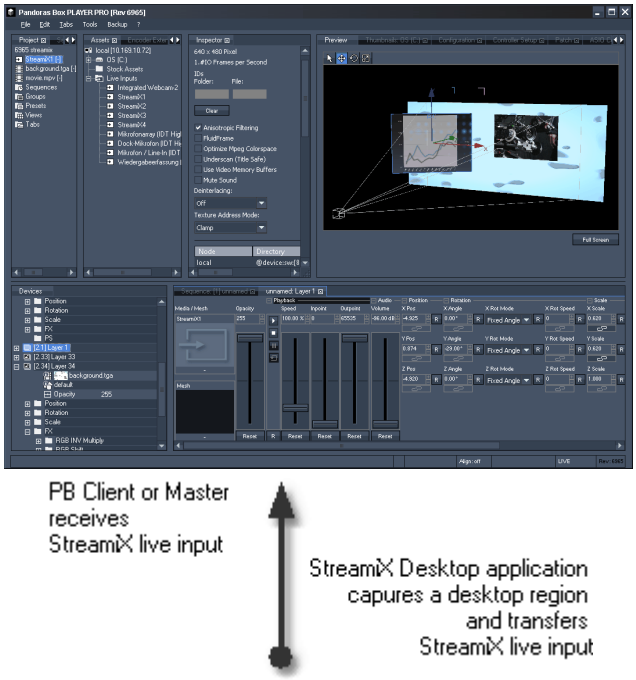
The TCP/IP output of Pandoras Box allows for simple communication to devices such as projectors or matrix switches which are remote controllable via their TCP/IP Server interface.

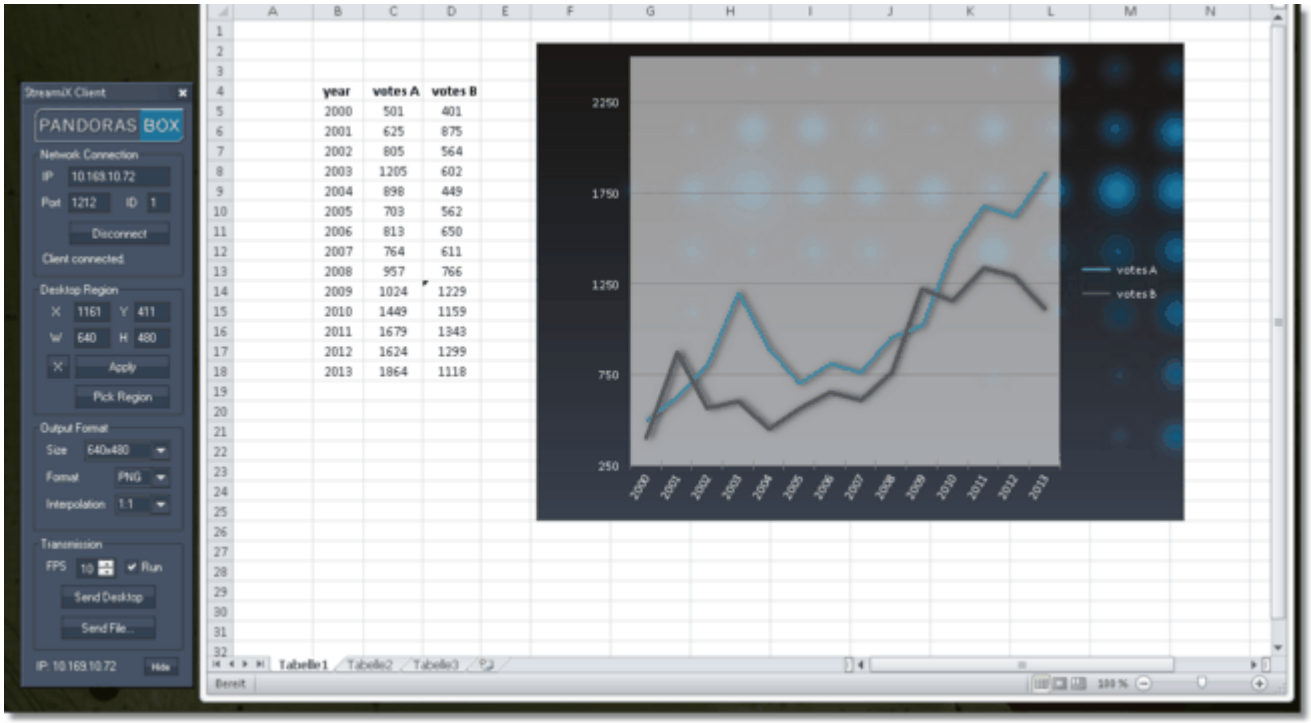
In order to control external TCP devices you need to add a "TCP Device" to your project. Any serial ASCII or hex string can be stored and copied to any point in the timeline. It can be recalled at any time the time cursor hits the stored serial key. This is described in detail in the chapter about the TCP Device in Pandoras Box [693].

6.7 StreamiX Live Input

StreamiX Desktop is a TCP based Desktop Streamer that works as a Live Input in Pandoras Box (PB). This tool streams any Windows desktop region, for example a dedicated area of an Excel sheet or any other application content, and may be used as a Live Input of a PB Client or PB Master directly. In addition it may stream a chosen file from the hard disk. In return you might as well stream the Preview area to other PB Clients.

This way you can send any desktop content directly into Pandoras Box without using capture cards. As the content is transferred via TCP please keep in mind that the tool is not designed to be a fully-fledged substitution for DVI input cards. It is rather meant to stream either low frame rate content or single images or low resolution content. To meet the systems performance it is possible to set up different frame-rates and sizes of the desktop region to be streamed.





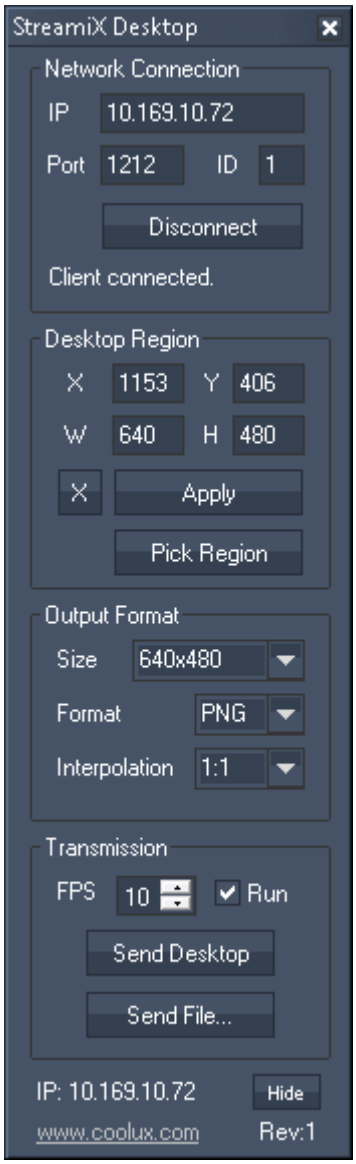
Even though StreamiX is designed to be used as a 1:1 connection, you may use several instances of StreamiX Desktop at the same time on one system. These may either send their data to a single PB Client or to several PB

Clients.
In return one PB Client may receive data from several StreamiX Desktop applications. However, keep your systems performance and network traffic in view, a 1:1 stream may be already the limit!
By rule of thumb one stream with a resolution of 1024x768 px and a frame rate of 30fps will run fluently on a gigabit network.

The total number of StreamiX Live Inputs depends on the license of your PB Client. See the Product Overview [62] for detailed differences.

The Pandoras Box CITP based Desktop Streamer for users of WYSIWYG or Capture Polar is described here [2052].

6.7.1 StreamiX Desktop User Interface



The User Interface is divided into the following sections:

Network Connection [726]

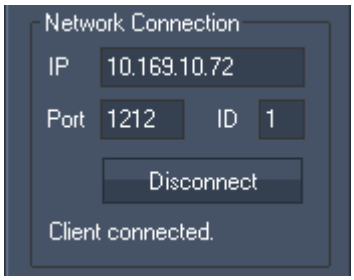
Desktop Region [726]

Output Format [727]

Transmission [727]

General information [728]

6.7.1.1 Network Connection



You may start several instances of StreamiX Desktop Streamer at the same time to pass several TCP streams to PB application. The number shown in the Window Title [] displays the stream number: Pandoras Box CITP Desktop Streamer [1] = Stream 1.

[IP]:
Enter the IP address of the PB Client (or PB Master if stand alone) you want to sent the stream as a live input to.

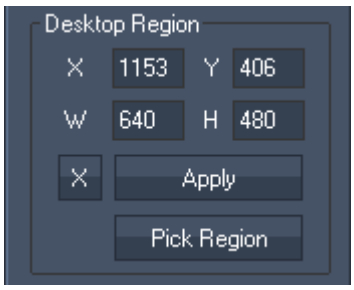
[Port]:
Enter the same port as the receiving PB application is set to.

[ID]:
Enter the ID. Each stream received from the PB Client (or PB Master) must have a unique ID and refers to the StreamiX Live Input with the same number.

[Connect / Disconnect]:
Click [Connect] to connect to the device with the entered IP and Port. As StreamiX Desktop is a TCP client, the connection must be built up in the first place from the PB Client (or PB Master) as this device poses as the TCP server. Please set up the stand alone Master first, or if a PB Client is used, it must additionally be part of a project and the live input must be added to it as an asset already.
After you have done so the TCP connection can be built up and the StreamiX Desktop can pick it up. The text "Waiting for connection..." will change to "Client connected". When the button [Disconnect] is pressed you will read "Client stopped".

In case of closing the PB Client or Master receiving the StreamiX live input "Server lost" will be displayed.

6.7.1.2 Desktop Region



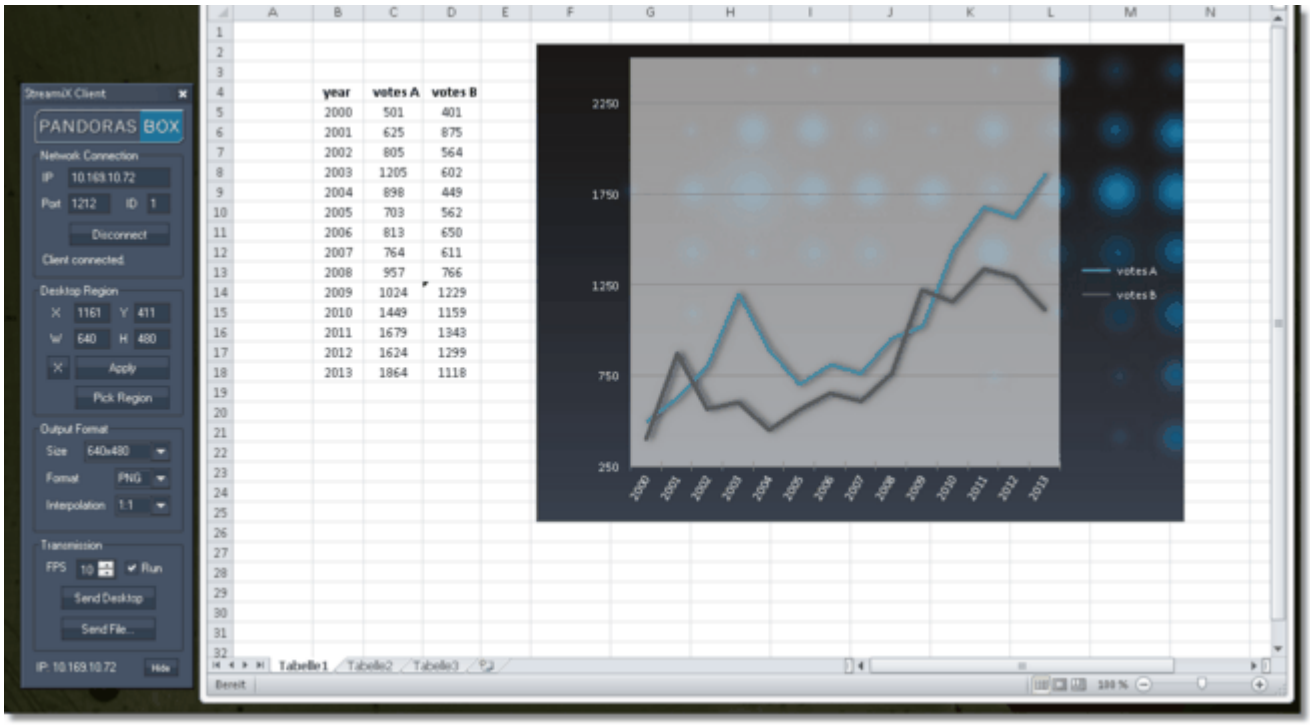
There are two possibilities to define the region of your desktop that is going to be captured for TCP streaming:

Enter the region's starting horizontal pixel [X] and the vertical pixel [Y] as well as the regions width [W] and height [H] into the text fields and press [Apply].

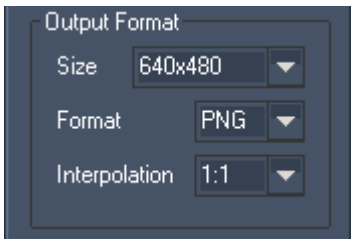
Or click [Pick Region]. The desktop will be overlaid with a transparent white mask and you may span the desired desktop region with the mouse.

Please note:
To capture a desktop region that is not on the primary monitor, enter the region's starting pixel and its size as described in option 1. This can't be done by picking a region with the mouse.

The example shows a typical use. By picking a smaller region, you will stream the graph only without the surrounding columns and rows within the table.



6.7.1.3 Output Format

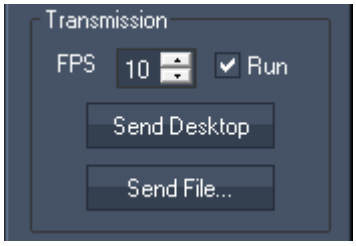


[Size]:
By choosing a size from the drop-down list you define the resolution and aspect ratio of the outgoing stream. Picking the same (or at least a multiple unit) in regards to the captured desktop region and Live Input setting will give you the best result.

[Format]:
Choose whether the frames of the stream are to be encoded as jpeg or PNG format.

[Interpolation]:
If a size has been picked that does not match the captured desktop region, the pixel within the new width and height have to recalculated. Choose whether the calculation should base upon 1:1, Cubic, Linear or Soft interpolation.

6.7.1.4 Transmission



[FPS]:
The Framerate can be set to values between 1 and 30 fps, according to your system's performance and network's capacity.

Please note:
You do not have to press [Apply] as your changes are applied immediately.

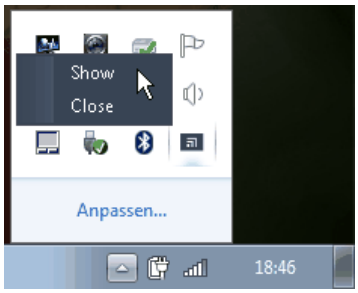
[Run]:
Enable the check box to stream successive frames of the desktop.
Disable the check box to stream a single image only. This may either apply to a desktop or file stream.
Please note that the check box is ticked automatically by the following buttons as well.

[Send Desktop]:
Captures the assigned desktop region with the assigned format and enables automatically the check box [Run]. However, you may tick it again in order to stream a single screen shot of the desktop. This will not interrupt the TCP connection itself but the capturing process.

[Send File...]:
Streams a file from the hard disk and disables the [Run] option. Please note that only single images (BMP, JPG, PNG, GIF) may be streamed.

6.7.1.5 General Information

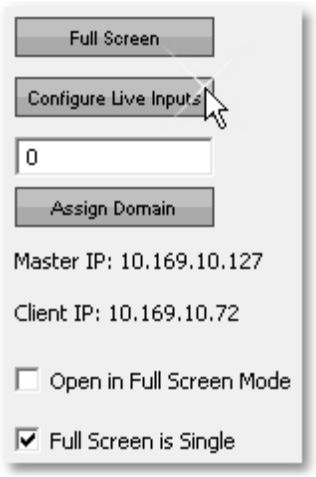
On the left bottom of the menu you find the local computer's IP address and a link to our website.
On the right bottom the revision is displayed.



The [Hide] button will hide the user interface (menu) of StreamiX Desktop. To unhide it again, open the system tray and make a right-click on the StreamiX Desktop icon or make a left click and choose "Show".

6.7.2 StreamiX Live Input Settings

As the StreamiX Desktop application can only pick connections up, it is necessary to establish them with the PB Client or Master beforehand. To do this you need to configure the live input.

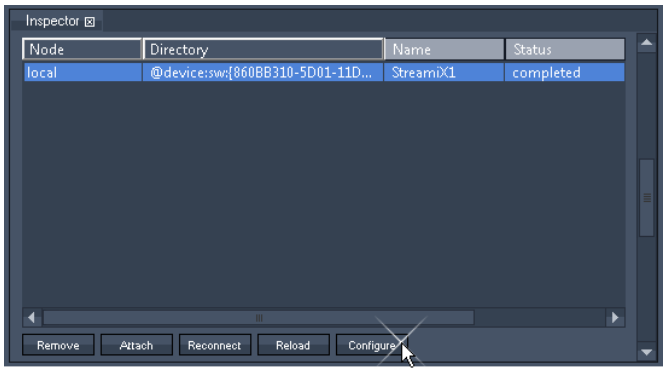


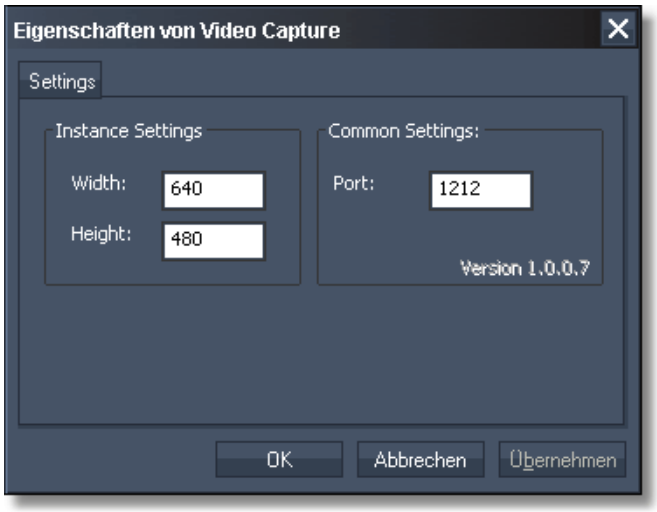
In order to configure the StreamiX live input on a PB Client, the Client must be part of a project and the live input must be added to it as an asset already.

As live inputs are only available (and render-able) on the system were the physical input is, all other system including PB Master cannot display the input.

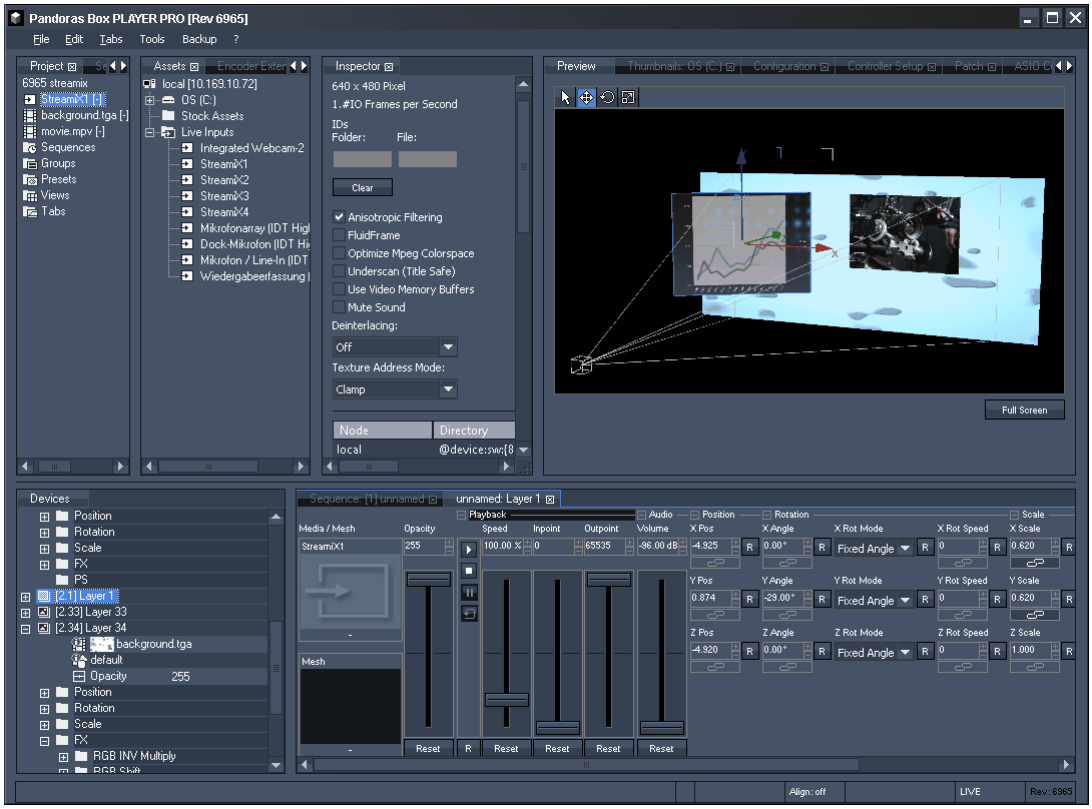
In order to configure the StreamiX live input on a PB Master drag the StreamiX live input from the Assets Tab to the Project Tab as well. Select the StreamiX asset and have a look at its options in the Inspector Tab. Scroll down, select the local entry and click the button "Configure".

Change the incoming resolution and the listening port.





Now the StreamiX live input may be used just like any other still image or video file. It is possible to assign it to a layer and use it in the sequence as usual.



6.8 SDK

The software development kit (SDK) including the PB Automation allows to integrate Pandoras Box interaction into your custom application. This can be realized with common programming languages such as Visual Basic scripts as well C# and C++.
The Pandoras Box Automation allows your custom program to control almost any aspect from the PB software, e.g. parameters on Video Layers, Sequence values and content data. As the Pandoras Box Clients are connected to the Master system, they are controlled indirectly. The control parameters can be sent using the network protocols TCP and UDP.

This chapter is written for:
- software programmers who have little knowledge of Pandoras Box and want to integrate it into their application
- Pandoras Box users who have little programming knowledge but need to write a custom application. Please have a look at the Widget Designer [786] too. One of the main purposes of Widget Designer is to enable users who do not have any programming skills, to create their own custom applications or interfaces. Widget Designer provides a graphical user interface and runtime environment. You can create user-controls such as faders, buttons, labels and many more and instantly use them without compiling any code. The PRO version allows to program visually with so called nodes. By simply connecting those visual control components you may create a dedicated interaction logic.

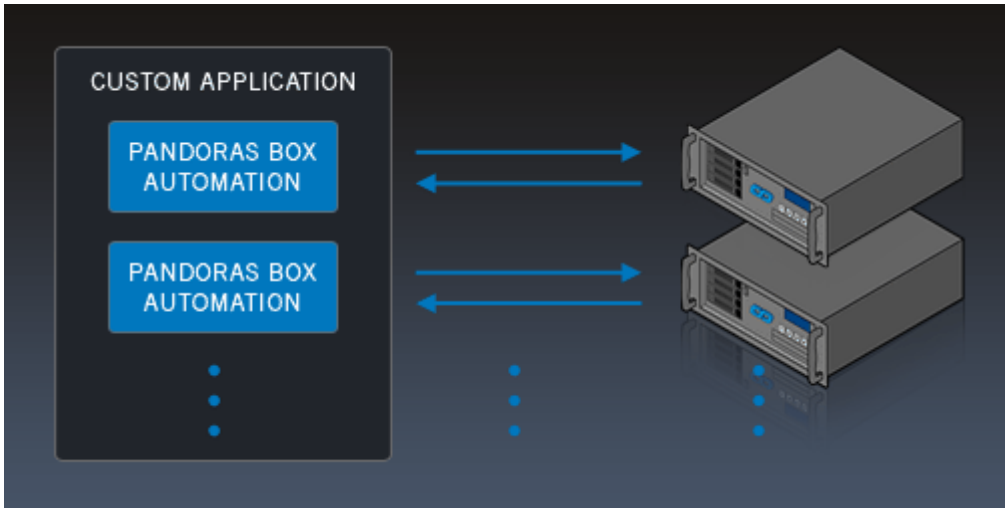
If you like to build your own custom application to interact with Pandoras Box, the following scenarios are possible.

One user application - One Pandoras Box Master system

One custom application with one integrated "PandorasAutomation.dll" sends and receives TCP or UDP commands to and from one Pandoras Box Master system.

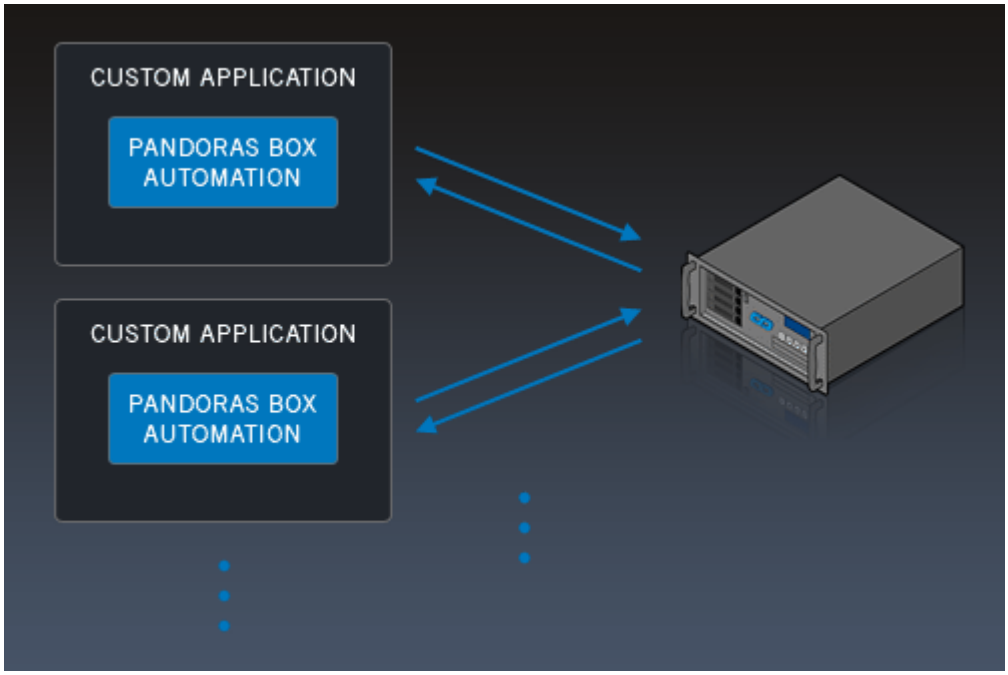
One user application - Multiple Pandoras Box Master systems

If you like to control more than one Pandoras Box Master system with your program, easily duplicate the "PandorasAutomation.dll" as many times as you wish.



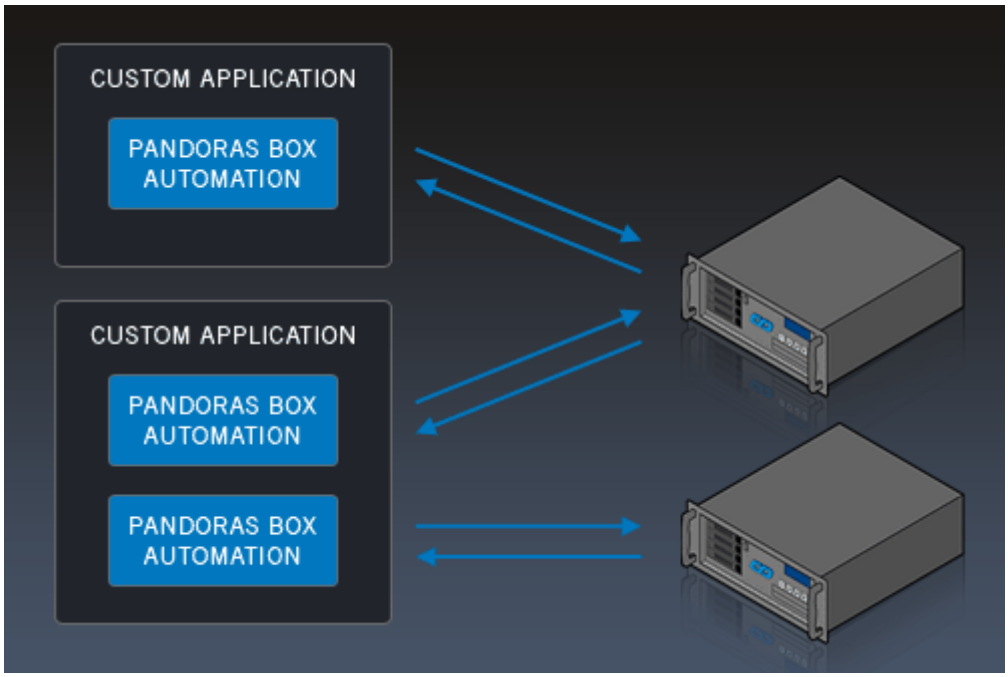
Multiple user applications - One Pandoras Box Master system

You can hook up as many applications to one Pandoras Box Master system as you wish. The commands will work in LTP mode (latest takes precedence).



Multiple user applications - Multiple Pandoras Box Master systems

This scenario combines the one-to-many and the many-to-one scenario. Pandoras Box does not set any limit regarding the application or PB Master count.



The following topics include a short Getting Started [732], some examples how to connect the SDK to a PB system [738], how to send [740] and receive [741] values as well as typical error messages [742]. Afterwards the general datatypes and structures [742] are explained and finally you find a list including all available functions [745].

6.8.1 Getting Started

Software Development Kits are available for the programming languages Visual Basic and C#. It is also possible to use JavaScript in Html pages. Other languages can also be used, but are not in the scope of this document.
We provide documented code for both Visual Basic and C# when used in combination with Visual Studio. You can obtain a copy of Visual Studio at the Microsoft website.

Required resources (Programming/Applications)
Step 1: Obtain PandorasAutomation.dll
You can find the PandorasAutomation.dll in every Pandoras Box installation folder (where PandorasBox.exe is located).
A typical location would be: C:\Program Files\Christie\Pandoras Box 8.1.0\PandorasAutomation.dll

Step 2: Download SDK
Eventhough the SDK includes a PandorasAutomation.dll, it is recommended to substitute it with the one from the current version!

Required resources (Scripting/Html+JS)
All the required files come with Pandoras Box.

The next topic describes the steps to get started with Visual Basic or C# [733]. For scripting with JavaScript read Using JavaScript [734]. Information about other languages can be found here [735].

6.8.1.1 Using Visual Basic / C#

To get started with Visual Basic or C# please follow these steps.

Installing the IDE (Integrated Development Environment)
To be able to convert the code you write to an actual executable program you will need a few tools. Fortunately, Microsoft provides all these tools bundled in one application called Visual Studio. Download Microsoft Visual Studio 2010 Express for Visual Basic or C#.
Download page: http://microsoft.com/visualstudio/eng/downloads#d-2010-express
Direct installer downloads: Visual Basic & C#

Creating a project
After starting Visual Studio click **File > New** to create a new Project. Choose your project type (ex: **Console Application**) from the list and name it using the text box at the bottom. When the project has finished loading, you need to add the Pandoras Box SDK. Select **Project > Add Existing Item** from the top menu. Choose **PbAutomation.*** in the file dialog . In case you have not previously downloaded it: Download SDK

Adding PandorasAutomation.dll
After saving your project, open the Windows Explorer and navigate to your project's folder. There you will find a couple of directories. Put PandorasAutomation.dll in the following directory: <ProjectPath>/bin/Debug/
Now you are ready to use Pandoras Automation with Visual Basic. When exporting applications, make sure you keep PandorasAutomation.dll next to the application.exe

Visual Basic Specific Information
The Visual Basic SDK provides an additional class to ease the workflow. The `Class Param` contains constants for the default parameter names (excluding effect parameters) used in all parameter functions.

C# Specific Information
It is required by C# that all functions are within a class. That means, that all C# functions are in class Auto and instead of calling `AutoGetContentIsConsistent(2,2)` you will have to put a dot in between Auto and the function name like this: `Auto.GetContentIsConsistent(2,2)`
When using `Auto.SetParamDouble` in combination with `enum TransportMode` you need to convert the enumerator to a `double` value.

Example (using explicit cast):
`Auto.SetParamDouble(1,1,"Playback Transport",TransportMode.Pause)` must be changed to
`Auto.SetParamDouble(1,1,"Playback Transport",(double)TransportMode.Pause)`

6.8.1.2 Using JavaScript

This chapter assumes that you already have a little knowledge about Html and JavaScript.

Setup of Pandoras Box web server
Open Pandoras Box and go to the Configuration tab. Choose Web Server from the categories list and change the settings as you like. Make sure to click the "Start" button as well. If you like, you can have a look at the demo pages which can be viewed from your web browser. Simply navigate to the IP address of your Pandoras Box Master using the port you previously set up in the Configuration tab.
URL example: Assuming your IP is 192.168.178.100 and the configured port is 6214: http://192.168.178.100:6214

Once you are familiar with using Pandoras Box web server you can go ahead and write your own html. The only thing required to be included is the pandora.js In your header section use
<script type="text/javascript" src="web_ui.js"></script>

The web server utilizes the very same commands as all the other languages. To be able to retrieve data, all commands ask for a callback function as the first parameter. This mechanism is explained in the demo files.

Using a dedicated web server
For advanced uses it is also possible to use a dedicated web server such as Apache to proxy the commands to Pandoras Box. We supplied a simple .php script for that. All you need to do is to change one line in pandora.js. Change "var usesProxy = false;" to "var usesProxy = true;"

Example - Sequence Control

```
// The first parameter is the callback, which is described later on.
// The second parameter is the sequence id, which in our case is simply 1
// The third parameter describes the desired state, which can be "Play","Pause" or "Stop" (case-sensitive!)
// "Play"-button code is
PBAutoCommands.setSequenceTransportMode(false,1,'Play')
```

Example - Retrieving Values

```
// callback_sequencemode is the callback (the function that is responsible to handle the incoming data)
PBAutoCommands.getSequenceTransportMode(callback_sequencemode,1)

function callback_sequencemode(response){
	// Quit if "false" is passed. That means that there is no response
	if(response === false) return;

	// Check for errors (On error, the first short is eAutoCmdError)
	if( && response.getNextAsShort() == eAutoCmdGetSequenceTransportMode){
		// Get the mode number
		var mode = response.getNextAsInt();
		var modeText = "???";
		// Translate numeric "mode" to text
		if(mode == 1) modeText = "Play";
		if(mode == 2) modeText = "Stop";
		if(mode == 3) modeText = "Pause";
		// Display
		document.getElementById("seq1mode").value = modeText;
	}
```

6.8.1.3 Other Languages

So far, we do not provide a complete SDK for other languages, but with a little knowledge you can use the PandorasAutomation.dll for them too. First, convert the following C++ export definitions to the language of choice in order to use the commands.
This list includes all commands PandorasAutomation.dll exports:

```
bool __stdcall AutoInitialize(char* pIpStr,int domain);
double __stdcall AutoGetParam(int siteNum,int deviceNum,char* pParamName);
bool __stdcall AutoSetParamInSelection(char* pParamName,int value);
bool __stdcall AutoSetParamInSelectiondouble(char* pParamName,double value);
bool __stdcall AutoSetContentAtTime(int siteNum,int deviceNum,int seqNum,int
hours,int minutes,int seconds,int frames,int dmxFolderId,int dmxId);
bool __stdcall AutoAssignMesh(int siteNum,int deviceNum,int dmxFolderId,int dmxId);
bool __stdcall AutoAssignMeshByName(int siteNum,int deviceNum,char* pMeshName,char*
pParamName);
bool __stdcall AutoAssignMeshToSelection(int dmxFolderId,int dmxId);
bool __stdcall AutoAssignMedia(int siteNum,int deviceNum,int dmxFolderId,int dmxId);
bool __stdcall AutoAssignMediaByName(int siteNum,int deviceNum,char*
pMediaName,char* pParamName);
bool __stdcall AutoAssignMediaToSelection(int dmxFolderId,int dmxId);
bool __stdcall AutoMoveContentToFolder(char* pContentName,char*pFolderName);
bool __stdcall AutoMoveTreeItem(int itemIdFrom,int itemIdTo);
bool __stdcall AutoSetSequenceTransportMode(int sequenceNum,char* pModeName);
bool __stdcall AutoMoveSequenceToCue(int sequenceNum,int cueId);
bool __stdcall AutoMoveSequenceToTime(int sequenceNum,int hours,int minutes,int
seconds,int frames);
bool __stdcall AutoMoveSequenceToLastNextFrame(int sequenceNum,bool isNext);
bool __stdcall AutoMoveSequenceToLastNextCue(int sequenceNum,bool isNext);
bool __stdcall AutoSetSequenceTransparency(int seqNum,int transparency);
int __stdcall  AutoGetSequenceTransparency(int seqNum);
bool __stdcall AutoSetSequenceTimeCodeMode(int seqNum,int timeCodeMode);
bool __stdcall AutoSetSequenceTimeCodeOffset(int seqNum,int hours,int minutes,int
seconds,int frames);
bool __stdcall AutoSetSequenceTimeCodeStopAction(int seqNum,int stopAction);
bool __stdcall AutoResetAll();
bool __stdcall AutoResetSite(int siteNum);
bool __stdcall AutoResetDevice(int siteNum,int deviceNum);
bool __stdcall AutoResetParam(int siteNum,int deviceNum,char* pParamName);
bool __stdcall AutoActivateAll();
bool __stdcall AutoActivateSite(int siteNum);
bool __stdcall AutoActivateDevice(int siteNum, int deviceNum);
bool __stdcall AutoActivateParam(int siteNum,int deviceNum,char* pParamName);
bool __stdcall AutoClearAllActive();
bool __stdcall AutoClearActiveSite(int siteNum);
bool __stdcall AutoClearActiveDevice(int siteNum,int deviceNum);
bool __stdcall AutoClearActiveParam(int siteNum,int deviceNum,char* pParamName);
bool __stdcall AutoToggleFullscreen(int siteNum);
bool __stdcall AutoSetParamRelative(int siteNum,int deviceNum,char* pParamName,int
value);
bool __stdcall AutoSetParamRelativedouble(int siteNum,int deviceNum,char*
pParamName,double value);
bool __stdcall AutoSetParamRelativedoubleExtended(int siteNum,int deviceNum,char*
pParamName,double value,bool silent,bool direct);
bool __stdcall AutoSetParamRelativeInSelection(char* pParamName,int value);
bool __stdcall AutoSetParamRelativeInSelectiondouble(char* pParamName,double value);
bool __stdcall AutoAddContent(char* pFullPath,int siteNum,int dmxFolderId,int
dmxId);
bool __stdcall AutoAddContentToFolder(char* pFullPath,int siteNum,int
dmxFolderId,int dmxId,char* pFoldername);
bool __stdcall AutoAddContentToTreeItem(char* pFullPath,int siteNum,int
dmxFolderId,int dmxId,int treeItemId);
bool __stdcall AutoAddContentFromLocalNode(char* pFullPath);
bool __stdcall AutoAddContentFromLocalNodeToFolder(char* pFullPath,char*
```

```
pFoldername);
bool __stdcall AutoAddContentFromLocalNodeToTreeItem(char* pFullPath,int
treeItemId);
bool __stdcall AutoAddContentFolder(char* pFolderPath,int siteNum,int
dmxFolderId,int dmxId,char* pProjectPath);
bool __stdcall AutoAddContentFolderFromLocalNode(char* pFolderPath);
bool __stdcall AutoAddContentFolderFromLocalNodeToFolder(char* pFolderPath,char*
pFoldername);
bool __stdcall AutoAddContentFolderFromLocalNodeToTreeItem(char* pFolderPath,int
treeItemId);
bool __stdcall AutoRemoveMediaById(int dmxFolderId,int dmxId);
bool __stdcall AutoRemoveMeshById(int dmxFolderId,int dmxId);
bool __stdcall AutoRemoveContentByName(char* pProjectPath,bool allEquallyNamed);
bool __stdcall AutoRemoveTreeItem(int treeItemId);
bool __stdcall AutoRemoveAllResources(bool removeFolder);
bool __stdcall AutoSpreadAll();
bool __stdcall AutoSpreadMediaById(int dmxFolderId,int dmxId);
bool __stdcall AutoSpreadMeshById(int dmxFolderId,int dmxId);
bool __stdcall AutoReloadMediaById(int dmxFolderId,int dmxId);
bool __stdcall AutoReloadMeshById(int dmxFolderId,int dmxId);
bool __stdcall AutoReloadResource(char* pProjectPath);
bool __stdcall AutoSpreadResource(char* pProjectPath);
bool __stdcall AutoReloadAndSpreadResourceByPath(char* pProjectPath);
bool __stdcall AutoReloadAndSpreadResourceByItemIndex(int treeItemId);
bool __stdcall AutoReloadAndSpreadResourceByDmxId(int dmxfolderId,int dmxId);
bool __stdcall AutoRemoveInconsistent();
bool __stdcall AutoStoreActive(int seqNum);
bool __stdcall AutoStoreActiveToTime(int seqNum,int hours,int minutes,int seconds,
int frames);
bool __stdcall AutoSetMediaFrameBlendingById(int dmxFolderId,int dmxId,bool
frameBlended);
bool __stdcall AutoSetMediaDeinterlacingById(int dmxFolderId,int dmxId,int
deinterlacer);
bool __stdcall AutoSetMediaAnisotropicFilteringById(int dmxFolderId,int dmxId,bool
useFiltering);
bool __stdcall AutoSetMediaUnderscanById(int dmxFolderId,int dmxId,bool
useUnderscan);
bool __stdcall AutoSetMediaMpegColourSpaceById(int dmxFolderId,int dmxId,bool
useMpegColourSpace);
bool __stdcall AutoSetMediaAlphaChannelById(int dmxFolderId,int dmxId,bool
useAlphaChannel);
bool __stdcall AutoCreateTextInput(int dmxFolderId,int dmxId,char* pText);
bool __stdcall AutoSetText(int dmxFolderId,int dmxId,char* pText);
bool __stdcall AutoLoadProject(char* pPath,char* pName,bool saveExisting);
bool __stdcall AutoCloseProject(bool save);
bool __stdcall AutoClearSelection();
bool __stdcall AutoSetDeviceAcceptDmxById(int siteNum,int deviceNum,bool acceptDmx);
bool __stdcall AutoSetSiteAcceptDmxById(int siteNum,bool acceptDmx);
bool __stdcall AutoSetDeviceDmxAddressById(int siteNum,int deviceNum,int index,int
id1,int id2);
bool __stdcall AutoSetSequenceCuePlayMode(int seqNum,int cueId,int playMode);
bool __stdcall AutoSetNextSequenceCuePlayMode(int seqNum,int playMode);
bool __stdcall AutoSetIgnoreNextSequenceCue(int seqNum,bool doIgnore);
bool __stdcall AutoSetChannelEvents(int ctEvents,int* pEvents);
bool __stdcall AutoSaveProject();
bool __stdcall AutoChangeFullscreenStateById(int siteNum,bool enterFullscreen);
bool __stdcall AutoChangeFullscreenStateByIp(char* pIp,bool enterFullscreen);
bool __stdcall AutoSetTextTextureSize(int dmxFolderId,int dmxId,int width,int
height);
bool __stdcall AutoSetTextProperties(int dmxFolderId,int dmxId,char* pFont,int
size,int style,int alignment,int colorRed,int colorGreen,int colorBlue);
bool __stdcall AutoSetTextCenterOnTexture(int dmxFolderId,int dmxId,bool
centerOnTexture);
bool __stdcall AutoCreateTextInputWide(int dmxFolderId,int dmxId,wchar_t* pText);
```

```
bool __stdcall AutoSetTextWide(int dmxFolderId,int dmxId,wchar_t* pText);
bool __stdcall AutoSetSiteIpById(int siteNum,char* pIp);
bool __stdcall AutoIsLayerReallySelected(int siteNum,int deviceNum);
int  __stdcall AutoGetNumMediaInProject();
int  __stdcall AutoGetNumTreeItemsInProject();
bool __stdcall AutoGetMediaInfo(int index,MediaStruct* pMediaInfo);
bool __stdcall AutoGetMediaInfo1(int index,MediaStruct1* pMediaInfo);
bool __stdcall AutoGetMediaInfoFromTreeItem(int treeItemIndex,MediaStruct1*
pMediaInfo);
bool __stdcall AutoGetTreeItemInfo(int index,TreeItemStruct* pItemInfo);
int  __stdcall AutoGetSequenceTransportMode(int seqNum);
bool __stdcall AutoGetSequenceTime(int seqNum,TimeStruct* pTime);
bool __stdcall AutoGetClipRemainingTime(int siteNum,int deviceNum,int
seqNum,TimeStruct* pTime);
bool __stdcall AutoGetRemainingTimeUntilNextCue(int seqNum,TimeStruct* pTime);
int  __stdcall AutoGetNumSelectedLayers();
bool __stdcall AutoGetSelectedLayer(int layerIndex,LayerStruct* layerInfo);
bool __stdcall AutoAddFolderToProject(char* pName);
bool __stdcall AutoAddFolderToProjectPath(char* pName, char* pFolderRoot);
bool __stdcall AutoAddFolderToTreeItem(char* pName,int treeItemId);
bool __stdcall AutoRemoveFolderFromProject(char* pFolderPath);
bool __stdcall AutoSetDeviceSelection(int siteNum,int deviceNum,int selectionMode);
bool __stdcall AutoSetClxControllerFaderMapping(int faderId,int seqNum);
bool __stdcall AutoSetClxControllerCueMapping(int cueBtnId,int seqNum,int cueId);
bool __stdcall AutoAddCue(int seqNum,int cueId,int hours,int minutes,int seconds,int
frames,char* pName,int cueKindId);
bool __stdcall AutoRemoveCueById(int seqNum,int cueId);
bool __stdcall AutoRemoveAllCues(int seqNum);
int __stdcall AutoAddGraphicLayer(int siteId);
int __stdcall AutoAddVideoLayer(int siteId);
bool __stdcall AutoRemoveGraphicLayer(int siteId,int layerId);
bool __stdcall AutoRemoveVideoLayer(int siteId,int layerId);
bool __stdcall AutoBackupMode(bool enable);
bool __stdcall AutoApplyView(int viewNum);
bool __stdcall AutoSetSpareFromSpread(int siteId,bool spareFromSpread);
bool __stdcall AutoGetParamMedia(int siteNum,int deviceNum,char*
pParamName,ParamResStruct* pInfo);
bool __stdcall AutoGetParamMedia1(int siteNum,int deviceNum,char*
pParamName,ParamResStruct1* pInfo);
bool __stdcall AutoGetParamObject(int siteNum,int deviceNum,char*
pParamName,ParamResStruct* pInfo);
bool __stdcall AutoGetParamObject1(int siteNum,int deviceNum,char*
pParamName,ParamResStruct1* pInfo);
bool __stdcall AutoAddMediaIncrementID(char* pMediaPath,int siteNum,ParamResStruct*
pInfo);
bool __stdcall AutoGetMediaTransportMode(int siteNum,int deviceNum,int*
pTransportMode);
bool __stdcall AutoIsSiteConnected(int siteNum);
bool __stdcall AutoMoveLayerUp(int siteNum,int deviceNum);
bool __stdcall AutoMoveLayerDown(int siteNum,int deviceNum);
bool __stdcall AutoMoveLayerToFirstPosition(int siteNum,int deviceNum);
bool __stdcall AutoMoveLayerToLastPosition(int siteNum,int deviceNum);
bool __stdcall AutoSetEnableClxJogShuttle(bool enable);
bool __stdcall AutoGetEnableClxJogShuttle();
bool __stdcall AutoSetEnableClxFaderExt(bool enable);
bool __stdcall AutoGetEnableClxFaderExt();
bool __stdcall AutoSetSequenceCueWaitTime(int seqNum,int cueId,int hours,int
minutes,int seconds,int frames);
bool __stdcall AutoSetSequenceCueJumpTargetTime(int seqNum,int cueId,int hours,int
minutes,int seconds,int frames);
bool __stdcall AutoSetSequenceCueJumpCount(int seqNum,int cueId,int jumpCount);
bool __stdcall AutoResetSequenceCueTriggerCount(int seqNum,int cueId);
int __stdcall AutoGetContentIsConsistent(int dmxFolderId,int dmxId);
int __stdcall AutoGetContentIsConsistentByName(char* pProjectPath);
```

```
int __stdcall AutoCreateSequence();
bool __stdcall AutoRemoveSequence(int seqNum);
bool __stdcall AutoGetIsConnected();
bool __stdcall AutoSendMouseInput(int siteNum,int eventType,int screenPosX,int
screenPosY,int screenWidth,int screenHeight);
bool __stdcall AutoSendTouchInput(int siteNum,int touchId,int touchType,int
screenPosX,int screenPosY,int screenWidth,int screenHeight);
bool __stdcall AutoSendKeyboardInput(int siteNum,int eventType,int keyCode);
bool __stdcall AutoSetShowCursorInFullscreen(int siteNum,bool showCursor);
bool __stdcall AutoSetNodeOfSiteIsAudioClockMaster(int siteNum,bool isMaster);
bool __stdcall AutoGetThumbnailByPath(char* pProjectPath,int* pWidth,int*
pHeight,VARIANT* pData);
bool __stdcall AutoGetThumbnailByItemIndex(int treeItemIndex,int* pWidth,int*
pHeight,VARIANT* pData);
bool __stdcall AutoAddEncryptionKey(char* pKey);
bool __stdcall AutoAddEncryptionPolicy(char* pPolicy);
int  __stdcall AutoGetLastError();
```

6.8.2 Examples

The following topics contain quick tutorials explaining the basic functionality of Pandoras Box Automation. They assume that you have already set up your programming environment and have added one of the SDKs to the project (as explained in Getting Started [732]).

To continue, go to Connect SDK to Pandoras Box [738]

6.8.2.1 Connect SDK to Pandoras Box

TCP versus UDP
The Pandoras Automation allows both, TCP and UDP connections. Both protocols allow sending data via a network connection. The ports that are used are 6211 for TCP and 6212 for UDP whilst the ports used for the TCP and UDP communication between a PB Master and PB Client are 1234 and 1235.

TCP has the advantage, that all packages are guaranteed to be delivered. To make that possible, TCP requires to send additional data, the so called acknowledge bits. The message sender asks the receiver for an acknowledge bit. If the receiver does not confirm, the message is send again. In addition TCP ensures the correct order of message packages. This is done by the network adapter; if packages are received in non-chronological order, it buffers the packages. The downside is, that the connection gets slower. The disadvantage of the acknowledging process is to be found when very large amounts of data is send as more network traffic is caused. Most sensors, for example send a lot of data, thus TCP would not be the protocol of choice. It is a different matter when you need to send "important" data and want to be sure that it arrives.
In short, decide for a TCP connection if the applications are connected in a stable network (by cable) and react to user interaction by sending a manageable amount of commands to PB Automation. When using TCP you can choose between automatic mode which will automatically reconnect or a "try once".

UDP is a better choice in scenarios where a lot of data is sent, or in case packages do not necessarily need to be acknowledged. In the UDP protocol the server does not know whether the clients received the messages, it rather sends the messages "blind". This way, UDP offers a faster response time for your application. The main disadvantage from a UDP protocol is that UDP packages are not as prioritized as TCP data. Thus each switch in the network may discard UDP data without further notice as soon as the network traffic reaches its limit.
In short, decide for a UDP connection if the application is allowed to get disconnected from the Master for short periods of time. Also, UDP is the better choice for applications that are updating values permanently, like in tracking systems or systems with permanent user interaction.

The following commands are needed to initialize the connection. The commands are listed with an example, assuming the Master system runs on a certain IP address and is set up with a certain domain number. The IP address can be found in the Asset tab [131] of the PB software, the domain channel is a Pandoras Box internal number set up in the Configuration tab [139].

Initializing TCP
Make sure you configured (or disabled) your firewall to allow your application to communicate with Pandoras Box.
Initialize the connection to the Pandoras Box Master running at IP 10.0.0.1 with Domain number 0. The third parameter allows to choose between
- False: instantly try to connect once
- True: wait and connect as soon as possible
Setting "waitForConnection" to True also enables automatic reconnecting on connection loss.
Note: You can later on enable the "waitForConnection" feature by using AutoWaitForConnection() and disable it by using AutoStopWaitingForConnection()

▾Visual Basic

```
AutoInitializeTCP("10.0.0.1", 0, True)
```

▾C#

```
Auto.InitializeTCP("10.0.0.1", 0, true);
```

Initializing UDP connections
Initialize the connection to Pandoras Box Master running at IP 10.0.0.1 with Domain number 0. Pandoras Automation will try to connect to Pandoras Box and verify the connection with a handshake.

▾Visual Basic

```
AutoInitialize("10.0.0.1", 0)
```

▾C#

```
Auto.Initialize("10.0.0.1", 0);
```

Now that you are connected to Pandoras Box, you can start Sending Parameter Values 740.

6.8.2.2 Sending Parameter Values

The following examples show you how to set parameter values from Pandoras Box using the SDK [730].
The examples require a working connection to Pandoras Box. (see Connect SDK to Pandoras Box [738])

▾Visual Basic

```
' Set the opacity of layer 3 of site 2 to 240.
' The parameter will be marked active and transition smoothing applies
AutoSetParamDouble(2, 3, Param.Opacity, 240)
' Set the x position of layer 1 of site 2 to -2 without setting the value active.
' Use Transition Smoothing
AutoSetParamDoubleExtended(2, 1, Param.Position.X, -2, True, False)
' Set the x position of layer 1 of site 2 to -2 and set value as active.
' Do not use Transition Smoothing
AutoSetParamDoubleExtended(2, 1, Param.Position.X, -2, False, True)
' Set the Mix parameter of the "White Key" effect on layer 1 of site 2 to a value of
255
AutoSetParamDouble(2, 1, "White Key|Mix", 255)
```

▾C#

```
// Set the opacity of layer 3 of site 2 to 240.
// The parameter will be marked active and transition smoothing applies
Auto.SetParamDouble(2, 3, "Opacity", 240)
// Set the x position of layer 1 of site 2 to -2 without setting the value active.
// Use Transition Smoothing
Auto.SetParamDoubleExtended(2, 1, "X Pos", -2, True, False)
// Set the x position of layer 1 of site 2 to -2 and set value as active.
// Do not use Transition Smoothing
Auto.SetParamDoubleExtended(2, 1, "X Pos", -2, False, True)
// Set the Mix parameter of the "White Key" effect on layer 1 of site 2 to a value
of 255
Auto.SetParamDouble(2, 1, "White Key|Mix", 255)
```

If you are not familiar with Pandoras Box "vocabulary", please follow the links. In short, a site refers to a Client e.g. a Server, in the Device Tree tab [173]. A site has many devices, also called layers, e.g. Video Layers [647]. A Video Layer has many parameters, e.g. position [651] on the X-axis. If a new value is applied to the parameters position / rotation / scaling, the layer would abruptly jump to it, which, does not look nice. Therefore, "Smoothing" applies a delay. It can be set up in the Device Inspector [210]. Other examples above refer to parameters "effects", more information about effects can be found in the tab Aeon FX [129] and FX on Layers [322], including an illustrated FX List [327] with all available effects. A list with all parameter names [1514] can be found here.

The next topic informs you about Receiving Parameter Values [741] from Pandoras Box.

6.8.2.3 Receiving Parameter Values

The following examples show you how to request information from Pandoras Box using the SDK [730] and how to handle incoming values.
The examples require a working connection to Pandoras Box (see Connect SDK to Pandoras Box) [738]. If you like to know, how to send values yourself please go back to the topic Sending Parameter Values [740])

▾Visual Basic

```
' Get the Y position of layer 1 of site 2
Dim y_pos As Double = AutoGetParam(2, 1, Param.Position.Y)
' Get the "Left" parameter value of effect "Crop Edges" on layer 3 of site 1
Dim crop_edges_left As Double = AutoGetParam(1, 3, "Crop Edges|Left")
' Get the transport mode of sequence 2 (Play/Stop/Pause)
Dim sequence_status As TransportMode = AutoGetSequenceTransportMode(2)
' Get the current time position of sequence 1 and extract the number of seconds
Dim sequence_time As TimeType
AutoGetSequenceTime(1, sequence_time)
Dim sequence_seconds As Integer = sequence_time.Seconds
```

▾C#

```
// Get the Y position of layer 1 of site 2
double y_pos = AutoGetParam(2, 1, "Pos Y")
// Get the "Left" parameter value of effect "Crop Edges" on layer 3 of site 1
double crop_edges_left As Double = AutoGetParam(1, 3, "Crop Edges|Left")
// Get the transport mode of sequence 2 (Play/Stop/Pause)
TransportMode sequence_status = Auto.GetSequenceTransportMode(2);
// Get the current time position of sequence 1 and extract the number of seconds
TimeType sequence_time = new TimeType();
Auto.GetSequenceTime(1, ref sequence_time);
int sequence_seconds = sequence_time.Seconds;
```

If you are not familiar with Pandoras Box "vocabulary", please follow the links. In short, a site refers to a Client e.g. a Server, in the Device Tree tab [173]. A site has many devices, also called layers, e.g. Video Layers [647]. A Video Layer has many parameters, e.g. position [651] on the X-axis. If a new value is applied to the parameters position / rotation / scaling, the layer would abruptly jump to it, which, does not look nice. Therefore, "Smoothing" applies a delay. It can be set up in the Device Inspector [210]. Other examples above refer to parameters "effects", more information about effects can be found in the tab Aeon FX [129] and FX on Layers [322], including an illustrated FX List [327] with all available effects. A list with all parameter names [1514] can be found here.

The next topic, Error Handling [742], covers the way the SDK reports problems.

6.8.2.4 Error Handling

Whilst using the SDK [730], e.g. sending [740] and receiving [741] parameter values you might encounter problems. The following function for error handling works only for functions which are not returning information directly. All other functions return a value true or false. If "false" was returned, you can get the type of problem with the function AutoGetLastError()

▼Visual Basic

```
// Store the error code in "error_code" and store a string describing the error in
"error_text"
Dim error_code As AutoError = AutoGetLastError()
Dim error_text As String = error_code.ToString()
```

▼C#

```
// Store the error code in "error_code" and store a string describing the error in
"error_text"
AutoError error_code = Auto.GetLastError();
string error_text = error_code.ToString();
```

6.8.3 Datatypes and Structures

This topic lists all available datatypes and their structures that are used by the SDK [730]. Please see the next topic for all available functions [745].

Enumeration AutoError - Contains the error codes which describe the problem that occurred

None = 0 ' No Error occurred
NoConnection = 1 ' Automation is not connected
WrongParam = 2 ' You have supplied a invalid parameter name (exceeded 100 characters)
AddressTranslation = 3 ' There seems to be a problem with the IP/Host you supplied
CouldNotConnectToSocket = 4 ' Failed to open a port for communication
HandshakeFailed = 5 ' The handshake was unsuccessful. For UDP connections: Make sure the connection is stable
RequestTimedOut = 6 ' The site failed to respond
WrongMessageReturned = 7 ' The Server returned a wrong message
ParamPointer = 8 ' Make sure you supplied all of the required arguments
WrongClient = 9 ' unused
HostInvalidLayer = 10 ' You supplied an invalid layer id
HostInvalidSequence = 11 ' You supplied an invalid sequence
HostInvalidPointer = 12 ' You supplied an invalid pointer
HostInvalidParameterName = 13 ' You supplied an invalid pointer name
HostInvalidParam = 14 ' You supplied an invalid parameter
InvalidPort = 15 ' Failed to connect to the Master
WrongNetworkProtocol = 16 ' unused
AlreadyConnected = 17 ' Failed to connect because the connection is already established
InvalidCueId = 18 ' You supplied an invalid cue
InvalidCueButtonId = 19 ' You supplied an invalid button id
InvalidDomainNr = 20 ' You supplied an invalid domain number
GraphicLayerNotCreated = 21 ' Failed to create a new graphics layer (which are discontinued since V8)
InvalidSiteId = 22 ' You supplied an invalid site id
InvalidViewId = 23 ' You supplied an invalid view id
InvalidCast = 24 ' (deprecated)
AddingVideoLayerNotAllowed = 25 ' Unable to add more Video Layers
InvalidLayerMoveTarget = 26 ' (deprecated)
InvalidFolderPath = 27 ' (deprecated)
DmxResourceNotFound = 28 '
NoAdditionalSequenceAllowed = 29
InvalidContentPath = 30
HandshakeTimeout = 31
FunctionNotSupportedByOS = 32 ' unused
TreeItemIndexNoMediaFile = 33 '

TreeItemNotFound = 34 ' The tree item was not found
InvalidTreeItemIndex = 35 ' The tree item was not found
NoThumbnailAvailable = 36 ' Indicates, that no thumbnail is available
EncryptionKeyNotValid = 37
EncryptionPolicyNotValid = 38
NoEncryptionManager = 39

Enumeration SequenceTimeCodeMode - Contains the codes for timecode modes

None = 0
Send = 1
Receive = 2

Enumeration SequenceTimeCodeStopAction - Contains the codes which determine what to do after a SMTPE timecode stop

None = 0
Stop = 1
Pause = 2
Continue = 3

Enumeration TransportMode - Contains the codes for play/pause/stop/playloop

Stop = 0
Pause = 128
Play = 64
PlayLoop = 192

Enumeration CuePlayMode - Contains the cue types (play,pause,stop,jump,wait)

Play = 0
Pause = 1
Stop = 2
Jump = 3
Wait = 4

Structure TimeType - Is used for functions that return time information

VersionNum As Integer
Hours As Integer
Minutes As Integer
Seconds As Integer
Frames As Integer

Structure MediaOptionsType - Contains options for media assets

anisotropicFiltering As Boolean
ignoreThumbnail As Boolean
alphaChannel As Boolean
fluidFrame As Boolean
optimizeMpegColorspace As Boolean
underscan As Boolean
optimizeLooping As Boolean
muteSound As Boolean

Structure MediaType1 - Contains information on media assets ("MediaType" is for legacy support)

dmxId As Integer
dmxFolderId As Integer
path As Byte()
projectPath As Byte()
width As Integer
height As Integer
fps As Integer
Length As TimeType
options As MediaOptionsType

Structure LayerType - Contains site and device number of a layer

VersionNum As Integer
siteNum As Integer
deviceNum As Integer

Structure ParamResourceType1 - Contains information about a resource ("ParamResourceType" is for legacy support)

folderId As Integer
fileId As Integer
path As Byte()
projectPath As Byte()

TreeItemType - Contains information about a tree item

projectPath As Byte()
idPath As Byte()
type As Integer

6.8.4 Function Reference

This topic lists all functions that are available in the SDK [730]. A list with the exact parameter names can be found here [1514].

AutoInitialize(IpStr As String, domain As Integer)

Initializes the connection to a Pandoras Box Master System using UDP

IpStr: IP Address of the machine running PB-Master
domain: Domain ID

True: No error ocurred False: Error ocurred

AutoInitializeTCP(IpStr As String, domain As Integer, waitForConnection As Boolean)

Initializes the connection to a Pandoras Box Master System using TCP

IpStr: IP Address of the machine running PB-Master
domain: Domain ID
waitForConnection: True: start a background thread to wait for a new connection False: only try once to connect

Success. When false is returned check AutoGetLastError()

AutoUnInitialize ()

Close any open connections

Success. When false is returned check AutoGetLastError()

AutoWaitForConnection ()

Starts a thread that keeps trying to connect to the PB-Master

Success. When false is returned check AutoGetLastError()

AutoStopWaitingForConnection ()

Stops the thread that keeps trying to connect to the PB-Master

Success. When false is returned check AutoGetLastError()

AutoGetIsConnected ()

Gets the current connection state

Success. When false is returned check AutoGetLastError()

AutoGetLastError ()

Gets the code for the last error that occurred

Error code

AutoSetParamDouble(siteNum As Integer, deviceNum As Integer, ParamName As String, value As Double)

Set a parameter to a specific value

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1514])
value: The value will be interpreted

Success. When false is returned check AutoGetLastError()

AutoSetParamDoubleExtended(siteNum As Integer, deviceNum As Integer, ParamName As String, value As Double, silent As Boolean, direct As Boolean)

Set a parameter to a specific value

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1514])
value: Value
silent: Do not mark parameter active
direct: Do not use transition smoothing

Success. When false is returned check AutoGetLastError()

AutoSetParamInSelectionDouble(ParamName As String, value As Double)

Sets the given parameter to given value for all layers in the current selection

ParamName: Name for the parameter. (see Class Param or this parameter list[1511])
value: The value will be interpreted

Success. When false is returned check AutoGetLastError()

AutoGetParam (siteNum As Integer, deviceNum As Integer, ParamName As String)

Gets the value of the parameter for given site/device

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1511])

Success. When false is returned check AutoGetLastError()

AutoSetContentAtTime (siteNum As Integer, deviceNum As Integer, seqNum As Integer, hours As Integer, minutes As Integer, seconds As Integer, frames As Integer, dmxFolderId As Integer, dmxId As Integer)

Sets the media for the container found at the given Site/Device/Sequence/Time combination. Note: This function will neither create new containers nor will it add keys. It only works with existing containers.

siteNum: Target Site ID
deviceNum: Target Device ID
seqNum: Sequence ID
hours: Time (Hours)
minutes: Time (Minutes)
seconds: Time (Seconds)
frames: Time (Frames)
dmxFolderId: Media Folder ID
dmxId: Media ID

Returns true if the PB-Master received the command

AutoAssignMesh(siteNum As Integer, deviceNum As Integer, dmxFolderId As Integer, dmxId As Integer)

Assign a Mesh to given site/device identified by DmxFolder and DmxId

siteNum: Site ID
deviceNum: Device ID
dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoAssignMeshByName(siteNum As Integer, deviceNum As Integer, MeshName As String, ParamName As String)

Assign a Mesh to given site/device identified by name. Also allows to assign eeshes to effects etc. using ParamName

siteNum: Site ID
deviceNum: Device ID
MeshName: Mesh Name
ParamName: The parameter to assign the mesh to. Use "Mesh" to assign to the device itself. (see Class Param or this parameter list [1511])

Success. When false is returned check AutoGetLastError()

AutoAssignMeshToSelection(dmxFolderId As Integer, dmxId As Integer)

Assign a Mesh to selected devices

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoAssignMedia(siteNum As Integer, deviceNum As Integer, dmxFolderId As Integer, dmxId As Integer)

Assign Media to given site/device identified by DmxFolder and DmxId

siteNum: Site ID
deviceNum: Device ID
dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoAssignMediaByName(siteNum As Integer, deviceNum As Integer, MediaName As String, ParamName As String)

Assign Media to given site/device identified by name

siteNum: Site ID
deviceNum: Device ID
MediaName: Media Name
ParamName: Name for the parameter. (see Class Param or this parameter list[1511])

Success. When false is returned check AutoGetLastError()

AutoAssignMediaToSelection(dmxFolderId As Integer, dmxId As Integer)

Assign Media to selected devices

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoMoveContentToFolder(ContentName As String, FolderName As String)

Moves content identified by name to given folder

ContentName: The path+name of the content in the projects tab. ex.: Myfolder/mySubfolder/somecontent.mpg
FolderName: Path to move to. ex.: SomeFolder/MyTargetFolder

Success. When false is returned check AutoGetLastError()

AutoMoveTreeItem(itemIdFrom As Integer, itemIdTo As Integer)

Move tree item to another tree item

itemIdFrom: tree item source
itemIdTo: tree item target

Success. When false is returned check AutoGetLastError()

AutoSetSequenceTransportMode(sequenceNum As Integer, ModeName As String)

Sets the transport mode for sequence with given ID

sequenceNum: Sequence ID
ModeName: Transport mode, case sensitive (Play/Pause/Stop)

Success. When false is returned check AutoGetLastError()

AutoMoveSequenceToCue(sequenceNum As Integer, cueId As Integer)

Moves the nowpointer to the position of the cue with given ID

sequenceNum: Sequence ID
cueId: Cue ID

Success. When false is returned check AutoGetLastError()

AutoMoveSequenceToTime(sequenceNum As Integer, hours As Integer, minutes As Integer, seconds As Integer, frames As Integer)

Moves the nowpointer to given time

sequenceNum: Sequence ID
hours: Time (Hours)
minutes: Time (Minutes)
seconds: Time (Seconds)
frames: Time (Frames)

Success. When false is returned check AutoGetLastError()

AutoMoveSequenceToLastNextFrame(sequenceNum As Integer, isNext As Boolean)

Moves to either the next/previous frame

sequenceNum: Sequence ID
isNext: true: next frame | false: previous frame

Success. When false is returned check AutoGetLastError()

AutoMoveSequenceToLastNextCue(sequenceNum As Integer, isNext As Boolean)

Moves to either the next/previous cue

sequenceNum: Sequence ID
isNext: true: next cue | false: previous cue

Success. When false is returned check AutoGetLastError()

AutoSetSequenceTransparency(seqNum As Integer, transparency As Integer)

Sets the transparency for given sequence

seqNum: Sequence ID
transparency: Opacity between 0 and 255

Success. When false is returned check AutoGetLastError()

AutoGetSequenceTransparency (seqNum As Integer)

Get the transparency of given sequence

seqNum: Sequence ID

Sequence Transparency (0-255)

AutoSetSequenceTimeCodeMode(seqNum As Integer, timeCodeMode As Param.SequenceTimeCodeMode)

Enables or disables a sequence to send/receive timecode

seqNum: Sequence ID
timeCodeMode: The timecode mode to use

Success. When false is returned check AutoGetLastError()

AutoSetSequenceTimeCodeOffset(seqNum As Integer, hours As Integer, minutes As Integer, seconds As Integer, frames As Integer)

Set the offset that will be added to the Pandoras Box Timecode. Negative values possible.

seqNum: Sequence ID
hours: Time (Hours)
minutes: Time (Minutes)
seconds: Time (Seconds)
frames: Time (Frames)

Success. When false is returned check AutoGetLastError()

AutoSetSequenceTimeCodeStopAction(seqNum As Integer, stopAction As Param.SequenceTimeCodeStopAction)

Set the behavior on timecode signal stop

seqNum: Sequence ID
stopAction: stop,pause or continue playback

Success. When false is returned check AutoGetLastError()

AutoResetAll ()

Reset all active values

Success. When false is returned check AutoGetLastError()

AutoResetSite(siteNum As Integer)

Reset all active values for given site

siteNum: Site ID

Success. When false is returned check AutoGetLastError()

AutoResetDevice(siteNum As Integer, deviceNum As Integer)

Reset all active values for given device

siteNum: Site ID
deviceNum: Device ID

Success. When false is returned check AutoGetLastError()

AutoResetParam(siteNum As Integer, deviceNum As Integer, ParamName As String)

Remove active value of specific parameter

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list 1511)

Success. When false is returned check AutoGetLastError()

AutoActivateAll ()

Select all parameters as active

Success. When false is returned check AutoGetLastError()

AutoActivateSite(siteNum As Integer)

Set all parameters of all devices of a whole site as active

siteNum: Site ID

Success. When false is returned check AutoGetLastError()

AutoActivateDevice(siteNum As Integer, deviceNum As Integer)

Set all parameters of a specific device as active

siteNum: Site ID
deviceNum: Device ID

Success. When false is returned check AutoGetLastError()

AutoActivateParam(siteNum As Integer, deviceNum As Integer, ParamName As String)

Set a specific parameter as active

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list 1511)

Success. When false is returned check AutoGetLastError()

AutoClearAllActive ()

Set all active values as inactive. The values themselves are preserved.

Success. When false is returned check AutoGetLastError()

AutoClearActiveSite(siteNum As Integer)

Set all active values of a site as inactive. The values themselves are preserved.

siteNum: Site ID

Success. When false is returned check AutoGetLastError()

AutoClearActiveDevice(siteNum As Integer, deviceNum As Integer)

Set all active values of a specific device as inactive. The values themselves are preserved.

siteNum: Site ID
deviceNum: Device ID

Success. When false is returned check AutoGetLastError()

AutoClearActiveParam(siteNum As Integer, deviceNum As Integer, ParamName As String)

Set a parameter value inactive. The values themselves are preserved.

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1511])

Success. When false is returned check AutoGetLastError()

AutoToggleFullscreen(siteNum As Integer)

Toggles full screen mode of a site

siteNum: Site ID

Success. When false is returned check AutoGetLastError()

AutoSetParamRelativeDouble(siteNum As Integer, deviceNum As Integer, ParamName As String, value As Double)

Add or subtract a value from a parameter

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1511])
value: Value to add, can be negative

Success. When false is returned check AutoGetLastError()

AutoSetParamRelativeDoubleExtended(siteNum As Integer, deviceNum As Integer, ParamName As String, value As Double, silent As Boolean, direct As Boolean)

Add or subtract a value from a parameter

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1511])
value: Value to add, can be negative
silent: Do not mark parameter active
direct: Do not use transition smoothing

Success. When false is returned check AutoGetLastError()

AutoSetParamRelativeInSelectionDouble (ParamName As String, value As Double)

Add or subtract a value from a parameter in the current selection

ParamName: Name for the parameter. (see Class Param or this parameter list[1511])
value: Value to add, can be negative

Success. When false is returned check AutoGetLastError()

AutoAddContent(FullPath As String, siteNum As Integer, dmxFolderId As Integer, dmxId As Integer)

Add content from given path and assign DmxIDs

FullPath: Absolute system path to media to add (`e.g. C:\Christie\content`)
siteNum: Site ID
dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoAddContentToFolder(FullPath As String, siteNum As Integer, dmxFolderId As Integer, dmxId As Integer, Foldername As String)

Adds content from given folder to a specific folder, also assigning DmxIDs

FullPath: Absolute system path to media to add (`e.g. C:\Christie\content`)
siteNum: Site ID
dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
Foldername: Target folder to add content to

Success. When false is returned check AutoGetLastError()

AutoAddContentToTreeItem(FullPath As String, siteNum As Integer, dmxFolderId As Integer, dmxId As Integer, treeItemId As Integer)

Add content to a tree item

FullPath: Absolute system path to media to add (`e.g. C:\Christie\content`)
siteNum: Site ID
dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
treeItemId: Target tree item

Success. When false is returned check AutoGetLastError()

AutoAddContentFromLocalNode (FullPath As String)

Add content from local node using an absolute path

FullPath: Absolute system path to media to add (`e.g. C:\Christie\content`)

Success. When false is returned check AutoGetLastError()

AutoAddContentFromLocalNodeToFolder(FullPath As String, Foldername As String)

FullPath: Absolute system path to media to add (`e.g. C:\Christie\content`)
Foldername: Target path in PB project

Success. When false is returned check AutoGetLastError()

AutoAddContentFromLocalNodeToTreeItem(FullPath As String, treeItemId As Integer)

Add content from an absolute system path to a specific tree node

FullPath: Absolute system path to media to add (`e.g. C:\Christie\content`)
treeItemId: target tree item

Success. When false is returned check AutoGetLastError()

AutoAddContentFolder(FolderPath As String, siteNum As Integer, dmxFolderId As Integer, dmxId As Integer, ProjectPath As String)

Add a complete folder with content to the project

FolderPath: Path to content
siteNum: Site ID
dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
ProjectPath: Project path

Success. When false is returned check AutoGetLastError()

AutoAddContentFolderFromLocalNode(FolderPath As String)

Add a complete folder with content to the projecta

FolderPath: Folder to add

Success. When false is returned check AutoGetLastError()

AutoAddContentFolderFromLocalNodeToFolder(FolderPath As String, Foldername As String)

Add a complete folder with content to the project at a specific folder

FolderPath: Folder to add
Foldername: Target folder in project tree

Success. When false is returned check AutoGetLastError()

AutoAddContentFolderFromLocalNodeToTreeItem(FolderPath As String, treeItemId As Integer)

Add a complete folder with content to the project to a specific TreeItem

FolderPath: Folder to add
treeItemId: tree item to add to

Success. When false is returned check AutoGetLastError()

AutoRemoveMediaById(dmxFolderId As Integer, dmxId As Integer)

Removes Media with given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoRemoveMeshById(dmxFolderId As Integer, dmxId As Integer)

Removes Mesh with given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoRemoveContentByName(ProjectPath As String, allEquallyNamed As Boolean)

Removes content by project path

ProjectPath: Path to the project content
allEquallyNamed: True: Removes all contents with the same name| False: Remove only one (the first) content with that name

Success. When false is returned check AutoGetLastError()

AutoRemoveTreeItem(treeItemId As Integer)

Remove a tree item

treeItemId: tree item to remove

Success. When false is returned check AutoGetLastError()

AutoRemoveAllResources(removeFolder As Boolean)

Remove all resources in specific folder

removeFolder: True: Removes everything | False: Remove files only. Folder structure stays intact

Success. When false is returned check AutoGetLastError()

AutoSpreadAll ()

Trigger a "Spread All Resources"

Success. When false is returned check AutoGetLastError()

AutoSpreadMediaById(dmxFolderId As Integer, dmxId As Integer)

Trigger spread for media identified by DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoSpreadMeshById(dmxFolderId As Integer, dmxId As Integer)

Trigger spread for mesh identified by DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoReloadMediaById(dmxFolderId As Integer, dmxId As Integer)

Trigger reload for media with given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoReloadMeshById(dmxFolderId As Integer, dmxId As Integer)

Trigger reload for mesh with given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoReloadResource(ProjectPath As String)

Trigger reload for resource with given name

ProjectPath: project path

Success. When false is returned check AutoGetLastError()

AutoSpreadResource(ProjectPath As String)

Trigger spread for resource with given name

ProjectPath: project path

Success. When false is returned check AutoGetLastError()

AutoReloadAndSpreadResourceByPath(ProjectPath As String)

Trigger reload and spread for resource identified by path name

ProjectPath: project path

Success. When false is returned check AutoGetLastError()

AutoReloadAndSpreadResourceByItemIndex(treeItemId As Integer)

Trigger reload and spread for resource identified by tree item index

treeItemId: tree item id of resource

Success. When false is returned check AutoGetLastError()

AutoReloadAndSpreadResourceByDmxId(dmxfolderId As Integer, dmxId As Integer)

Trigger reload and spread for resource identified by DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Success. When false is returned check AutoGetLastError()

AutoRemoveInconsistent ()

Remove all inconsistent files

Success. When false is returned check AutoGetLastError()

AutoStoreActive(seqNum As Integer)

Stores active values in given sequence at the current timecode

seqNum: Sequence ID

Success. When false is returned check AutoGetLastError()

AutoStoreActiveToTime(seqNum As Integer, hours As Integer, minutes As Integer, seconds As Integer, frames As Integer)

Stores active values in given sequence at given time

seqNum: Sequence ID
hours: Time (Hours)
minutes: Time (Minutes)
seconds: Time (Seconds)
frames: Time (Frames)

Success. When false is returned check AutoGetLastError()

AutoSetMediaFrameBlendingById(dmxFolderId As Integer, dmxId As Integer, frameBlended As Boolean)

Sets the Frame Blending for media identified by given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
frameBlended: Use fra

Success. When false is returned check AutoGetLastError()

AutoSetMediaDeinterlacingById(dmxFolderId As Integer, dmxId As Integer, deinterlacer As Integer)

Sets the Deinterlacing for media identified by given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
deinterlacer: Use deinterlacing?

Success. When false is returned check AutoGetLastError()

AutoSetMediaAnisotropicFilteringById(dmxFolderId As Integer, dmxId As Integer, useFiltering As Boolean)

Sets the Anisostropic Filtering for media identified by given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
useFiltering: Use Anisostrophicfiltering?

Success. When false is returned check AutoGetLastError()

AutoSetMediaUnderscanById(dmxFolderId As Integer, dmxId As Integer, useUnderscan As Boolean)

Sets the Underscan for media identified by given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
useUnderscan: Use underscan?

Success. When false is returned check AutoGetLastError()

AutoSetMediaMpegColourSpaceById(dmxFolderId As Integer, dmxId As Integer, useMpegColourSpace As Boolean)

Sets wether to use mpeg color space conversion for media identified by given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
useMpegColourSpace: Use mpeg color space conversion?

Success. When false is returned check AutoGetLastError()

AutoSetMediaAlphaChannelById(dmxFolderId As Integer, dmxId As Integer, useAlphaChannel As Boolean)

Sets the Alpha Channel usage for media identified by given DmxID

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
useAlphaChannel: Use alpha channel?

Success. When false is returned check AutoGetLastError()

AutoCreateTextInput(dmxFolderId As Integer, dmxId As Integer, Text As String)

Creates a new text asset with given DmxIDs

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
Text: Asset text

Success. When false is returned check AutoGetLastError()

AutoSetText(dmxFolderId As Integer, dmxId As Integer, Text As String)

Sets the text for the text asset with given DmxIDs

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
Text: Text

Success. When false is returned check AutoGetLastError()

AutoLoadProject(Path As String, Name As String, saveExisting As Boolean)

Loads a project from given path with given filename.

Path: Folder name
Name: Project file name
saveExisting: Save the existing project before opening the new one?

Success. When false is returned check AutoGetLastError()

AutoCloseProject(save As Boolean)

Closes the current project

save: Save project before close?

Success. When false is returned check AutoGetLastError()

AutoClearSelection()

Clears the device selection

Success. When false is returned check AutoGetLastError()

AutoSetDeviceAcceptDmxById(siteNum As Integer, deviceNum As Integer, acceptDmx As Boolean)

Set wether given device should accept Dmx values

siteNum: Site ID
deviceNum: Device ID
acceptDmx: Accept Dmx?

Success. When false is returned check AutoGetLastError()

AutoSetSiteAcceptDmxById(siteNum As Integer, acceptDmx As Boolean)

Set wether given site should accept Dmx values

siteNum: Site ID
acceptDmx: Accept Dmx?

Success. When false is returned check AutoGetLastError()

AutoSetDeviceDmxAddressById(siteNum As Integer, deviceNum As Integer, index As Integer, id1 As Integer, id2 As Integer)

Sets the Dmx address for given site/device

siteNum: Site ID
deviceNum: Device ID
index: Channel Id
id1: Dmx Subnet
id2: Dmx Universe

Success. When false is returned check AutoGetLastError()

AutoSetSequenceCuePlayMode(seqNum As Integer, cueId As Integer, playMode As Param.CuePlayMode)

Sets the play mode for a cue at given sequence

seqNum: Sequence ID
cueId: Cue ID
playMode: The cue play mode

Success. When false is returned check AutoGetLastError()

AutoSetNextSequenceCuePlayMode(seqNum As Integer, playMode As Integer)

Sets the play mode for the next cue at given sequence

seqNum: Sequence ID
playMode: The cue play mode

Success. When false is returned check AutoGetLastError()

AutoSetIgnoreNextSequenceCue(seqNum As Integer, doIgnore As Boolean)

Sets wether to ignore the next cue in given sequence

seqNum: Sequence ID
doIgnore: Ignore next cue?

Success. When false is returned check AutoGetLastError()

AutoSaveProject ()

Saves the current project

Success. When false is returned check AutoGetLastError()

AutoChangeFullscreenStateById(siteNum As Integer, enterFullscreen As Boolean)

Change the fullscreen mode for given site identified by Id

siteNum: Site ID
enterFullscreen: Enter fullscreen?

Success. When false is returned check AutoGetLastError()

AutoChangeFullscreenStateByIp(Ip As String, enterFullscreen As Boolean)

Change the fullscreen mode for given site identified by Ip

Ip: IP Address
enterFullscreen: Enter fullscreen?

Success. When false is returned check AutoGetLastError()

AutoSetTextTextureSize(dmxFolderId As Integer, dmxId As Integer, width As Integer, height As Integer)

Sets the texture size for text asset identified by DmxId

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
width: New texture width
height: New texture height

Success. When false is returned check AutoGetLastError()

AutoSetTextProperties(dmxFolderId As Integer, dmxId As Integer, Font As String, size As Integer, style As Integer, alignment As Integer, colorRed As Integer, colorGreen As Integer, colorBlue As Integer)

Set the style for an existing text asset

dmxFolderId: Dmx ID (Folder)
dmxId: Dmx ID (Item)
Font: Font name
size: size in pixels
style: text style
alignment: text alignment
colorRed: Color (Red)
colorGreen: Color (Green)
colorBlue: Color (Blue)

Success. When false is returned check AutoGetLastError()

AutoSetTextCenterOnTexture(dmxFolderId As Integer, dmxId As Integer, centerOnTexture As Boolean)

Set the center on texture for text asset identified by DmxId

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
centerOnTexture: Center text on texture?

Success. When false is returned check AutoGetLastError()

AutoCreateTextInputWide(dmxFolderId As Integer, dmxId As Integer, Text As String)

Creates new text asset. Adjusts width automatically.

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
Text: Text

Success. When false is returned check AutoGetLastError()

AutoSetTextWide(dmxFolderId As Integer, dmxId As Integer, Text As String)

Sets the text of a text asset and adjusts width automatically

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)
Text: Text

Success. When false is returned check AutoGetLastError()

AutoSetSiteIpById(siteNum As Integer, Ip As String)

Changes the IP address of a site.

siteNum: Site ID
Ip: IP Address

Success. When false is returned check AutoGetLastError()

AutoIsLayerReallySelected(siteNum As Integer, deviceNum As Integer)

Check if layer is in current selection

siteNum: Site ID
deviceNum: Device ID

Is the given layer currently selected?

AutoGetNumMediaInProject ()

Gets the number of media in the project

Number of media in project

AutoGetNumTreeItemsInProject ()

Gets the number of tree items in project

Number of tree items in project

AutoGetMediaInfo1(index As Integer, MediaInfo As MediaType1)

Gets information on media identified by index

index: Index
MediaInfo: MediaInfo1 object to write information to

Success. When false is returned check AutoGetLastError()

AutoGetMediaInfoFromTreeItem(treeItemIndex As Integer, MediaInfo As MediaType1)

Gets information on media identified by index

treeItemindex: Index
MediaInfo: MediaInfo1 object to write information to

Success. When false is returned check AutoGetLastError()

AutoGetTreeItemInfo(index As Integer, ItemInfo As TreeItemType)

Get information on tree item identified by index

index: Index
ItemInfo: TreeItemType object to write results to

Success. When false is returned check AutoGetLastError()

AutoGetSequenceTransportMode(seqNum As Integer)

Get the transport mode of a sequence

seqNum: Sequence ID

Success. When false is returned check AutoGetLastError()

AutoGetSequenceTime(seqNum As Integer, Time As TimeType)

Get the time of a sequence

seqNum: Sequence ID
Time: TimeType to write information to

Success. When false is returned check AutoGetLastError()

AutoGetClipRemainingTime(siteNum As Integer, deviceNum As Integer, seqNum As Integer, Time As TimeType)

Get the remaining time for a clip on a specific layer for given sequence

siteNum: Site ID
deviceNum: Device ID
seqNum: Sequence ID
Time: TimeType to write information to

Success. When false is returned check AutoGetLastError()

AutoGetRemainingTimeUntilNextCue(seqNum As Integer, Time As TimeType)

Get remaining time until the next cue for given sequence

seqNum: Sequence ID
Time: TimeType to write information to

Success. When false is returned check AutoGetLastError()

AutoGetNumSelectedLayers ()

Get the number of selected layers

Number of selected layers

AutoGetSelectedLayer(layerIndex As Integer, layerInfo As LayerType)

Get the selected layer with given number in selection.

layerIndex: Index of the layer in selection. Values between 0 and (AutoGetNumSelectedLayers - 1)
layerInfo: LayerType object to write results to

Success. When false is returned check AutoGetLastError()

AutoAddFolderToProject(Name As String)

Create a new folder

Name: Name for the new folder

Success. When false is returned check AutoGetLastError()

AutoAddFolderToProjectPath(Name As String, FolderRoot As String)

Create new folder in given path

Name: Name for the new folder
FolderRoot: Path to create folder in

Success. When false is returned check AutoGetLastError()

AutoAddFolderToTreeItem(Name As String, treeItemId As Integer)

Create new folder in given tree item

Name: Name for the new folder
treeItemId: Tree item to create folder in

Success. When false is returned check AutoGetLastError()

AutoRemoveFolderFromProject(FolderPath As String)

Remove a folder from the project

FolderPath: Folder with path

Success. When false is returned check AutoGetLastError()

AutoSetDeviceSelection(siteNum As Integer, deviceNum As Integer, selectionMode As Integer)

Select / Deselect a device

siteNum: Site ID
deviceNum: Device ID
selectionMode: True: Selected| False: Not selected

Success. When false is returned check AutoGetLastError()

AutoSetClxControllerFaderMapping(faderId As Integer, seqNum As Integer)

Map a Pandoras Box controller fader to a sequence

faderId: Fader ID
seqNum: Sequence ID

Success. When false is returned check AutoGetLastError()

AutoSetClxControllerCueMapping(cueBtnId As Integer, seqNum As Integer, cueId As Integer)

Map a Pandoras Box controller button to a cue

cueBtnId: Button ID
seqNum: Sequence ID
cueId: Cue ID

Success. When false is returned check AutoGetLastError()

AutoAddCue(seqNum As Integer, cueId As Integer, hours As Integer, minutes As Integer, seconds As Integer, frames As Integer, Name As String, cueKindId As Param.CuePlayMode)

Add cue to given sequence at given time

seqNum: Sequence ID
cueId: Cue ID
hours: Time (Hours)
minutes: Time (Minutes)
seconds: Time (Seconds)
frames: Time (Frames)
Name: Cue name
cueKindId: Kind of cue

Success. When false is returned check AutoGetLastError()

AutoRemoveCueById(seqNum As Integer, cueId As Integer)

Remove cue with given ID on given sequence

seqNum: Sequence ID
cueId: Cue ID

Success. When false is returned check AutoGetLastError()

AutoRemoveAllCues(seqNum As Integer)

Remove all cues for a sequence

seqNum: Sequence ID

Success. When false is returned check AutoGetLastError()

AutoAddGraphicLayer(siteId As Integer)

Add new graphic layer (discontinued since version 8!)

siteId: Site ID

Success. When false is returned check AutoGetLastError()

AutoAddVideoLayer(siteId As Integer)

Add new video layer

siteId: Site ID

Success. When false is returned check AutoGetLastError()

AutoRemoveGraphicLayer(siteId As Integer, layerId As Integer)

Remove a graphic layer by id

siteId: Site ID
layerId: Layer ID

Success. When false is returned check AutoGetLastError()

AutoRemoveVideoLayer(siteId As Integer, layerId As Integer)

Remove a video layer by id

siteId: Site ID
layerId: Layer ID

Success. When false is returned check AutoGetLastError()

AutoBackupMode(enable As Boolean)

Enables/Disables the backup mode

enable: Enable backup mode?

Success. When false is returned check AutoGetLastError()

AutoApplyView(viewNum As Integer)

Applies view identified by given number

viewNum: The view number

Success. When false is returned check AutoGetLastError()

AutoSetSpareFromSpread(siteId As Integer, spareFromSpread As Boolean)

Set the spare from spread option

siteId: Site ID
spareFromSpread: True: Do not spread resources to this site | False: Spreading resources possible

Success. When false is returned check AutoGetLastError()

AutoGetParamMedia1(siteNum As Integer, deviceNum As Integer, ParamName As String, Info As ParamResourceType1)

Get resource information of a device

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1511])
Info: ResourceType1 object to write information to

Success. When false is returned check AutoGetLastError()

AutoGetParamObject1(siteNum As Integer, deviceNum As Integer, ParamName As String, Info As ParamResourceType1)

Get information of a specific parameter

siteNum: Site ID
deviceNum: Device ID
ParamName: Name for the parameter. (see Class Param or this parameter list[1511])
Info: ParamResourceType1 object to write information to

Success. When false is returned check AutoGetLastError()

AutoGetMediaTransportMode(siteNum As Integer, deviceNum As Integer, TransportMode As TransportMode)

Checks wether given site is connected

siteNum: Site ID
deviceNum: Device ID
TransportMode: TransportMode object to write the result to

Success. When false is returned check AutoGetLastError()

AutoIsSiteConnected (siteNum As Integer)

Checks wether given site is connected

siteNum: Site ID

Success. When false is returned check AutoGetLastError()

AutoMoveLayerUp(siteNum As Integer, deviceNum As Integer)

Move given layer up by one in the device tree

siteNum: Site ID
deviceNum: Device ID

Success. When false is returned check AutoGetLastError()

AutoMoveLayerDown(siteNum As Integer, deviceNum As Integer)

Move given layer down by one in the device tree

siteNum: Site ID
deviceNum: Device ID

Success. When false is returned check AutoGetLastError()

AutoMoveLayerToFirstPosition(siteNum As Integer, deviceNum As Integer)

Move given layer to the first position in the device tree

siteNum: Site ID
deviceNum: Device ID

Success. When false is returned check AutoGetLastError()

AutoMoveLayerToLastPosition(siteNum As Integer, deviceNum As Integer)

Move given layer to the last position in the device tree

siteNum: Site ID
deviceNum: Device ID

Success. When false is returned check AutoGetLastError()

AutoSetEnableClxJogShuttle(enable As Boolean)

Enable/Disable the Jog/Shuttle

enable: Enable?

Success. When false is returned check AutoGetLastError()

AutoGetEnableClxJogShuttle()

Get wether the Jog/Shuttle is enabled

Jog/Shuttle enabled?

AutoSetEnableClxFaderExt(enable As Boolean)

Enable/Disable the FaderExtention

enable: Enable?

Success. When false is returned check AutoGetLastError()

AutoGetEnableClxFaderExt()

Gets wether the FaderExtention is enabled

FaderExtention enabled?

AutoSetSequenceCueWaitTime(seqNum As Integer, cueId As Integer, hours As Integer, minutes As Integer, seconds As Integer, frames As Integer)

Set the wait time for given cue

seqNum: Sequence ID
cueId: Cue ID
hours: Time (Hours)
minutes: Time (Minutes)
seconds: Time (Seconds)
frames: Time (Frames)

Success. When false is returned check AutoGetLastError()

AutoSetSequenceCueJumpTargetTime(seqNum As Integer, cueId As Integer, hours As Integer, minutes As Integer, seconds As Integer, frames As Integer)

Set the jump target for given cue

seqNum: Sequence ID
cueId: Cue ID
hours: Time (Hours)
minutes: Time (Minutes)
seconds: Time (Seconds)
frames: Time (Frames)

Success. When false is returned check AutoGetLastError()

AutoSetSequenceCueJumpCount(seqNum As Integer, cueId As Integer, jumpCount As Integer)

Set the jump count for given cue

seqNum: Sequence ID
cueId: Cue ID
jumpCount: Number of jumps

Success. When false is returned check AutoGetLastError()

AutoResetSequenceCueTriggerCount(seqNum As Integer, cueId As Integer)

Reset the trigger count for given cue

seqNum: Sequence ID
cueId: Cue ID

Success. When false is returned check AutoGetLastError()

AutoGetContentIsConsistent(dmxFolderId As Integer, dmxId As Integer)

Get wether given content is consistent

dmxFolderId: DmxID (Folder)
dmxId: DmxID (Item)

Is content consistent?

AutoGetContentIsConsistentByName(ProjectPath As String)

Get wether content identified by name is consistent

ProjectPath: Path to project item

Is content consistent?

AutoCreateSequence ()

Create a new sequence

Sequence ID

AutoRemoveSequence (seqNum As Integer)

Remove sequence

seqNum: Sequence ID

Success. When false is returned check AutoGetLastError()

AutoSetShowCursorInFullscreen(siteNum As Integer, showCursor As Boolean)

Enable/Disable cursor in fullscreen

siteNum: Site ID
showCursor: Show cursor?

Success. When false is returned check AutoGetLastError()

AutoSetNodeOfSiteIsAudioClockMaster(siteNum As Integer, isMaster As Boolean)

Set site as audio clock Master

siteNum: Site ID
isMaster: Is site Master?

Success. When false is returned check AutoGetLastError()

AutoGetThumbnailByPath(ProjectPath As String, Width As Integer, Height As Integer, Data As Object)

Get thumbnail for given path

ProjectPath: Path to project item
Width: Integer to write width to
Height: Integer to write height to
Data: Image Data

Success. When false is returned check AutoGetLastError()

AutoGetThumbnailByItemIndex(treeItemIndex As Integer, Width As Integer, Height As Integer, Data As Object)

Get thumbnail for given item index

treeItemIndex: Item index
Width: Integer to write width to
Height: Integer to write height to
Data: Image Data

Success. When false is returned check AutoGetLastError()

AutoAddEncryptionKey(Key As String)

Add encryption key

Key: The encrption key

Success. When false is returned check AutoGetLastError()

AutoAddEncryptionPolicy(Policy As String)

Add encryption policy

Policy: The encryption policy

Success. When false is returned check AutoGetLastError()

AutoSetRouteInputToLayer(ByVal siteNum As Integer,ByVal enable As Boolean)

Sets input routing for given layer

AutoSetRouteInputToWidgetDesigner(ByVal siteNum As Integer,ByVal enable As Boolean)

Sets input routing to the Widget Designer

AutoEnableOutputForPicking(ByVal siteNum As Integer,ByVal outputNum As Integer,ByVal enable As Boolean)

Enable/Disable Output for layer picking feature

AutoSetASIOMasterVolume(ByVal siteNum As Integer,ByVal value As Double)

Sets the ASIO master volume

AutoCreatePlaylist(ByVal setDmxIds As Boolean,ByVal newDmxFolderId As Integer,ByVal newDmxId As Integer)

Creates a new Playlist

AutoCreatePlaylistByPath(ByRef PlaylistFolderPath As String PlaylistFolderPath,ByVal setDmxIds As Boolean,ByVal newDmxFolderId As Integer,ByVal newDmxId As Integer)

Creates a new Playlist in given path

AutoCreatePlaylistByItemId(ByVal playlistFolderItemId As Integer,ByVal setDmxIds As Boolean,ByVal newDmxFolderId As Integer,ByVal newDmxId As Integer)

Creates a new Playlist by given item id

AutoCreatePlaylistFromFolderByPath(ByRef PlaylistFolderPath As String PlaylistFolderPath,ByRef ResourceFolderPath As String ResourceFolderPath,ByVal setDmxIds As Boolean,ByVal newDmxFolderId As Integer,ByVal newDmxId As Integer)

Creates a new playlist from files in given folder

AutoCreatePlaylistFromFolderByItemId(ByVal playlistFolderItemId As Integer,ByVal resourceFolderItemId As Integer,ByVal setDmxIds As Boolean,ByVal newDmxFolderId As Integer,ByVal newDmxId As Integer)

Creates a new playlist from files in given folder

AutoPushBackPlaylistEntryByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal resourceDmxFolderId As Integer,ByVal resourceDmxId As Integer)

Pushes back Playlist entry by given dmx id

AutoPushBackPlaylistEntryByPath(ByRef PlaylistPath As String PlaylistPath,ByRef ResourcePath As String ResourcePath)

Pushes back Playlist entry by given path

AutoPushBackPlaylistEntryByItemId(ByVal playlistItemId As Integer,ByVal resourceItemId As Integer)

Pushes back Playlist entry by given item id

AutoInsertPlaylistEntryByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal resourceDmxFolderId As Integer,ByVal resourceDmxId As Integer,ByVal index As Integer)

Inserts new Playlist entry by dmx id

AutoInsertPlaylistEntryByPath(ByRef PlaylistPath As String PlaylistPath,ByRef ResourcePath As String ResourcePath,ByVal index As Integer)

Inserts Playlist entry by path

AutoInsertPlaylistEntryByItemId(ByVal playlistItemId As Integer,ByVal resourceItemId As Integer,ByVal index As Integer)

Inserts playlist entry by item id

AutoRemovePlaylistEntryByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal index As Integer)

Removes Playlist entry by dmx id

AutoRemovePlaylistEntryByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer)

Removes Playlist entry by path

AutoRemovePlaylistEntryByItemId(ByVal playlistItemId As Integer,ByVal index As Integer)

Removes Playlist entry by item id

AutoGetPlaylistSizeByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer)

Gets the number of items in Playlist

AutoGetPlaylistSizeByPath(ByRef PlaylistPath As String PlaylistPath)

Gets the number of items in Playlist

AutoGetPlaylistSizeByItemId(ByVal playlistItemId As Integer)

Gets the number of items in Playlist

AutoGetPlaylistEntryByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal entryIndex As Integer,ByVal pPlaylistEntryInfo As PlayListEntry)

Gets the Playlist entry of given Playlist at given position

AutoGetPlaylistEntryByPath(ByRef PlaylistPath As String PlaylistPath,ByVal entryIndex As Integer,ByVal pPlaylistEntryInfo As PlayListEntry)

Gets the Playlist entry of given Playlist at given position

AutoGetPlaylistEntryByItemId(ByVal playlistItemId As Integer,ByVal entryIndex As Integer,ByVal pPlaylistEntryInfo As PlayListEntry)

Gets the Playlist entry of given Playlist at given position

AutoGetPlaylistEntryIndicesByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal pData() As Byte,ByRef pSize As Integer)

Not yet documented.

AutoGetPlaylistEntryIndicesByPath(ByRef PlaylistPath As String PlaylistPath,ByVal pData() As Byte,ByRef pSize As Integer)

Not yet documented.

AutoGetPlaylistEntryIndicesByItemId(ByVal playlistItemId As Integer,ByVal pData() As Byte,ByRef pSize As Integer)

Not yet documented.

AutoSetPlayListEntryIndexByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal index As Integer,ByVal newIndex As Integer)

Sets Playlist entry by dmx id

AutoSetPlayListEntryIndexByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer,ByVal newIndex As Integer)

Sets Playlist entry by path

AutoSetPlayListEntryIndexByItemId(ByVal playlistItemId As Integer,ByVal index As Integer,ByVal newIndex As Integer)

Sets Playlist entry by item id

AutoSetPlayListEntryDurationByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the duration of given Playlist item

AutoSetPlayListEntryDurationByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the duration of given Playlist item

AutoSetPlayListEntryDurationByItemId(ByVal playlistItemId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the duration of given Playlist item

AutoSetPlayListEntryFadeOutTimeByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the fade out duration of given Playlist item

AutoSetPlayListEntryFadeOutTimeByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the fade out duration of given Playlist item

AutoSetPlayListEntryFadeOutTimeByItemId(ByVal playlistItemId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the fade out duration of given Playlist item

AutoSetPlayListEntryInPointByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the entry's inpoint

AutoSetPlayListEntryInPointByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the entry's inpoint

AutoSetPlayListEntryInPointByItemId(ByVal playlistItemId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the entry's inpoint

AutoSetPlayListEntryOutPointByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the entry's outpoint

AutoSetPlayListEntryOutPointByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the entry's outpoint

AutoSetPlayListEntryOutPointByItemId(ByVal playlistItemId As Integer,ByVal index As Integer,ByVal hours As Integer,ByVal minutes As Integer,ByVal seconds As Integer,ByVal frames As Integer)

Sets the entry's outpoint

AutoSetPlayListEntryTransitionByDmxId(ByVal playlistDmxFolderId As Integer,ByVal fadeFxId As Integer)

Sets the entry's transition

AutoSetPlayListEntryTransitionByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer,ByVal fadeFxId As Integer)

Sets the entry's transition

AutoSetPlayListEntryTransitionByItemId(ByVal playlistItemId As Integer,ByVal index As Integer,ByVal fadeFxId As Integer)

Sets the entry's transition

AutoSetPlayListEntryNoteByDmxId(ByVal playlistDmxFolderId As Integer,ByVal playlistDmxId As Integer,ByVal index As Integer,ByRef Note As String Note)

Sets the entry's note

AutoSetPlayListEntryNoteByPath(ByRef PlaylistPath As String PlaylistPath,ByVal index As Integer,ByRef Note As String Note)

Sets the entry's note

AutoSetPlayListEntryNoteByItemId(ByVal playlistItemId As Integer,ByVal index As Integer,ByRef Note As String Note)

Sets the entry's note

AutoRecordLiveInputById(ByVal folderID As Integer,ByVal fileID As Integer,ByRef Filename As String Filename,ByRef EncodingPreset As String EncodingPreset,ByVal durationHour As Integer,ByVal durationMin As Integer,ByVal durationSec As Integer,ByVal durationFrames As Integer)

Records live input with given id

AutoRecordLiveInputByIdStart(ByVal folderID As Integer,ByVal fileID As Integer,ByRef Filename As String Filename,ByRef EncodingPreset As String EncodingPreset)

Starts recording live input with given id

AutoRecordLiveInputStop()

Stops recording live input

AutoRecordLiveInputByName(ByRef LiveInput As String LiveInput,ByRef Filename As String Filename,ByRef EncodingPreset As String EncodingPreset,ByVal durationHour As Integer,ByVal durationMin As Integer,ByVal durationSec As Integer,ByVal durationFrames As Integer)

Records live input with given name

AutoRecordLiveInputByNameStart(ByRef LiveInput As String LiveInput,ByRef Filename As String Filename,ByRef EncodingPreset As String EncodingPreset)

Starts recording live input with given name

AutoExportVideo(ByRef Filename As String Filename,ByRef EncodingPreset As String EncodingPreset,ByVal seqNum As Integer,ByVal startHour As Integer,ByVal startMinute As Integer,ByVal startSec As Integer,ByVal startFrame As Integer,ByVal endHour As Integer,ByVal endMinute As Integer,ByVal endSec As Integer,ByVal endFrame As Integer)

Exports video with given parameters

AutoEncodeFileByName(ByRef Filename As String Filename,ByRef Preset As String Preset)

Encodes file by name

AutoEncodeFileById(ByVal folderID As Integer,ByVal fileID As Integer,ByRef Preset As String Preset)

Encodes file by id

AutoEncodeFileToTargetByName(ByRef Filename As String Filename,ByRef TargetPath As String TargetPath,ByVal overwriteExisting As Boolean,ByRef Preset As String Preset)

Encodes file to target by name

AutoEncodeFileToTargetById(ByVal folderID As Integer,ByVal fileID As Integer,ByRef TargetPath As String TargetPath,ByVal overwriteExisting As Boolean,ByRef Preset As String Preset)

Encodes file to target by id

6.9 Ports Used by PB and WD

This chapter lists all ports used by Pandoras Box and Widget Designer that are of interest when setting up a firewall or network router. Most ports are used for TCP and UDP unicast packets unless it is specified otherwise in the table. Please make sure that our multicast communication is not blocked.

If you are using configurable switches and would like to use IGMP Snooping, make sure that ALL switches in your network are configured accordingly. Otherwise, IGMP Snooping can be deactivated as well.

Ports	Protocol	Purpose
23	TCP	SerialDevice in Pandoras Box
80	TCP	Pandoras Box Webserver [153]
80 / 30300	TCP	Widget Designer HTTP Port (Listener) If port 80 is occupied, port 30300 or higher is used per default. This port can be adjusted in the WebServer Settings [1929].
81 / 30500	TCP	Widget Designer Upload Port (Listener) If port 81 is occupied, port 30500 or higher is used per default. This port can be adjusted in the WebServer Settings [1929].
1234 1235	UDP Multicast 239.0.0.10	Pandoras Box Communication
1300 1301 1302 1303	TCP	Pandoras Box Communication
2234 2235	UDP	Pandoras Box Communication
5900	TCP	VNC / Server Management Application [2109]
6211	TCP	Pandoras Box Automation [730] (SDK)
6212	UDP	Pandoras Box Automation [730] (SDK)
6213	TCP	WD Device [696] in Pandoras Box
6214	TCP	Pandoras Box Webserver [153]
6920	TCP	Pandoras Box CITP [148] and Widget Designer CITP [1256]
7000 7002 7004	UDP Multicast 239.0.0.30	Pandoras Box Communication
8080 / 30400	TCP	Widget Designer Websockets [1929] If port 8080 is occupied, port 30400 or higher is used per default. This port can be adjusted in the WebServer Settings [1929].
8992	UDP Multicast 239.0.0.11	PB Engine Health Report
49998 49999	UDP Broadcast 255.255.255.255	PB Menu [2097] Communication
50000	TCP	PB Menu [2097] Communication
50001	UDP Broadcast 255.255.255.255	Widget Designer Network Broadcasting Widgets [1274]

54300 54444	UDP	Pandoras Box Dante [166]

6.10 Copyright Pandoras Box

http://www.christiedigital.com

Pandoras Box uses the following licenses and libraries

3Dconnexion:
3D input device development tools and related technology are provided under license from 3Dconnexion.

Awesomium Technologies LLC:

Awesomium is a trademark of Awesomium Technologies LLC

Chromium Embedded Framework (New BSD License):

Crazy Eddie´s GUI (MIT License):

Peperoni-Light:

wxWidgets (wxWindows licence):

Microsoft DirectX:

Microsoft Visual C++ 2005, 2008, 2010, 2012, 2013, 2015, 2017, 2019 Redistributable:

SafeNet SDK:

Autodesk FBX:
This software contains Autodesk® FBX® code developed by Autodesk, Inc. Copyright 2008 Autodesk, Inc. All rights, reserved. Such code is provided "as is" and Autodesk, Inc. disclaims any and all warranties, whether express or implied, including without limitation the implied warranties of merchantability, fitness for a particular purpose or non-infringement of third party rights. In no event shall Autodesk, Inc. be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of such code.

FreeImage (FreeImage Public License):
This software uses the FreeImage open source image library. See http://freeimage.sourceforge.net for details. FreeImage is used under the FIPL.

Mainconcept Mpeg2:

NVAPI SDK:

Oculus SDK:

OpenSSL (dual license: OpenSSL License and SSLeay license):

Apple Quicktime SDK:
© 2016 Apple Inc. All rights reserved.

Deltacast SDK:
© 2015 Deltacast. All rights reserved.

Stardock DirectSkin SDK:
© 2016 Stardock Corporation. All Rights Reserved.

Hap:
© 2011-2013, Tom Butterworth and Vidvox LLC. All rights reserved.

Snappy:
© 2011 and onwards Google Inc. All rights reserved.

Squish:
© 2006 Simon Brown si@sjbrown.co.uk

YCoCg-DXT Compression:
© 2007 Id Software, Inc. Written by J.M.P. van Waveren

Streamix:
© Streamix

Telerik SDK:
© 2016, Progress Software Corporation and/or its subsidiaries or affiliates. All Rights Reserved.

zlib:
© 1995-2013 Jean-loup Gailly and Mark Adler.

libpng (libpng license):
libpng versions 1.0.7, July 1, 2000 through 1.6.23, June 9, 2016 are © 2000-2002, 2004, 2006-2016 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux, Eric S. Raymond, Mans Rullgard, Cosmin Truta, Gilles Vollant, James Yu.

SVN:
Apache License version 2.0.

GIT:
LGPL-2.1

NSIS (zlib/libpng license):
http://nsis.sourceforge.net/Main_Page

TinyTiff:
© 2014-2015 by Jan W. Krieger. GNU Lesser General Public License 3.0.

Roboto Font:
© Google Inc. Apache License, Version 2.0.

FontForge:
© George Williams and https://github.com/fontforge/fontforge/blob/master/AUTHORS. GNU Lesser General Public License 3.0.

Medialooks MFormats SDK:
© https://www.medialooks.com/

FFmpeg:
This software uses libraries from the FFmpeg project under the LGPL version 2.1.

Christie
Pandoras Box

Part 7
Widget Designer

7 Widget Designer

The Christie® Pandoras Box Widget Designer is an advanced control surface creation framework, that lets you create dedicated user interfaces and interaction logic. The great number of features are all based around the idea, that even non-programming specialists should be given the tools to create truly immersive interactive experiences for their customers.

You can add Widgets to your interface, e.g. buttons and faders. The Widget Designer interface is based on HTML-5 and features the possibility to style and manipulate complete pages or single widgets due to internally used CSS3 styles. Those can be imported and applied to the local user interface. Web designers can easily style customized control panels for a variety of different users and applications.

In addition, visual node based programming is available to create customized show-control scenarios. With the embedded node programming environment users can route and set up almost any possible control scenario. Easily interact with sensors and data sources to route input data to any other output protocol such as Art-Net, Midi/MSC, TCP/UDP, DMX or RS-232/-422 devices. Node programming can be an alternative or an addition to scripting.

One of the most powerful concepts in WD is the script language [1511]. Anywhere in WD where commands can be entered a Script Assistant helps you to find and use the desired command. Currently over 1500 commands [1520] are available to control specific features of Pandoras Box, Widget Designer or other 3rd party products and protocols.

Regarding the Pandoras Box [67] integration, Widget Designer can be directly used as a sequence device as part of the Pandoras Box GUI. One also has the option of reading out a great variety of Pandoras Box values.

The Widget Designer comes in three versions :

Widget Designer Free Edition (formerly known as STD)
This edition is provided as a freeware license and can be downloaded on the official Christie website. It allows you to create custom user interfaces and to work with commands.
Using the build-in Web Server [1929] you may publish one page to be accessed by one user.
You may open a project that was originally programmed with a licensed Widget Designer. All project elements, that are also available in of the Free edition will be displayed and will run in the dedicated way. If licensed features were used (e.g. the AirScan) you will get a message that these elements are not available in the Free version. Other features, like nodes [936] or variables [1900] for example still exist even though they are not part of the Free package, but are restricted in their use. In short, the Free version can run but not change the settings of those features. Only a licensed Widget Designer provides full editing access.

Widget Designer (formerly with the addition PRO)
The licensed Widget Designer offers various advanced features and tools to give you the most flexible and easy way of programming user interfaces and interactive show elements. This version allows to program visually with so called nodes. By simply connecting those visual control components you may create a dedicated interaction logic.
The Script Language [1511] is fully supported including variables [1900], functions and macros [1897].
Using the build-in Web Server [1929] you may publish one page to be accessed by several users at the same time whereas the sessions are all synchronized. This version also offers to choose a dedicated network adapter for various protocols.

Unlimited Web Clients Option (formerly known as the ULT edition)
This option extends the Web Server feature of a licensed Widget Designer.
Using the build-in Web Server [1929] you may publish an unlimited number of pages to be accessed by several users at the same time. In addition you may decide whether the users see a synchronized interface of an individual one. This is perfect for installations where several remote controls are needed with a centralized control station, or other large multi-user projects that focus on the integrated Web Server

Per default, those three version are started in a mode where you have full access to all features that come with your version. But you have also the choice to start a project in the so called **Viewer Mode**
In this mode the project cannot be change anymore, it can only be used as intended and hence is perfect for end customers in museum-like scenarios. More information... [787]

Please see the following topics for working with the Widget Designer:
Installation and Launching [787]
User Interface [789]
Widgets [818]
Nodes [936]
Tools [1483]

7.1 Installation and Launching

Installation

All WD versions require a 64-bit processor hardware and run on Windows only. Windows 8.1 or 10 Professional are recommended.

The C++ Redistributables 2017 are included in the installation package and need to be installed.

Microsoft .NET Framework needs to be downloaded and installed manually. The Download-Center provides the installers under Drivers > .Net Framework.
All WD versions require .NET Framework 3.5. In addition, WD versions 6.0.x require .Net Framework 4.5.2 or higher and WD versions 6.1.x .require Net Framework 4.8.
If you have any issues with the installer itself, you may need to update your windows installer from the Microsoft homepage.

As well, we recommend to use the newest dongle driver version (from our Download-Center) when installing the software Widget Designer or Widget Designer with unlimited web clients.
The WD Free Version does not require any dongle, neither for the installation nor for running the software.

Launching

Once WD is installed on your PC, you may launch the application from the start menu or create a shortcut on your desktop. Depending on your license, you may start Widget Designer in three editions which were explained in the main chapter 786:
WD Free
WD
WD Unlimited

Another way to start the WD in a defined edition is the so called command line argument from the application, which is explained below.

Per default, those three version are started in a mode where you have full access to all features that come with your version. But you have also the choice to start a project in the so called Viewer Mode.

All files created at runtime that are not associated with a specific project are saved under `%PROGRAMDATA% > WidgetDesigner`. This includes the following files:

- Default Project (default.wd)
- Settings (settings.json)
- User Profiles
- Dialog Cache
- Websession Storage
- Logs

Viewer Mode

You can start each WD edition in the Viewer Mode. In this mode the project cannot be change anymore, it can only be used as intended and hence is perfect for end customers in museum-like scenarios.

Usually, you would prepare a project, save it and load it in the Viewer Mode. The File Menu 792 offers the option "Create Viewer Shortcut" to saves a shortcut which opens the current project in Viewer Mode.

In detail, the Viewer Mode loads a project with the following restrictions:
- the Tool and Menu Bar are missing
- there are no context menus / right-click menus
- all WD (!) shortcuts are disabled, this includes shortcuts that open and save projects or open certain Widget dialogs. There are only two exceptions: [Alt + F4] closes the application (this can be password protected). [F11] is optional and toggles the Kiosk View. Note that other Windows shortcuts still work!

As a result, the project cannot be toggled from "Run Mode" to the "Create" or "Edit Mode" [789] and there is no option to change the user interface if it was not prepared beforehand with scripts. All scripts, Nodes, Widgets, connection etc. are working as they do in the "Run mode".

The Status Bar and the Window Titlebar are displayed according to the Edit Windows dialog [803]. If you are interested in a fullscreen view of your project, you can also activate the "Kiosk View [801]" from the View Menu or allow the shortcut [F11] to toggle it. This can be done in the Security Settings dialog [796]. There, you can also setup a password for the Viewer Mode. It needs to be entered when a project is loaded, saved or closed.

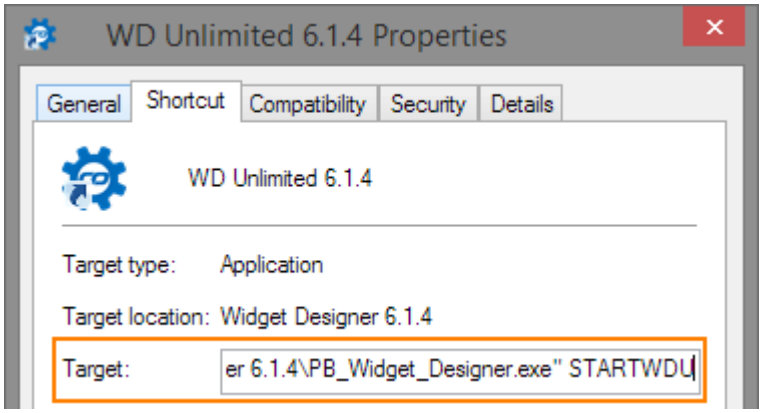
Regarding the saving process, the Viewer Mode offers no options to save and overwrite a project. Even when the project is being closed, Widget Designer does not ask whether the changes should be saved. The only possibility to allow saving is to script this beforehand, e.g. with the command `WDSave` [1810]. Depending on the Security Settings, the user needs to confirm this with a password. Same applies to scripts that allow to load another project or close it.

Command Line Argument

You can use command line arguments in the command prompt, within a BAT file or simply in a shortcut as described below. You may also use these additions with the command `ApplicationStart(Filepath,Optional Commandline Arguments)` [1522].

The chapter Widget Designer [786] explains the different editions and the Viewer Mode, which can also be secured with a password [796].

Create a shortcut of the Widget Designer .exe file (for example on your desktop). Right-click on this shortcut > Properties. On the tab "Shortcut" attach the following to the Target text field (with a space in between, see example below).



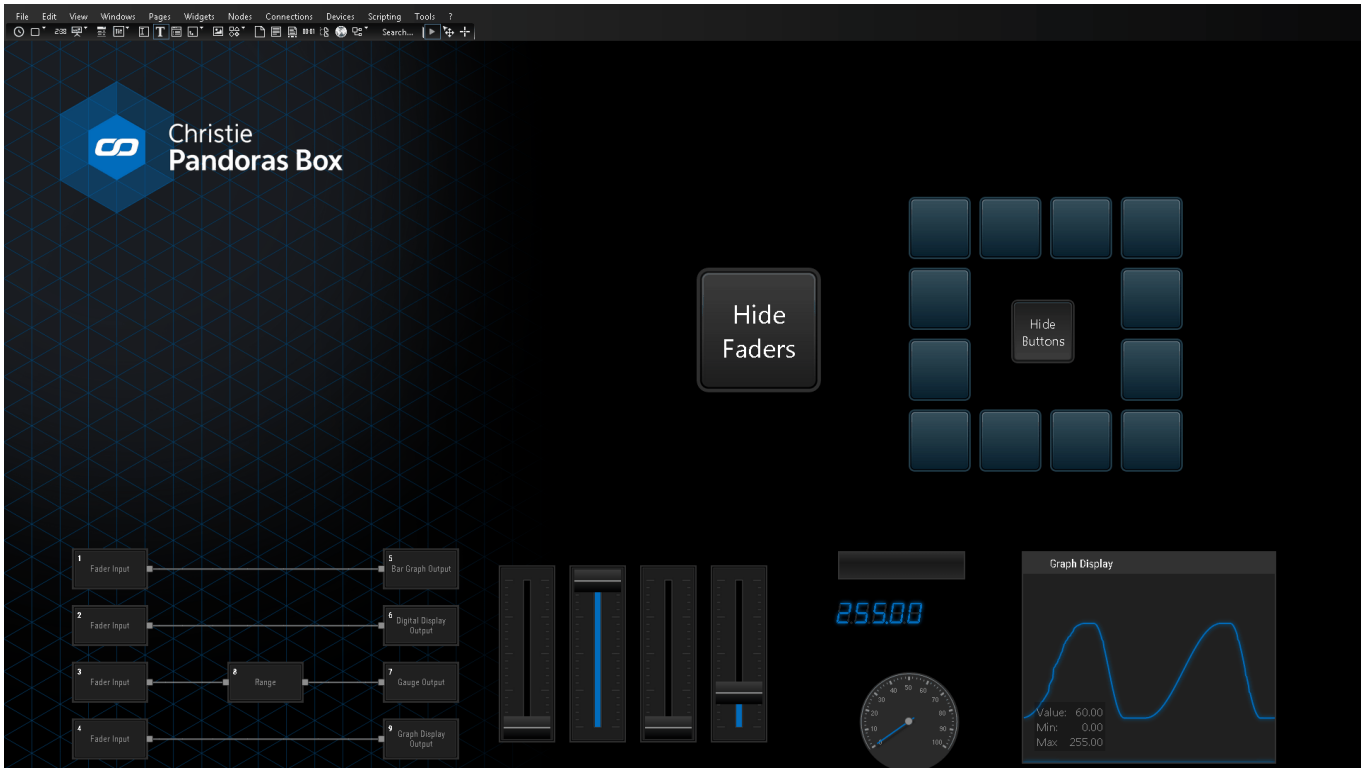
STARTWDF: launches the free edition
STARTWD: launches the regular edition, or leave the path as it is
STARTWDU: launches the edition with unlimited web clients

--viewer: launches the Viewer Mode, declare the edition first if necessary

"project path": loads the specified project; declare the edition first if necessary

Example:
"C:\Program Files\Christie\Widget Designer 6.1.4\PB_Widget_Designer.exe" STARTWDF --viewer
"C:\Program Files\Christie\Widget Designer 6.1.4\PB_Widget_Designer.exe" STARTWDU "C:\Christie\projects WD\6-1-4 Show\6-1-4 Show.wd"

7.2 User Interface

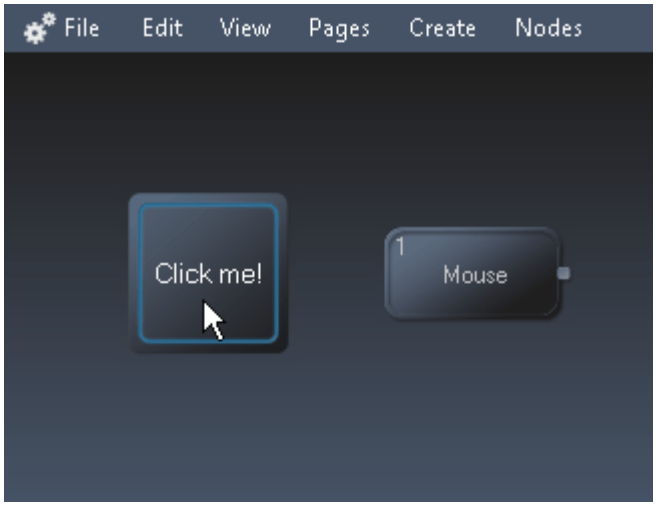


When starting WD, you will see an empty screen with a main menu bar and a toolbar containing all available widgets. The empty screen represents the main background of your application window. A right-click here opens a context menu that offers ALL options from the main toolbar as well.

Please see the following topics for detailed description regarding the different menus, from File Menu [792] until Tools Menu [813].

The topics Project Settings [798] and Window Settings [803] explain how general settings can be influenced. For example, if you are creating a touch screen user interface, you can hide the standard windows borders and the icons to close, maximize or minimize the application.

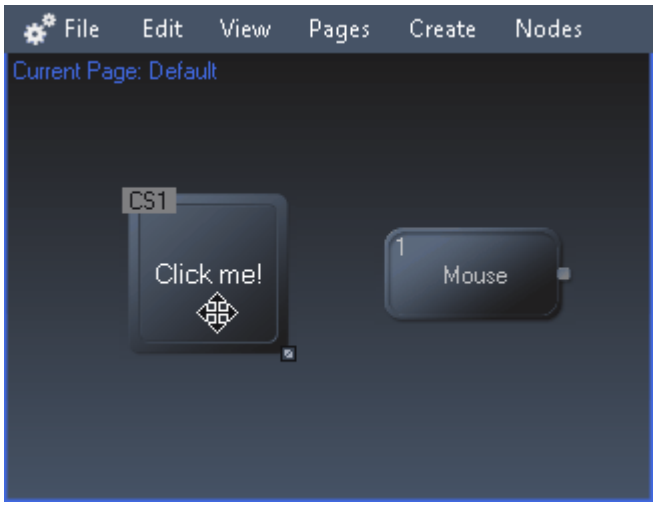
The last topic within this chapter lists all available keyboard shortcuts [816].



You may use the entire background area to create and position any controls you wish to be part of your user interface (UI) to remote control Pandoras Box or any other device connected.

Widget Designer has three operating modes:
The default "run mode" makes the *function* of each tool available. For example, buttons may be clicked, fader bars may be dragged. Tools that are remoted by incoming data behave accordingly.

Shortcut to enter the run mode: [F8]
Or click the play symbol in the toolbar.



In the "edit mode", also called "move mode" you may move, resize and copy any controls and - if talking about nodes - connect them.

In order to select an item, simply click on it. In order to select multiple elements, left-click and drag a selection frame or click on additional items whilst holding the [Ctrl] key.

To draw a selection frame, hold the left mouse button whilst dragging the mouse. If you start somewhere and drag to the bottom right side, a blue selection frame appears selecting all items completely surrounded by the frame. All other directions create a green frame, selecting all items it touches.

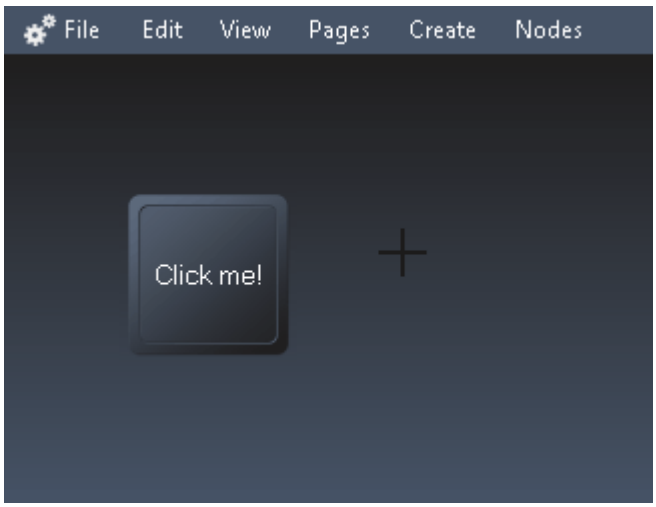
Most widgets will show a small diagonal double arrow icon at their bottom right corner. Click and drag it to change the size.

All standard user control and nodes support copy and paste with [Ctrl + C] and [Ctrl + V].

If you need to delete an item, right-click on the element and click "Delete" or select it and press the delete key on the keyboard.

Even though data is still calculated and processed in the background whilst being in the edit mode, tools and labels will only be updated as soon as you switch back into the run mode.

Shortcut to enter the edit / move mode: [F9]
Or click the arrow cross symbol in the toolbar.



The third operating mode is only active if you have chosen an item from the widgets menu [818] or nodes menu [936]. WD will automatically change the mouse cursor to a crosshairs icon. Wherever you left-click now, the chosen item is generated. When you have as many copies from the item as you need, enter the edit mode to edit, e.g. resize them or enter run mode in order to use them.

It is possible to deactivate the create mode after creating a widget per default, please refer to Profile Settings [795].

Shortcut to enter create mode (last widget created): [Ctrl + Shift]
Or click the crosshairs symbol in the toolbar.

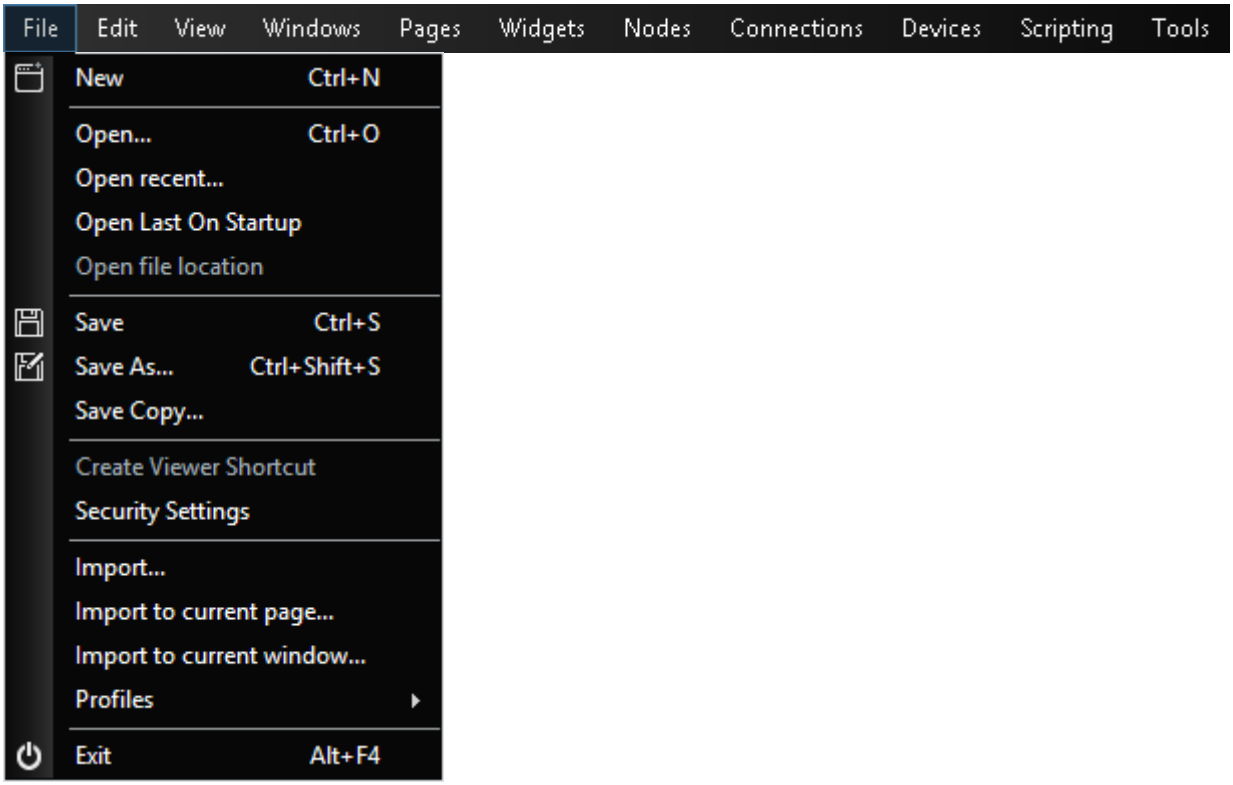
The edit dialogs contain different, extendable sections. Widget Designer remembers which panel is extended and which not for all other newly opened dialogs. The check box at the right side of each bar allows you to keep the panel opened when it is ticked. If it is unchecked, the panel will close as soon as another one is opened.

The default behavior can be set in the User Profile [795].

All edit dialogs offer the possibility to "Automatically apply changes", this mode is ticked by default. All changes involving check boxes, buttons and drop-downs will be applied directly. Changing the content of a text box will take place as soon as you leave the box or press [Enter].

7.2.1 File Menu

The File menu in WD lists commands that influence your WD project.



New
Creates a new WD project file.The shortcut is: [Ctrl + N]

Open
A click on "Open..." (shortcut [Ctrl + O]) opens a file browser where you can choose the WD project file to be loaded.

Open recent
This opens a list of recent projects. Items from the list can be removed manually.

Open last on Startup
To load the current project on next start, click on "Open Last On Startup".

Open file location
This opens the Windows File Explorer showing the folder where the current project is saved in.

Save
Click on save or use the shortcut [Ctrl + S] to save your WD project file.
If you choose this option, a folder will be created at the location you specified. Inside this folder, you will find the WD project file as well as a data-folder containing all images applied to different controls in WD, as well as a folder containing custom CSS templates [814] if used. Without the extra folder, the data-folder will be saved at the same location as the project file if there is any such data used at the project.

Save as...
Use this option to save the current WD project file under a different name.
Please note: WD only asks you to save the current project file when exiting the application, there is NO AUTOSAVE.

Save copy
"Save Copy" creates a copy of the current project and saves it at the specified location while you keep working in the current file. This is especially useful when saving backup versions of your work.

Create Viewer Shortcut
This opens a dialog where you can choose to save a shortcut which would open the currently used Widget Designer version in Viewer Mode and load the current project. This option is only available if the project has been saved before. Optionally, you can set up a password for the Viewer Mode as described in the topic "Security Settings [796]".

Security Settings
This allows to set up passwords for the Viewer Mode and general editing, see topic "Security Settings [796]".

Import...
With WD V6, it is possible to import widgets, nodes, pages or whole windows from another project. If you choose this option, all parent elements of the desired element(s) will be imported, too. E.g.: Importing a single custom script button would import the page and the window as well, unless you define something else in the Import Settings [793].

Import to current page...
This option imports the selected widgets and / or nodes to the current page, for more information see Import Settings [793].

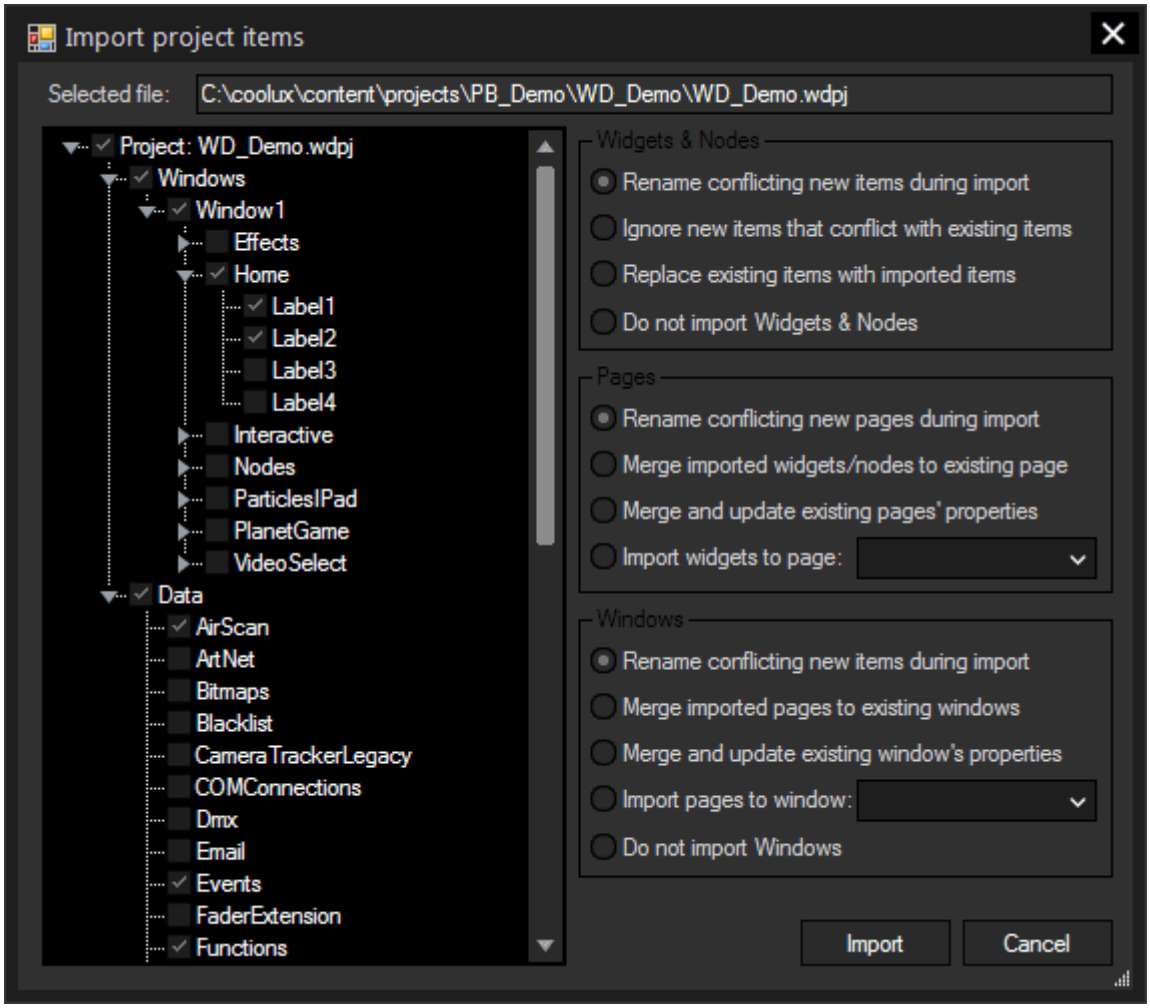
Import to current window...
This option imports the selected pages, widgets and / or nodes to the current window, as well as selected data. For more information see Import Settings [793].

Profiles
Choose between different user profiles or edit existing ones in the Profile Settings [795]. A profile stores your working environment.

Exit
Closes the application (with asking if changes made within a project should be saved).

7.2.1.1 Import Settings

With WD V6, it is now possible to import objects and data from other WD project files. As explained in the previous chapter, the File menu [792] in WD offers several Import commands. When selecting one of them you are asked to choose a project and then a dialog opens which is explained below.
By the way, if one or more objects that have to be imported are encrypted with a password, they can only be imported after entering the correct password.



Widgets & Nodes

In this section you can choose between three ways of handling conflicting items. If you tick "Rename conflicting new items during import", the imported item will be renamed with a "_2" at the end (or higher numbers if this one exists already).
Example: CustomScript1 -> CustomScript1_2

You can also choose to ignore any item from the imported project that conflicts with the current project, or to replace existing items with the new imported ones.

If you only want to import pages or windows without any widgets and nodes, tick "Do not import Widgets & Nodes".

Pages

If you tick "Rename conflicting new items during import", the imported page will be renamed with a "_2" at the end (or higher numbers if this one exists already).
Example: Page1 -> Page1_2

Like above, you can also choose to merge existing pages while keeping the properties of the current project ("Merge imported widgets/nodes to existing pages") or adopt the imported pages' properties ("Merge and update existing pages' properties"). With both modes, only the widgets and nodes are transferred directly.

With "Import widgets to page ..." you can even import all selected widgets and nodes to a specified page from your current project.

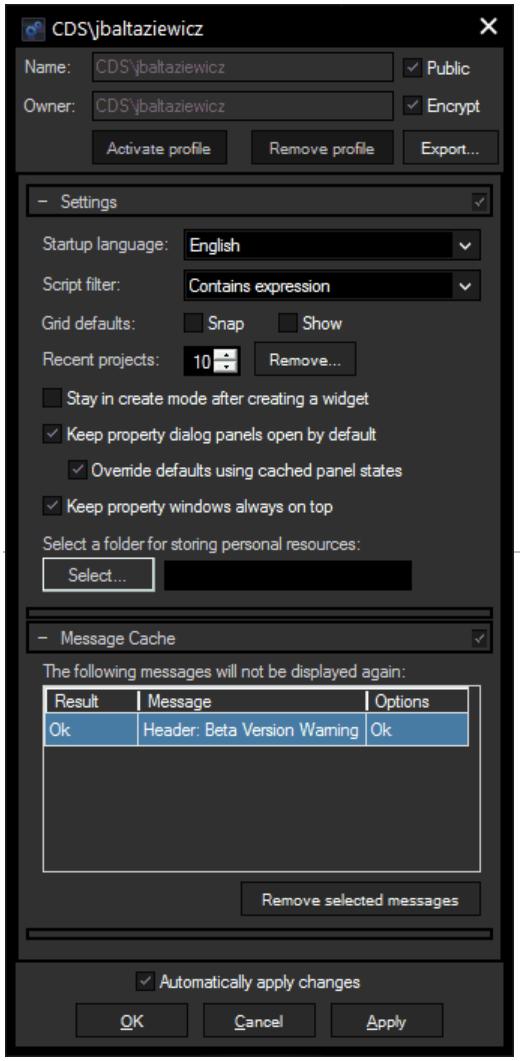
Windows

If you tick "Rename conflicting new items during import", the imported window will be renamed with a "_2" at the end (or higher numbers if this one exists already).
Example: Window1 -> Window1_2

Like above, you can also choose to merge existing windows while keeping the properties of the current project ("Merge imported pages to existing windows") or adopt the imported windows' properties ("Merge and update existing windows' properties"). With both modes, only the pages are transferred directly.

With "Import pages to window ..." you can even import all selected pages to a specified window from your current project.
Last, you can choose to not import any windows at all.

7.2.1.2 Profile Settings



With the user profiles, you can specify some general settings concerning the usage of the WD application itself. To open the profile dialog, go to the File menu [792], choose "Profile" and one from the list. Other project Settings can be found under Edit menu > Project Settings [798].

The **"Owner"** is the Windows account you are logged in with. When creating new profiles, you can define the **"Name"** yourself.
If the **Public** box is ticked, everyone using this PC will be able to access this profile. If it is not ticked, only the "Owner" can see and edit the profile.
The **Encrypt** box encrypts the profile information. This might come in useful in later versions, when passwords can be saved with the user profile.
"Activate profile" activates the profile you are currently editing, **"Remove profile"** deletes it. With **"Export"** you can save it.

Settings

Changing the **Startup language** (English or German) takes effect after restarting Widget Designer.

The **Script filter** affects the results shown in the Script Assistant [1511]. You can choose between "Starts with expression", "Contains expression" and "Camel Case Search". The last option allows you to abbreviate your search entry to the letters used at the command's camel case nomenclature.
E.g.: "wdcstc" for "WDCustomScriptTextColour"

If you like working with a **grid**, you can also set your grid to automatically be visible when opening the Widget Designer, or to start in snap mode.

The list of **recent projects** can be edited, entries can be removed.

If you uncheck the box **"Stay in create mode after creating a widget"**, the mode changes to edit/move mode [789] automatically after creating a widget. Multiple widgets of the same kind then can be produced at once while pressing [Ctrl + Shift].

"Keep property dialog panels open by default" automatically ticks or unchecks all check boxes at the expandable property dialog panels. Without the check box a section closes when another one is opened. If you have preferred layouts of extended sections for certain types of widgets, tick **"Override defaults using cached panel states"** to open all respective property dialogs with the same layout.

You can also decide to **keep the property windows always on top**, this is especially useful while adjusting and testing different settings.

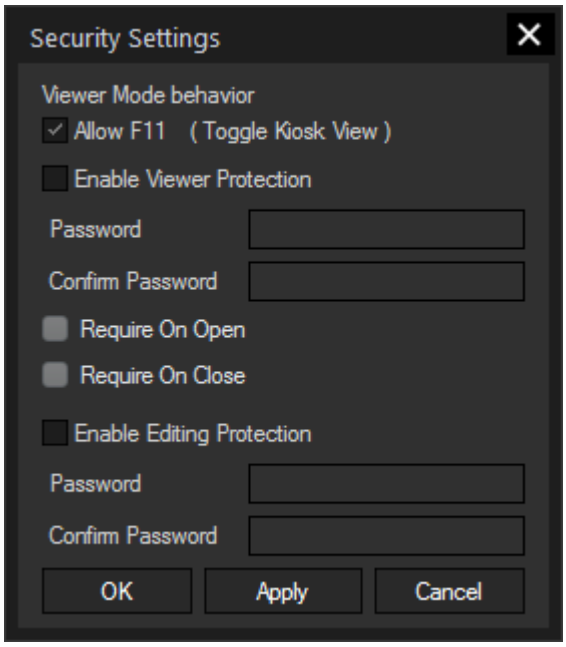
Message Cache

Most of the pop-up dialogs offer the possibility "Do not ask again". The answer to this dialog will be cached for future occasions. If you want to display the message again, select it at the Message Cache and press **"Remove selected messages"**.

7.2.1.3 Security Settings

The Security settings allow you to secure your entire project with a password. If you are rather interested in protecting a single page, open its Edit Page dialog [805]. Another option is to protect (parts of) scripts with the command `WDPasswordDialog` [1791].

Open the File Menu [792] and choose "Security Settings" to open the following dialog. It allows to setup two passwords, one for the Viewer Mode and one for general editing. In case you like to protect the "Viewer Mode", it is a good idea to activate both options but choose a different password. If you are not interested in the "Viewer Protection" you can use the Editing Protection independently.



Viewer Mode behavior

The option "Allow F11" applies only when the project is started in the Viewer Mode [787]. It is enabled per default. The key [F11] toggles the Kiosk View [799], which shows the interface in fullscreen and hides all other bars.

Viewer Protection

First, tick the check box "Enable Viewer Protection", then enter a password and confirm it. You can use as many characters as you wish. Lastly, choose whether the password needs to be entered when the project is opened and / or when it is closed. It is also needed when saving the project. Note, that the dialog asking for the password only includes an on-screen-keyboard when the project is closed. In any case, you can enter the password with a connected keyboard, an on-screen keyboard or via a VNC connection.

Click "OK" to close the dialog and save your project. If you now open it in the Viewer Mode, Widget Designer asks for the password.

Note that the File Menu offers the option "Create Viewer Shortcut" which saves a shortcut to open the currently used Widget Designer version in Viewer Mode and load the current project. Read more about the Viewer Mode... [787]

Editing Protection

First, tick the check box "Enable Editing Protection", then enter a password and confirm it. You can use as many characters as you wish. Click "OK" to close the dialog and save your project. From now on, you need to enter the password when the project is loaded, saved or closed.

7.2.2 Edit Menu

The Edit menu in WD lists commands influencing the look and behavior of your working environment.

Item Properties
Opens the Item Properties dialog of the selected widget or node.

Run
Activates the operating mode called "run mode". (Shortcut: [F8])
The operating modes are explained here [789].

Edit / Move
Activates the operating mode called "edit / move mode" (Shortcut: [F9]).

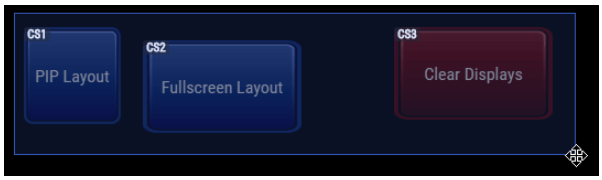
Copy
Copies widgets that have been selected before.

Paste
Pastes widgets that have been copied before. The controls are inserted according to the mouse pointer's position. If you like to paste controls to another page use the shortcut [Ctrl + P] while being in the edit / move mode.

Delete
Deletes widgets that have been selected before.

Selection
This opens a list of actions that can be executed for single or multiple selected widgets. They can be moved pixel-wise with the [arrow keys], or snap to other widgets with the shortcut [Ctrl + arrow key].

Align Selection
To align multiple UI elements to each other, you will first need to select the items while being in the edit mode and then apply the desired alignment method.
To select multiple widgets, you can either hold the [Ctrl] key and click another button to add it to the selection or you can draw a selection box with the left mouse button.



Please head to the chapter "User Interface" if you like to learn more about the edit mode. [789]

Z-Order
When you need to place one element on top of another, you may click on the desired element(s) and choose "Send to Back" or "Bring to Front".

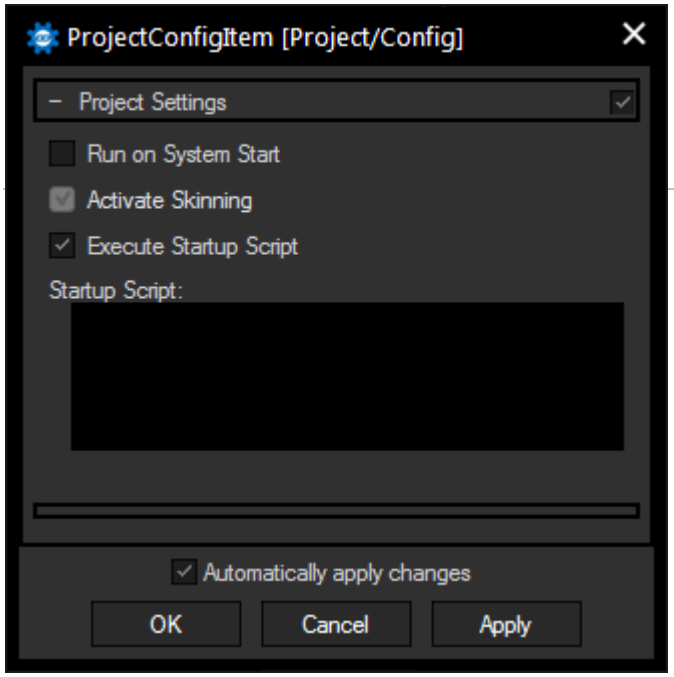
Reset protection passwords
Deletes all passwords from the project cache. If you have objects protected by passwords, you will have to enter each password again for accessing those objects. For more information on the usage of passwords, please refer to the Project Settings dialog [798].

Project Settings
This opens the Project Settings dialog [798].

Web Server Settings
This opens the Web Server Settings dialog [1929].

Web Styles
This opens the Effects and Animations dialog, please see the chapter Effects & Animations (Web Styles) [814].

7.2.2.1 Project Settings



The Project Settings include adjustments for the whole project. To open the dialog go to the Edit menu [797] in WD and choose "Project Settings". More project settings can be found under File menu > Profile [795].

Project Properties Panel

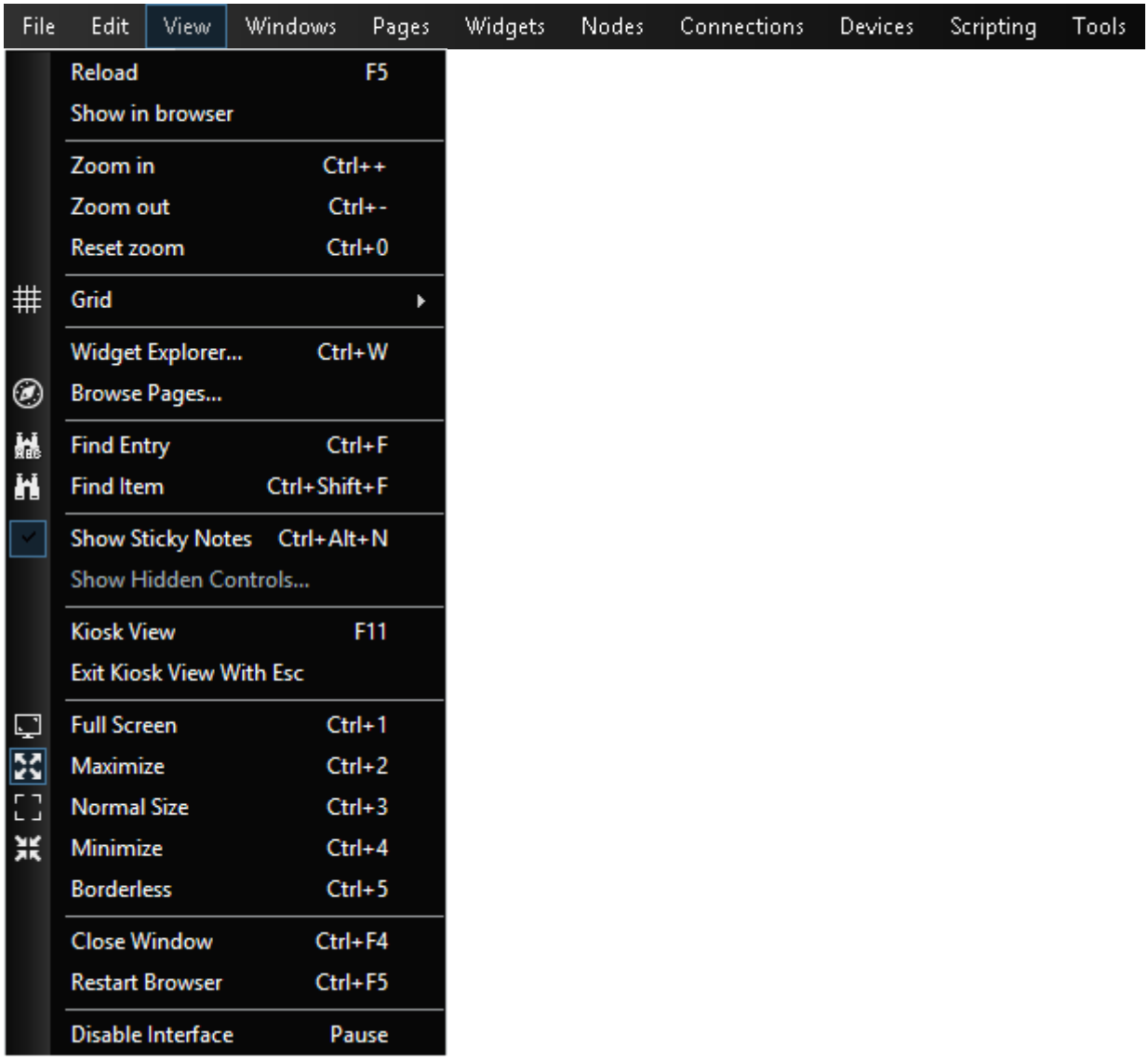
Tick **"Run on System Start"** if you would like the project to automatically open after your system finished booting.

"Activate Skinning" loads the dark skinning of the WD application. Restart WD after changing this option.

If you like to run a script (consisting of one or more commands) each time the project is loaded, add the commands to the text box and tick the check box **"Execute Startup Script"**. The topic Script Language [1511] explains commands and how to use them in more detail. Tip: You may use a macro [1897].

7.2.3 View Menu

The View menu in WD lists commands influencing the setup of your working environment.



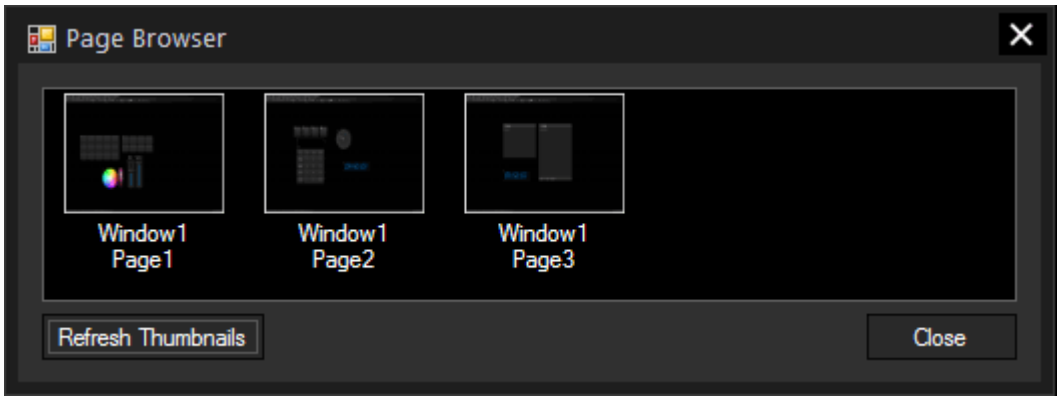
Reload
Reloads the current page.

Zoom
The maximum zoom factor is 300%, minimum 25%. To reset the zoom, press [Ctrl + 0].

Grid
"Show Grid" will display a pattern in the background. In the "Settings" dialog you can adjust its size. With the "Snap to Grid" feature, you may use the (in-)visible grid as magnetic guides.

Widget Explorer
Opens the Widget Explorer [802] dialog that displays a tree view of all items of your project.

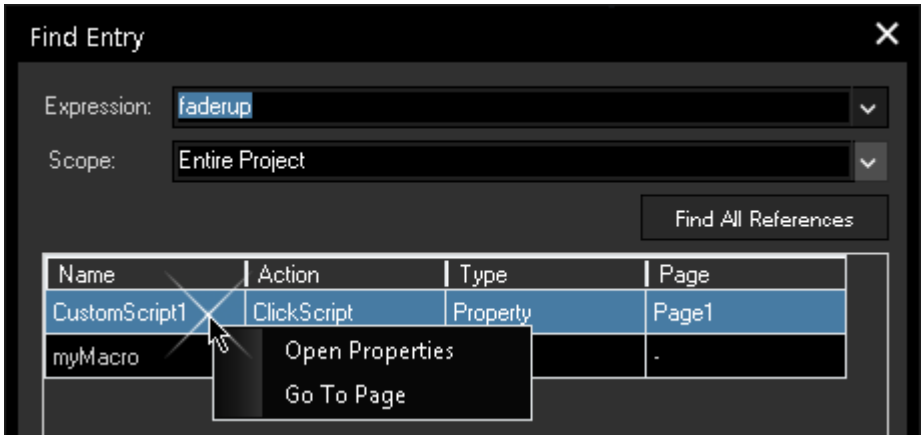
Browse Page
Opens a browser where all pages created are displayed with a thumbnail and their name. This makes it possible to navigate easily through many pages. By clicking on one of the pages you will switch to it. You may open the page browser via a command: OpenPageBrowser [1563]



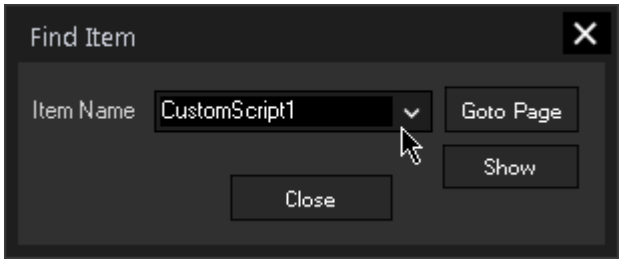
Find Entry
Opens a dialog where you may enter a (part) of a command.

"Scope" opens a drop-down list where you may choose to search the entire project or only on the current page / window.
"Find all References" starts the search and displays where the command was found, for example in a Custom Script Button [822], a Script Output node [1207], a Function/Macro [1897], or Page [805].
Right-click on the result to open the according properties or go to the page where it is located.

If you look for a tool that helps you debugging, i.e. finding errors in your command programming, open Tools > Options > Debug [1510].



Find Item
Opens a dialog where you may enter a widget´s name.
"Goto Page" calls the page where the control is located.
"Show" will not switch pages but overlays them with the control. The control will flash three times and then stay in front of them until you go to its source page. Thus you may use the control or edit its item properties.



Show Sticky Notes
Sticky Notes [921] are only visible if enabled in the menu or by using [Ctrl + Alt + N].

Show Hidden Controls
If there are any hidden widgets or nodes in the project, this option opens a dialog where all hidden items are listed. You can choose which ones of them you want to be visible again.

Kiosk View
This option removes all bars and menus and resizes the main GUI to full screen. For leaving the Kiosk mode, press [F11] again. Additionally, you can decide whether to use the [Esc] key for exiting Kiosk mode.
The Kiosk view gives you the largest available space to arrange any controls forming an individual user interface.

Full Screen
Maximizes the WD window over the whole screen and hides the standard Windows border and application title bar for sure. You may as well use the shortcut [Ctrl + 1]

Maximize
Maximizes the WD window over the whole screen. Shortcut: [Ctrl + 2]

Normal Size
Brings the WD to the size and style it had before maximizing, or the default window size of 800x600px. Shortcut: [Ctrl + 3]

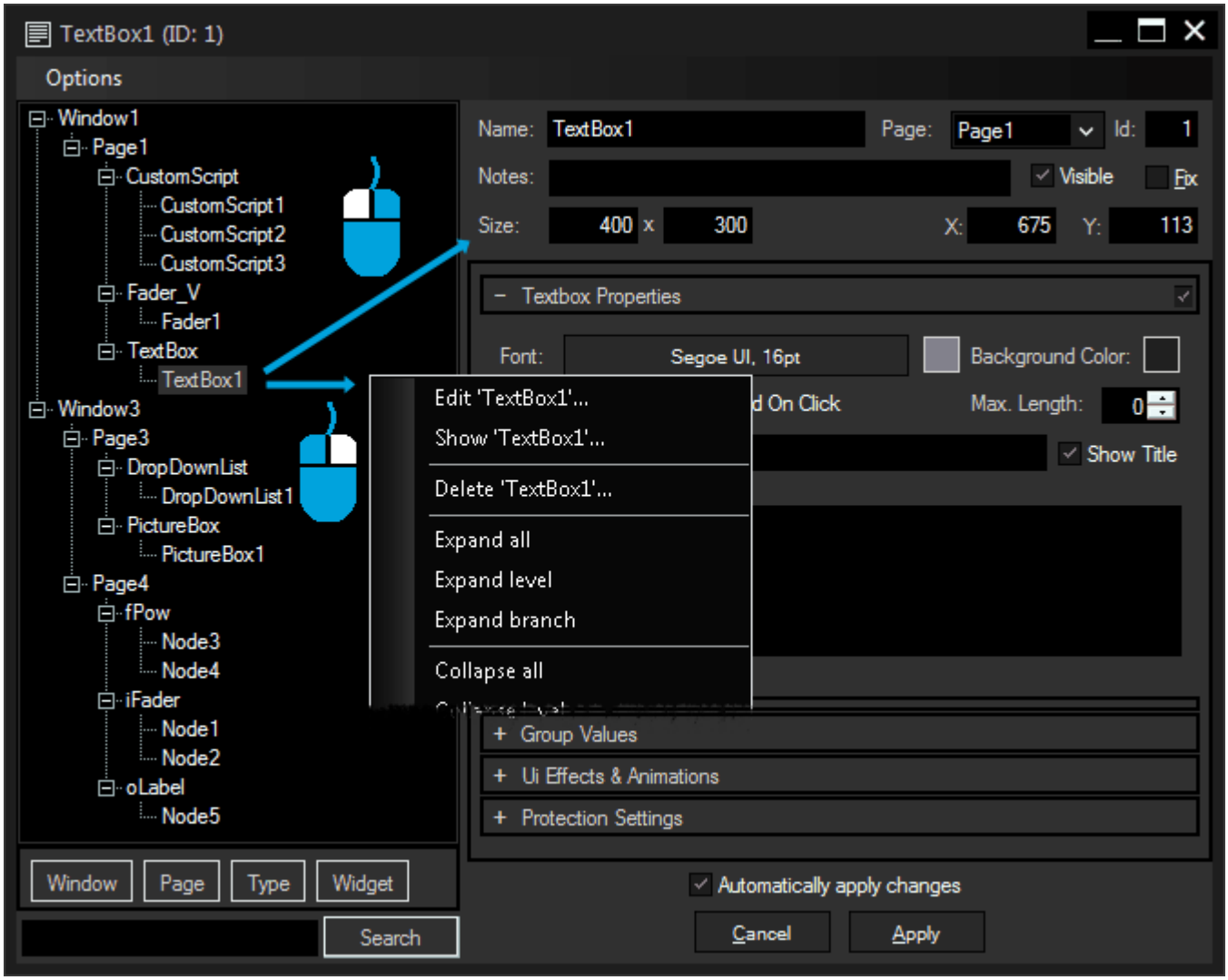
Minimize
Minimizes the WD window. Shortcut: [Ctrl + 4]

Close Window
Closes the current WD window. If there is no other window opened, the program will be closed after asking for the project to be saved.

Disable Interface
With "Disable Interface" you can disable all (!) mouse, touch and keyboard input and nothing can be clicked or entered anymore. When an input is detected, a Windows error sound is played. This mode is for example perfect for programmers who want to make sure, nobody changes the interface or clicks anything. You can enter and leave this mode with the shortcut [Pause]. Note that the [Pause] key differs from the [Play/Pause] key for the Windows Media Player and cannot be found on all (hardware) keyboards anymore but it is available in the Windows onscreen keyboard.

7.2.3.1 Widget Explorer

The widget explorer [Ctrl + W] displays a tree view of all items of your project. They are sorted by windows [803], pages [805], types (of widgets [818] or nodes [936]) and individual items. The four buttons below the tree view allow you to blank out the corresponding type. You can also search for a special item.



Left-clicking on an item will load the item's properties directly into the widget explorer. Of course the properties can be edited directly. You can also left-click on items in the main user interface to load their properties into the widget explorer. Note that the Item Properties dialog can not be opened in addition to the widget explorer.

Right-clicking on an item will show you some additional options for interacting with the tree view itself. For example you can show or delete items and expand, collapse or delete branches.
Please bear in mind that deleting branches will result in deleting the respective items as well.

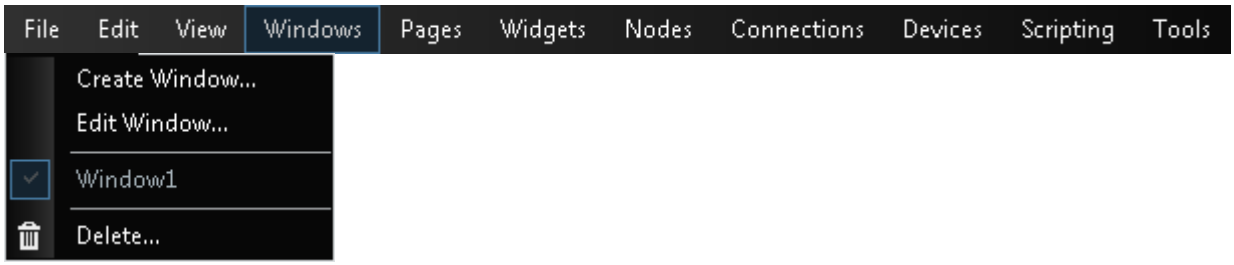
The lower case letter in front of each node indicates its type (e.g.: iFader = Fader Input Node, oFader = Fader Output Node)

The widget explorer is being refreshed automatically.

7.2.4 Windows Menu

The Windows menu in WD allows you to create a new window, edit or delete the current one, and to open already created windows.

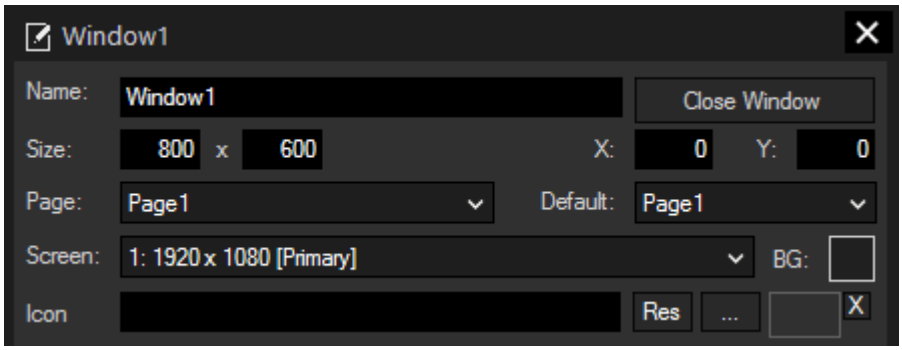
Multiple Windows enable you to split your interface, for example if you have two or more monitors you want to work on, or because you want to work parallel on several pages. Each window can be edited separately, each page can be assigned to each window, which allows a flexible usage of your interface. Please note that multiple windows still belong to the same instance of the application. Exiting the program in one window will close all others as well!



When you **create** a new window, a new page is also being created. A dialog asks you to name both of them.

When you **delete** a window, all contained pages will also be erased. In case you like to move them to another window first, go to Page > Edit Page [805].

Edit Window



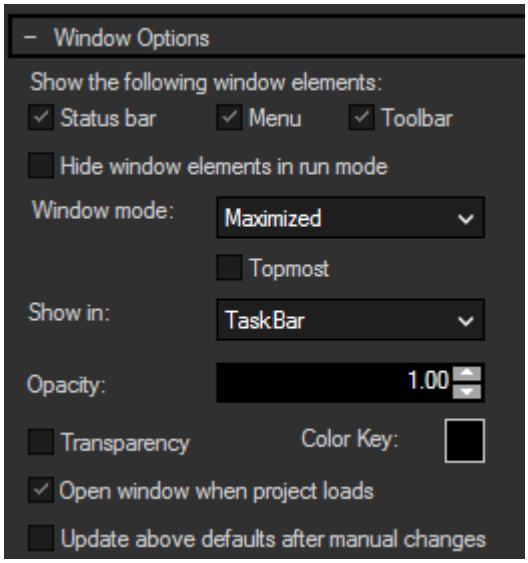
In the first section, there are some settings concerning the outer appearance of your current window, like **name**, **size and XY position**. You can either enter the absolute values, or adjust the size of your window by dragging the right bottom corner. This updates the values in the boxes immediately if the box **"Update above defaults after manual change"** further down in the Window Options is ticked.
The maximum window size depends on your maximum desktop size (e.g. If you use two FullHD displays with your PC, the maximum window size would be 2x1920px = 3840px).

There are also two drop-downs for managing the Window's pages. The first **Page** drop-down selects the currently displayed page for this Window, the **Default** drop-down sets the page that is initially displayed when the Window opens or when this Window is requested in an external web client.

The **Screen** drop-down lists all available displays attached to your PC. Choose on which one you like to see WD.

The **background color** is only visible if the page is smaller than the window itself.

If you like to give your window a special **icon** (it is displayed at the left corner of the window's title bar), you can do so either with one of the Widget Designer default icons or with your own image. Choose the directory with the **[...]** button or choose an image within the Image Resource Manager [1509] that is opened with the **[Res]** button.



Window Options:
Choose to show or hide the **status bar** (at the bottom), the **menu** bar (topmost bar) and the **toolbar**. If you check **"Hide window elements in run mode"**, those three bars are hidden as soon as the run mode is activated. Press either [F9] (edit / move mode) or [Ctrl + Shift] (create mode) to show them again.

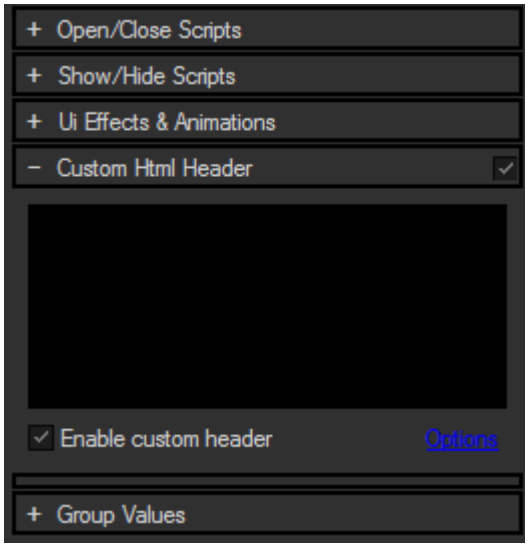
Additional to the settings from the View menu [799] (Kiosk Mode, Minimize, Maximize, Full Screen, Normal size), you can choose to display your window at the center of the monitor and to keep your Window always on top (tick "**Topmost**" for this option).

If your application is not supposed to be visible in the Taskbar, you can choose to hide the symbol in the System Tray (Taskbar Notification Area) with the "**Show in**" parameter.

You can adjust the **opacity** of the whole window including all bars. The smaller the value is the more transparent the window gets. This does not apply to any dialog, such as properties or drop downs.

The color key applies to anything inside your window that matches the chosen color. For example, if you set your color key to blue and add a Shape filled with the same color, the blue area will be completely removed from the window so that you can click on items lying somewhere behind this Widget Designer window. To use this feature, set the opacity to 1.00 and check the **"Transparancy"** option.

The **"Open window when project loads"** option is especially useful if there is more than one window in the project, and some of them should open automatically when loading the project while others should not. This box is ticked by default, so if you only want to open your main window on startup, uncheck the box for every further window. Please note that at least one window must be set to open when project loads.



Open / Close Scripts:
If you like to run a script (consisting of one or more commands) each time the window is opened or closed, add the commands to the text box. The topic Script Language [1511] explains commands and how to use them in more detail.
Tip: You may use a macro or a function [1897].

Ui Effects & Animations:
CSS based effects and animations can be applied to this widget. If you have three dimensional effects on some of your Widgets, you have to apply the effect "3D\Perspective" to your Window in order to activate the Z-axis. Please refer to the topic Effects & Animations [814] for more information.

Custom HTML Header:
Due to its nature, being set up in HTML5, it is possible to create your application's own design using this language, CSS3 and JavaScript. The Custom Header affects all objects within the respective Window including Pages. You can set additional style properties for all classes, may it be all Widgets of one type or one special Widget by referring to its modifier. If you open your interface in an external browser, you can find out the modifier name by using the browser's developer tools (most commonly opened with pressing [F12] or the entry within the browser settings) and analyzing the displayed source code.

Example 1:

```
<style>
.WdWindow{
position: relative;
width: 1200px;
margin: 0 auto;
}
</style>
```

This Script has the effect that all pages are always displayed in the middle of the Window (provided that the page has a width of 1200px), local GUI as well as external browser.

Example 2:

```
<style>
#WrapperWidgetXyPanel1{
opacity: 0.25;
}
</style>
```

This Script has the effect that only XYPanel1 is displayed with 0.25% opacity. This can be especially useful, when you have a graphic or an object, within which you would like to be able to read out a position upon a mouse-click. Simply place the graphic or object in the background, an XYPanel of the same size in front of it, and set the opacity to 0 with the example script.
The panel is invisible, but you can read out the clicked coordinates with the XYPanel input node [1092].

It is also possible to import and embed your own CSS and JS files. Simply click on the "Options" label and "**Import Files**". The files you import will be saved in the project's "Data" folder and the header will automatically be supplemented with a link to this file. If you have several files already stored, you can easily add the respective links by clicking "**Update header from files**".
The check box "**Enable custom header**" enables you to quickly switch between your own design and the unaffected Widget interface.

Group Values:
This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

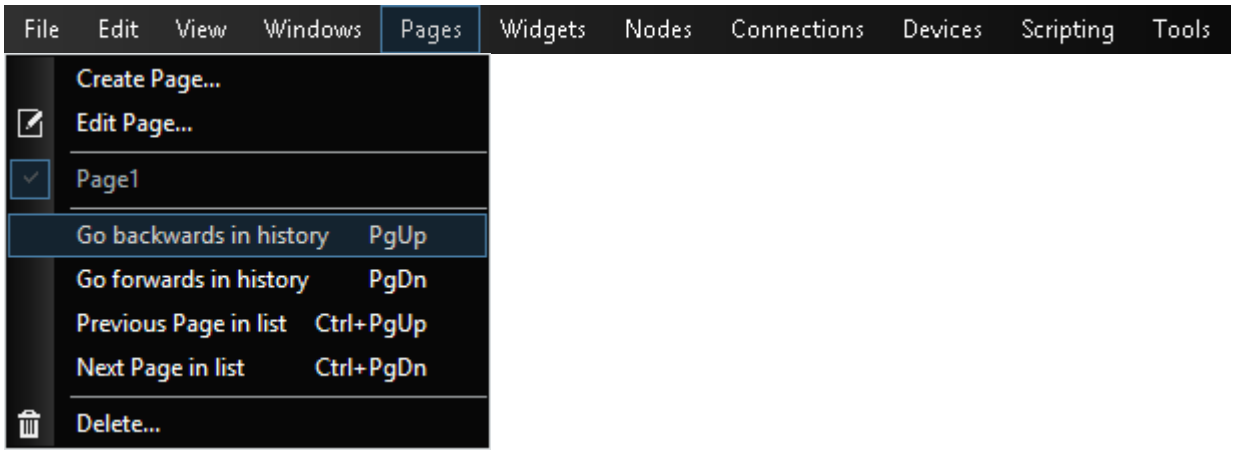
7.2.5 Pages Menu

The Pages menu in WD lists all pages you have already created in the current window [803] and eases the navigation through your custom user interface.

Widget Designer supports unlimited pages to create dynamic menu systems. Buttons and commands [1511] can be used to change from one page to another.

Specifics about Pages: When nodes are set up, different pages can help to keep an overview. If you like to deactivate (respectively mute) an output node when the page is not active, you can do so in the node's Item Properties. Upon Page enter all nodes of the entered page are automatically activated. When VNC Panels are used with pages, the page change will cause all invisible VNC panels to disconnect their VNC Connection for performance reasons.

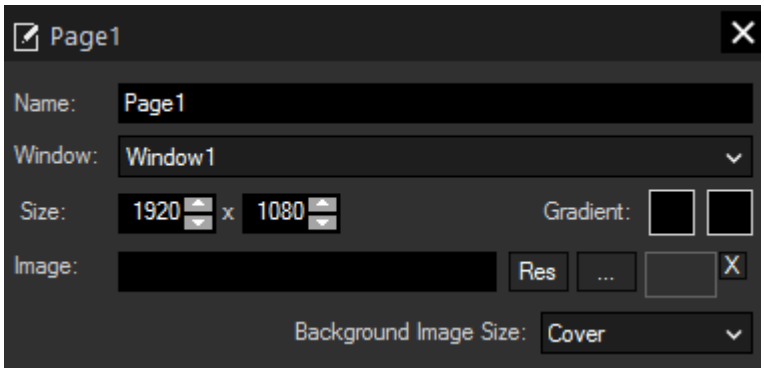
The command Browse Pages [799] in the View menu opens a page browser, where you will see all pages as thumbnails.



When you **create** a new page, an empty page is added to the window. The dialog with the page's properties opens where you can choose a name and all other properties as explained further down. To edit the current page, click on "Edit page...".

When you **delete** a page all contained items will also be erased. In case you like to move them first to another page, open their Item Properties.

Edit Page



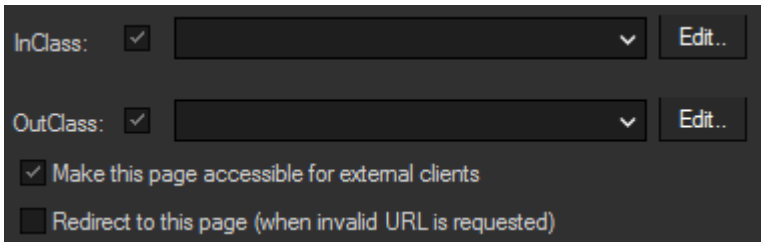
First you can define a **name** for the page. The name may contain letters, capital or lower-case, as well as numbers and underscore, the first character has to be a letter.

If you use multiple windows [803] for your project, you can assign your page to any **window** from the drop-down list.

Define the **size** of your page, widgets and nodes are not visible outside that area. If you reduce the size of an already used page and happen to lose some of the widgets, they can still be accessed with the widget explorer, they also still react to being dragged in the move mode, even if you can't see them. The maximum size of a page is 8192x8192px.

You may set up a background image. Choose the directory with the **[...]** button or choose an image within the Image Resource Manager [1509] that is opened with the **[Res]** button. Alternatively, a solid color or gradient can be set with the **color fields**.

"Background Image Size" offers different aspect modes for the background texture:
- Stretch: stretches the images so that it fits into the page size
- Cover: resizes the image so that it fits either horizontally or vertically, the other side will be cropped according to the page size
- Contain: resizes the image so that the larger side fits into the page size
- None: maintains the original picture size and positions it in the center of the page

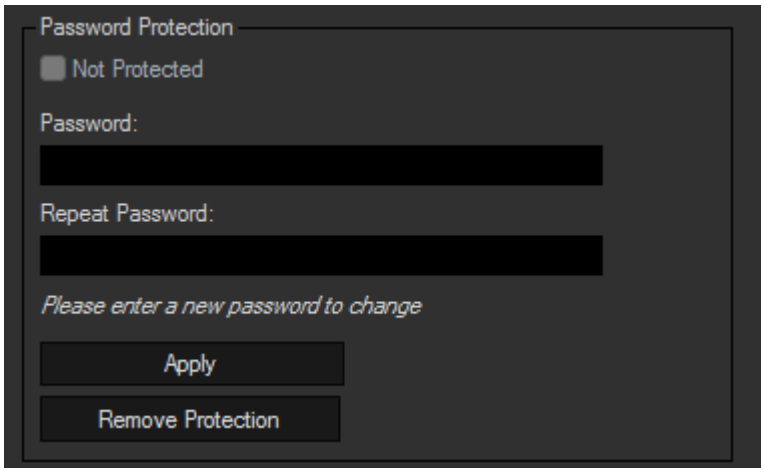


The **In/OutClass** enables you to set up transition effects for page changes. Choose one of the effects from the drop-down and edit them according to your needs. Those effects are often designed to work in pairs and may not produce the intended results if combined in other ways.

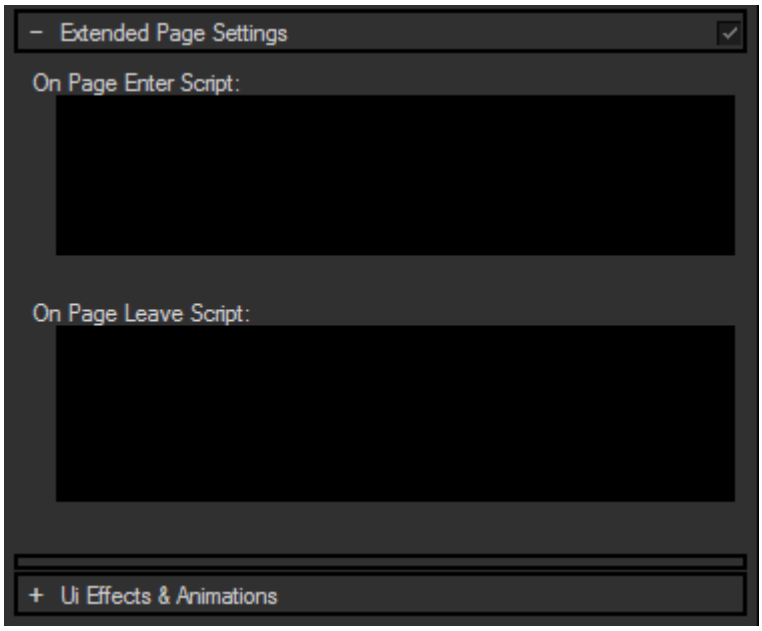
You can also import your own effects. CSS Import is covered in the chapter Effects & Animations (Web Styles) [814].

With the two additional check boxes, you can determine the behavior for the Web Server [1929]. If "**Make this page accessible...**" is not ticked, you cannot access it from an external client. This is especially useful for pages with nodes and configuration items, someone from outside is not supposed to see.

One page in the project can be set to an index page, check "**Redirect to this page...**" for the chosen page. Any web server client will then be redirected to this page first when accessing the project. checking this box will automatically uncheck the former index page.



Password
Set up a password that needs to be entered every time this page is called in Widget Designer itself or in the Web Browser. The on-screen keyboard will appear upon each page load process and will forward to the page only when the correct password was entered.
The checkbox Protected / Not Protected informs you about the current protection state.
To activate this feature, enter any character string in the **Password** and the **Repeat Password** field. The status message below shows whether both are matching or not.
Press "**Apply**" to set this password for the page, press "**Remove Protection**" for deleting the password setting.



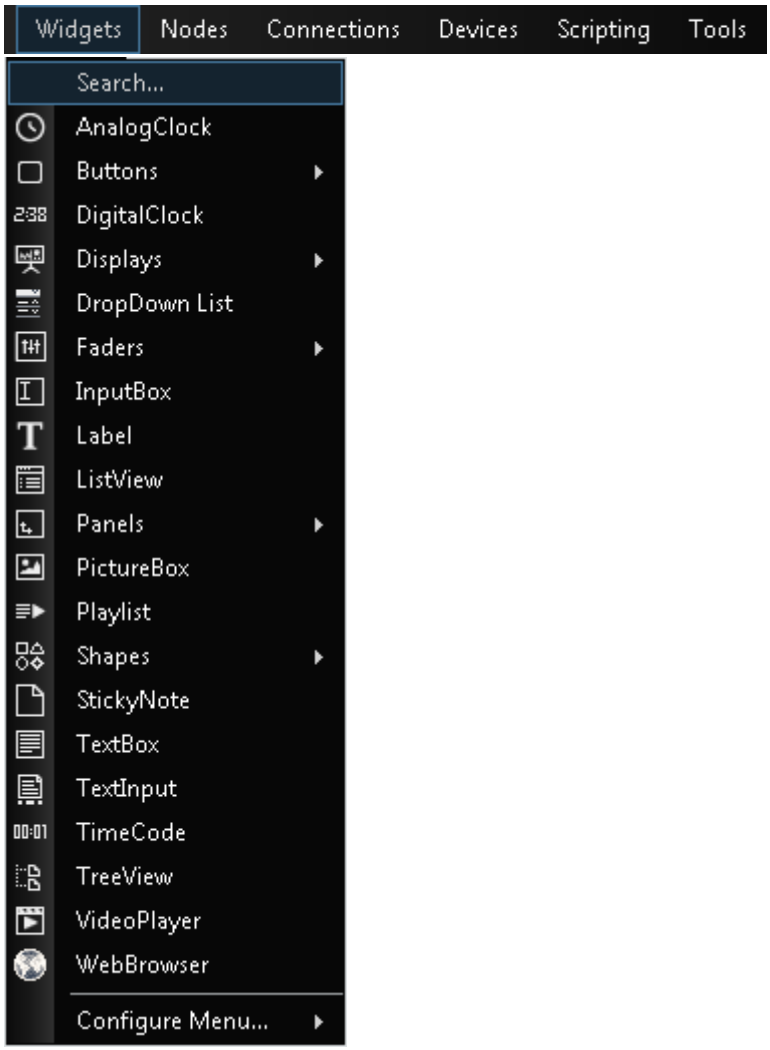
Extended Page Settings
Enter custom commands [1520] to the On-Page-Enter- or On-Page-Leave-Script to assign an additional action to every page change. The topic Script Language [1511] explains this in more detail. Tip: You may use a macro [1897].

Ui Effects & Animations
You can set up advanced render. Please see the chapter Effects & Animations (Web Styles) [814].

7.2.6 Widgets Menu

The Widgets menu lists all controls you may create in Widget Designer. Since Widget Designer version 6 all widgets are also supported by the Web Server [1929].

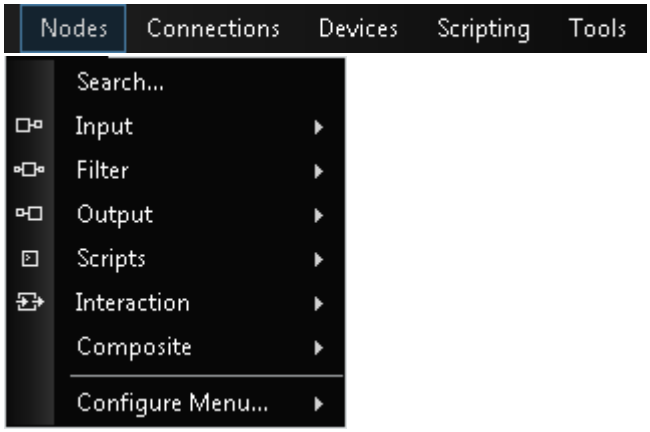
Once you have chosen an item, the mouse cursor will change to a crosshair icon. Wherever you left-click now, the chosen item is generated. More information about each control can be found in the Widgets chapter [818].



7.2.7 Nodes Menu

The Nodes menu lists all node categories including all nodes you may create in Widget Designer PRO and ULT, currently there are over 250 nodes available.

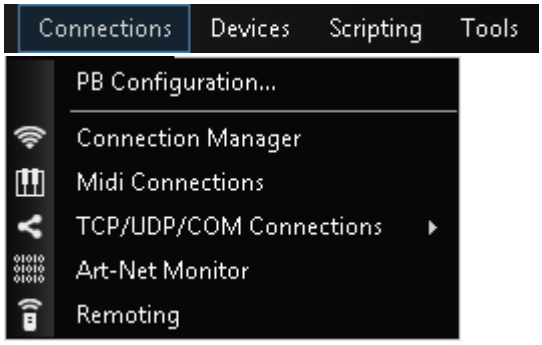
Once you have chosen a node, the mouse cursor will change to a crosshairs icon. Wherever you left-click now, the chosen node is generated. More information about each node and how to work with them can be found in the Nodes chapter [936].



7.2.8 Connections Menu

The Connections menu in WD lists all available connection possibilities you can set up in Widget Designer. Other than the listed input and output protocols, you can connect to devices [1277] and other tools [1483].

Once you have chosen an entry a dialog opens with more options.

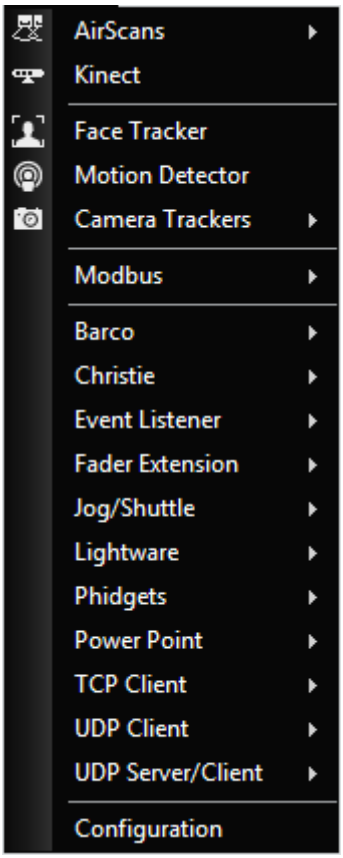


For a detailed description please see the following pages from the Connections chapter [1255]:
PB Network Configuration [1256]
Connection Manager [1258]
Midi Input [1485]
TCP/UDP/COM Connections [1272]
Art-Net Monitor [2051]
Remoting [1273]

7.2.9 Devices Menu

The Devices menu lists all (physical hardware) devices you can work with in Widget Designer. In case your device is not available, bear in mind that it is also possible to set up a connection [1255], e.g. a TCP connection. Additionally, WD supports several Tools [1483]. Note that the Devices are not supported in the Widget Designer Free Edition.

Once you have chosen an entry, a dialog opens with more options.



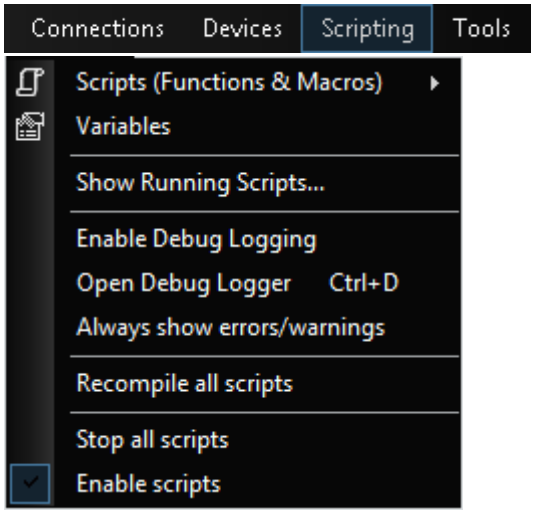
For a detailed description please see the following pages from the Devices chapter [1277]:

AirScan [1277]
Kinect [1283]
Face Tracker [1287] (not available in the Free Version)
Motion Detector [1289] (not available in the Free Version)
Camera Point Tracker [1291] (not available in the Free Version)
Phidgets [1383] (not available in the Free Version)
Modbus [1298] (not available in the Free Version)

7.2.10 Scripting Menu

The Scripting menu lists all helping tools and other actions regarding scripting in Widget Designer.

Once you have chosen an entry, either a dialog opens with more options or the command is executed. More information about the Script Language can be found in the Scripting chapter [1511].



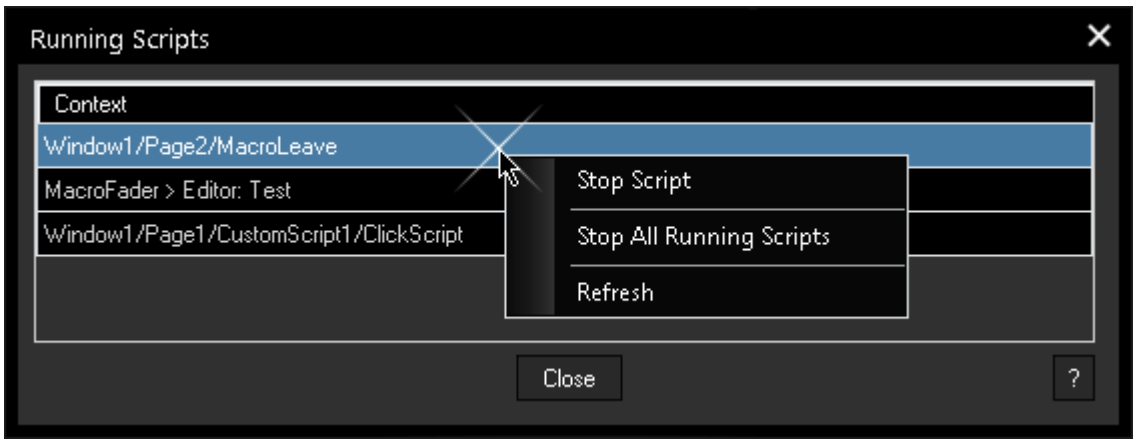
Scripts (Functions & Macros) [1897]
Here you can create Macros and Functions with "New Macro..." or "New Function...". To open the macro/function editor for editing existing ones, simply click on the respective macro//function name.
To delete one, press "Delete" at the respective menu entry and a dialog will open where you can tick all scripts you want to erase.

Variables [1900]
This opens the Variable List [1900] where you can create global variables or edit existing ones.

Show Running Scripts
Opens a dialog where you may see all currently running scripts. The Context shows where a script is executed from, e.g. from a Custom Script Button, a Script Output node, a Macro, or Page.

The dialog is meant to help finding out whether there is a script being executed and allows to stop it. Right-click on an entry to get access to the stop and refresh commands.

The commands `ScriptCancel` [1600] or `ScriptCancelAll` [1601] might also be of interest.

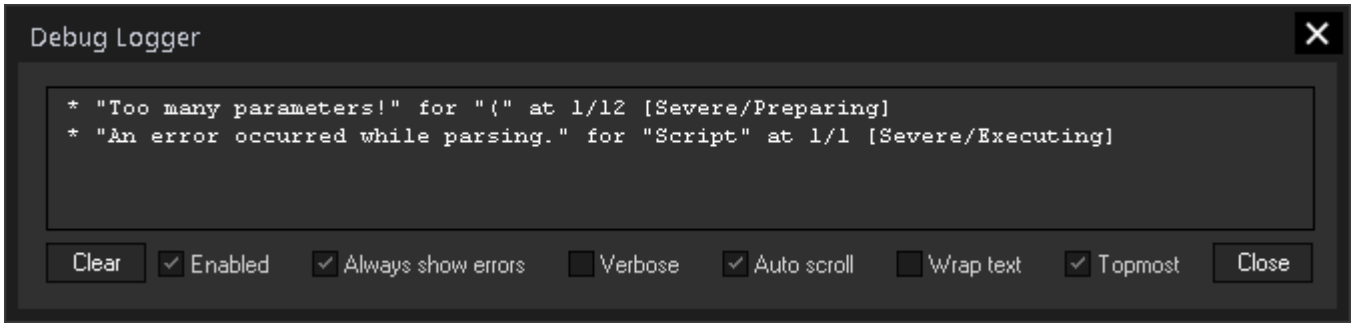


Enable Debug Logging
This toggles the debug logging on and off to show or hide possible error messages in the dialog Debug Logger.

Open Debug Logger
This opens the dialog Debug Logger and enables debug logging to display error messages. Its shortcut is [Ctrl + D]. It is recommended to turn the logging off during a show.

As soon as you are working with more complex scripts, it might me useful to generate your own debug messages with the command "DebugMessage(message text)" that will be displayed here, too.

This feature is currently not available for the WD Free Version.



This helps to find errors in your programmed scripts. The logger includes indications whether..:

- a command is wrongly spelled or an arguments are not enclosed in double quotation marks

  e.g. `WDLabelText(1,giraffe)` instead of `WDLabelText(1,"giraffe")`
  => * "Unknown token will be processed as a literal!" for "giraffe" at 1/15 [Warning/Executing]
- a command argument is not put in round brackets

  e.g.`WDLabelText,1,"giraffe"`
  => * "Parameters must be enclosed in '()'!" for "WDLabelText" at 0/0 [Severe/Parsing]
- a command has not enough arguments

  e.g. `WDLabelText(1)`
  * "A required parameter is missing." for "(" at 1/12 [Severe/Preparing]
- a command has too many arguments

  e.g. `WDLabelText(1,"giraffe","toomuch")`
  * "Too many parameters!" for "(" at 1/12 [Severe/Preparing]
- a command includes a reference to a nonexistent ID

  => Label '2' does not exist. Command 'WDLabelText' exception: Object reference not set to an instance of an object.

When you check the **Verbose** option, more details are listed including where the error happened, e.g. > Window1/Page1/CustomScript1/ClickScript

Alternatively you can copy the text into the "Find Entry" dialog 800 (from the View menu) or work with Find Item 800.

Always show errors/warnings
Per default, the Debug Logger does not open automatically to show whether a script includes an error. It opens only if you choose the "Test"command from the right-click menu. If you like to always see errors and warnings, enable this option.

Recompile all scripts
Scripts are normally compiled as soon as they are written.
However, if you have changed something, e.g. a variable type, it might be necessary to recompile all scripts to update the adjustment.

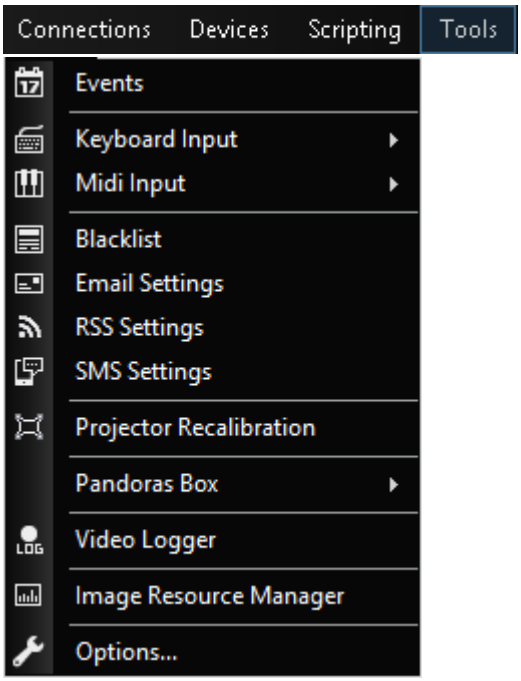
Stop all scripts
Stops all running scripts and aborts all fades, macros and functions.
This is especially useful if you have programmed very long scripts, maybe with some wait-commands, and have to abort them prematurely because something is not working correctly.

Enable scripts
Per default this option is enabled to allow scripts (including macros and functions) to be executed. If it is disabled, no script will be executed.

7.2.11 Tools Menu

The Tools menu lists all tools available in Widget Designer. In contrast to a physical hardware device [1277], a tool is rather a software tool. In addition, it is also possible to set up a connection [1255], e.g. a TCP connection.

Once you have chosen an entry a dialog opens with more options.



For a detailed description please see the following pages:

Events [1483]
Keyboard Input [1483]
Midi Input [1485]
Blacklist [1490] (not included in the WD Free Edition)
Email Settings [1490] (not included in the WD Free Edition)
RSS Settings [1493] (not included in the WD Free Edition)
SMS Settings [1494] (not included in the WD Free Edition)
Projector Recalibration [1496]

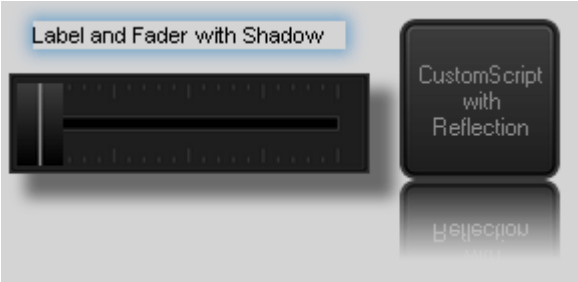
Pandoras Box >
Add Layers to Pandoras Box [1503]
Array Align Layers [1504]
Circle Align Layers [1505]
Cue Generator [1506]

Video Logger [1507]
Image Resource Manager [1509]

Options… [1510]

7.2.12 Effects & Animations (Web Styles)

Effects & Animations (Web Styles)



Since Widget Designer 6, the graphical user interface you build relies on modern web technologies. It is based on HTML-5 and features the possibility to style and manipulate complete pages or single widgets due to internally used CSS3 (Cascading Style Sheets) styles, a "language" created for design. You can utilize the full power of CSS in WD6, with or without any technical knowledge of CSS specifics.
Web designers can easily style customized control panels for a variety of different users and applications.

Applying styles

Most widgets (and pages and windows) can be modified using web styles. This includes layout, design, filters, images, animations, basically anything you can do with CSS.
In the widgets' properties dialog, these styles can be found in the panel "Effects & Animations". There you can add and apply several styles that you can choose from a large selection of out-of-the-box styles, but you can also add your own styles, of course.

After adding a new (empty) item to the list with the **Add button**, you can determine the style (effect or animation) for that item by choosing from a drop-down list. The check box in front of the line enables and disables the style. With the **Edit button** you can change its settings. The available effects and animations are stored in the following structure:
- Effects: a collection of effects that can be applied to a widget, page or window. Commonly split further into groups like 2D, 3D and Design.
- Pages: a folder for page transitions, grouped by "Enter" and "Leave" transitions.

Depending on the element you add a style to (window, page or widget), you will only be able to choose styles that are applicable for that specific context in the property dialogs. Within the folders above, you can organize the styles as you wish (and it is highly suggested that you keep your own custom styles in separate folders).

You can add as many styles as you like, but be aware that not all combinations make sense, i.e. the settings can contradict or influence each other:
- Page enter- and leave-transitions are often designed to work in pairs and may not produce the intended results, if combined in other ways.
- Both transitions will start at the same time and the shorter transition determines the total length of both transition effects.
That is why page transitions will only be animated, if both pages have defined and activated transitions.
- 2D effects are often intended to be combined with "Transition speed".
- 3D effects often assume that the parent element (page and/or window) has defined a "Perspective" style.
- Some effects may not be able to override the styles defined for an element in other ways, like styles that are usually set using the item's properties (e.g. label's fore color). Sometimes reloading the page ([F5]) helps, or extending the rendered selector in the templates so that it is more specific.

Note also that, except for page transitions, the **styles will only be applied in run mode** [F8]!

Editing parameters

From within the properties dialog, you can edit the parameter values used for each style. Editing a style (in an item's properties dialog) will only effect the respective item. All other items using the same style will remain unaffected. In the application, this is called editing "local overrides". If a value was changed to be different than the default value (defined in the style's template), will be added behind the value's label for your information.

Style templates

Each style is defined by a template file that is stored in one of the folders described below. These templates are rendered into CSS files, they can contain any valid CSS code. Plus, they may contain special placeholders for

inserting parameters that can be changed from within the application, using local or global overrides. These placeholders are detected and used to a) create a dynamic dialog for editing the values and b) render different values for the different contexts in which a template is used.

Example for a CSS template (effect scale):

```
<<Meta|Description=Resize the element. Requires TransitionSettings style to be added and activated.|Elements=Widget>>
.RunMode .<<Selector>>{
   transform: scale(<<XFactor=2>>,<<YFactor=2>>); }
```

This description is depicted below the chosen effect

Placeholder for the specific widget

Placeholders for local overrides with default values

Creating new templates, of course, requires a certain amount of knowledge about CSS. But, since it is an open web technology, there are sheer endless amounts of examples and tutorials that make creating impressive effects and animations feasible even for complete CSS-newbies.

"w3 schools" for example offers tutorials for starters as well as additional information on expressions for advanced users: http://www.w3schools.com/css/

Please note: The commands that can edit the CSS styles for widgets, like e.g. WDCustomScriptCssStyleEdit, are not meant to do so continuously. Every time this commands changes values, a new CSS file has to be generated from the template. If you want to build complex animations, it is rather recommended to write your own CSS template than successively alternating the values of a single effect.

Managing styles

All available styles are managed in the "Web Style Explorer" (Menu > Edit > Web Styles...). The explorer shows a tree view of all styles using the structure described above (Effects, Pages/Enter, Pages/Leave). All styles are defined by template files that are saved in one of the following locations:
- System (out-of-the-box) styles are stored with the application.
- Project styles are stored in the project's folder in "Data\Styles".
- Profile styles are stored in the user's profile's resource folder in "Styles."

Widget Designer expects the above folder structure to be in one or all of the locations above. Identical paths and names will be handled as only one style and the location from which the template is loaded, is determined by the following order: first check for Project files, then Profile files, then System files.

From within the Web Style Explorer, you can copy styles between the locations:
- Copy them to the project, if it is to be shared.
- Copy them to the profile, if the profile data is used on several computers, or to use different styles for different users.
- Using the System styles (without copying them) has the advantage that they can be extended/enhanced by future updates. (Hence they are not intended to be modified.)

You can also edit the parameter values of each style. These are "global overrides" that will effect all other elements that use the same style - unless they have their own "local overrides".

7.2.13 Widget Designer Keyboard Shortcuts

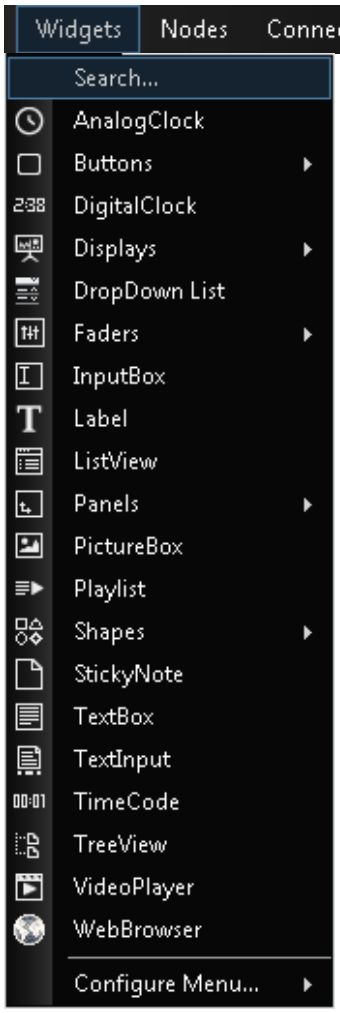
File	Ctrl + N	Create new project
	Ctrl + O	Open project
	Ctrl + S	Save
	Ctrl + Shift + S	Save as
	Alt + F4	Exit Application
	F1	Help
User Interface	F8	Run Mode
	F9	Edit / Move Mode
	Ctrl + Shift	Create last Widget
	F5	Reload
	Ctrl + ⊞ / ⊟	Zoom in / out
	Ctrl + 0	Reset Zoom
	F11	Toggle Kiosk Mode
	Ctrl + 1	Full Screen
	Ctrl + 2	Maximize Window
	Ctrl + 3	Set Window to normal Size
	Ctrl + 4	Minimize Window
	Ctrl + F4	Close Window
	Alt + G	Show / Hide Grid
	Alt + S	Toggle Snap to Grid
	Alt + N	Create Node
	Ctrl + W	Open Widget Explorer
	Ctrl + F	Find Entry
	Ctrl + Shift + F	Find Item
	Ctrl + Alt + N	Show / Hide Sticky Notes
	Pause	Disable / Enable Interface
	Page Up / Down	Go backward / forward in history (Pages)
	Ctrl + Page Up / Down	Previous / Next Page in list
Scripting	Ctrl + T	Test - Executes the whole script inside this field
	Ctrl + Shift + T	Test selected lines - Executes all highlighted lines or the line with cursor
	Ctrl + D	Open Debug Logger
	Ctrl + F	Open Search Dialog to find expression
	F3	Find Next - Highlights next search-expression
	Shift + F3	Find Previous - Highlights previous search-expression
Selection	Ctrl + A	Select all
	Ctrl + Mouse	Multi-select widgets and nodes
	Esc	Clear selection
	Ctrl + C	Copy
	Ctrl + V	Paste (if in edit / move mode)
	Arrow key ↑ → ↓ ←	Move selected items
	Ctrl + Arrow key	Snap selected items
	Ctrl + Home	Bring selected items to front in the Z-order
	Ctrl + End	Send selected items to back in the Z-order
	Alt + P	Open Item Properties of selected item
	Del	Delete selected items

Align Selection	Alt + Shift + L	Align selection to the left border
	Alt + Shift + R	Align selection to the right border
	Alt + Shift + T	Align selection to the top border
	Alt + Shift + B	Align selection to the bottom border
	Alt + Shift + H	Align selection to its horizontal center
	Alt + Shift + V	Align selection to its vertical center
	Ctrl + Shift + H	Align each item of the selection to the horizontal center of the UI
	Ctrl + Shift + V	Align each item of the selection to the vertical center of the UI
	Ctrl + Alt + H	Align the selection as a Group to the horizontal center of the UI
	Ctrl + Alt + V	Align the selection as a Group to the vertical center of the UI
	Ctrl + Alt + Shift + H	Spread selection horizontally
	Ctrl + Alt + Shift + H	Spread selection vertically

7.3 Widgets

The Widgets menu lists all controls you may create in Widget Designer.

This chapter includes general information about creating and editing widgets. The following chapters explain each Widget in detail. The Widgets marked with (*) are not included in the Widget Designer Free Edition.



"Search" helps you navigate to the correct (sub) menu.

Analog Clock [819]
Buttons [821]
Digital Clock [850]
Displays [851] (*)
Drop Down List [868]
Fader Controls [874]
Input Box [886]
Label [888]
List View [890] (*)
Panels [892]
Picture Box [911]
Playlist [913] (*)
Shapes [919]
Sticky Note [921]
Textbox [923]
Text Input [926]
Timecode [928]
Tree View [929]
Video Player [932] (*)
Web Browser [934] (*)

Here you can change the order of the widgets in the menu.

To create a control
a) open the Widgets menu from the main menu bar...
b) click on the widget symbols in the toolbar...
c) right-click anywhere in the empty main background and open the Widgets menu there...

...and choose the desired widget. The mouse cursor will change to a crosshairs icon, telling you that you are in the operating mode [789] called "create mode". Wherever you left-click now, the chosen item is generated. When you have as many copies from the item as you need, enter the edit / move mode (e.g. with [F9]) to edit, e.g. move or resize them or enter the run mode [F8] in order to use their function.

If you want to set up the properties of the newly built control, right-click on it and choose the menu entry "Item Properties" or press [Alt + P]. The widget's property dialog opens up. It contains all properties influencing the widget's behavior and look, as well as its ID which is important when you want to use it together with a node [936] or command [1511]. As you will need them many times, the Control IDs are displayed when you are in the edit / move mode [F9].
Use the keyboard shortcuts [Ctrl + C] and [Ctrl + V] in order to copy and paste items, use the [Delete] key if you want to erase them. These commands can be found in the edit menu as well.
Additionally, all properties can be accessed via the Widget Explorer [802], too.

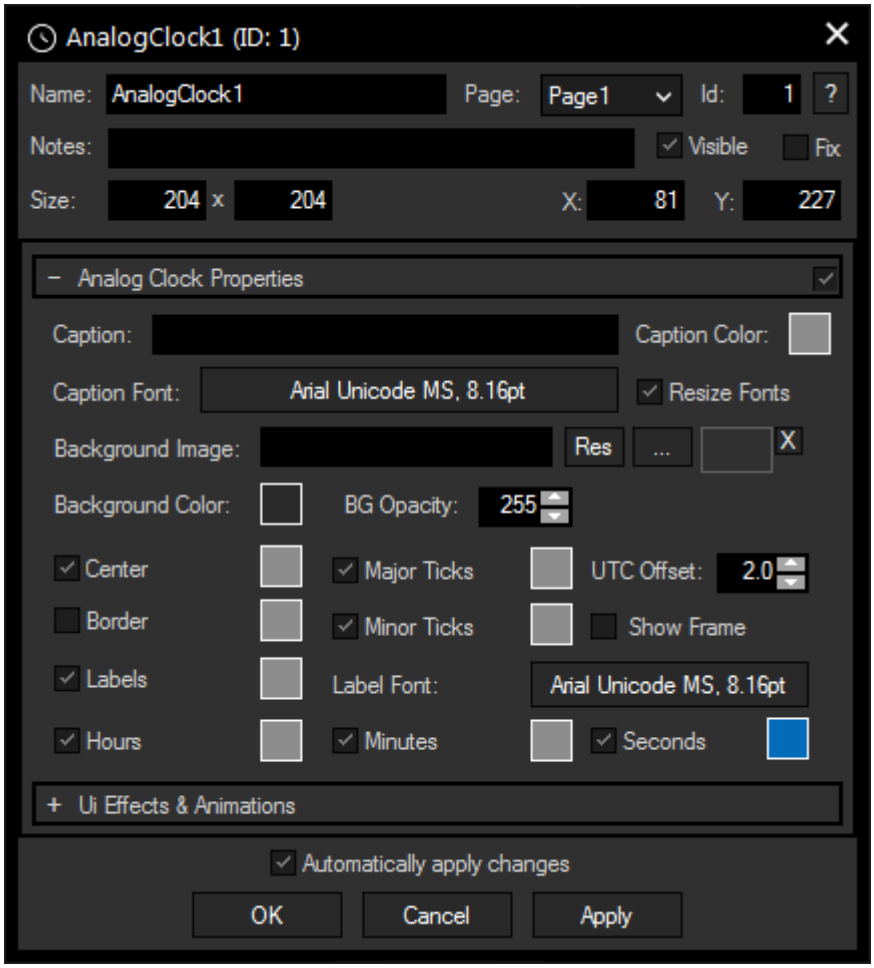
The image shown above gives an overview how many widgets Widget Designer supports. Since Widget Designer version 6 all widgets are also supported by the Web Server [1929].

7.3.1 Analog Clock

Use this control to display the current computer time via an AnalogClock.

To create an AnalogClock widget choose "Widgets > AnalogClock". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the AnalogClock properties simply right-click it and choose the first menu entry "AnalogClock Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Analog Clock property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The AnalogClock's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the AnalogClock will be displayed on every page.

Size:
Enter a pixel size for the AnalogClock's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Caption:
Here, you can enter a label for the clock, color and font can be selected with the respective buttons.

Resize Fonts:
This features adjusts the fonts of caption and numbers when changing the size of the widget.

Background:
An image can be loaded as a background. You can either browse your system for a picture or choose one out of the Resource Manager [1509].
It is also possible to set a background color and transparency.

You may design your Analog Clock using different colors for **Center, Border, Labels, Hours, Major Ticks, Minor Ticks, Minutes** and **Seconds**.
Tick the check boxes to hide / display the mentioned units. A decorative **frame** can be displayed as well.

UTC Offset:
Enter here the time offset the AnalogClock should have from your computer's time.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.2 Buttons

Choose between various kinds of buttons:

Custom Script Buttons: 822
The Custom Script Button control is the most flexible button control. With this control you can set up any button style and any click-able user interaction. Over 1000 commands are available to remote control all important Pandoras Box features as well as Widget Designer interface elements and much more.

Script Timer 826
The Script Timer control lets you execute commands after a certain amount of time. This can be done once or continuously.

Media Control 828
The Media Panel control is designed to create thumbnail button tables based on the thumbnail downloads of the CITP feature of Pandoras Box.

Cue Control 831
The Cue Control is designed to easily get access over the sequence control of Pandoras Box.

Image Loader 833
The Image Loader control is designed to easily exchange an image file that is added to the Pandoras Box project with any other image file.

Video Snapshot 837
The Video Snapshot control is designed to easily exchange an image file within a Pandoras Box project by a Video Snapshot from your attached Video Input.

Art-Net Snapshot 839
The Art-Net Snapshot Button is designed to record a full Art-Net DMX universe state, with the ability to record from Universe A and output to Universe B.

Art-Net Recorder 841
The Art-Net Recorder Button is designed to record a full Art-Net DMX universe over a certain time, with the ability to record from Universe A and output to Universe B.

Scroller Horizontal / Vertical 844
The Scroller is designed to change the X or Y position of assigned media files in your Pandoras Box project, so that you can scroll through a certain amount of pictures.

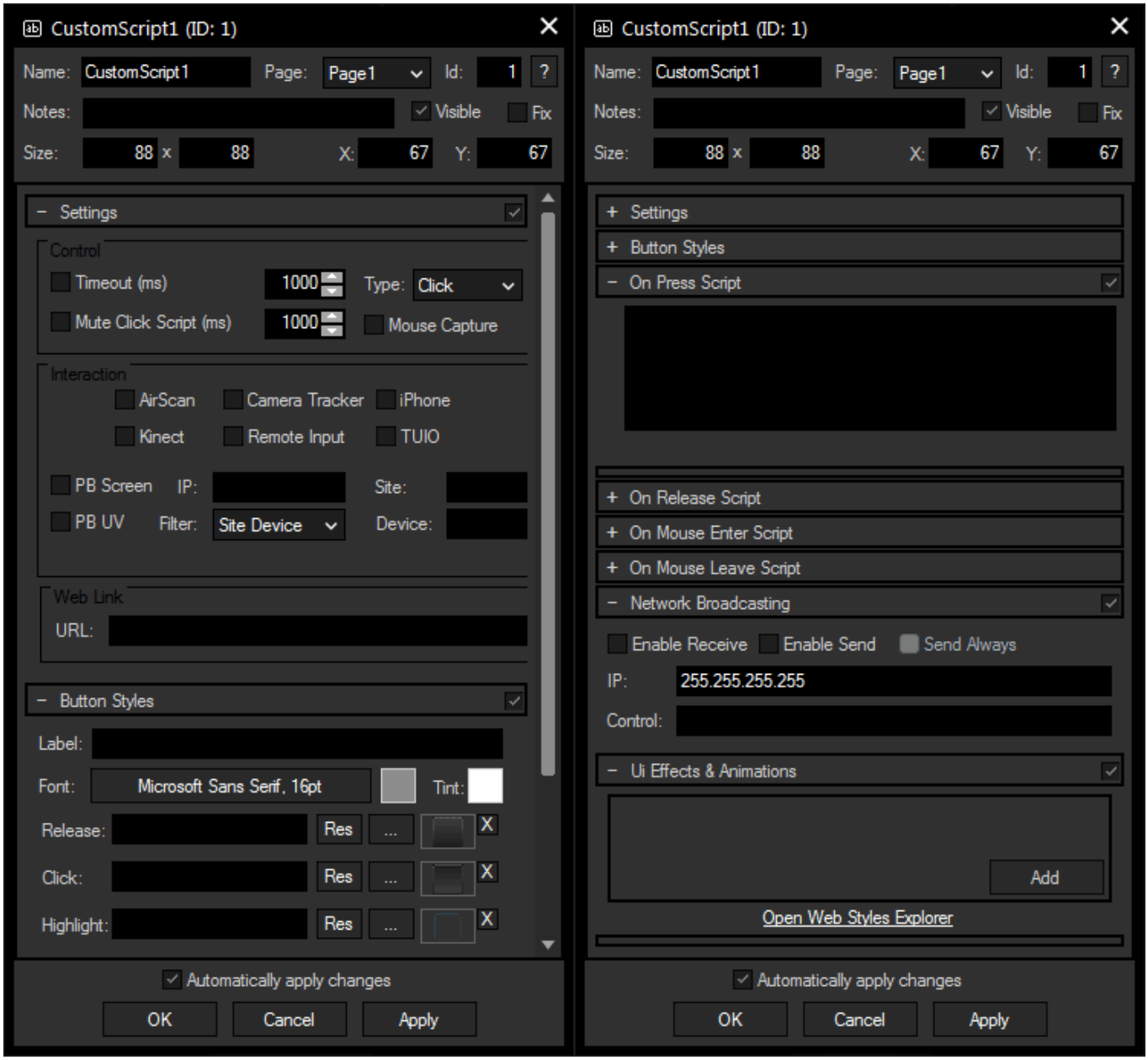
FlipSwitch Horizontal / Vertical 848
The Flipswitch is designed to execute commands only when the control reaches the specified unlock position.

7.3.2.1 Custom Script Button

The CustomScript button allows you to execute one or more commands whenever the button is clicked or activated in a different way. You can set up any button style and any click-able user interaction. Over 1500 commands are available to remote control all important Pandoras Box features as well as Widget Designer interface elements and much more.

To create a CustomScript button choose "Widgets > Buttons > Custom Script". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the CustomScript button properties simply right-click it and choose the first menu entry "CustomScript Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it.
The CustomScript button property dialog opens up.



The property dialog is divided into eight sections additional to the general widget properties on the top: Settings, Button Styles, four different kinds of Scripts, Network Broadcasting and Ui Effects & Animations.

General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The CustomScript button's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the CustomScript button will be displayed on every page.

Size:
Enter a pixel size for the CustomScript button's size.

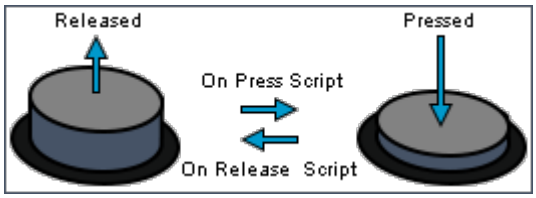
X and Y:
Enter the location of the widget (upper left corner) in pixels

Settings

Timeout (ms):
The timeout setting can be used to call the click script after a given timeout. The timeout is always restarted upon mouse enter on the control.
This feature is particularly useful for AirScan or touch applications where the user can only position the mouse but not cause a click with the input device.

Mute Click Script:
Set a specific time in milliseconds for a CustomScript click script to be muted (not executed).
This protects a CustomScript button from accidental double clicks.

Type:
You may set the button type to be Click or Toggle. The toggle state of the CS button will be kept when the project is saved and loaded.



After creating a button it is always in the "Released" mode. A button with the Click type has no other status. If you click on it, each time, the "On Press" script is executed. When releasing the mouse click, the button is again into the "Released" mode and the Release Script executed (if defined).

The toggle button has two modes. First it is "Released". When clicking on it, the "On Press" script is executed and the button changes to the "Pressed" mode. The next time it is clicked, the "On Release" script is executed and the button changes to the "Released" mode again.

The modes can be indicated by an image (see below). There are commands [1691] that press a button, with or without executing the associated script and with or without changing the mode.

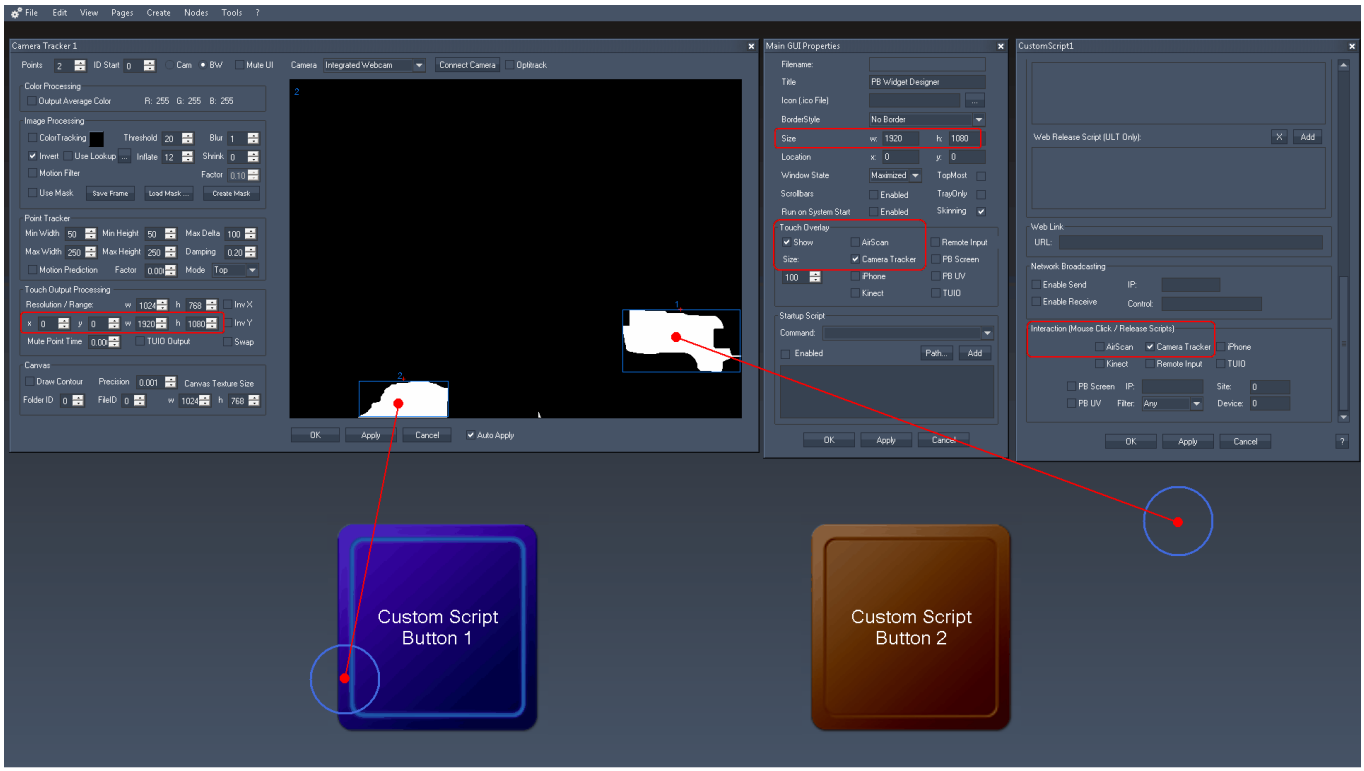
Mouse Capture:
CustomScript buttons allow to capture a mouse left click even when WD GUI is in the background, as well as

Mouse Enter and Mouse Leave events. This is useful when WD is used on the same machine as the Pandoras Box Client (e.g. a Server). Please note that the WD interface needs to be minimized for this feature to work!

Interaction (Mouse Click / Release Scripts):
This section is of interest if you like to "click" a CustomScript button without using a mouse device, including node solutions addressing a mouse. It is possible to use the input device directly to execute the "On Press / Release" or "On Mouse Enter / Leave" script. If the device generates more than one input event you can use all of them even if running a Win XP system that is normally restricted to one input event. The following input devices are supported: AirScan [1277], Camera Tracker [1291], iPhone [1275], Kinect [1283], Remote Input [1273], TUIO [1258].
In addition you can also receive inputs from a device that is physically connected and set up on the Pandoras Box Master or even Client system. The Client sends the mouse / touch / etc. inputs to the Master system which sends it through the Widget Designer Device [696] to the Widget Designer interface.

Set up the supported device connection as usual to generate input events.

Last, open the Item Properties of the Custom Script Button that should receive the input data. Enter the script as usual and in the section "Interaction" enable then the check box for the input device, e.g Camera Tracker as shown in the below screenshot. If you like to click the button with data from a PB Master or Client, decide whether the screen (i.e. output resolution) or UV data of a layer is of interest. If you like to filter data from a specific site / layer only, enter the Master's IP address, then the Site (and Device ID) and the filter type. For more info about UV data see the chapter about Layer Picking [252].



Web Link URL:
The Web Link section is of special interest when working with the Web Server [1929] feature, i.e. clicking the CustomScript button in a web browser. Enter a URL e.g. "https://www.christiepandorasbox.com" that your browser should call when clicking the button.
But it is also possible to achieve a quick page change with the URL, simply enter a hash tag "#" and the page name. E.g.: #Page2

Button Styles

Label:
Enter a label that will be displayed on the button. The text's color can be set via the color field and its font via the according button.

Tint:
CustomScript buttons can be colored by selecting a tint color, this applies only to the default button images.

Three images can be assigned to the different states of the button (the modes are explained above).
"Release" refers to no click or not toggled.
"Click" is on press or toggled.
"Highlight" can be used to draw a PNG with alpha over either the click or release image.

Click on the "..." button to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

For a long time, Custom Script Buttons [822] needed to be rectangular but now they can have any shape you wish. Simply create an according image with transparency and assign it to the Release, Click or Highlight state of the CS button. When you click on a fully transparent area in Run mode, the button does not execute any script. In case you placed another Widget underneath that area, this one will receive that click (however certain drag'n'drop actions to this Widget are not supported).

It is recommended to assign images with the same size to the three states and to reduce the transparent area to the borders of visible pixels.

If you like to work with transparent buttons, it might be faster or more convenient to use Labels [888] (with an on-click-script) instead.

Script

In the Script section you may enter commands to be executed. You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

The scripts for Press, Release, Mouse Enter and Mouse Leave are executed when the Widget Designer button is being used, on the local interface as well as on any web client.
You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts
If you are interested in the Web Server feature and some small examples, please read the topics Web Server [1929] and Object and Member Notation [1904] .

See here a list of all commands [1520].

Network Broadcasting

The NBS (Network Broadcast Service) allows transmitting and updating Faders, Labels and CustomScript buttons across multiple WD Designers instances on the network. To activate this service, please refer to the Remoting Menu [1273]!

Enable Send:
To send the CustomScript buttons state (clicked or released) as broadcast into the network, you only have to check "Enable Send".

Enable Receive:
To receive values from other CustomScript buttons, please check "Enable Receive".

Now you have to specify which item should update your CustomScript button:
Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the CustomScript button through another CustomScript button on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the CustomScript button you want to take the status from. This could be e.g. "CustomScript1" or "CustomScript2".

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

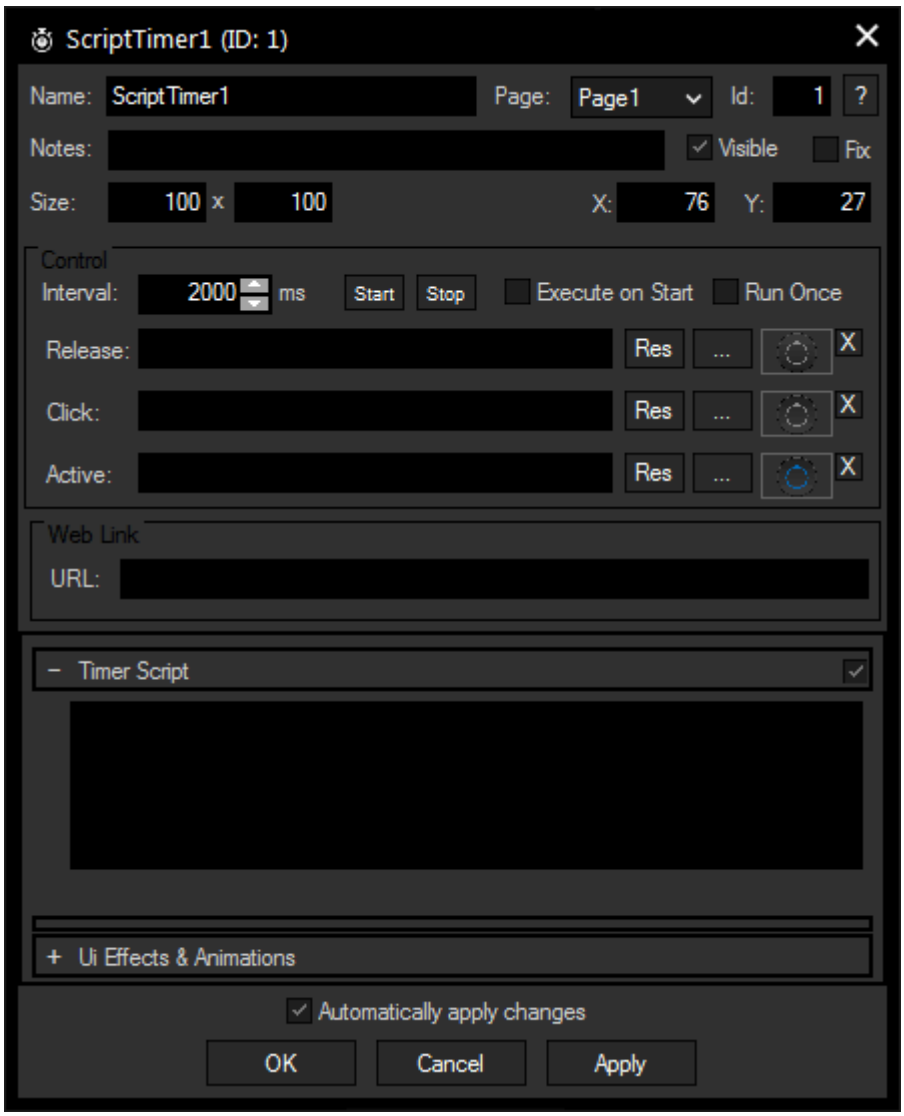
7.3.2.2 Script Timer

The Script Timer control lets you execute a script after a certain amount of time. This can be done once or continuously.

By the first click on the ScriptTimer it will be started. A second click on it stops it again.

To create a Script Timer widget choose "Widgets > Buttons > Script Timer". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Script Timer properties simply right-click it and choose the first menu entry "ScriptTimer Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The ScriptTimer property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The ScriptTimer's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the ScriptTimer will be displayed on every page.

Size:
Enter a pixel size for the ScriptTimer's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Interval:
Enter here the Interval time in ms, after which the ScriptTimer executes the script once or continuously.

Start/Stop:
Starts and stops the ScriptTimer.

Execute on Start:
If this box is checked, the script will be executed immediately when the ScriptTimer is started. Otherwise, the first script will be executed in the second interval.

Run Once:
When "Run Once" is enabled, the Script Timer will execute the script once after the Interval Time is expired and then stops again.
When "Run Once" is disabled (by default) the Script Timer will execute the script continuously in the Interval you entered in the text field to the right (in ms).

Three images can be assigned to the different states of the ScriptTimer:
"**Release**" refers to no click or not toggled.
"**Click**" is on press or toggled.
"**Active**" is the image appearing shortly at the beginning of each interval.

Click on the image place holder itself to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

Web Link URL:
The Web Link section is of special interest when working with the Web Server [1929] feature, i.e. clicking the ScriptTimer button in a web browser. Enter a URL e.g. "https://www.christiepandorasbox.com" that your browser should call when clicking the button.
But it is also possible to achieve a quick page change with the URL, simply enter a hash tag "#" and the page name. E.g.: #Page2.

Script

In the Script section you may enter commands to be executed. You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

The scripts are executed when the Widget Designer button is being used, on the local interface as well as on any web client.

You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts
If you are interested in the Web Server feature and some small examples, please read the topics Web Server [1929] and Object and Member Notation [1904].

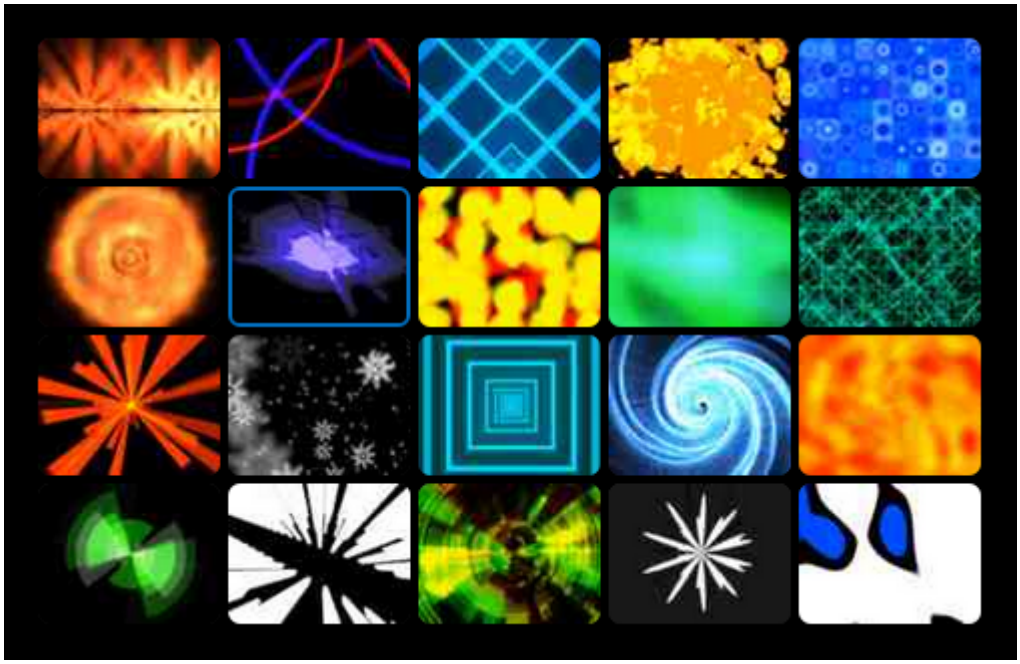
See here a list of all commands [1520].

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.2.3 Media Control Panel

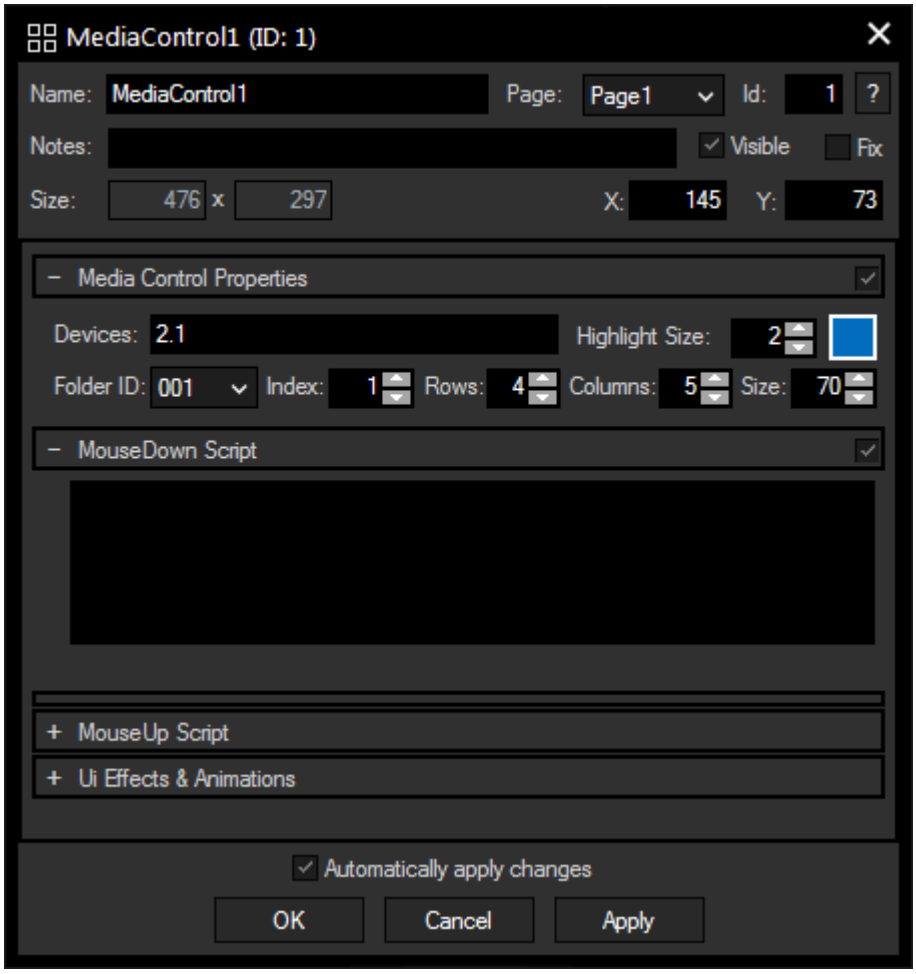
The MediaControl is designed to create thumbnail button tables based on the thumbnail downloads of the CITP feature of Pandoras Box.



To create a MediaControl widget choose "Widgets > Buttons > MediaControl". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window, you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

Please note that CITP needs to be enabled in Pandoras Box and the respective resources need to be assigned a file and folder ID. To download or update thumbnails, open the Pandoras Box Network Configuration [1256] dialog, connect to the application and press "Download Thumbnails".

To edit the MediaControl properties simply right-click it and choose the first menu entry "MediaControl Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The MediaControl property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The MediaControl's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the MediaControl will be displayed on every page.

Size:
Enter a pixel size for the MediaControl's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Media Control Settings

The Control Settings section lets you design and set up the MediaControl button

Devices:
To assign the media files to one or multiple devices, enter the site and device ID into the device text field.

Example:
To assign the media files to Device1 of Server1 and Device1 of Server2 enter "1.1 2.1"

Highlight Size and Color:
If a thumbnail button is clicked, it is highlighted with a colored line. Here you can define the line width in pixels and the color.

Folder ID, Index, Rows and Columns:
The MediaControl is automatically built based on the selected FolderID, the start index of the files as well as the rows and columns count.

Size:
The size sets the width of every thumbnail button.

Script

In the Script section you may enter commands to be executed. You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

The scripts are executed when the Widget Designer button is being used, on the local interface as well as on any web client.
You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts
If you are interested in the Web Server feature and some small examples, please read the topics Web Server [1929] and Object and Member Notation [1904].

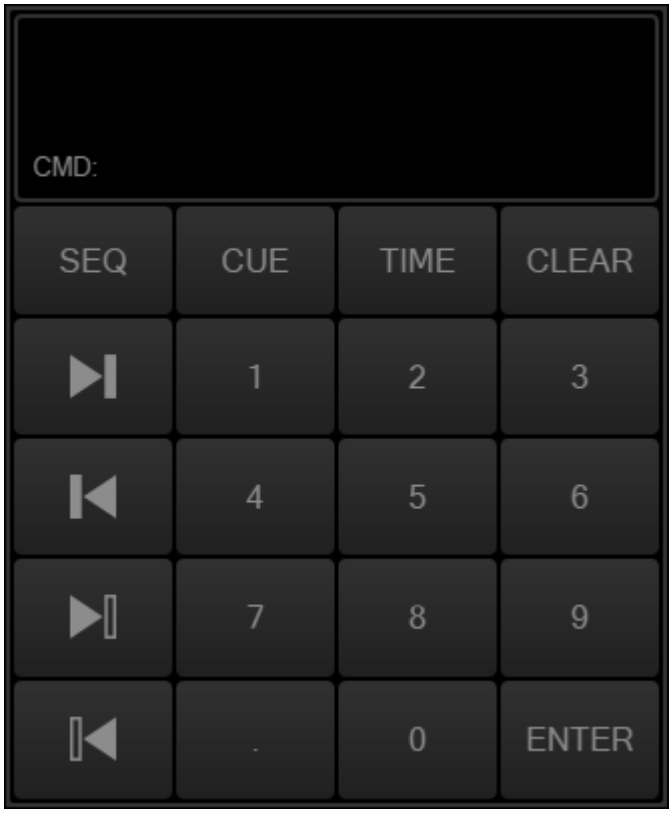
See here a list of all commands [1520].

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.2.4 Cue Control

The CueControl is designed to easily get access over the sequence control of Pandoras Box.



To create a CueControl widget choose "Widgets > Buttons > CueControl". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

Widget Control

Sequence:
By default the CueControl is assigned to Sequence 1 (displayed down right in the command window).
To change the sequence you want to control, please press
[SEQ+SeqID+ENTER], e.g. SEQ+2 will control Seq. 2 as long as you do not change the sequence.

Cues:
To jump to the next / last Cue or frame in the sequence, use the arrow buttons on the left.

To jump to a dedicated Cue, please press
[CUE+CueID+ENTER], e.g. [CUE+5+ENTER] and the nowpointer jumps to Cue 5.
As long as you do not press another command like SEQ, TIME or CLEAR entering any number + ENTER will be interpreted as "jump to Cue xx".

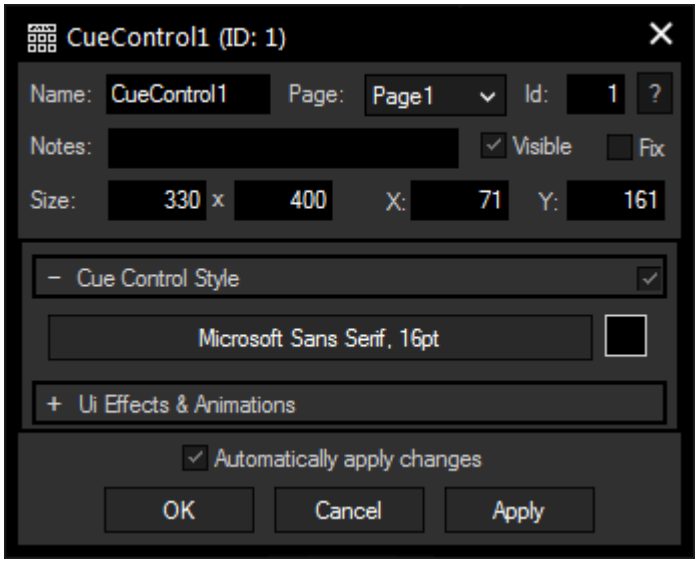
Time:
To jump to a specific timecode, please press
[TIME+hh:mm:ss:ff+ENTER], e.g. [TIME+1041513+ENTER] lets the nowpointer jumps to the timecode 1:04:15:13.

A more simple way to enter the timecode is to just enter the last numbers different from 0,
e.g. [TIME+213+ENTER] lets the nowpointer jumps to the timecode 0:00:02:13.

As long as you do not press another command like SEQ, CUE or CLEAR entering any number + ENTER will be interpreted as "jump to Timecode xx".

Clear:
Press [Clear] to clear the current entered values.

To edit the CueControl properties simply right-click it and choose the first menu entry "CueControl Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The CueControl property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The CueControl's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the CueControl will be displayed on every page.

Size:
Enter a pixel size for the CueControl's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Font and Color:
Changes the font and its color of the text displayed on the buttons and the command window.

Ui Effects & Animations

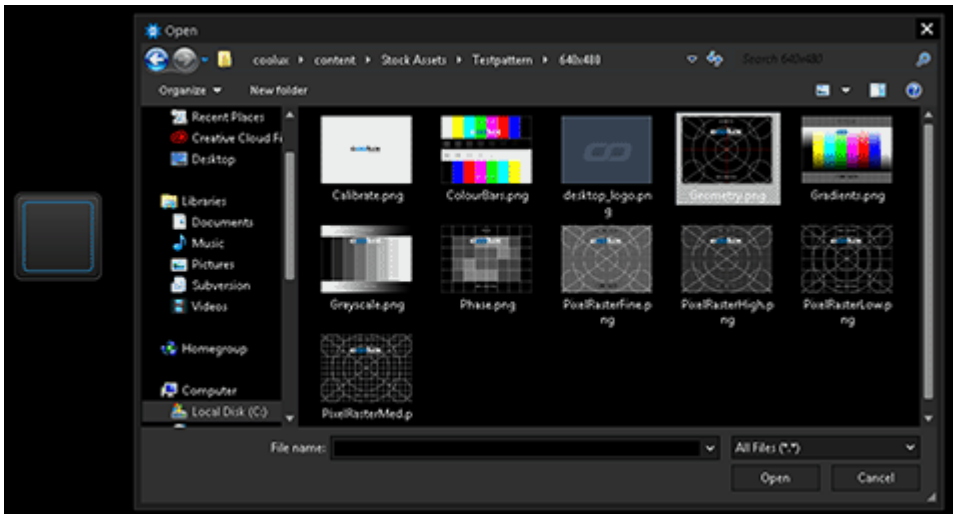
The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.2.5 Image Loader

The ImageLoader control is designed to easily exchange an image file that is added to the Pandoras Box project with any other image file on your PC. An interesting application for this could be e.g. if pictures of the audience are made and by button click are displayed on a Pandoras Box output.

Click on the ImageLoader button and a Windows explorer opens at the locations, from where you want to send image files to Pandoras Box.

The newly loaded image file will automatically spread to all connected Clients.



To create an ImageLoader widget choose "Widgets > Buttons > Image Loader". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the ImageLoader properties simply right-click it and choose the first menu entry "ImageLoader Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The ImageLoader property dialog opens up.


ImageLoader1 (ID: 1)
Name: ImageLoader1
Page: Page1
Id: 1
Notes:
Visible
Fix
Size: 88 x 88
X: 57
Y: 220
Reload Settings
Control
Folder ID:
File ID:
Width: 1024
Height: 768
Delay: 2.0
Fit
Target Path...
Start Path...
Web Link
URL:
Button Styles
Label:
Font: Roboto Condensed, 16pt
Tint:
Release:
Res
...
Click:
Res
...
Highlight:
Res
...
On Click Script
On Mouse Enter Script
On Mouse Leave Script
Network Broadcasting
Enable Receive
Enable Send
Send Always
IP: 255.255.255.255
Control:
Ui Effects & Animations
Automatically apply changes
OK
Cancel
Apply

General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The ImageLoader button's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the ImageLoader button will be displayed on every page.

Size:
Enter a pixel size for the ImageLoader button's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Reload Settings

Folder and File ID:
Enter here the File and Folder ID of the image in Pandoras Box, to which the new image should be loaded.

Width, Height, Fit:
You may enter the Width and Height the image should get. Press Fit to enable this size option, otherwise the image will be fitted to the size of the original image file inside Pandoras Box.

Target Path:
Press [Target Path...] and browse to the image file which should be exchanged through the new loaded images.

Start Path:
Press [Start Path...] and choose or create a folder, from where you later choose the new image files, that should be transferred to Pandoras Box.

Delay:
The Delay between copying the new image to the original image file and loading it inside Pandoras Box is set to 2.0 seconds. This delay is set to be sure that the copy process is finished before the image is loaded into Pandoras Box. This value can be edited by entering a new delay time.

Web Link URL:
The Web Link section is of special interest when working with the Web Server [1929] feature, i.e. clicking the ImageLoader button in a web browser. Enter a URL e.g. "https://www.christiepandorasbox.com" that your browser should call when clicking the button.
It is also possible to achieve a quick page change with the URL, simply enter a hash tag "#" and the page name. E.g.: #Page2

Button Style

Label:
Enter a label that will be displayed on the button. The text's color can be set via the color field and its font via the according button.

Tint:
ImageLoader buttons can be colored by selecting a tint color, this applies only to the default button images.

Three images can be assigned to the different states of the button (the modes are explained above).
"**Release**" refers to no click or not toggled.
"**Click**" is on press or toggled.
"**Highlight**" can be used to draw a PNG with alpha over either the click or release image.

Click on the "..." button to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

Script

In the Script section you may enter commands to be executed. You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

The scripts for Click, Mouse Enter and Mouse Leave are executed when the Widget Designer button is being used, on the local interface as well as on any web client.
You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts
If you are interested in the Web Server feature and some small examples, please read the topics Web Server [1929] and Object and Member Notation [1904] .

See here a list of all commands [1520].

Network Broadcasting

The NBS (Network Broadcast Service) allows transmitting and updating Faders, Labels and buttons across multiple WD Designers instances on the network. To activate this service, please refer to the Remoting dialog [1274]!

Enable Send:
To send the ImageLoader buttons state (clicked or released) as broadcast into the network, you only have to check "Enable Send".

Enable Receive:
To receive values from other ImageLoader buttons, please check "Enable Receive".

Now you have to specify which item should update your ImageLoader button:
Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the ImageLoader button through another ImageLoader button on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the ImageLoader button you want to take the status from. This could be e.g. "ImageLoader1" or " ImageLoader2".

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.2.6 Video Snapshot

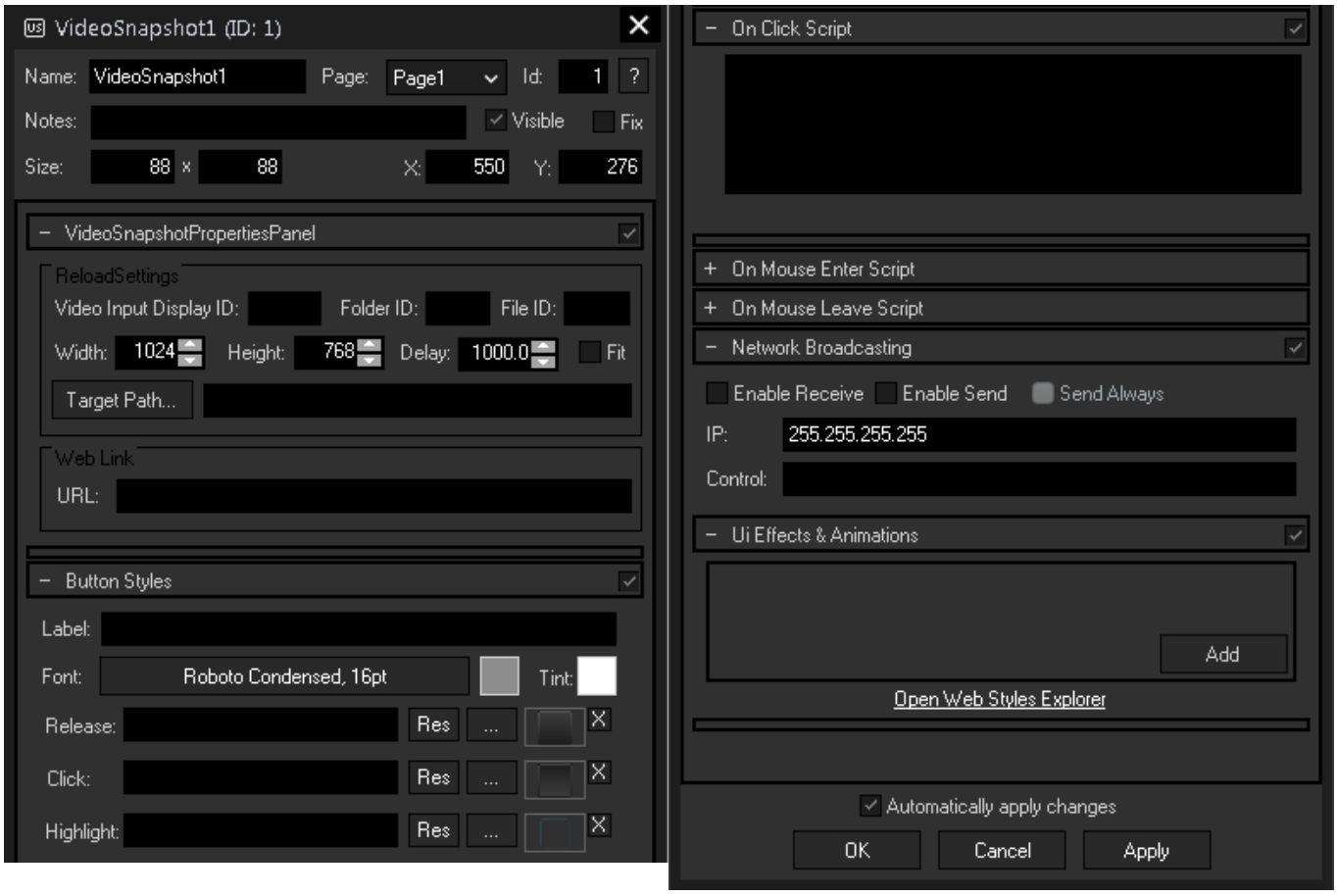
The VideoSnapshot widget is designed to easily exchange an image file within a Pandoras Box project by a Video Snapshot from your attached Video Input.

A click on the VideoSnapshot button and the created snapshot is immediately send to an image file within Pandoras Box.
The newly loaded image file will automatically spread to all connected Clients.

To send a Video Snapshot to Pandoras Box, you need to
1. Have an image file added into Pandoras Box project with assigned File and Folder ID [191].
2. Create a Video Input Display [865] and enter the Video Input ID of your Video Source.
3. Set up the VideoSnapshot Properties, see below.

To create an VideoSnapshot widget choose "Widgets > Buttons > VideoSnapshot". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the VideoSnapshot properties simply right-click it and choose the first menu entry "VideoSnapshot Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The VideoSnapshot property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The VideoSnapshot button's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the VideoSnapshot button will be displayed on every page.

Size:
Enter a pixel size for the VideoSnapshot button's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Reload Settings

Video Input Display ID:
Enter here the ID of the VideoInput display you want to generate the snaphots from.

Folder and File ID:
Enter here the File and Folder ID of the image in Pandoras Box, to which the new image should be loaded.

Width, Height, Fit:
You may enter the Width and Height the image should get. Press Fit to enable this size option, otherwise the image will be fitted to the size of the original image file inside Pandoras Box.

Delay:
The Delay between copying the new image to the original image file and loading it inside Pandoras Box is set to 1000ms. This delay is set to be sure that the copy process is finished before the image is loaded into Pandoras Box. This value can be edited by entering a new delay time.

Target Path:
Press [Target Path...] and browse to the image file which should be exchanged through the new loaded images.

Web Link URL:
The Web Link section is of special interest when working with the Web Server [1929] feature, i.e. clicking the VideoSnapshot button in a web browser. Enter a URL e.g. "https://www.christiepandorasbox.com" that your browser should call when clicking the button.
But it is also possible to achieve a quick page change with the URL, simply enter a hash tag "#" and the page name. E.g.: #Page2

Button Style

Label:
Enter a label that will be displayed on the button. The text's color can be set via the color field and its font via the according button.

Tint:
VideoSnapshot buttons can be colored by selecting a tint color, this applies only to the default button images.

Three images can be assigned to the different states of the button (the modes are explained above).
"**Release**" refers to no click or not toggled.
"**Click**" is on press or toggled.
"**Highlight**" can be used to draw a PNG with alpha over either the click or release image.

Click on the "..." button to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you

can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

Script

In the Script section you may enter commands to be executed. You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

The scripts for Click, Mouse Enter and Mouse Leave are executed when the Widget Designer button is being used, on the local interface as well as on any web client.
You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts
If you are interested in the Web Server feature and some small examples, please read the topics Web Server [1929] and Object and Member Notation [1904] .

See here a list of all commands [1520].

Network Broadcasting

The NBS (Network Broadcast Service) allows transmitting and updating Faders, Labels and buttons across multiple WD Designers instances on the network. To activate this service, please refer to the Remoting dialog [1274]!

Enable Send:
To send the VideoSnapshot buttons state (clicked or released) as broadcast into the network, you only have to check "Enable Send".

Enable Receive:
To receive values from other VideoSnapshot buttons, please check "Enable Receive".

Now you have to specify which item should update your VideoSnapshot button:
Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the VideoSnapshot button through another VideoSnapshot button on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the VideoSnapshot button you want to take the status from. This could be e.g. "VideoSnapshot1" or " VideoSnapshot2".

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

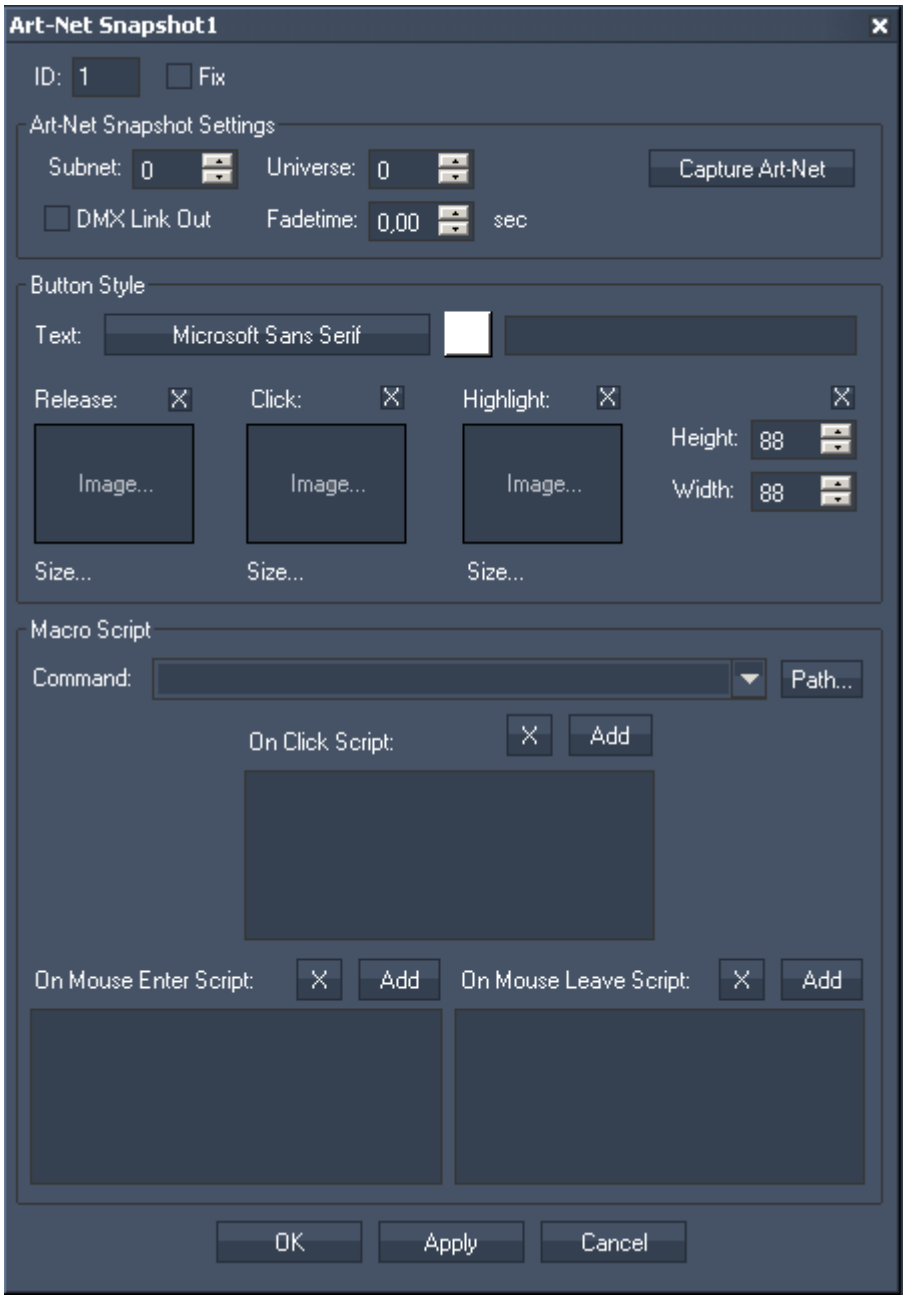
7.3.2.7 Art-Net Snapshot

The Art-Net Snapshot Button is designed to record a full Art-Net DMX universe state, with the ability to record from Universe A and output to Universe B.
A click on the Art-Net Snapshot Button and the captured Art-Net values are send out to the specified Universe.

Please note:
In order to use the Art-Net Snapshot Button Art-Net Input and Art-Net Output needs to be enabled in the Connection Manager [1258]!

To create an Art-Net Snapshot widget choose "Widgets > Buttons > ArtNetSnapshot". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Art-Net Snapshot properties simply right-click it and choose the first menu entry "Art-NetSnapshot Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Art-Net Snapshot property dialog opens up.



ID:
The Art-Net Snapshot's ID may be changed by entering a new one in the text field top left.

Fix:
When the option "Fix" is checked, the Art-Net Snapshot Button will be displayed on every page.

The Art-Net Snapshot Property dialog is divided into three sections: Art-Net Snapshot Settings, Button Style and Script.

Art-Net Snapshot Settings

Subnet and Universe:
Enter the Subnet and Universe the Art-Net values should be captured from.

DMX Link Out:
Check this option if you want to play out the Art-Net values via the DMX Link 1 Port Output device. Please note that the device has to be enabled in the Connection Manager[1258].

Fade Time:
If you want to fade to the Art-Net values stored in this button instead of sending them out abruptly, you may define a fade time here (in sec).

[Capture Art-Net]:
Press [Capture Art-Net] and the current state of the specified Universe will be recorded.
To send out these Art-Net values on the same Universe, close the Item Properties dialog and just press the Art-Net Snapshot Button.
To send out these Art-Net values on a different Universe, do it as following:
Enter the Subnet and Universe the Art-Net values should be captured from.
Press [Capture Art-Net] and the current state of the specified Universe will be recorded.

Now change the Art-Net settings to the Universe on which you want to send out the recorded Art-Net values (do NOT press [Capture Art-Net] again) and press [OK] or [Apply] at the bottom of the Item Properties dialog. Close this dialog and click on the Art-Net Snapshot Button.

Button Style

In the Button Style you may set up the button's text as well as its font and colour.
Three images can be assigned to the different states of the button.
"Release" refers to no click.
"Click" is on press.
"Highlight" can be used to draw a PNG with alpha over either the click or release image.

You may also set the Buttons Height and Width.

Script

In the Script section you may enter commands to be executed. You may use the drop-down list and the "Add" button or type directly in the text field. The topic Script Language [1511] explains this in more detail.
See here a list of all commands [1520]. These ones [1665] are available to control the Art-Net Snapshot Button itself.

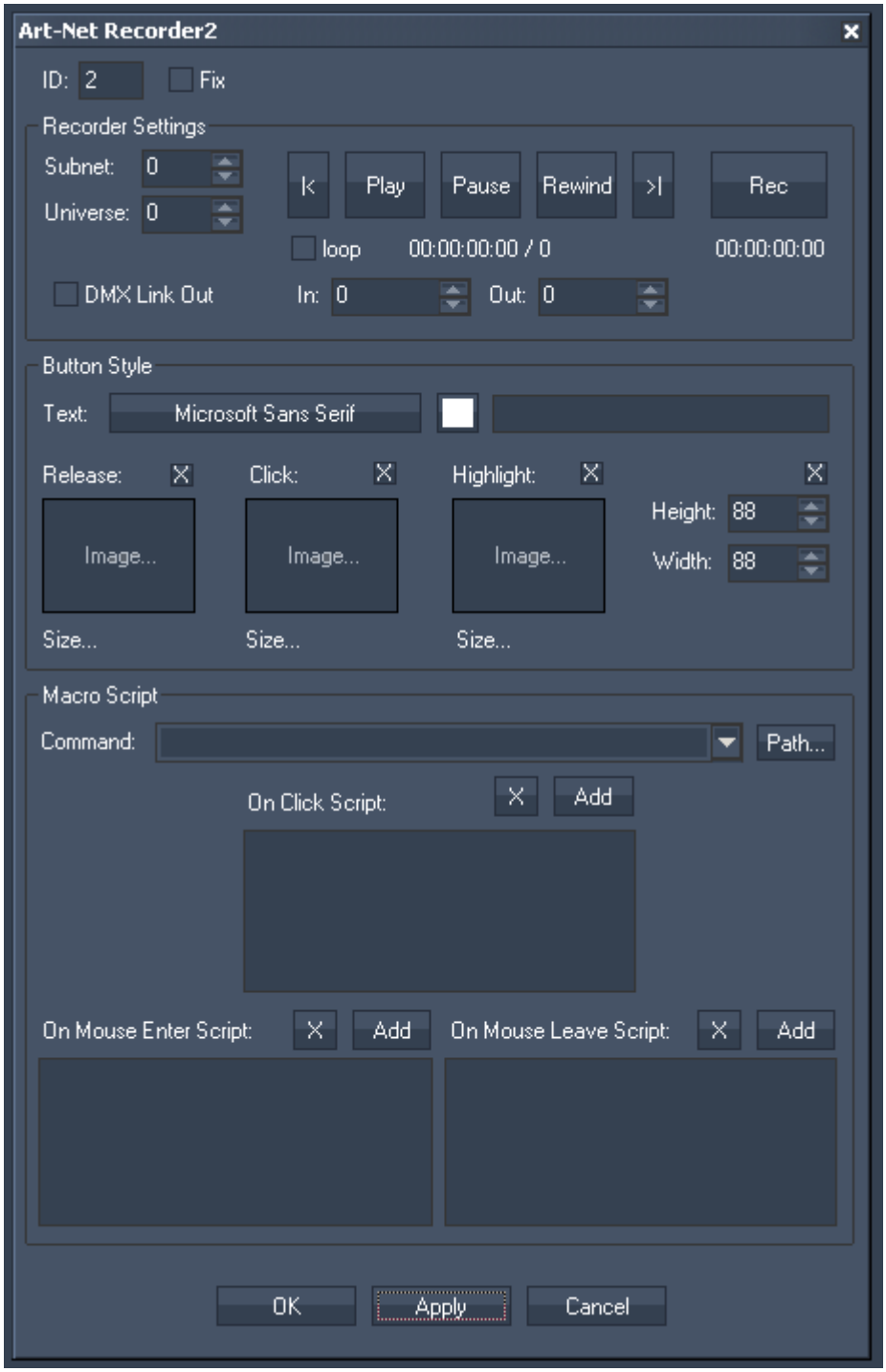
7.3.2.8 Art-Net Recorder

The Art-Net Recorder button is designed to record a full Art-Net DMX universe over a certain time, with the ability to record from Universe A and output to Universe B.
A click on the Art-Net Recorder and the captured Art-Net sequence is send out to the specified Universe.

Please note:
In order to use the Art-Net Recorder Art-Net Input and Art-Net Output needs to be enabled in the Connection Manager [1258]!

To create an Art-Net Recorder widget choose "Widgets > Buttons > ArtNetRecorder". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Art-Net Recorder properties simply right-click it and choose the first menu entry "ArtNet Recorder Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Art-Net Recorder property dialog opens up.



ID:
The Art-Net Recorder's ID may be changed by entering a new one in the text field top left.

Fix:
When the option "Fix" is checked, the Art-Net Recorder Button will be displayed on every page.

The Art-Net Recorder Property dialog is divided into three sections.

Recorder Settings

Subnet and Universe:
Enter the Subnet and Universe the Art-Net values should be recorded from.

[Rec]:
Press [Rec] and the specified Universe will be recorded until you press the button again, that meanwhile changes

its name to [Stop].
To send out the Art-Net values on the same Universe they were recorded from, close the Item Properties dialog and just press the Art-Net Recorder Button.
To send out these Art-Net values on a different Universe, do it as following:
Enter the Subnet and Universe the Art-Net values should be captured from.
Press [Rec] and the current state of the specified Universe will be recorded.
Press [Stop] to stop the recording.

Now change the Art-Net settings to the Universe on which you want to send out the recorded Art-Net values (do NOT press [Rec] again) and press [OK] or [Apply] at the bottom of the Item Properties dialog. Close this dialog and click on the Art-Net Recorder Button.

[|>] and [<|]:
With these two buttons you may browse through the recorded Art-Net sequence step by step.

[Play]:
Press [Play] to play the recorded Art-Net sequence.

[Pause]:
Press [Pause] to pause the recorded Art-Net sequence.

[Rewind]:
Press [Rewind] to bring the sequence to ist Inpoint.

[Loop]:
Activate this checkbox and the Art-Net sequence will be looped instead of played once.

In & Out:
Here you may define an In- and Output different from the originally reccorded sequence. To cut off 2 seconds from the beginning of the recording, enter "50" as inpoint (25 frames = 1 second).

[DMX Link Out]:
To play the recorded Universe via the DMX Link Out, aktivate this checkbox.
Please note that "DMX LINK Out" needs to be enabled in the Connection Manager [1258].

Button Style

In the Button Style you may set up the button's text as well as its font and colour.
Three images can be assigned to the different states of the button.
"Release" refers to no click.
"Click" is on press.
"Highlight" can be used to draw a PNG with alpha over either the click or release image.

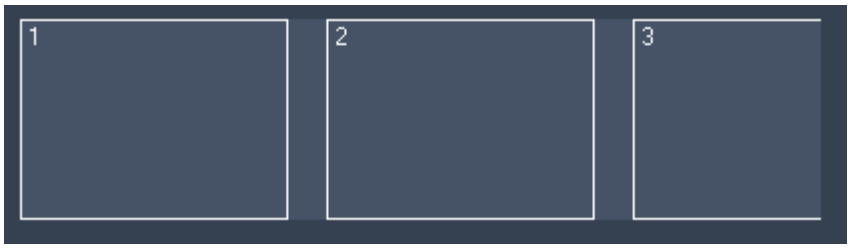
You may also set the Buttons Height and Width.

Script

In the Script section you may enter commands to be executed. You may use the drop-down list and the "Add" button or type directly in the text field. The topic Script Language [1511] explains this in more detail.
See here a list of all commands [1520]. To control the ArtNet Recorder via commands itself, please use these ones [1662]
.

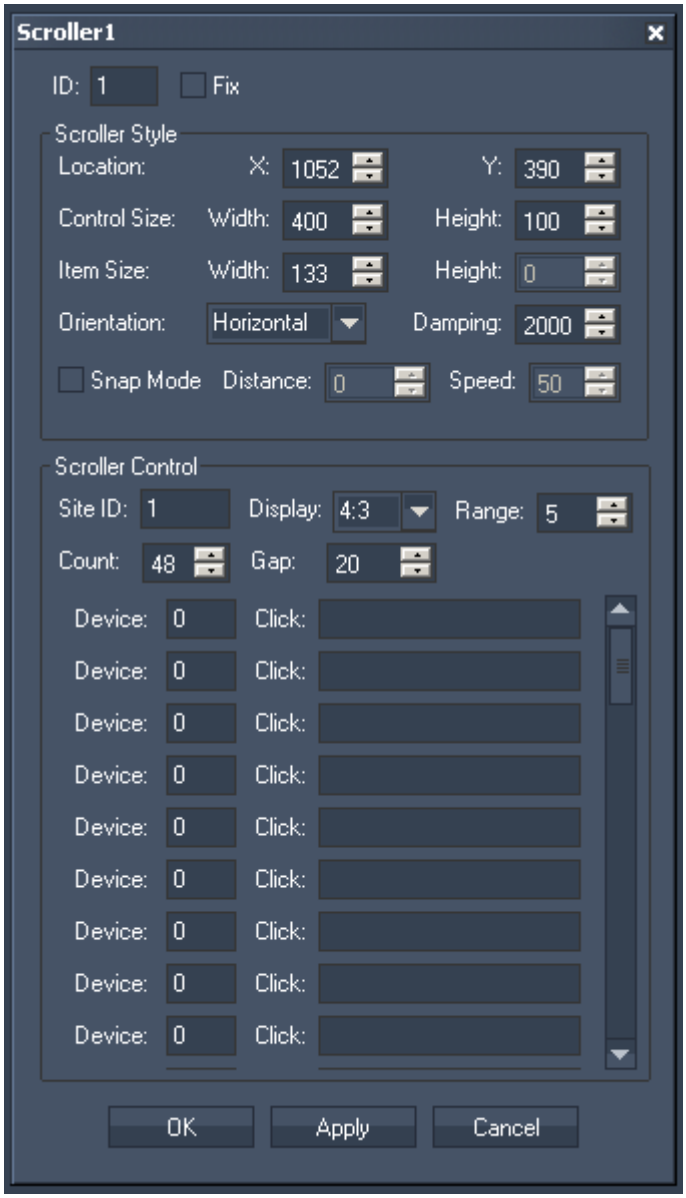
7.3.2.9 Scroller

The Scroller is designed to change the X or Y position of assigned media files in your Pandoras Box project, so that you can scroll through a certain amount of pictures.



To create a Scroller widget choose "Widgets > Buttons > Scroller (Horizontal or Vertical)". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Scroller properties simply right-click it and choose the first menu entry "Scroller Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Scroller property dialog opens up.



The property dialog is divided into two sections: the Scroller Style and the Scroller Control.

ID:
The Scroller's ID may be changed by entering a new one in the text field top left.

Fix:
When the option "Fix" is checked, the Scroller will be displayed on every page.

Scroller Style

Location:
To position the scroller enter here the Scroller's X and Y Position (in px) or move it manually with the mouse over the WD user interface when being in the edit / move mode [F9].

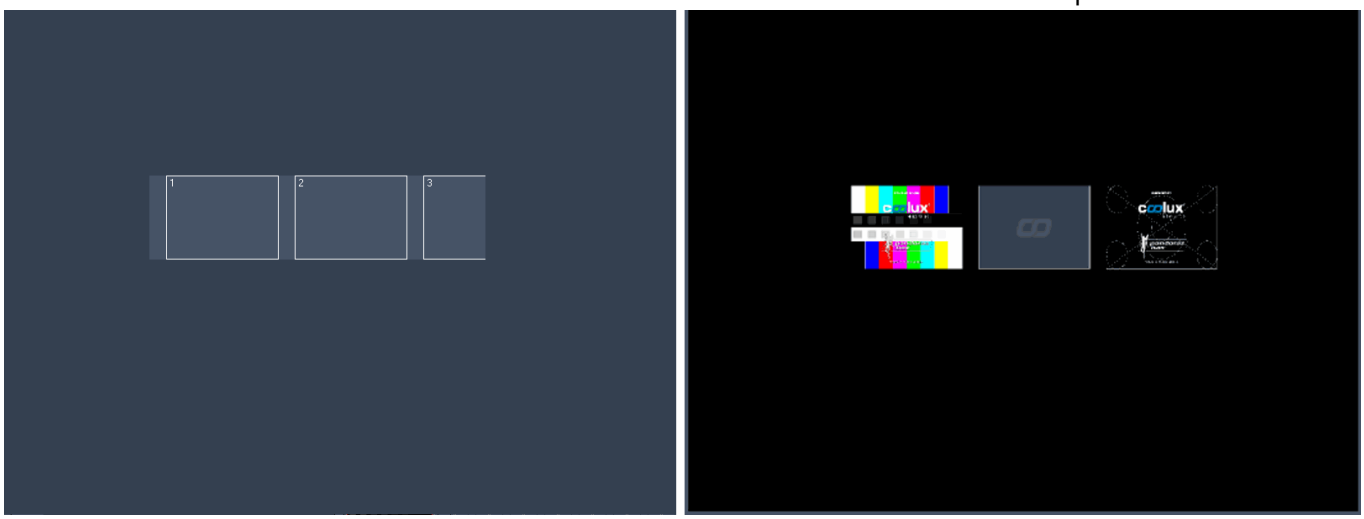
Control Size:
To adjust the Scroller's size enter the desired values (in px) for Width and Height. Or scale the Scroller manually with the mouse at its corner down right when being in the edit / move mode [F9].

Item Size:
Choose here how wide an item on the scroller should be. According to the width of the items you may see more or less items on the Scroller's interface. The Height of an item can't be influenced here – it always refers to the Height of the Scroller's Size itself. To adjust the Height please change the Height for the Control Size.

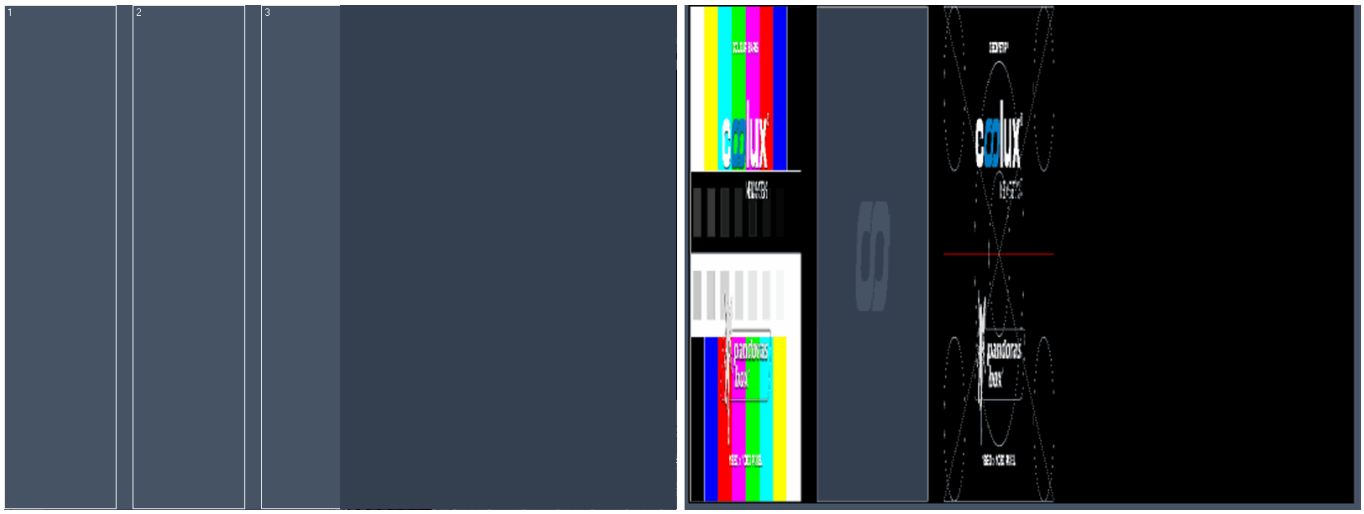
Relation of the Control / Item Size and the layers in Pandoras Box:

The size of an item influences the size of a layer inside Pandoras Box. To have a layer in Pandoras Box (that is controlled via an Scroller Item) covering the whole output, the Scroller size has to cover the whole Height of the WD interface. The Width of the Scroller does not necessarily have to cover the whole width of the WD interface as long as the Item's width gets the width resolution of the WD interface.

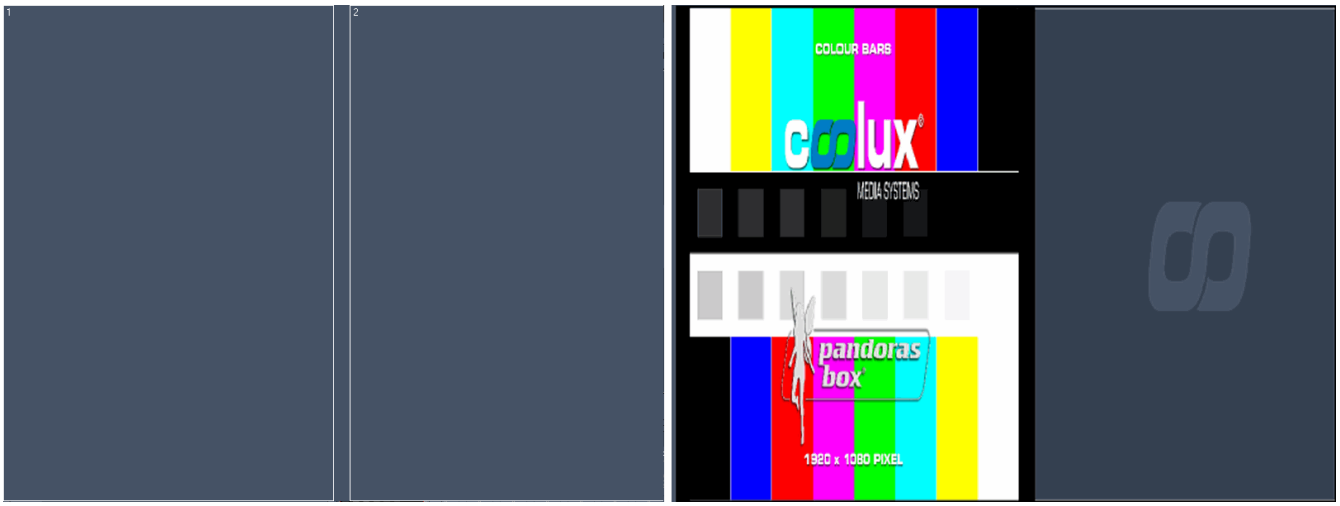
Example 1:
The Scroller in this example controls layers 1 through 3 in Pandoras Box. As you see the relation between the Scroller's size and the size of the whole WD interface is reflected in the Pandoras Box output.



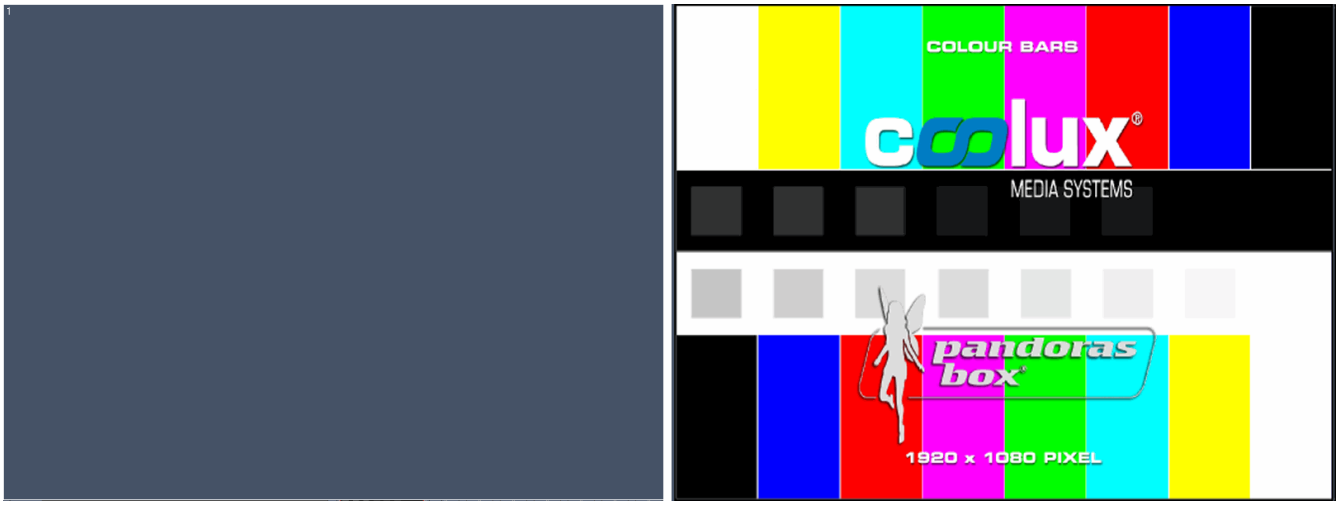
Example 2:
The Scroller's location is now set to X/Y = 0/0 and the Scroller's Height got the same Height the WD interface covers (without adjusting any width value). To bring the layer in Pandoras Box to aspect ratio, at least the Width of the Scroller's Items has to be adjusted to the Width of the WD interface, see next example.



Example 3a:
The Scroller now covers the whole WD interface. The Item's Width is set to half of the Scroller's Width.



Example 3b:
The Scroller covers the whole WD interface. The Item's Width is set to the Scroller's Width.



Orientation:
The Scroller can work horizontally and vertically, you may change its orientation here.

Damping:
The damping value influences the acceleration of the follow-up movement when the Scroller is released. The

default value is set to 2000. The minimum value is 0 (no follow-up movement), the maximum value is 10000 (longest follow-up movement).

Snap Mode
Without the Snap Mode enabled the Items will be moved over the Scroller as far as you scroll the Scroller. When the Snap Mode is enabled and an Item is e.g.. moved about half of its width to the left, the Scroller will automatically go on scrolling until the next Item snaps at the Scroller's left border.

--> Distance:
The distance value (in px) defines how far you have to scroll the scroller until the snap mode snaps to the next Item. By default it is set to 0, so each small movement to the left or to the right will activate the snap mode to snap to the next Item to the left or to the right. If the distance is set to e.g.. the value 100, you have to scroll the scroller about 100 px to the left / right to let the snap mode snap to the next item to the left/right.

--> Speed:
The speed value defines how fast the scroller is scrolled to the next item when the snap mode is being activated: 100= max. speed, 1 = min. speed.

Scroller Control

Site ID:
Enter here the Site ID of the site in Pandoras Box the Scroller should be linked to.

Display:
Please choose here the aspect ratio of the Pandoras Box output. This is important to bring the layers in Pandoras Box to the correct aspect ratio, according to the Scroller Item sizes.

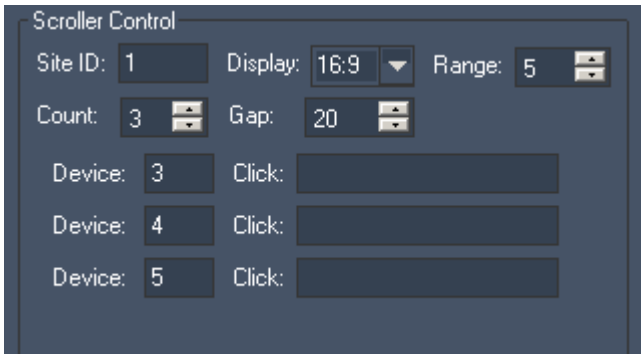
Range:
The range value defines how far a selected Scroller Item has to be moved in order to generate a click on it, see explanation under Device / Click. The default value is 5, that means that within moving the Item 5 pixels to the left / right the Item won't be clicked. This range function is useful not to activate an item by fault.

Count:
Set here the amount of Items visible in the Scroller and controllable in Pandoras Box.

Gap:
You may increase or decrease the gap between each Item by entering a new value (in px). It ranges from 0 (=no gap) up to 2048 (=max. gap).

Device / Click:
Set here the PB Layer ID (Device ID) which should be linked to each Item inside the Scroller.

Example:
Here the Count is set to 3, so only 3 Items are visible inside the Scroller and only 3 layers in PB can be controlled through the Scroller Items.

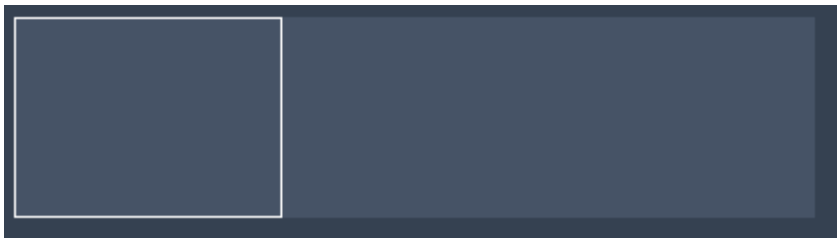


The devices are set so that:
Item 1 controls Layer 3 of site 1,
Item 2 controls Layer 4 of site 1,
Item 2 controls Layer 5 of site 1.

To execute a command when an Item is clicked please enter the command in the Click text field. The topic Script Language [1511] explains commands and how to use them in more detail.
As these small text fields do not offer a good overview when using multiple commands, working with macros [1897] is a good option.
See here a list of all commands [1520]. To reset the Scroller via a command (bringing it back to display the first Item), please use the following one:
WDScrollerReset,'ID' [1818]

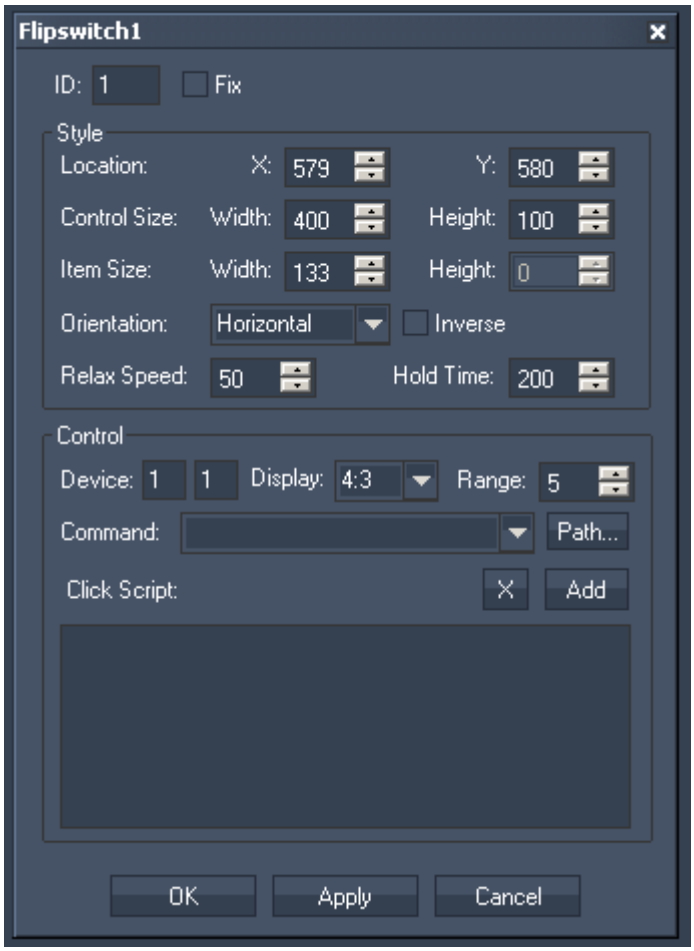
7.3.2.10 Flipswitch

The Flipswitch is designed to execute scripts only when the control reaches the specified unlock position.



To create a Flipswitch widget choose "Widgets > Buttons > Flipswitch (Horizontal or Vertical)". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Flipswitch properties simply right-click it and choose the first menu entry "Flipswitch Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Flipswitch property dialog opens up.



The property dialog is divided into two sections: Style and Control.

ID:
The Flipswitch's ID may be changed by entering a new one in the text field top left.

Fix:
When the option "Fix" is checked, the Flipswitch will be displayed on every page.

Flipswitch Style

Location:
To position the Flipswitch enter here the Flipswitch ‘s X and Y Position (in px) or move it manually with the mouse over the WD user interface when being in the edit / move mode [F9].

Control Size:
To adjust the Flipswitch's size enter the desired values (in px) for Width and Height. Or scale the Flipswitch manually with the mouse at its corner down right when being in the edit / move mode [F9].

Item Size:
Choose here how wide the handle-item on the Flipswitch should be. The height of an the item can't be influenced here – it always refers to the Height of the Flipswitch‘s Size itself. To adjust the height please change the height for the Control Size.

Relation of the Control / Item Size and the linked layer in Pandoras Box:
The size of the handle item influences the size of the linked layer inside Pandoras Box. To have the layer in Pandoras Box (that is controlled via the handle item) covering the whole output, the Scroller size has to cover the whole height of the WD interface. The width of the Scroller does not necessarily have to cover the whole width of the WD interface as long as the Item's width gets the width resolution of the WD interface.

Orientation:
The Flipswitch can work horizontally and vertically, you may change its orientation here.

Inverse:
Tick the check box to bring the Flipswitch Item to the opposite border of the Control.

Relax Speed:
The Relax Speed defines how fast the Flipswitch Item will move back to its default position after it was moved and the Click Script was NOT executed. The default value for the Relax Speed is 50. Decrease this value to get a slower move, increase it to quicken it.

Hold Time:
The Hold Time (in ms) defines how long the Item will stay on the position where the click script will be executed before it jumps back to its default position.

Flipswitch Control

Device ID:
Set here the PB Layer ID (Device ID) which should be linked to the Flipswitch.

Example:
With the Device ID 1 | 1 the Layer 1.1 in PB will be controlled.

Display:
Please choose here the aspect ratio of the Pandoras Box output. This is important to bring the layers in Pandoras Box to the correct aspect ratio, according to the Flipswitch Item size.

Range:
The range value defines how far the Flipswitch Item has to be moved to the opposite border (in pixels). The default value is 5, that means that within moving the Item 5 pixels to the border opposite the Flipswitch Item the click script will be executed. This range function is useful to define how far the Flipswitch Item has to be moved to execute the script.

Click Script:
To execute a command when the Flipswitch Item is moved into the click-area, please enter the command in the Click Script field. You may use the drop-down list and the "Add" button or type directly in the text field. The topic

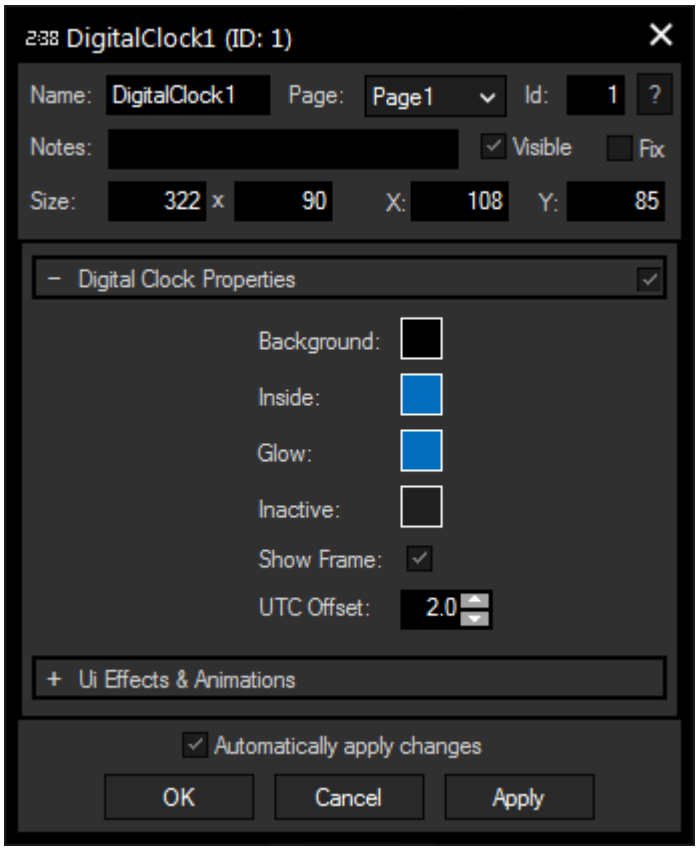
Script Language [1511] explains this in more detail.
See here a list of all commands [1520].

7.3.3 Digital Clock

Use this control to display the current time via a DigitalClock.

To create a DigitalClock widget choose "Widgets > DigitalClock". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the DigitalClock properties simply right-click it and choose the first menu entry "DigitalClock Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The DigitalClock property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The DigitalClock's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the DigitalClock will be displayed on every page.

Size:
Enter a pixel size for the DigitalClock's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

You may design your DigitalClock using different colors for **Background, Inside, Glow** and **Inactive**, you can also show or hide the **border**.

UTC Offset:
Enter here the time offset the DigitalClock should have from your computer's time.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.4 Displays

Bar Graph [852]
The Bar Graph allows showing any numeric value that is routed to the Bar Graph Output Node [1226].

Digital Display [854]
The Digital Display allows showing any numeric value that is routed to the Digital Display Output Node [1227].

Gauge [856]
The Angular Display allows showing any numeric value that is routed to the Gauge Output Node [1229].

Graph Display [858]
The GraphDisplay allows showing the graphical procession of a value over time that is routed to the Graph Display Output Node.

Terra Display Array [863]
The Terra Display shows the display arrangement of a connected Terra device.

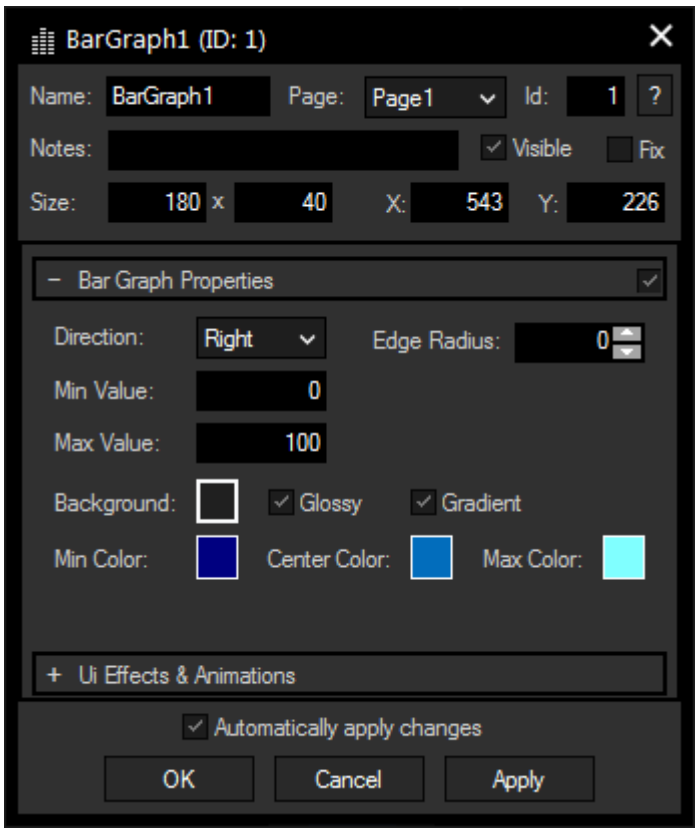
Video Input Display [865]
Use the Video Input Display displaying a video source on the WD user interface or to send a Video Snapshot [837] to Pandoras Box.

7.3.4.1 Bar Graph

The BarGraph allows showing any numeric value that is routed to the Bar Graph Output Node [1226].

To create a BarGraph widget choose "Widgets > Displays > BarGraph". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the BarGraph properties simply right-click it and choose the first menu entry "BarGraph Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The BarGraph Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The BarGraph's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the BarGraph will be displayed on every page.

Size:
Enter a pixel size for the BarGraph's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Bar Graph Properties

Direction:
Set the orientation of the BarGraph to up, down, left or right. You may also want to edit the **Edge Radius**.

Max / Min:
Set the value range for incoming data using Max and Min.

Design the BarGraph for your own needs by choosing different type of colors (for **background**, **Max, Center** and **Min Colour**). Max, Center and Min Color are only applicable when "**Gradient**" is checked. Otherwise, the BarGraph will have only the Center Color.

Check "**Glossy**" to add a 3D gloss to the widget.

Ui Effects & Animations

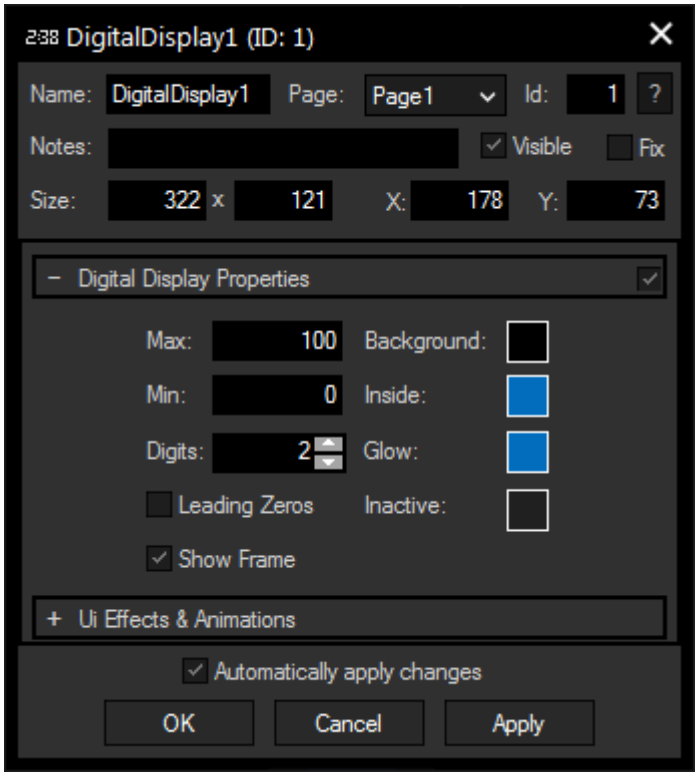
The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.4.2 Digital Display

The DigitalDisplay allows to show any numeric value that is routed to the DigitalDisplay Output Node [1227].

To create a DigitalDisplay widget choose "Widgets > Displays > DigitalDisplay". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the DigitalDisplays properties simply right-click it and choose the first menu entry "DigitalDisplays Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The DigitalDisplays Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The DigitalDisplay's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the DigitalDisplay will be displayed on every page.

Size:
Enter a pixel size for the DigitalDisplay's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Digital Display Properties

Max / Min:
Set the value range for incoming data using Max and Min. Enter the amount of digits.

Leading Zeros:
The option "Leading Zeros" fills all empty digits with zeros if amount of digits is smaller than the maximum, e.g. "7" will be displayed as "007" if maximum value is set to 100.

You may design your DigitalDisplay using different colors for **Background, Inside, Glow** and **Inactive**, you can also show or hide the **frame**.

Ui Effects & Animations

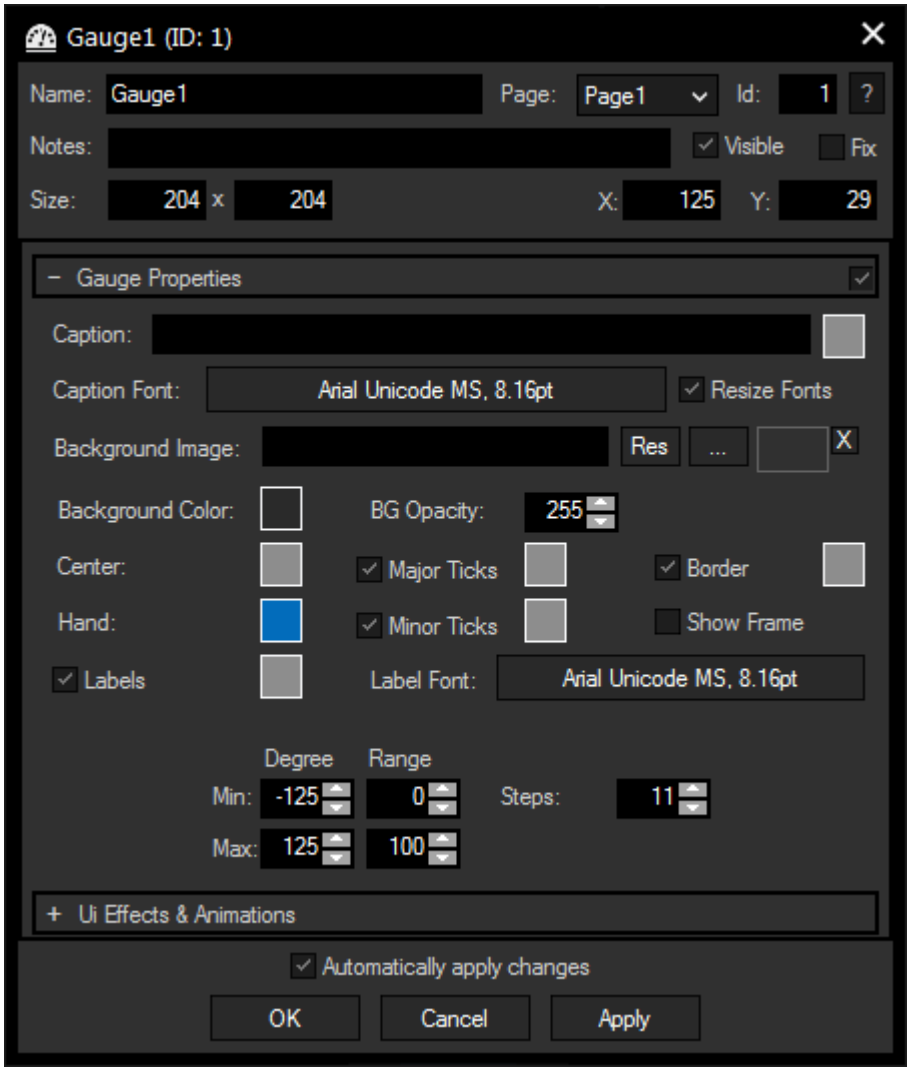
The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.4.3 Gauge

The Gauge allows showing any numeric value that is routed to the Gauge Output Node [1229].

To create a Gauge widget choose "Widgets > Displays > Gauge". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Gauge properties simply right-click it and choose the first menu entry "Gauge Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Gauge Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Gauge's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Gauge will be displayed on every page.

Size:
Enter a pixel size for the Gauge's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Gauge Properties

Caption:
Here, you can enter a label for the display, color and font can be selected with the respective buttons.

Resize Fonts:
This features adjusts the fonts of caption and numbers when changing the size of the widget.

Background:
An image can be loaded as a background. You can either browse your system for a picture or choose one out of the Resource Manager [1509].
It is also possible to set a background color and transparency.

You may design your Gauge using different colors for **Center, Hand, Label, Major Ticks, Minor Ticks** and **Border**.
Tick the check boxes to hide / display the mentioned units. A decorative **frame** can be displayed as well.

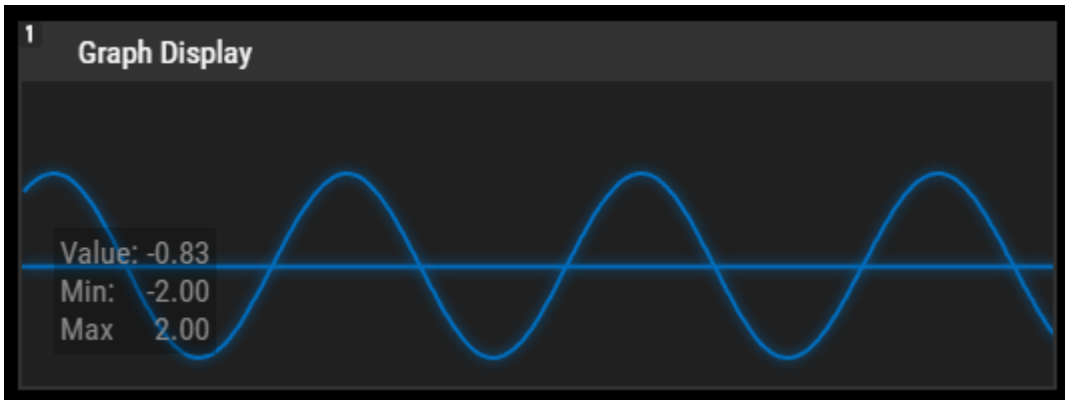
Degree / Range:
Set Min and Max for Degree and Range to change the Gauge's size and range. Enter the amount of steps for your range.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

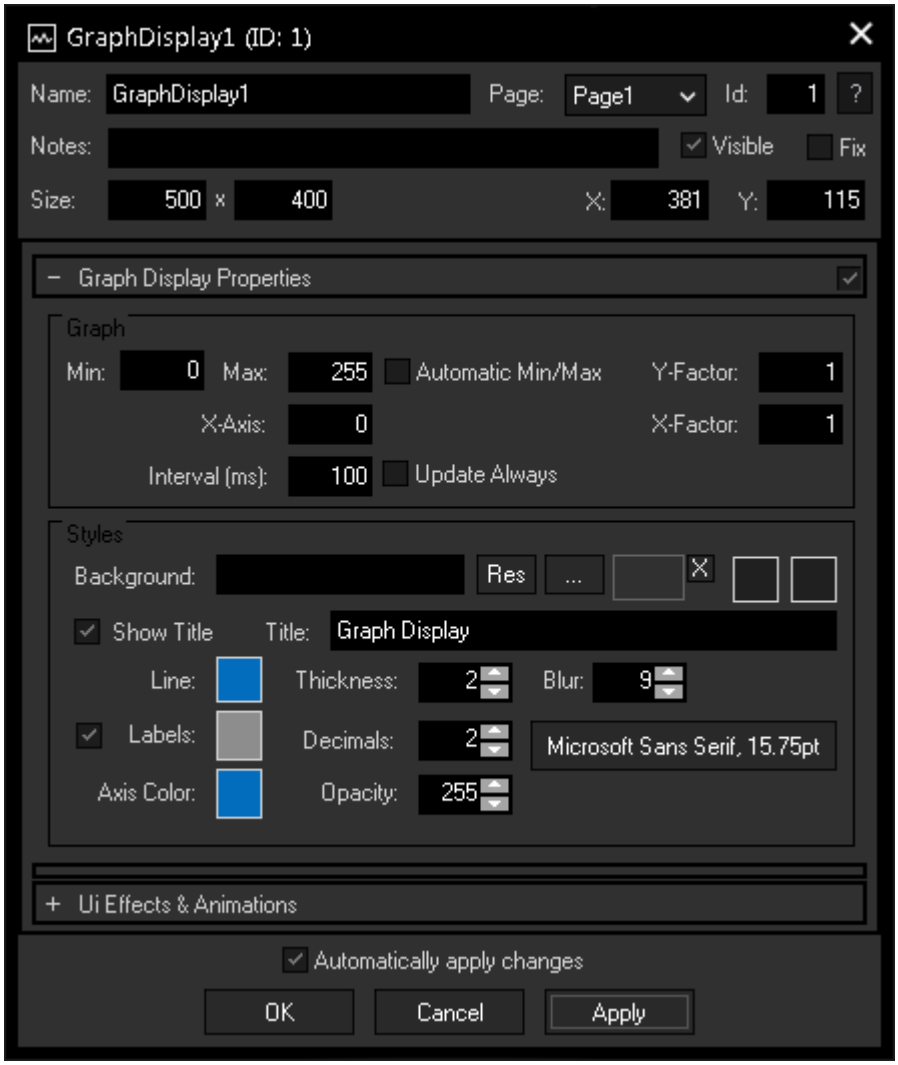
7.3.4.4 Graph Display

The GraphDisplay allows showing the graphical procession of a value over time that is routed to the Graph Display Output Node.



To create a GraphDisplay widget choose "Widgets > Displays > GraphDisplay". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the GraphDisplay properties simply right-click it and choose the first menu entry "GraphDisplay Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The GraphDisplay Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The BarGraph's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the BarGraph will be displayed on every page.

Size:
Enter a pixel size for the BarGraph's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Graph Display Properties

Max / Min:
Set the value range for incoming data using Max and Min. If you checked "**Automatic Min/Max**", those values are automatically to the incoming values.

X-/Y-Factor:
Set here a factor to adjust the coordinate system by compressing or stretching the graph.

X-Axis:
Enter here an offset for your X-axis.

Interval:
If you check "**Update Always**", the graph is being constantly updated within the given interval. Otherwise, it will only be updated on value change.

Background:
An image can be loaded as a background. You can either browse your system for a picture or choose one out of the Resource Manager [1509]. Additionally, you can define two different colors for a background gradient by clicking on the two far right color buttons.

Title:
Enter here a title that is displayed at the top of the GraphDisplay. Uncheck the box "Show Title" if you want to remove the title bar.

Labels:
The labels show the the current value as well as Min and Max. Check the box on the left to show the data and set **color** and **font** with the respective buttons. Adjust the exactness of the displayed values with the **Decimal** field.

Customize your GraphDisplay design by adjusting the **Line Color** and **Thickness**, the amount of **Blur** for the line and the **Axis Color**, as well as the **Opacity** of the graph.

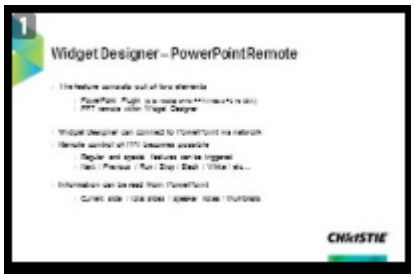
Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.4.5 PowerPoint Display

The PowerPoint Display shows a thumbnail sent from PowerPoint and allows to execute a script if you click on the image, hence it is a perfect control to see the current slide and call the next one for example.

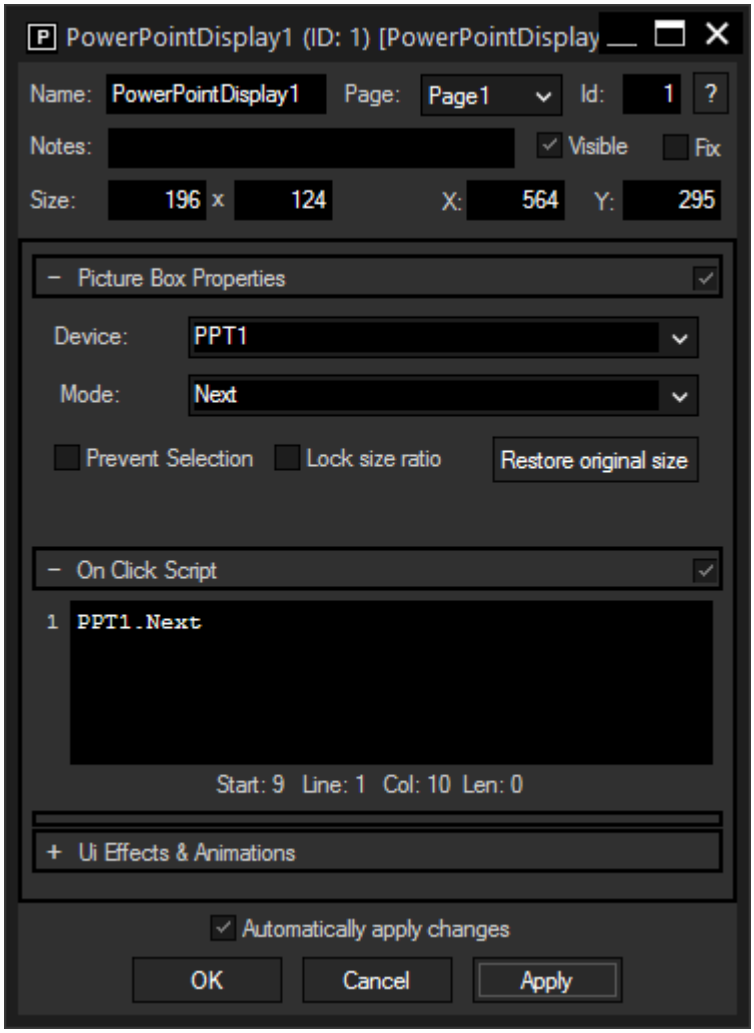
The display requires a connected PowerPoint device in the Configuration dialog. The chapter "PowerPoint 1468" explains how to do this and how to setup the PowerPoint plugin to send thumbnails.



Example with connected PowerPoint

To create a PowerPoint Display widget choose "Widgets > Displays > PowerPointDisplay". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the PowerPoint Display properties simply right-click it and choose the first menu entry "PowerPointDisplay Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The PowerPointDisplay Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The PictureBox' ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the PictureBox will be displayed on every page.

Size:
Enter a pixel size for the PictureBox' size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

PowerPoint Display Properties

Device:
Open the drop-down list and choose the name of a PowerPoint device which you have already connected using the Configuration dialog as explained in the chapter "PowerPoint [1468]".

Mode:
Choose which thumbnail should be shown. You can choose the previous, current or next slide but make sure that the plugin sends those thumbnails (see "PowerPoint [1468]").

Prevent Selection:
Check this box to avoid selecting the widget. It then can't be selected when in moving mode, except with a right-click directly on the PictureBox. Ticking this box also has the effect that the PictureBox' Z-position is being sent to the back.

Lock Size Ratio:
If this box is checked, the current size ratio will be maintained, even if you change the size manually.

Restore original size:
Click this button restore the image's original size. The original size can be displayed when hovering the mouse over the the image's thumbnail. Again, the size depends on the PowerPoint plugin. There, you can setup to send the thumbnails in three different **sizes**: Small-192x108, Medium-320x180 or Large-640x360 pixels.

Web Link URL:
The Web Link section is of special interest when working with the Web Server [1929] feature, i.e. clicking the PictureBox in a web browser. Enter a URL e.g. "https://www.christiepandorasbox.com" that your browser should call when executing the On Click Script.
It is also possible to achieve a quick page change with the URL, simply enter a hash tag "#" and the page name. E.g.: #Page2

Starting with Widget Designer version 6.0.6 the URL is editable from any version or license.

Script

In the Script section you may enter commands to be executed. To call the next slide, you could use the command:
`PPT1.Next`

You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts

See here a list of all commands [1520].

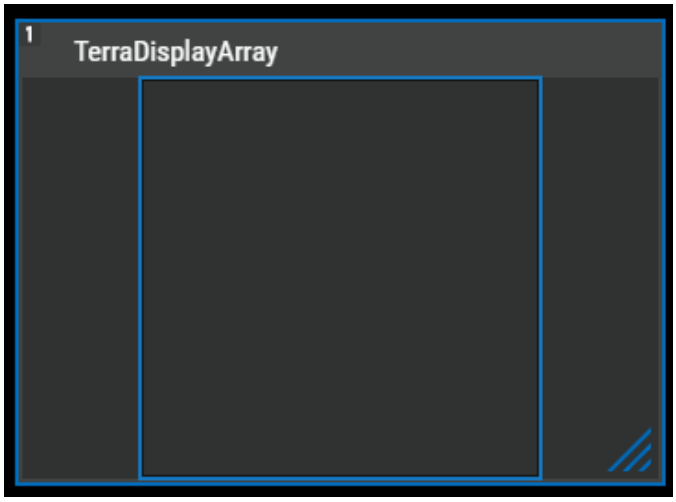
On Click Script:
Enter a script that will be executed when clicking on the image. Functions and Macros [1897] are a good option to manage large and sophisticated scripts.

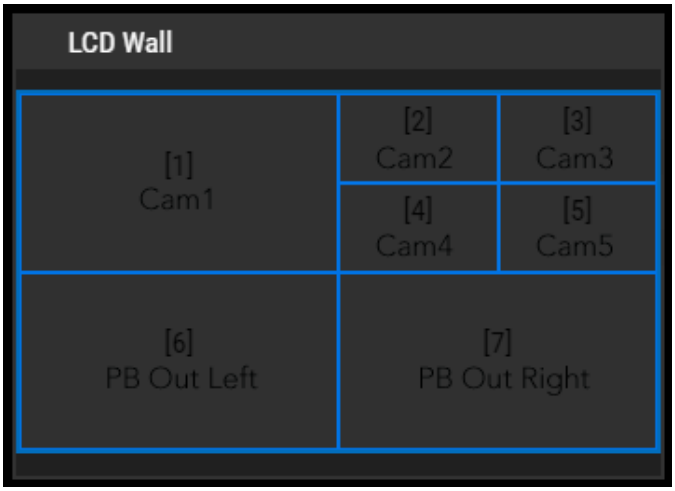
Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.4.6 Terra Display Array

The TerraDisplayArray shows the display arrangement of a connected Christie Terra device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Terra" [1332] which also shows how to work with the device using the scripting language.

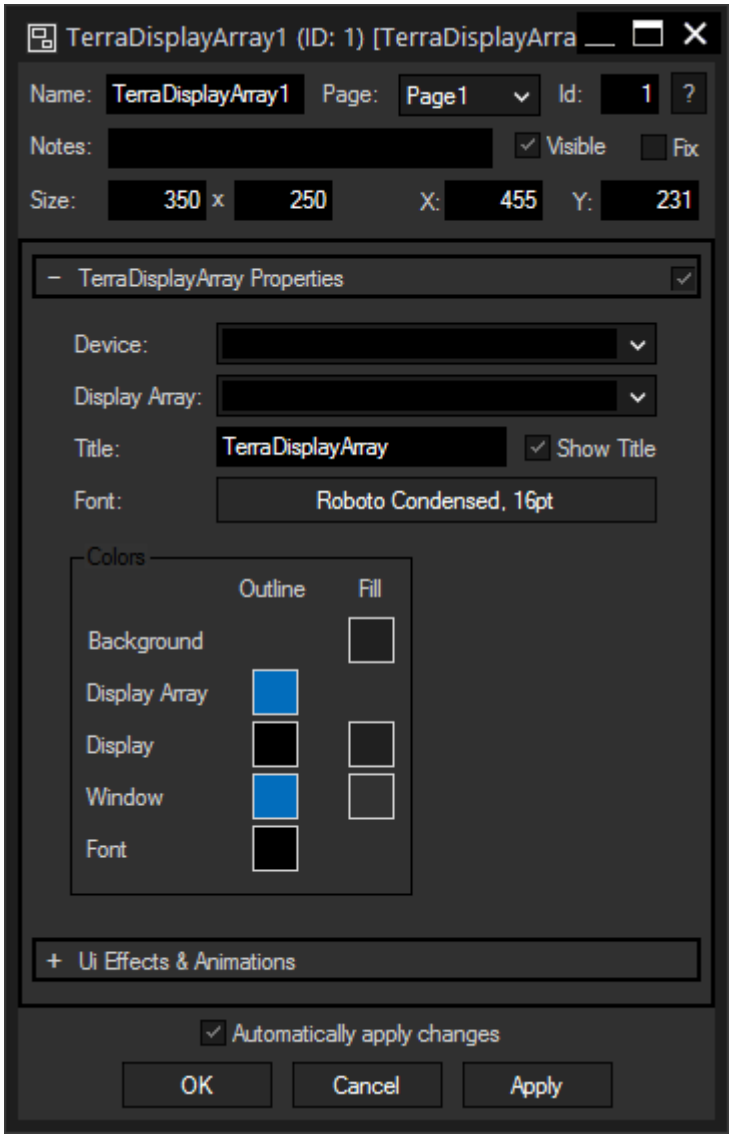




Left image: Example with connected Terra device

To create a TerraDisplayArray widget choose "Widgets > Displays > TerraDisplayArray". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the TerraDisplayArray properties simply right-click it and choose the first menu entry "TerraDisplayArray Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The TerraDisplayArray Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the widget will be displayed on every page.

Size:
Enter a pixel size for the widget's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels.

Terra Display Array Properties

Device:
Open the drop-down list and choose the name of a Terra device which you have already connected using the Configuration dialog as explained in the chapter "Terra [1332]".

DisplayArray:
Open the drop-down list and choose the name of the display array you like to see. The list updates automatically according to the connected Terra device.

Title:
Enter here a title that is displayed at the top of the widget. Uncheck the box "Show Title" if you want to remove the title bar.

Font:
Edit the font and the font size by clicking on the button with the current font. This font refers to the text written across each display.

Colors:
Click on the colored fields to choose another color. You can change the fill and / or the outline color for the background, the display array, the individual displays, the window and font.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

Using the Widget via the Scripting Language

Most widgets are accessible in the scripting language. You may either execute a function which they support or change a property or retrieve information from it. The scripting language supports regular commands, like:

```
WDTerraDisplayArraySize(1,320,180)
```

as well as widget specific members, like:

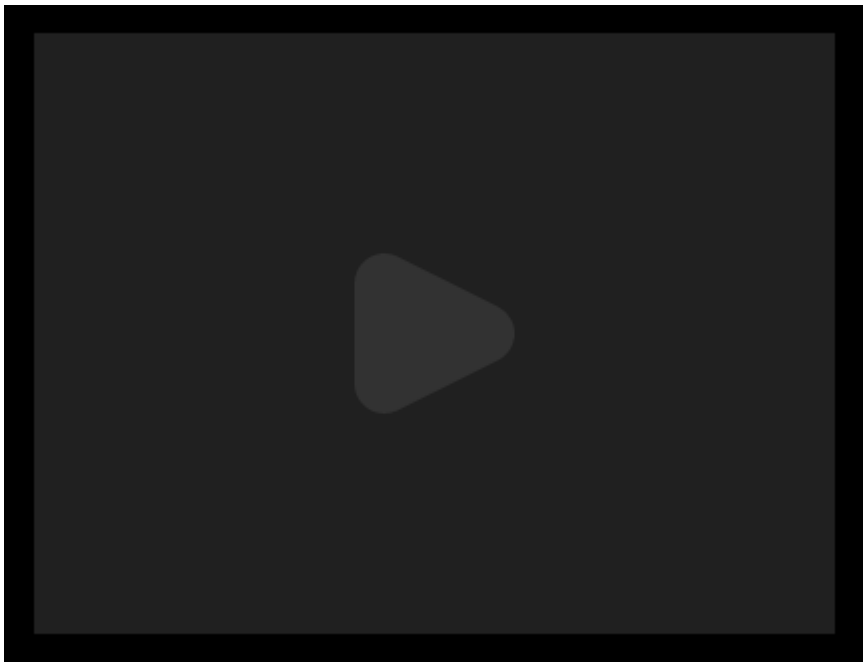
```
TerraDisplayArray1.SetSize(320,180)
```

To write a script with commands or members, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. Now simply start writing WDTerra... to see all commands referring to it or type the widget's name followed by a dot to see all supported members. As soon as you select a command or member, ScriptAssist offers more help. More information can be found in the chapters "Script Language" [1511] or "Object and Member Notation" [1904].

7.3.4.7 Video Input Display

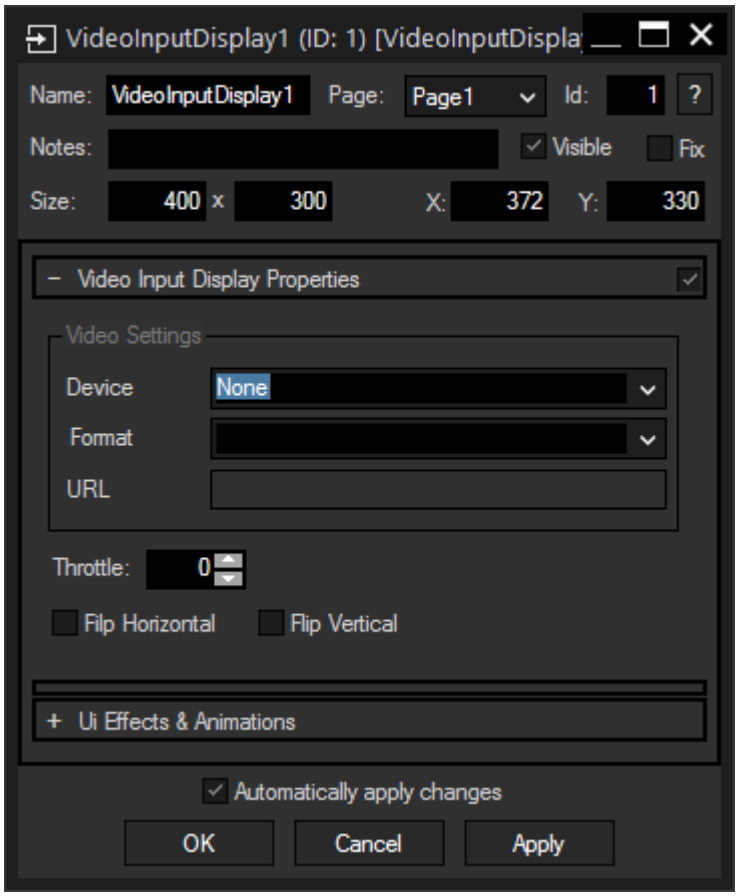
Use the Video Input Display to display a video source, such as a live input, a web cam or a desktop stream, on the WD user interface and to send a snapshot to Pandoras Box via the VideoSnapshot button [837].

In principle, all video inputs providing a DirectShow interface are usable.
Please bear in mind that a video input can only be used by one application at a time. This means that you can display the stream either in Widget Designer or Pandoras Box, but not both.



To create a Video Input Display widget choose "Widgets > Displays > Video Input Display". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Video Input Display properties simply right-click it and choose the first menu entry "VideoInputDisplay Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Video Input Display Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Video Input Display's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Video Input Display will be displayed on every page.

Size:
Enter a pixel size for the Video Input Display's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Video Input Display Properties

Device:
Select the desired video stream source from the drop-down list. All available live inputs are listed, e.g. web cams and network streams like StreamiX [725], NDI [47] or RTSP [47], . In case of choosing "RTSP", you also need to fill the "URL" textfield.

Format:
Select one of the offered formats which depends on the chosen Device.

URL:
Enter an URL in case you chose "RTSP" from the Device list. This can be an address from a web cam within your local network for example or a link to a public stream.

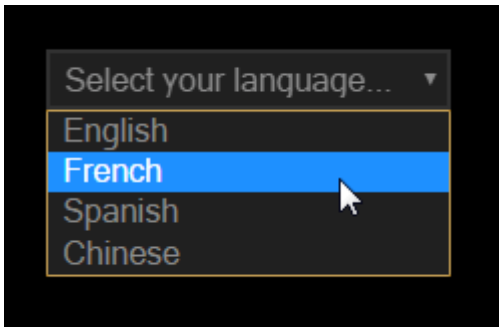
Throttle:
Enter here the amount (in ms) of throttle that is supposed to be applied to the refresh request rate of the video input.

Flip Horizontal / Vertical:
Check the according option if you want the video input to be mirrored.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.5 Drop Down List

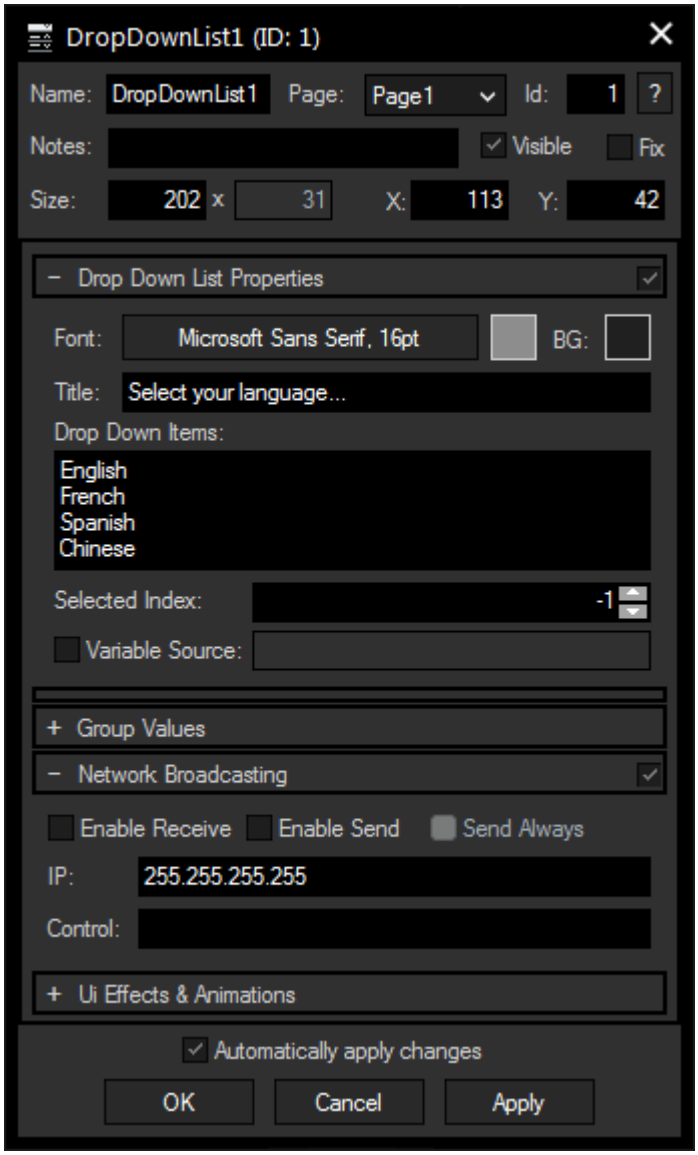


A DropDown List is a control element that allows to select one value from a series of predefined options. First, the DropDown List displays a single line, e.g. "Select a language". When the user clicks on the small down arrow, a list is displayed from which one entry can be chosen. Once this is done, Widget Designer displays the chosen value and in addition returns an internal index number. For the above depicted example, the operator gets the information that "French" and the index "1" has been chosen. Note that the indexing starts with "0". Depending on this information further commands can be executed accordingly. For example, Widget Designer toggles to a certain page with French labels or Pandoras Box calls a cue with French content.

To create a DropDown List widget choose "Widgets > DropDown List". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the DropDown List properties simply right-click it and choose the first menu entry "DropDownList Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The DropDown List property dialog opens up.

General Widget Settings



Name:

A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The DropDown List's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the DropDown List will be displayed on every page.

Size:
Enter a pixel size for the DropDown List's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Drop Down List Properties

Title:
Enter the text that is displayed per default before the Drop Down List is opened.
To set this text back you can use the command `WDDropDownListSetText(ID,Text)`

Font and Color:
Pick a font setting and color for the displayed text including the DropDown Items.

Drop Down Items:
Enter the items for the list to be displayed when the Drop Down List is opened with the down arrow.
The "Command List [1708]" lists all available commands.

Selected Index:
Choose here the selected item you would like to display per default, e.g. when you do not want to show a Title but one of the items.

Variable Source:
Check this option and enter the name from an existing list variable [1900].

Group Values

This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Network Broadcasting

The NBS (Network Broadcast Service) allows transmitting and updating Faders, Labels and CustomScript buttons etc. across multiple WD Designers instances on the network. To activate this service, please refer to the Remoting dialog [1274]!

Enable Send:
To send the DropDown List's state as broadcast into the network, you only have to check "Enable Send".

Enable Receive:
To receive values from other DropDown Lists, please check "Enable Receive".

Now you have to specify which item should update your DropDown List:
Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the DropDown List through another DropDown List on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the DropDown List you want to take the status from. This could be e.g. "DropDown1" or "DropDown2").

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

Programming with Drop Down Lists

This is a short description how you can program with the DropDown List.

First of all there is a DropDownList Input node [1078] which can be used to forward the chosen text and index number to a following filter or output node.

Second, the DropDown control returns member values, allowing for example the following script. Please find more information in the topic Object and Member Notation [1904] .

```
If DropDownList1.Index = 1
{
Label1.Text = "Bonjour"
}
```

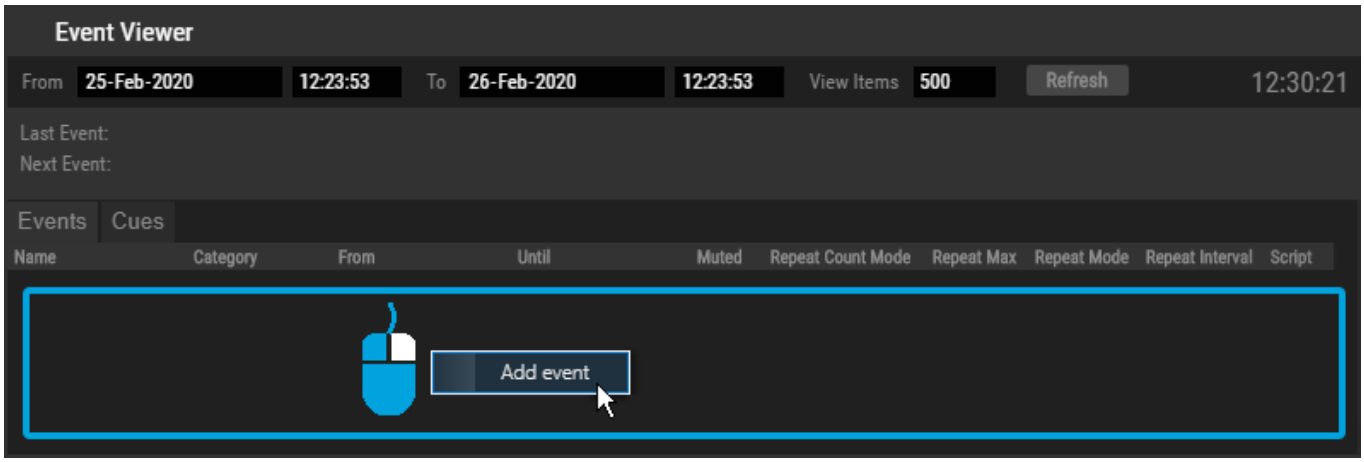
Lastly, the "Command List [1708]" lists all available commands.

7.3.6 Event Viewer

The EventViewer widget allows you to create new Events and gives an overview of them. An Event is a scheduled script, i.e. it allows to execute commands at a certain time and can also be repeated in a certain time interval.

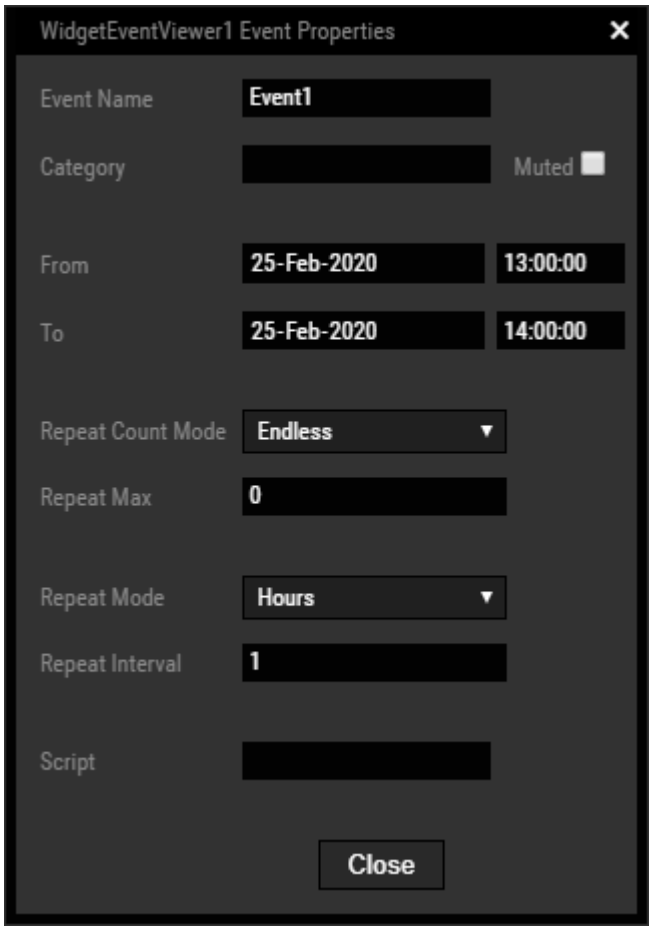
To create an EventViewer widget choose "Widgets > EventViewer". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

Now, you can add an Event or Event series. Simply right-click into the area where the Events are listed, and choose "Add event". The Event Properties dialog comes up.



Add Event - Properties Dialog

All changes done in the dialog are applied immediately. There is no dedicated "Apply" button, you simply close the dialog when you are done.



Event Name

The name simply helps to identify the Event. Please choose a unique name.

Category
The category simply helps to allocate and identify the Event.

Muted
If this option is checked, the Event is not executed.

From
Choose a start date and time for the scheduled Event. For the date, you can either de-/increase the date by one day with the small up/down arrow keys, or choose one from the calender that opens as a drop-down list, or you can enter the date manually.

To
The end date is only of interest when the "Repeat Count Mode" option is set to "Until Date". Choose an end date and time for the scheduled Event whereas the same options apply as described for the Start date.

Repeat Count Mode
The mode influences if and how the Event repeats. You can choose between three options:
- Endless: The Event repeats according to the options "Repeat Mode" and "Repeat Interval" for an unlimited number of times. The options for the end date and "Repeat Max" are ignored.
- Until Date: The Event repeats according to the options "Repeat Mode" and "Repeat Interval" but only until the end date and time is reached. The "Repeat Max" option is ignored.
- Max Count: The Event repeats according to the options "Repeat Mode" and "Repeat Interval" for the number of times set up under "Repeat Max". The end date is ignored.

Repeat Max
This number is only of interest when the "Repeat Count Mode" option is set to "Max Count" and sets up how many times the Event happens according to the options "Repeat Mode" and "Repeat Interval".

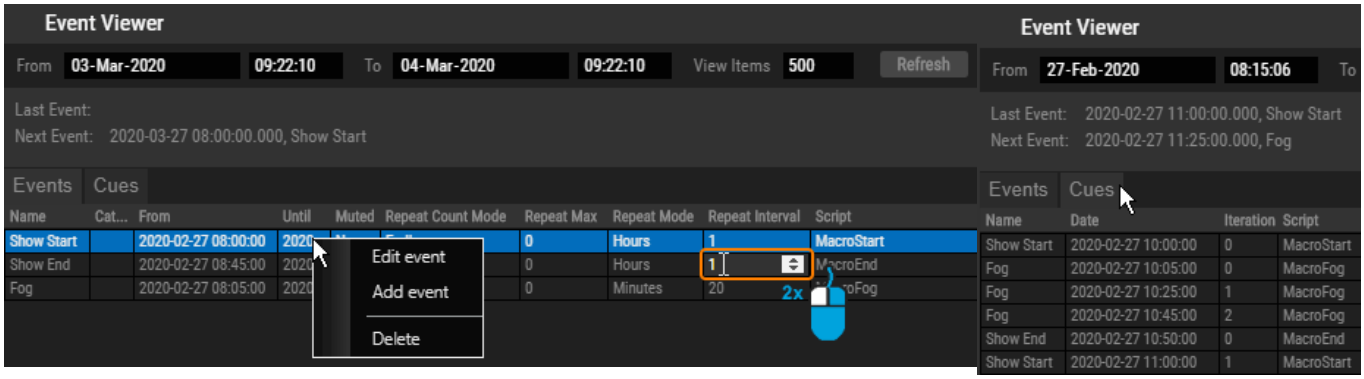
Repeat Mode, Repeat Interval
With both settings you can setup, that the Event is scheduled every 5 minutes for example.
The "Repeat Mode" sets up the time unit and the "Repeat Interval" the count of those units after which the Event repeats. Choose "Single Event" if it should only happen ones. Choose Seconds, Minutes, Hours, Days, Weeks, Months, or Years and a "Repeat Interval" if the Event should be scheduled multiple times.

Script
Enter the command that should be executed at the scheduled time. Currently, the script field only supports one line of code. To execute a longer script, write it first into a Macro [1897] or CustomScript button [822] and call it then with the Event. To call a macro, simply enter the macro's name into the Script field. To "click" a CustomScript button, enter the command `CustomScript1.Click`

Event Viewer

Once you created an Event, the EventViewer will display it in two different ways in the tabs "Events" and "Cues".

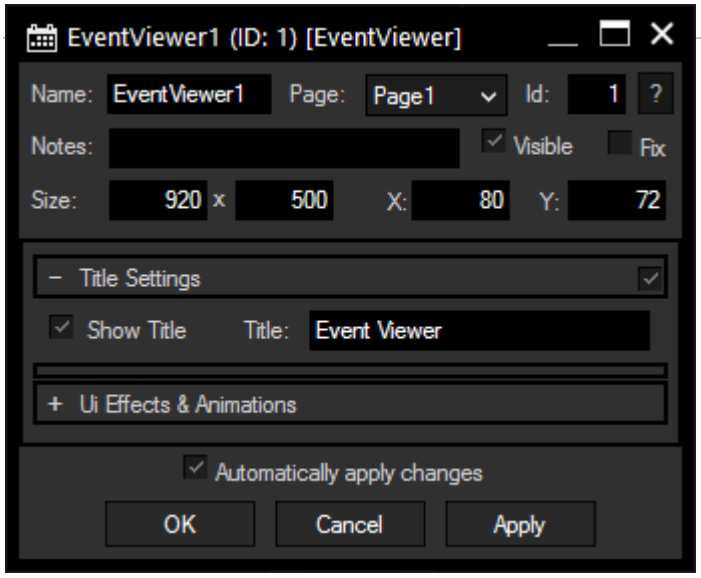
"Events" lists all Events once in the order in which they where created and the top date filters do not apply so that you really can see every Event. You can right-click one to edit or delete it. You can also double-click a single field to edit it.



The "Cues" tab on the other hand lists all instances belonging to one Event. So if you created an Event that happens every hour, you will see 24 entries per day. All instances of all Events are sorted by date and time so that you can see the upcoming Event right away. You can filter this view by using the options at the top of the widget, like the start and end date / time.
Again, you can right-click an Event instance to edit or delete the superordinate Event, the tab switches to "Events" and selects the according entry there. You can also select an Event and press the [Delete] key.

Event Viewer - Properties Dialog

To edit the EventViewer properties simply right-click the upper half and choose the first menu entry "EventViewer Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The EventViewer property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The EventViewer's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the InputBox will be displayed on every page.

Size:
Enter a pixel size for the EventViewer's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Title Settings

Show Title and Title:
Decide whether a title should be shown at the top of the widget and enter one.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.7 Fader Controls

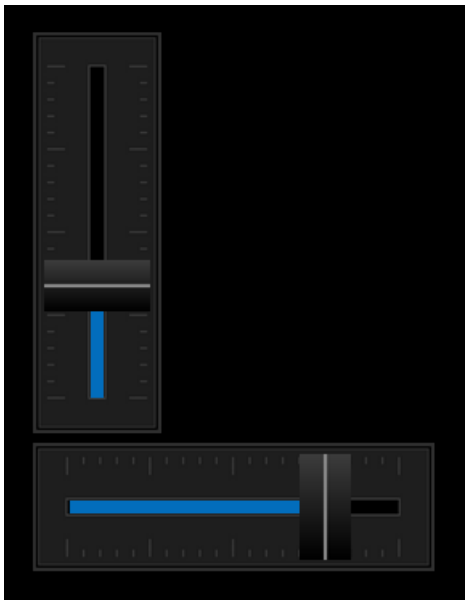
A fader can be created to control Pandoras Box device parameters or sequence parameters.

Choose between four kinds of faders:

Fader Vertical / Horizontal [874]
Encoder [877]
Color Picker [880]
Wheel Vertical / Horizontal [883]

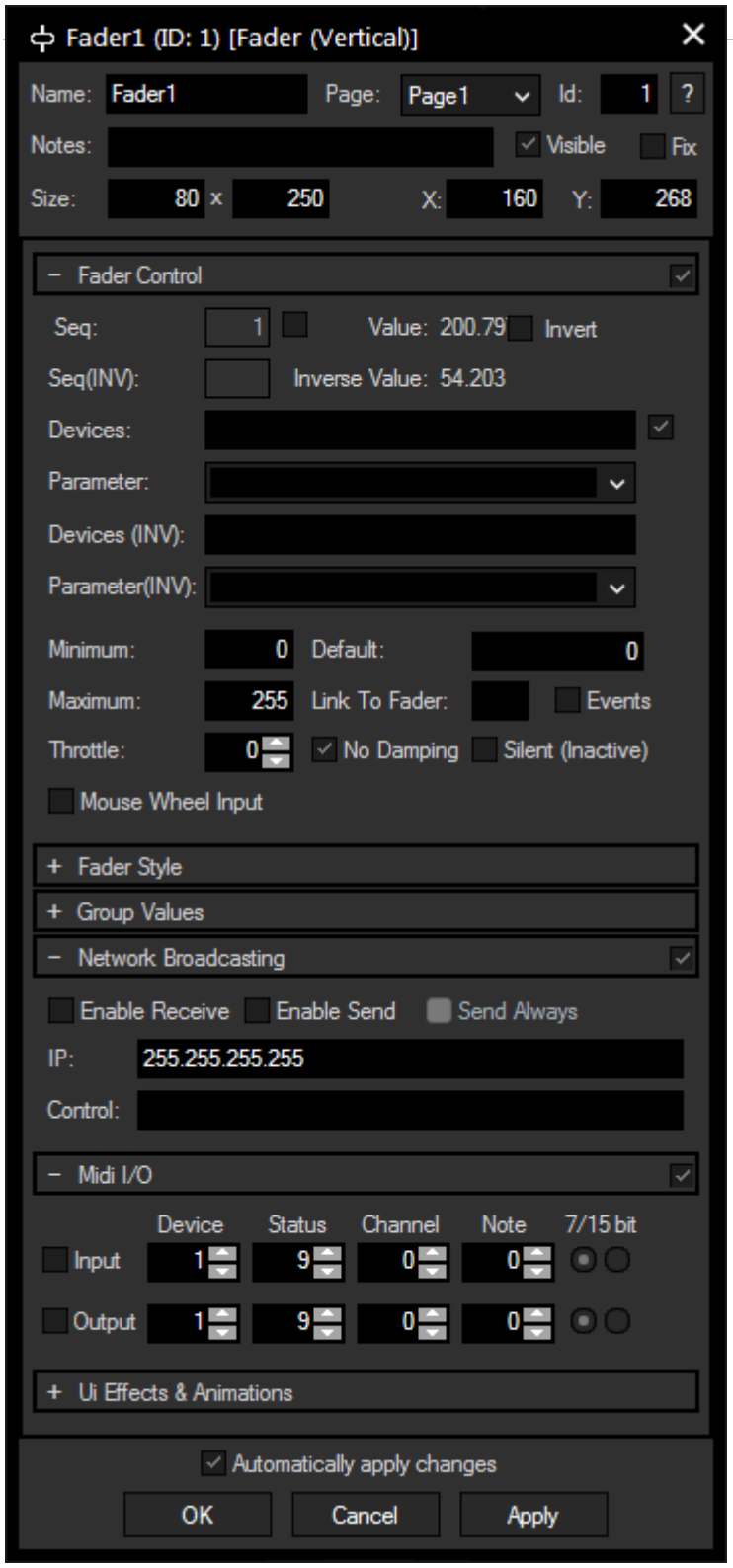
7.3.7.1 Fader

A Fader can be created to control Pandoras Box device parameters or a sequence opacity, additionally you can retrieve and set its value with nodes [936] and Object and Member Notation [1904].



To create a Fader widget choose "Widgets > Faders > Fader (Vertical or Horizontal)". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Fader properties simply right-click it and choose the first menu entry "Fader Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Fader Property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Fader's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Fader will be displayed on every page.

Size:
Enter a pixel size for the Fader's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Fader Control Settings

Seq:
Check this box to directly control the opacity of a Sequence and enter the respective sequence ID.

Seq(INV):
Choose a second sequence to send the inverse value to. This is especially useful if you like to cross-fade two sequences.

Invert:
Check this box to invert your value.

Devices:
Check the box on the right to send the Fader value to a Device (layer) parameter. Enter the Site and Device ID of the respective Device, multiple Devices can be entered too, with a white space as separator (e.g.: "2.1 2.2 2.17")

Parameter:
This drop-down offers you all available Device parameters of Pandoras Box. If you want to access an FX parameter, use the FX name and the pipe symbol to specify it (e.g.: "RGB Multiply|Blue"). If the parameter list does not show the desired parameter (in case that you use a custom device) simply type in the name of the parameter here.

The inverse Fader value can also be assigned to device parameters.

Minimum / Maximum:
Enter here the minimum and maximum value your Fader should have, as well as the **Default** position of the handle.

Link To Fader:
Enter here the ID of a second Fader, this Fader should listen to. The actual Fader can be used as it is, but as soon as the linked Fader is moved, the first one will follow.
This feature is especially useful when requiring the same Fader on several pages and using the "Fix" checkbox is not an option.

Events:
If two Faders are mutually linked via the Link To Fader option, they can interfere with each other if both are listening to the other's input events. To ensure a smooth movement, uncheck at least one of the Events boxes. Please note that if a Fader is moved from the remote Fader and the Events box is not checked, no values are transmitted from this Fader to the node system.

Throttle:
The Throttle parameter limits the amount of data transmitted to Pandoras Box. The higher this value is, the fewer times the Device's parameters are updated. Throttling the data can help saving performance, but also reduces smoothness.

No Damping:
Check this box to turn off the parameter value smoothing in Pandoras Box.

Silent (Inactive):
Check this box if you temporarily want to mute its output

Mouse Wheel Input:
If you check this box, the Fader will react to your mouse's scroll wheel when hovering over the widget.

Fader Style

Two images can be assigned to the different parts of the Fader, "**Handle**" and "**Background**".

Click on the "..." button to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

With the **Orientation** parameter, you can set your Fader either vertically or horizontally.

Group Values

This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Network Broadcasting

The NBS (Network Broadcast Service) allows to transmit and update Faders, Labels and Custom Script Buttons across multiple WD Designers instances on the network.

To activate this service, please refer to the Remoting Menu [1273]!

Enable Send
To send the faders values as broadcast into the network, you only have to check "Enable Send".

Enable Receive
To receive values from other faders, please check "Enable Receive".

Now you have to specify which item should update your fader:

Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the fader through another item on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the item you want to take the values from. This could e.g. be "Fader1" or "Fader2".

Once all values are set, the fader is ready to be used or edited at any time.

Midi I/O

The Fader widget is able to receive Midi Notes and / or send its own value as Midi note velocity. A Midi input / output device has to be connected via the Connection Manager [716] or the Midi Connection Manager.

For sending and receiving Midi, please select which one you would like to do and enter the respective settings:

Device:
If you have several connected Midi devices, enter here the ID of a specific device assigned in the Midi Connection Manager.

Status / Channel / Note:
Specify the values of the note you want to listen to. The values are numeric, so note E4 would be represented by 64. Please note that different translations of Midi notes can exist for different devices, so you might need to add or subtract an octave.

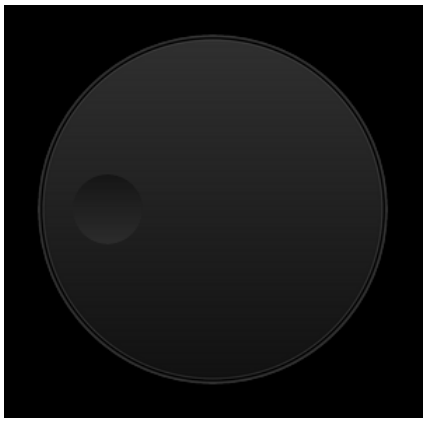
7/15bit:
Some systems support 15 bit Midi values instead of 7bit, select the respective radio box here.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

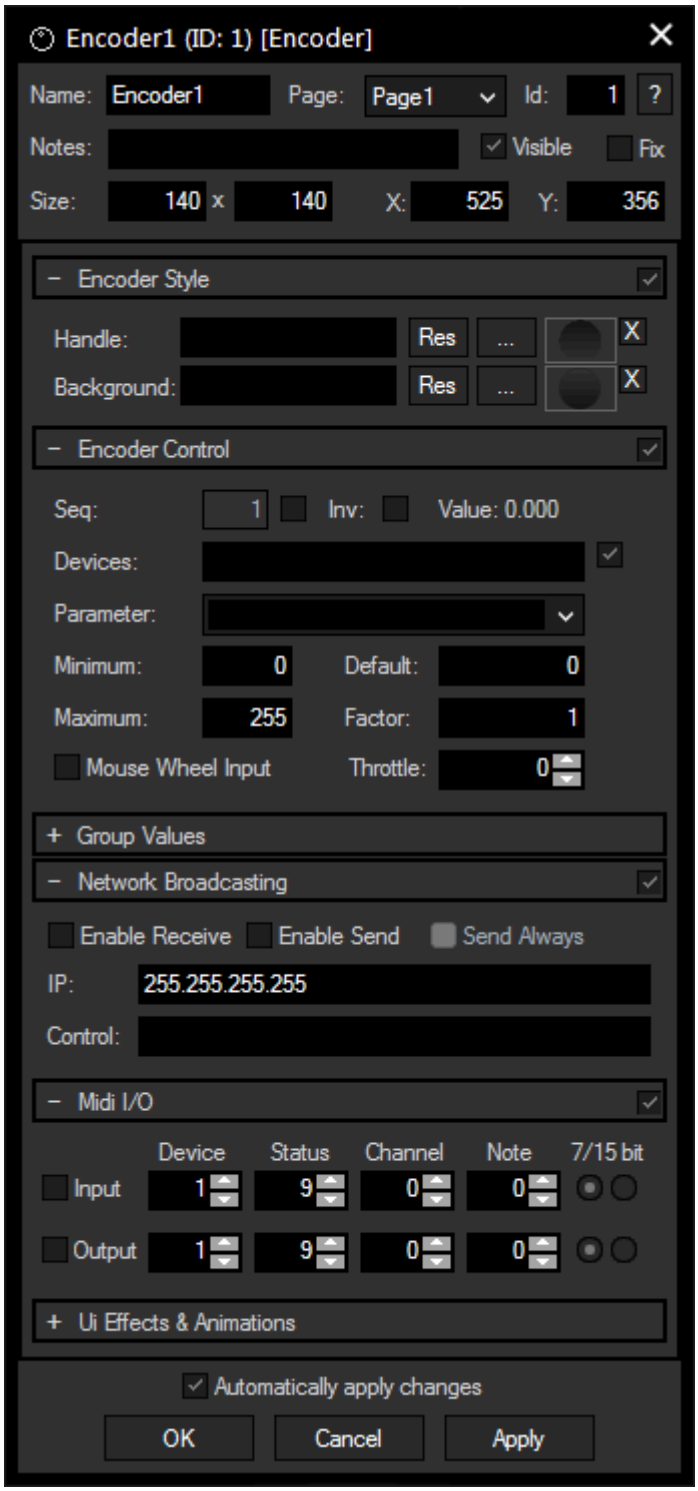
7.3.7.2 Encoder

An Encoder can be created to control Pandoras Box device parameters or a sequence opacity, additionally you can retrieve and set its value with nodes [936] and Object and Member Notation [1904].



To create an Encoder widget choose "Widgets > Faders > Encoder". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Encoder properties simply right-click it and choose the first menu entry "Encoder Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Encoder property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Encoder's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Encoder will be displayed on every page.

Size:
Enter a pixel size for the Encoder's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Encoder Style

Two images can be assigned to the different parts of the Encoder, "**Handle**" and "**Background**".

Click on the "..." button to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

Encoder Control

Seq:
Check this box to directly control the opacity of a Sequence and enter the respective sequence ID.

Invert:
Check this box to invert your value.

Devices:
Check the box on the right to send the Encoder value to a Device (layer) parameter. Enter the Site and Device ID

of the respective Device, multiple Devices can be entered too, with a white space as separator (e.g.: "2.1 2.2 2.17")

Parameter:
This drop-down offers you all available Device parameters of Pandoras Box. If you want to access an FX parameter, use the FX name and the pipe symbol to specify it (e.g.: "RGB Multiply|Blue"). If the parameter list does not show the desired parameter (in case that you use a custom device) simply type in the name of the parameter here.

Minimum / Maximum:
Enter here the minimum and maximum value your Encoder should have, as well as the **Default** position of the handle.

Factor:
Enter here a factor with which the step size should be multiplied. If you set a factor of 7, each movement of the Encoder will increase/decrease the value by 7 instead of 1.

Throttle:
The Throttle parameter limits the amount of data transmitted to Pandoras Box. The higher this value is, the fewer times the Device's parameters are updated. Throttling the data can help saving performance, but also reduces smoothness.

Mouse Wheel Input:
If you check this box, the Encoder will react to your mouse's scroll wheel when hovering over the widget.

Group Values

This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Network Broadcasting

The NBS (Network Broadcast Service) allows to transmit and update Faders, Labels and Custom Script Buttons across multiple WD Designers instances on the network.

To activate this service, please refer to the Remoting Menu [1273]!

Enable Send:
To send the Encoder values as broadcast into the network, you only have to check "Enable Send".

Enable Receive:
To receive values from other Encoders, please check "Enable Receive".

Now you have to specify which item should update your Encoder:

Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the Encoder through another item on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the item you want to take the values from. This could e.g. be "Encoder1" or "Encoder2".

Once all values are set, the Encoder is ready to be used or edited at any time.

Midi I/O

The Encoder widget is able to receive Midi Notes and / or send its own value as Midi note velocity. A Midi input / output device has to be connected via the Connection Manager [716] or the Midi Connection Manager.

For sending and receiving Midi, please select which one you would like to do and enter the respective settings:

Device:
If you have several connected Midi devices, enter here the ID of a specific device assigned in the Midi Connection Manager.

Status / Channel / Note:
Specify the values of the note you want to listen to or send. The values are numeric, so note E4 would be represented by 64. Please note that different translations of Midi notes can exist for different devices, so you might need to add or subtract an octave.

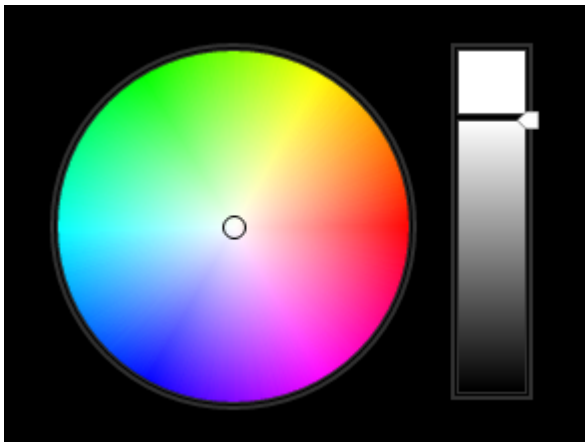
7/15bit:
Some systems support 15 bit Midi values instead of 7bit, select the respective radio box here.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

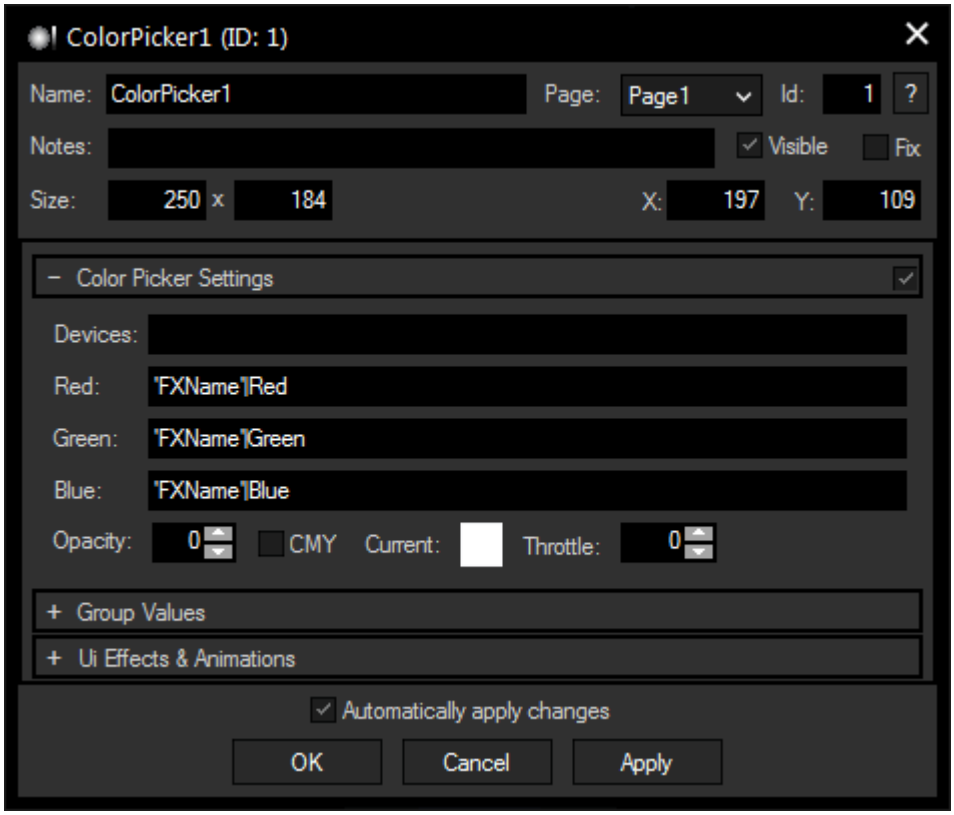
7.3.7.3 Color Picker

The ColorPicker allows you to choose a RGB / CMY color and its brightness. The values of this RGB / CMY color can be transferred to the Color FX faders on a layer in Pandoras Box.



To create a ColorPicker widget choose "Widgets > Faders > ColorPicker". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the ColorPicker properties simply right-click it and choose the first menu entry "ColorPicker Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The ColorPicker property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The ColorPicker's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the ColorPicker will be displayed on every page.

Size:
Enter a pixel size for the ColorPicker's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Color Picker Settings

Devices:
Enter single or multiple devices the ColorPicker should effect. Separate multiple devices with a space character (e.g.: "1.1 1.2 1.3")

Red / Green / Blue:
The ColorPicker can send its RGB values directly to an FX in Pandoras Box. Just enter the name of the FX (as it is written in Pandoras Box) and the parameter name, i.e. the color, separated by a pipe symbol.

Example:
RGB Multiply|Red
RGB Multiply|Green
RGB Multiply|Blue

CMY:
Check this box if you do not want to assign the Color Picker's red, green and blue value, but the cyan, magenta and yellow value to the Pandoras Box FX.

Current:
The current color can also be set with the standard Windows color picker here.

Throttle:
The Throttle parameter limits the amount of data transmitted to Pandoras Box. The higher this value is, the fewer times the Device's parameters are updated. Throttling the data can help saving performance, but also reduces smoothness.

Group Values

This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.7.4 Wheel

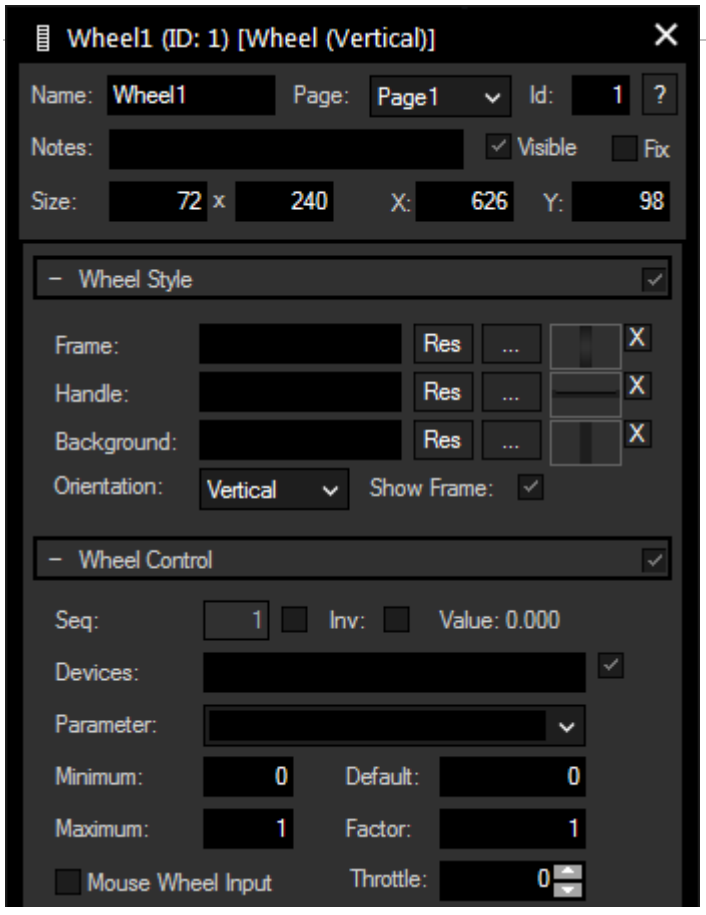
A Wheel can be created to control Pandoras Box device parameters or the sequence opacity, additionally you can retrieve and set its value with nodes [936] and Object and Member Notation [1904].

To create a wheel right-click anywhere on the empty main background window and go to "Widgets">"Fader Controls">"Wheel Vertical" or "Wheel Horizontal".

Once you have clicked on the last menu item the mouse cursor will change to a cross hairs icon. This icon tells you that you are in the create mode.
To create one or multiple wheels just left click anywhere on the empty main background window and you will see a new wheel being assigned and displayed.
The wheel may now be clicked on and moved up and down.

To create a Wheel widget choose "Widgets > Faders > Wheel (Vertical or Horizontal)". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Wheel properties simply right-click it and choose the first menu entry "Wheel Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Wheel property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Wheel's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Wheel will be displayed on every page.

Size:
Enter a pixel size for the Wheel's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Wheel Style

Three images can be assigned to the different parts of the Wheel, "**Frame**", "**Handle**" and "**Background**".

Click on the "..." button to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

With the **Orientation** parameter, you can set your Wheel either vertically or horizontally.

Show Frame:
Uncheck this box to hide the frame.

Wheel Control

Seq:
Check this box to directly control the opacity of a Sequence and enter the respective sequence ID.

Invert:
Check this box to invert your value.

Devices:
Check the box on the right to send the Wheel value to a Device (layer) parameter. Enter the Site and Device ID of the respective Device, multiple Devices can be entered too, with a white space as separator (e.g.: "2.1 2.2 2.17")

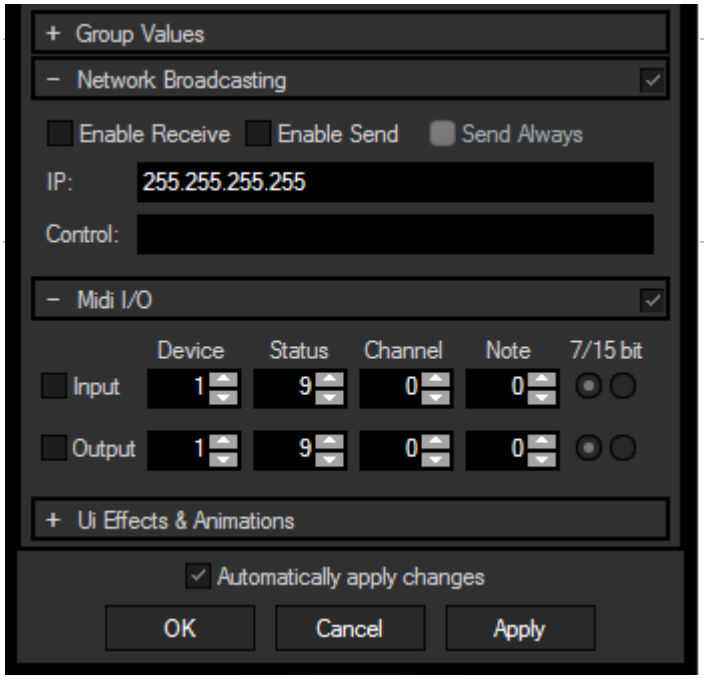
Parameter:
This drop-down offers you all available Device parameters of Pandoras Box. If you want to access an FX parameter, use the FX name and the pipe symbol to specify it (e.g.: "RGB Multiply|Blue"). If the parameter list does not show the desired parameter (in case that you use a custom device) simply type in the name of the parameter here.

Minimum / Maximum:
Enter here the minimum and maximum value your Wheel should have, as well as the **Default** position.

Factor:
Enter here a factor with which the step size should be multiplied. If you set a factor of 7, each movement of the Wheel will increase/decrease the value by 7 instead of 1.

Throttle:
The Throttle parameter limits the amount of data transmitted to Pandoras Box. The higher this value is, the fewer times the Device's parameters are updated. Throttling the data can help saving performance, but also reduces smoothness.

Mouse Wheel Input:
If you check this box, the Wheel will react to your mouse's scroll wheel when hovering over the widget.



Group Values

This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Network Broadcasting

The NBS (Network Broadcast Service) allows to transmit and update Faders, Labels and Custom Script Buttons across multiple WD Designers instances on the network.

To activate this service, please refer to the Remoting Menu [1273]!

Enable Send
To send the Wheel values as broadcast into the network, you only have to check "Enable Send".

Enable Receive
To receive values from other Wheels, please check "Enable Receive".

Now you have to specify which item should update your Wheel:

Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the Wheel through another item on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the item you want to take the values from. This could e.g. be "Wheel1" or "Wheel2".

Once all values are set, the Wheel is ready to be used or edited at any time.

Midi I/O

The Wheel widget is able to receive Midi Notes and / or send its own value as Midi note velocity. A Midi input / output device has to be connected via the Connection Manager [716] or the Midi Connection Manager.

For sending and receiving Midi, please select which one you would like to do and enter the respective settings:

Device:
If you have several connected Midi devices, enter here the ID of a specific device assigned in the Midi Connection Manager.

Status / Channel / Note:
Specify the values of the note you want to listen to or send. The values are numeric, so note E4 would be represented by 64. Please note that different translations of Midi notes can exist for different devices, so you might need to add or subtract an octave.

7/15bit:
Some systems support 15 bit Midi values instead of 7bit, select the respective radio box here.

Ui Effects & Animations

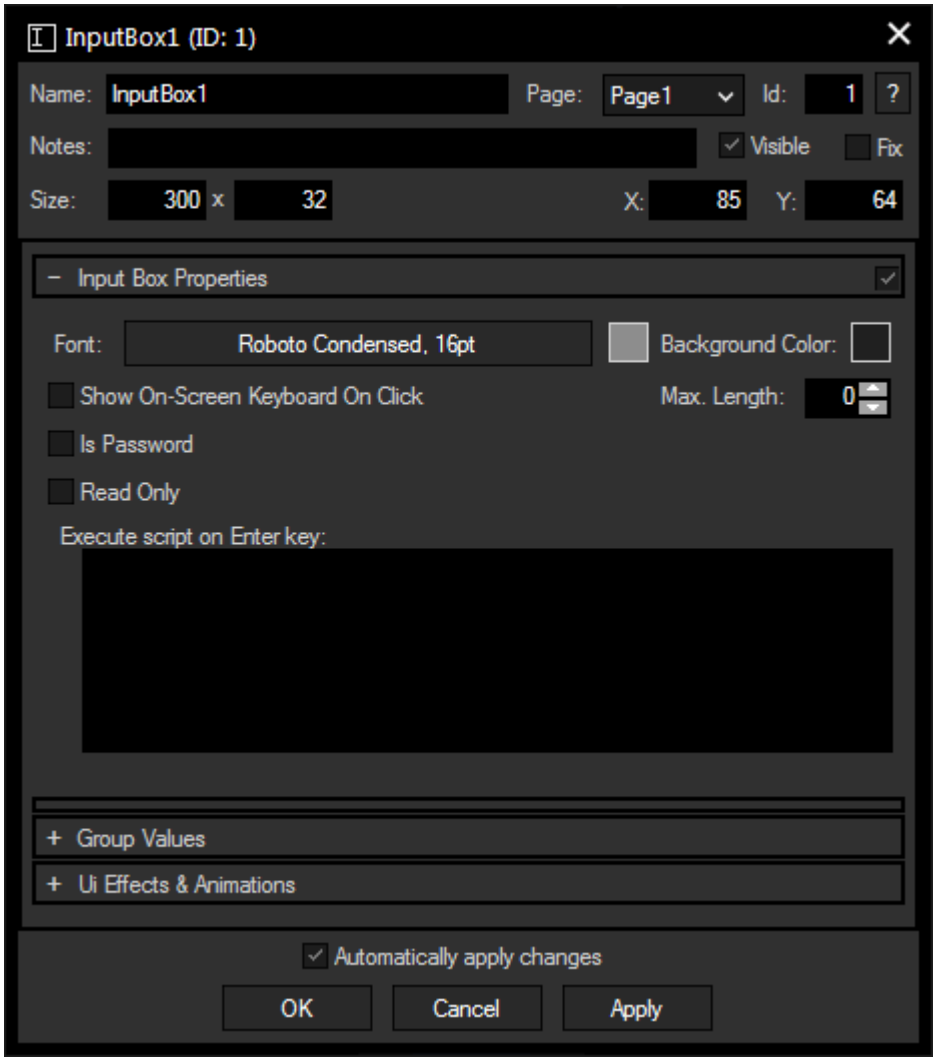
The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.8 Input Box

The InputBox enables you, like the TextBox, to enter text and retrieve the string for further use with the respective member [1904] or node [936].

To create an InputBox widget choose "Widgets > InputBox". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the InputBox properties simply right-click it and choose the first menu entry "InputBox Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The InputBox property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The InputBox' ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the InputBox will be displayed on every page.

Size:
Enter a pixel size for the InputBox' size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Input Box Properties

Font:
Edit the InputBox text font and the font size by clicking on the button with the current font.

To change the text color click in the small box on the right side.

Background Color:
Click the box to open a color picker dialog for the background color.

Show On-Screen Keyboard on click:
Check this box to open the integrated on-screen keyboard as soon as a click is performed inside the InputBox. This is especially useful for touch applications such as tablet PCs or touch monitors.

Is Password:
This check box turns all entered characters to dot characters.

Read Only:
Check this box to prevent the InputBox text from being changed manually. Changing the text per script is still possible.

Max.Length:
Enter the maximum length of the entered character string. "0" stands for an infinite number of characters.

Execute script on Enter key:
Enter a script that will be executed when the [Enter] key is pressed. Functions and Macros [1897] are a good option to manage large and sophisticated scripts.

Group Values

This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

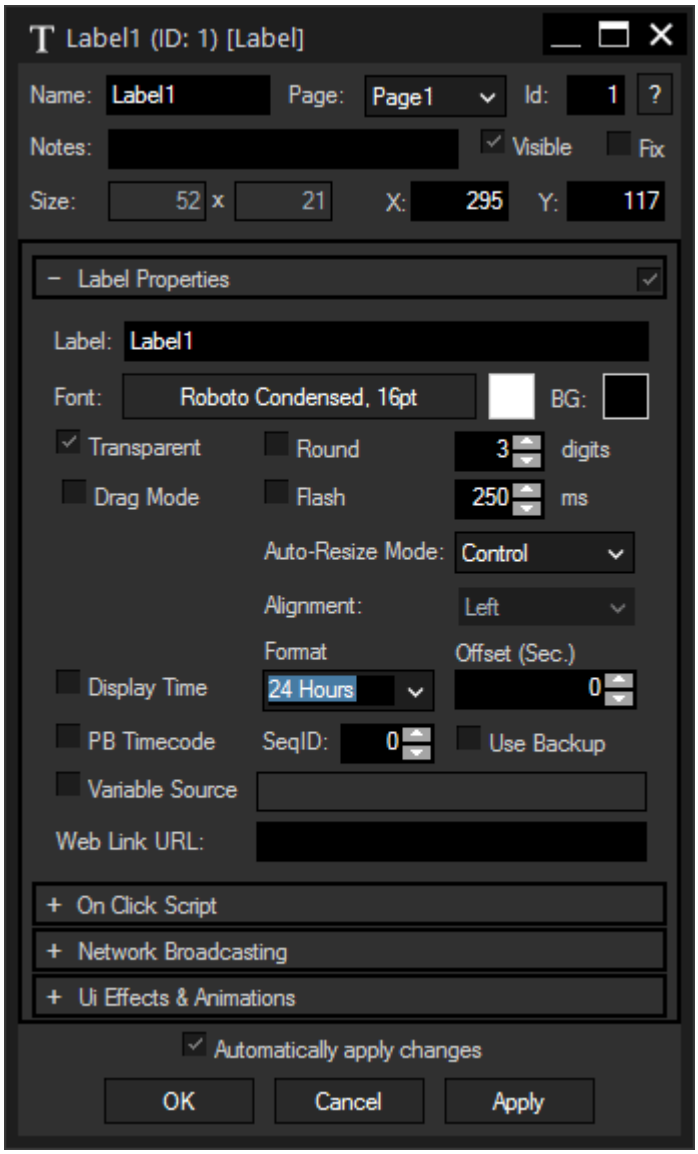
7.3.9 Label

The Label control lets you add text labels to your user interface.

To create a Label widget choose "Widgets > Label". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Label properties simply right-click it and choose the first menu entry "Label Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Label property dialog opens up.

General Widget Settings



Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Label's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Label will be displayed on every page.

Size:
Enter a pixel size for the Label's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Label Properties

Label:
Edit the Label text by entering it in the text field.

Font:
Edit the Label font and the font size by clicking on the button with the current font.

To change the Label text color click in the small box on the right side.

BG:
The background color is transparent by default. If you want to underlay the text with a color, uncheck "**Transparent**" and choose a color for the background by clicking in the small box.

Round:
Check this box if you want to display floating point values with a reduced amount of digits. Enter the desired amount of digits in the field at the right.

Flash:
When this option "Flash" is checked, the Label will flash (visible / not visible). The flash interval time is set to 250 ms by default. To increase or decrease the flashing interval time, change the value here or use the command WDLabelFlashInterval(ID,Value) [1749].

Auto Resize Mode:
Choose here one out of three modes to correlate the widget's size and the font:
Disabled: The size of the widget is adjustable, the text will not be adjusted automatically. Text that overlaps the borders will not be displayed.
Font: The font will be adjusted automatically to the widget's size so that the text always fits in.
Control: The widget's size will follow the text, it cannot be adjusted manually except by setting the font size.

The options "Disabled" and "Font" also offer the possibility to set the text alignment within the widget to "**Left**", "**Center**" and "**Right**".

Display Time:
Choose this option to display the local computer's time, either in the "24 Hours" or in the "AM/PM" **format**. An **offset** can also be set if necessary.

PB Timecode:
If a Pandoras Box is connected, the Label can be used to display the timecode of a sequence. Specify the **Sequence ID** in the respective field and check "**Use Backup**" if you need to backup machine's timecode instead of the main machine. Please see also the dialog PB Network Configuration [1256].

Variable Source:
If the value of a variable [1900] should be displayed as Label text, check the option "Variable Source" and enter the name of the variable. If the variable's value is going to be updated, the Label text will be updated as well.

If one of the options "Display Time", "PB Timecode" or "Variable Source" is checked, the Label's text will be overwritten by those values.

Web Link URL:
The Web Link section is of special interest when working with the Web Server [1929] feature, i.e. clicking the Label in a web browser. Enter a URL e.g. "https://www.christiepandorasbox.com" that your browser should call when executing the On Click Script.
It is also possible to achieve a quick page change with the URL, simply enter a hash tag "#" and the page name. E.g.: #Page2

On Click Script:
Enter a script that will be executed when clicking on the Label. Functions and Macros [1897] are a good option to manage large and sophisticated scripts.

Network Broadcasting

The NBS (Network Broadcast Service) allows to transmit and update Faders, Labels and Custom Script Buttons across multiple WD Designers instances on the network.

To activate this service, please refer to the Remoting Menu [1273]!

Enable Send:
To send the labels values as broadcast into the network, you only have to check "Enable Send".

Enable Receive:
To receive values from other faders, labels or Custom Script buttons, please check "Enable Receive".

Now you have to specify which item should update your label:

Enter the **IP address** of the computer you want to listen to. This could be the local computer (to control the fader through another item on the same Widget Designer) or a different computer in the network. If you do not want to specify the computer but want to listen to all computers in the network, enter "255.255.255.255".

As next step specify the item you want to take the values from. This could be a fader (e.g. "Fader1" or "Fader2"), a label (e.g. "Label1" or "Label2"), or a Custom Script Button (e.g. "CustomScript1" or "CustomScript2" – you will be able get its status: 1/0).

Group Values

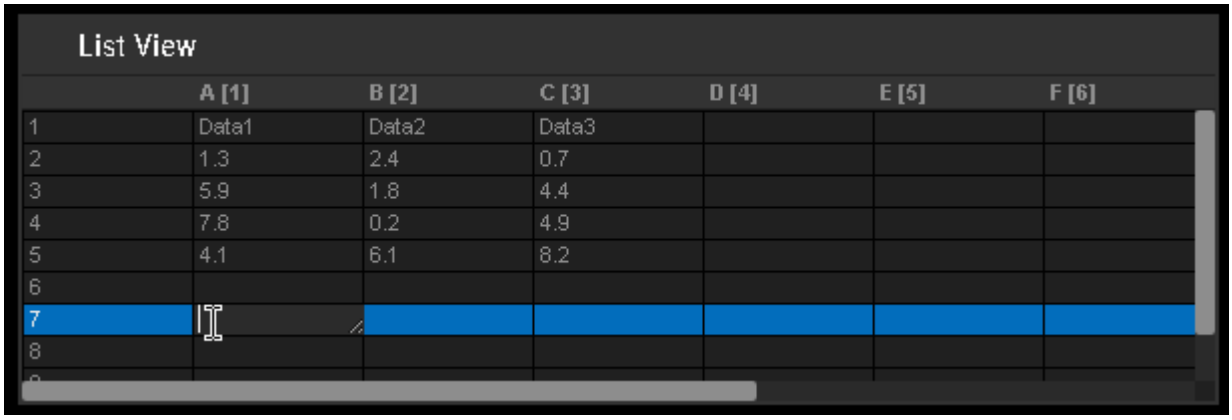
This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.10 List View

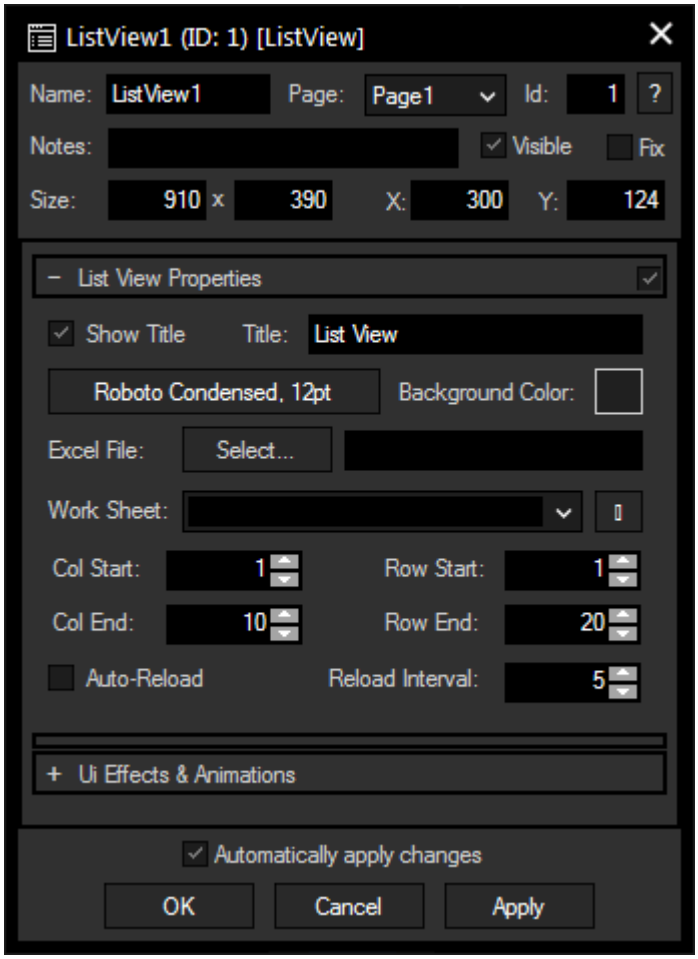
A List View is a control element that displays data in a table with rows and columns. To fill the List View with a new database, you can either import it from other sources (e.g. Microsoft Excel) or create it in Widget Designer. Furthermore in WD, you can edit every cell, add new rows or columns, or use one of the offered WD table operations. You can use the cell values for your Widget Designer programming, e.g. through variables.



To create a List View widget choose "Widgets > ListView". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the ListView properties simply right-click it and choose the first menu entry "ListView Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The ListView property dialog opens up.

General Widget Settings



Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The widgets's unique ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

?:
This displays the chapter about this widget in a newly opened helpfile window.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID. Note the property member [1904].

Visible:
Uncheck the "Visible" option to hide the widget.

Fix:
Check the option "Fix" to display the widget on every page. If the option is unchecked (again) the widget is only displayed on the page chosen with the Page drop-down list.

Size:
Enter the width and height for the widget in pixels.

X and Y:
Enter the position in pixels of the widget on the page in pixels. 0,0 is the top left corner of the Page and the position refers to the top left corner of the widget.

List View Properties

Show Title and Title:
The check box toggles the visibility of the title at the very top of the List View.
Column Names
The commands `WDListViewSetColumnName` and `WDListViewSetRowName` let you assign names for columns and rows.

Font and Background Color:
Click the Font button to choose another font and style for all cells including the column and row names. For a new color, click the Background Color button to open the Color dialog .

Excel File, Worksheet, Reload button and Saving:
In case you like to import an existing database from an Excel Sheet, enter the path (or select one with the [...] button) and select one of the loaded names from the sheets. With the Reload button, you can reload this file, e.g. if you like to undo changes made in Widget Designer after loading the file.
The supported import formats are: XLS or XLSX as well as CSV for comma separated values and TSV for tab separated values.

Alternatively, you can also use the commands `WDListViewLoadCSVFile(ID,FileName,Separator)` to import data from a CSV-file which is especially interesting when another separator than a comma is used. If the file contains tabs to separate cell values, the command to use would be `WDListViewLoadCSVFile(1,"C:`

`\data\file1.csv","\t".Unescape)`. As explained in the chapter "Data Type specific members [1914]", the "Unescape" member is needed to return the object string with the correct formatting when using "\n" (new line), "\r" (carriage return) and "\t" (tab).

If you like to save the table with the current data, you can use the SaveCSVFile member [1904] from the respective ListView. The following script saves the current data in ListView with ID 2 and separates the cells with a semicolon.

```
ListView2.SaveCSVFile("C:\data\file1.csv",";")
```

Please note that Microsoft Excel has to be installed on the computer if you want to use the ListView for importing or exporting files!

Col / Row Start and End:
Pick the start and end column and row if you like to reduce the displayed cells.

Auto-Reload and Reload Interval:
If you would like to update your database in a certain interval, check the option "Auto-Reload" and enter the time in seconds after which your data is updated. Please note, that the file must be saved under the same path.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

Programming with List Views

This is a short description how you can program with a List View.

If you like to manually edit a cell value, make sure to be in the run mode [F8] and double-click into a cell. Enter the new value into the small text field and hit [Enter].

There are commands available that write a value, or set another property from a List View. The description of all WDListView... commands [1760] is included in the command list.

In addition there are various commands that read a value (or column/row) and write it into a(n array) variable. Further, WD offers operations that calculate the sum or average from a column or row. Again, the detailed description of all VGetListView... commands [1634] is included in the command list.

Last, the List View control offers member values, allowing for example the following script. Please find more information in the topic Object and Member Notation [1904].

```
ListView2.SetCell(1,1,"text")
ListView2.SetCell(1,1,123)
ListView2.SetCell(1,1,var1)
ListView2.SetCell(1,1,Math.Pow(var1,2))
var2 = ListView2.GetCell(1,1)
```

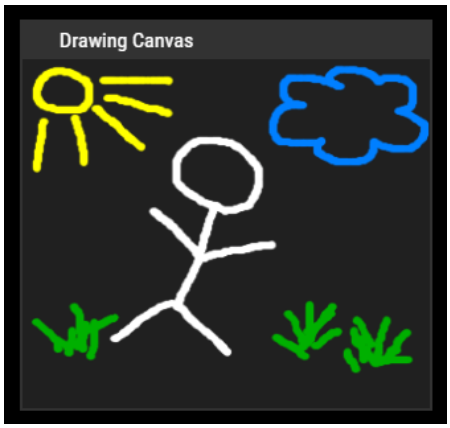
7.3.11 Panel

See here the available panel types:
Drawing Canvas [893]
VNC Panel [894]
XY Panel [896]
MultiTouch Panel [899]

7.3.11.1 Drawing Canvas

The Drawing Canvas lets you draw with the mouse, it can be used as entertaining and creative item on the WD interface.

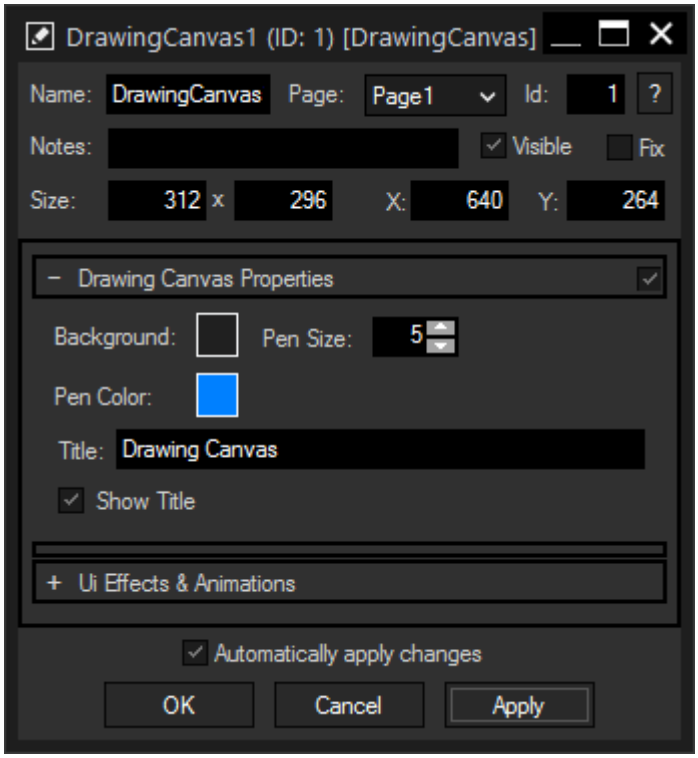
Please note that the Drawing Canvas will be reset on reloading the GUI or on page change.



To create a Drawing Canvas widget choose "Widgets > Panels > Drawing Canvas". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the Drawing Canvas properties simply right-click it and choose the first menu entry "DrawingCanvas Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Drawing Canvas property dialog opens up.

General Widget Settings



Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Drawing Canvas's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Drawing Canvas will be displayed on every page.

Size:
Enter a pixel size for the Drawing Canvas's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Drawing Canvas Properties

Background:
Click on the color button to the right to open a color picker and select the background color. The background color can also be changed via script, e.g. with `DrawingCanvas1.SetBGCol(128,0,255)`.

Pen Color:
Click on the color button to the right to open a color picker and select the pen color. The pen color can also be changed via script, e.g. with `DrawingCanvas1.SetPenCol(255,128,0).`

Pen Size:
Enter here the thickness (in px) of the line you are drawing.

Title:
Enter a title for your Drawing Canvas and display it with checking the **Show Title** box.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.11.2 VNC Panel

The VNC Panel control lets establish a VNC session to a remote computer and have full remote access.
It is also possible to host multiple VNC sessions within one page of the interface.

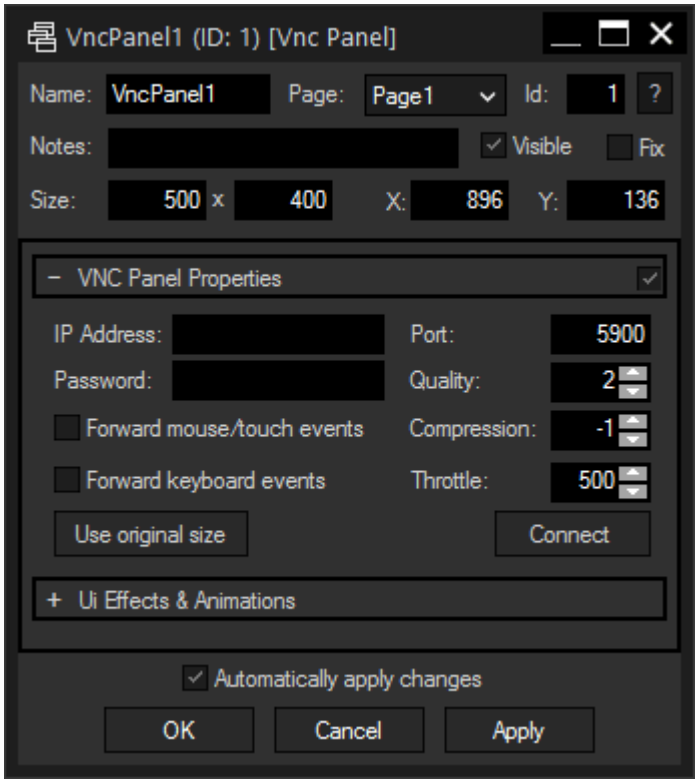
Please bear in mind that transmitting a desktop stream uses up resources in your network and can lead to a high amount of traffic, especially when several sessions are open at the same time!
A VNC server, e.g. the PB Menu [2097], needs to be running on the remote PC.

If you want to remote into different remote computers, you can set up button scripts to automatically change the IP address and connect to different places.

To create a VNC Panel widget choose "Widgets > Panels > VNC Panel". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the VNC Panel properties simply right-click it and choose the first menu entry "VncPanel Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The VNC Panel property dialog opens up.

General Widget Settings



Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The VNC Panel's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the VNC Panel will be displayed on every page.

Size:
Enter a pixel size for the VNC Panel's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

VNC Panel Properties

IP Address:
Enter here the IP address of the remote PC to be accessed via VNC.

Port:
Enter the port for your VNC connection if it deviates from the standard port 5900.

Password:
If your VNC server requires a password for allowing a connection, you can enter it here.

Quality:
The higher this value (min 0, max 9), the higher the quality of the desktop stream.
A higher quality results in higher network traffic!

Forward mouse/touch events:
Check this box if you want to be able to perform mouse events on the remote computer.

Forward keyboard events:
Check this box if you require keyboard data to be sent from your local to the remote computer.

Throttle (ms):
To save network resources, it is recommended to throttle the stream to limit the transmitted frame rate. Keep this value as high as possible unless you need a more fluid stream.

Use original size:
Press this button to apply the size of the desktop stream to the VNC Panel and view the remote computer with its native resolution.

Connect / Disconnect:
Press this button to start or stop the VNC session. This action can also be performed from a script, like e.g. `VncPanel1.Connect` and `VncPanel1.Disconnect`.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.11.3 XY Panel

The XY Panel represents a graphical touch pad with relative or absolute value controls. It can either be used to directly link Pandoras Box Device parameters to the generated position data, or for retrieving position values for further processing.
If the Mouse Over Mode is not selected, position data is only sent upon a mouse left-click.

To create a XY Panel widget choose "Widgets > Panels > XYPanel". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the XY Panel properties simply right-click it and choose the first menu entry "XyPanel Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The XY Panel property dialog opens up.

General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The XY Panel's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the XY Panel will be displayed on every page.

Size:
Enter a pixel size for the XY Panel's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

XY Panel Properties

Devices:
Enter the Site and Device ID of the Device that shall be controlled by the XY Panel. Multiple Devices can be entered too, with a white space as separator (e.g.: "2.1 2.2 2.17")

Mouse X / Mouse Y:
The following properties are applied for the X as well as for the Y coordinate of the mouse within the XY Panel.

Distance / Angle:
If you are using an external multi-touch device such as an AirScan [1277] for controlling the XY Panel, distance and angle of two active points can be used for interacting with Pandoras Box, too.

Name:
Select the Pandoras Box Device parameter to be controlled by the XY Panel from the drop-down.

Default:
Enter here a default value to which the XY Panel shall be reset. A reset can be triggered e.g. via the command `XYPanel1.ResetDefault`.

Min / Max:
These are the values referring to the absolute minimum and maximum values the XY Panel can produce. The default direction for the X coordinate is from left to right and for the Y coordinate from top to bottom. The direction can be reversed by switching the Min and Max values.
This setting is only available for the **Absolute** mode (see below).

Factor:
Apply a factor to affect the strength of the value increment based on the mouse input motion when you are using **Relative** mode.

Mode:
Select the **Absolute** mode from the drop-down if you want your Pandoras Box Device to directly correspond to the absolute values delivered from the XY Panel.
Select the **Relative** mode if you want to add the XY Panel values to the current Pandoras Box Device values.

Throttle (ms):
Sliding movements on the XY Panel creates a high amount of values being transmitted to Pandoras Box. To reduce the amount, e.g. if you experience a delay or jitter in the PB response, you can set up a throttle that limits the value output to a certain interval.

Reset Values On Page Change:
The entered **Default** values are sent on page enter when this option is checked.

Mouse Over Mode:
Check this box if you want the XY Panel to mouse over events instead of just reacting upon left-click.

Show Info:
Uncheck this box if you do not want the position data to be displayed on the widget.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

Script

In the Script section you may enter commands to be executed. You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

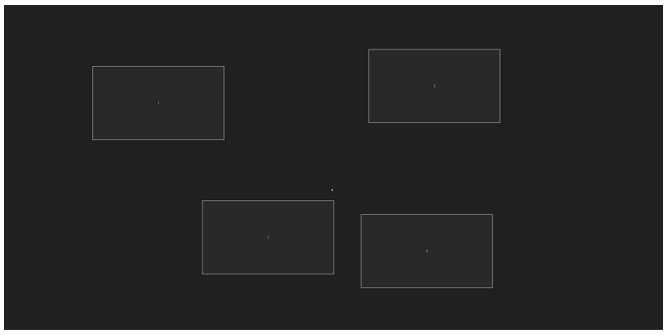
The scripts for Press, Release, Mouse Enter and Mouse Leave are executed when the Widget Designer button is being used, on the local interface as well as on any web client.
You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts
If you are interested in the Web Server feature and some small examples, please read the topics Web Server [1929] and Object and Member Notation [1904] .

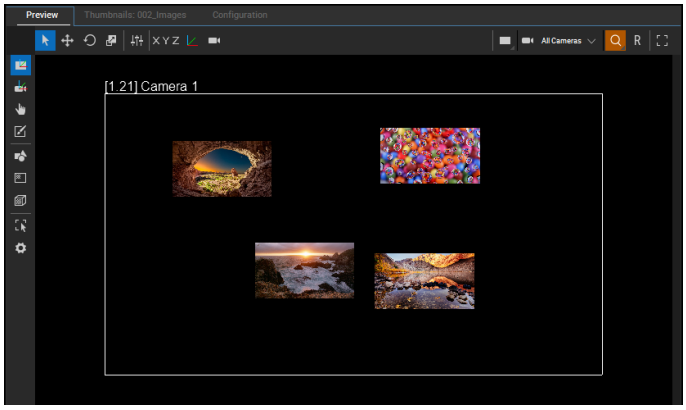
See here a list of all commands [1520].

"On Click" is processed when a mouse left button down click on the panel control is being performed by the user.
"On Release" is processed when a mouse left button up click on the panel control is being performed by the user.
"On Mouse Enter" is processed when the mouse cursor enters the panel control.
"On Mouse Leave" is processed when the mouse cursor leaves the panel control.

7.3.11.4 Multitouch Panel

The MultiTouch Panel allows direct control of up to 48 layers in PB as interactive touch surfaces.





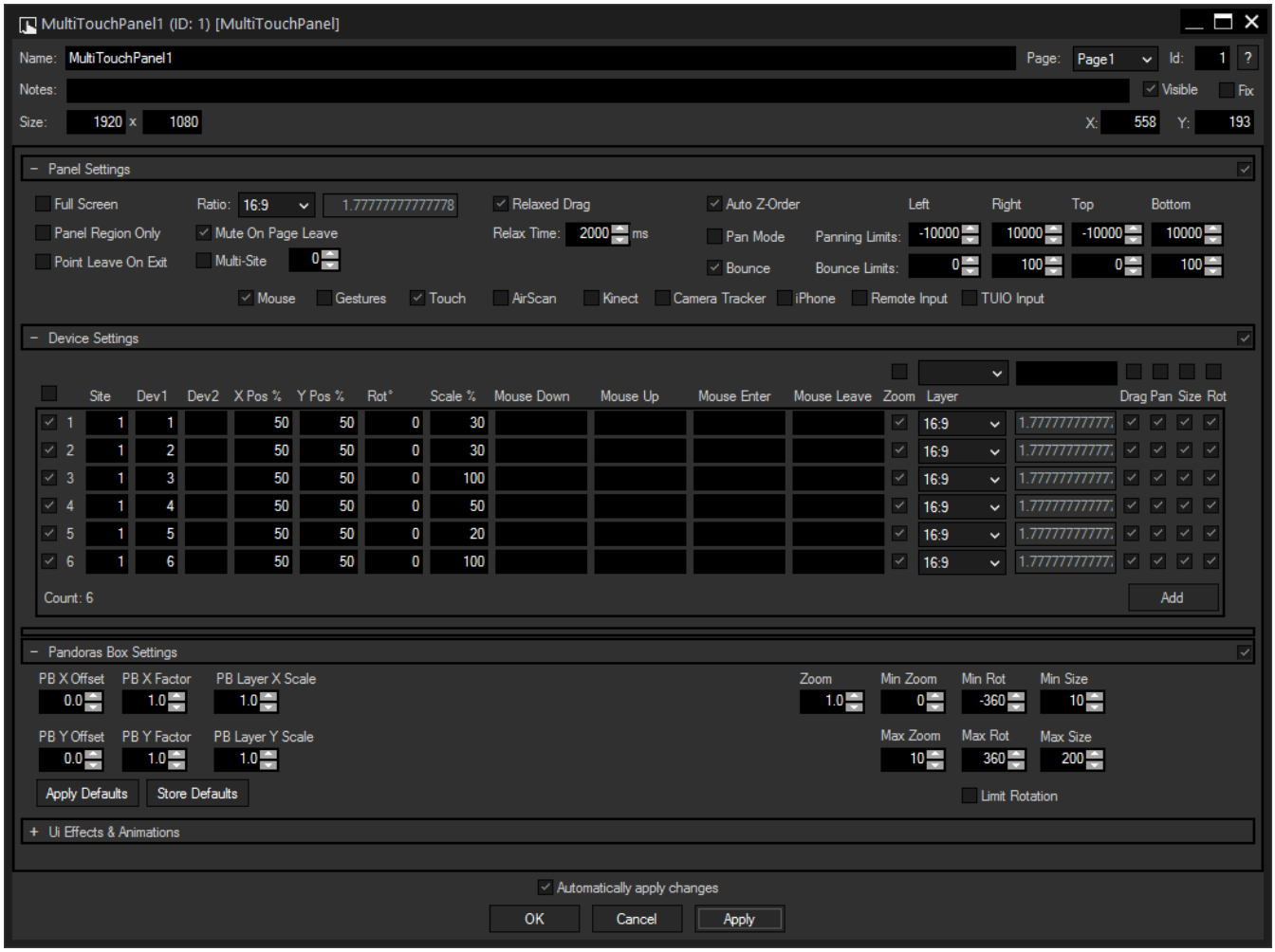
Any changes done to these items in the MultiTouch Panel (in the example above numbered 1-4), like moving, scaling or rotating them will be applied to the Layers in Pandoras Box as well:

See here information about the Item Properties of the MultiTouch Panel [899].

7.3.11.4.1 Multitouch Item Properties

To create a MultiTouchPanel widget choose "Widgets > Panels > MultiTouchPanel". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets. Most widgets however need to be set up first.

To edit the MultiTouchPanel properties simply right-click it and choose the first menu entry "MultiTouchPanel Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The MultiTouch Dialog opens up. Don't worry if this takes a bit longer than opening the Item Properties of other controls.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The VNC Panel's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the VNC Panel will be displayed on every page.

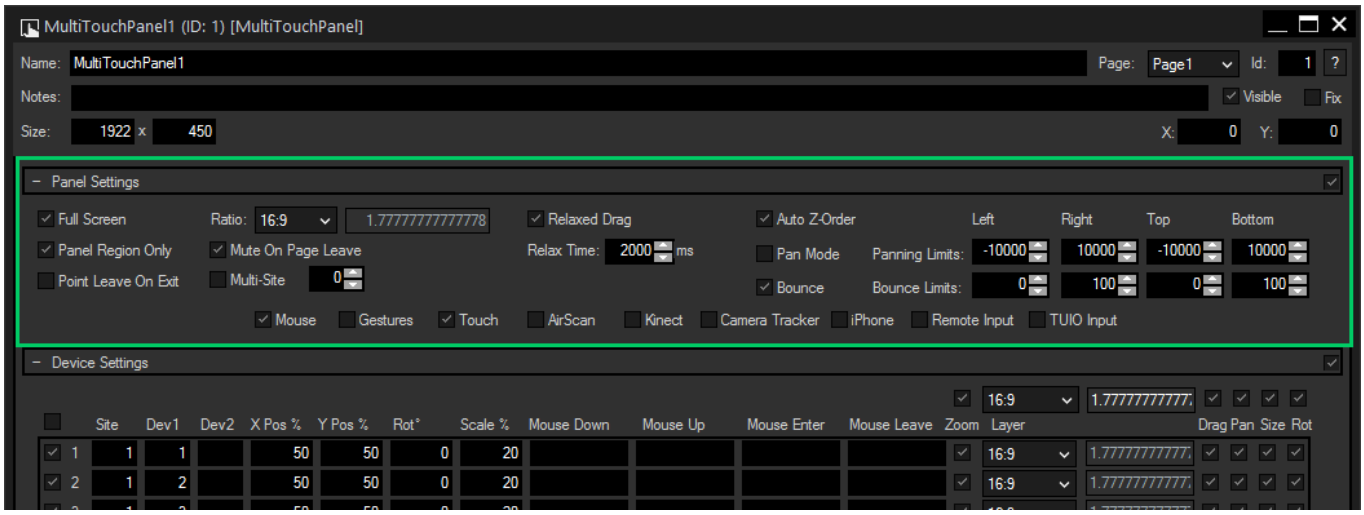
Size:
Enter a pixel size for the VNC Panel's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

The Item Properties dialog is divided into three different sections:
Panel Settings, [901]
Device Settings [905],
Pandoras Box Settings. [909] and
Effects & Animations. [814]

7.3.11.4.1.1 Panel Settings

This chapter explains the Panel Settings from the Multitouch Panel [899] in Widget Designer.



Full Screen
Check this option so that the MultiTouchPanel covers the full screen of your WD computer.

Ratio
Please choose between the Display Modes Custom, 16:9 and 16:10, 4:3, 1:1 according to the aspect ratio of your WD computer screen. This is needed to have the devices in the correct aspect ratio inside the MultiTouch Panel.

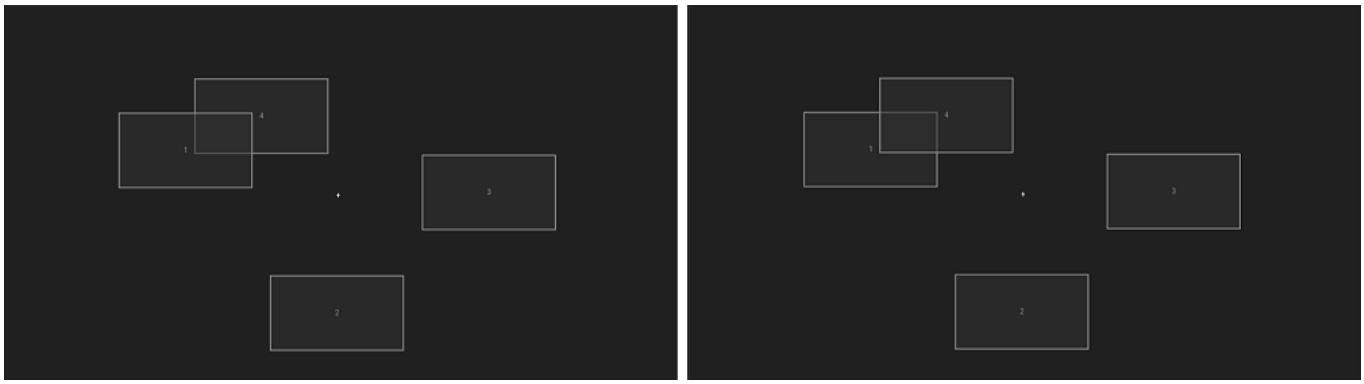
Relaxed Drag
When Relaxed Drag is enabled, an item will not stop hard at a position to which it was dragged, but it will run out smoothly depending on the dragging speed. This option is active by default.

The Relax time (in ms) and the maximum Relax Delta (in px) may be adjusted in the text fields below "Relaxed Drag". The higher the Relax time, the longer the Item will be dragged. The Relax Delta defines the max. distance in px the Item will be dragged.

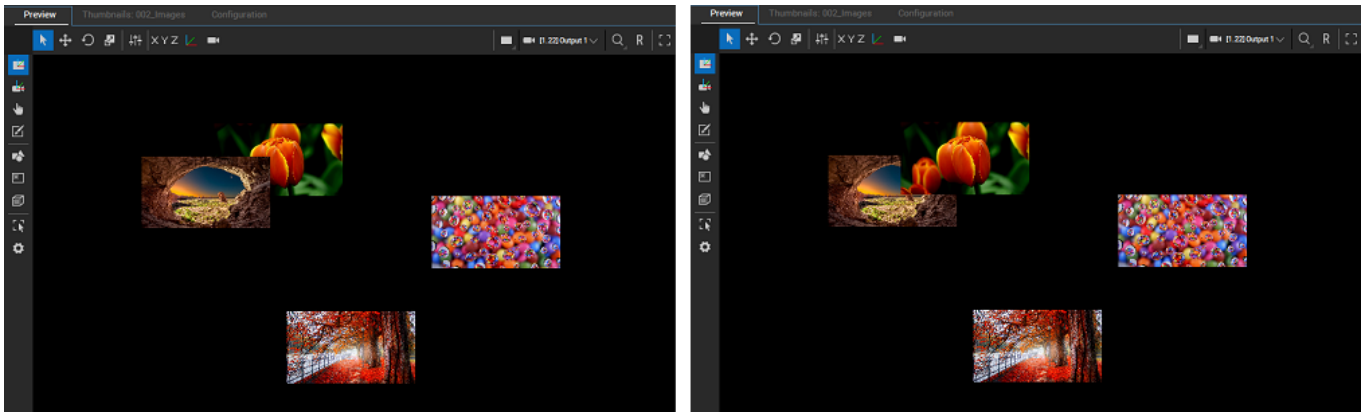
Auto Z-Order
When this option is checked, the selected item inside the MultiTouch Panel will automatically be placed in front of other items inside Pandoras Box (if one item overlays another one). This option influences the Z Position of the layer in PB the devices are linked to. Do not use Auto Z-Order, if you have assigned non-default Z Positions to the layers manually.

Please note: this option is not working when controlling former PB products which do not offer the Z-axis.

Example



Left: Item1 is in front of Item4. **Right**: After selecting Item4 the Z Order changes and Item1 is behind Item4.



Left and right: see this effect in PB.

Panel Region Only

Mute On Page Leave

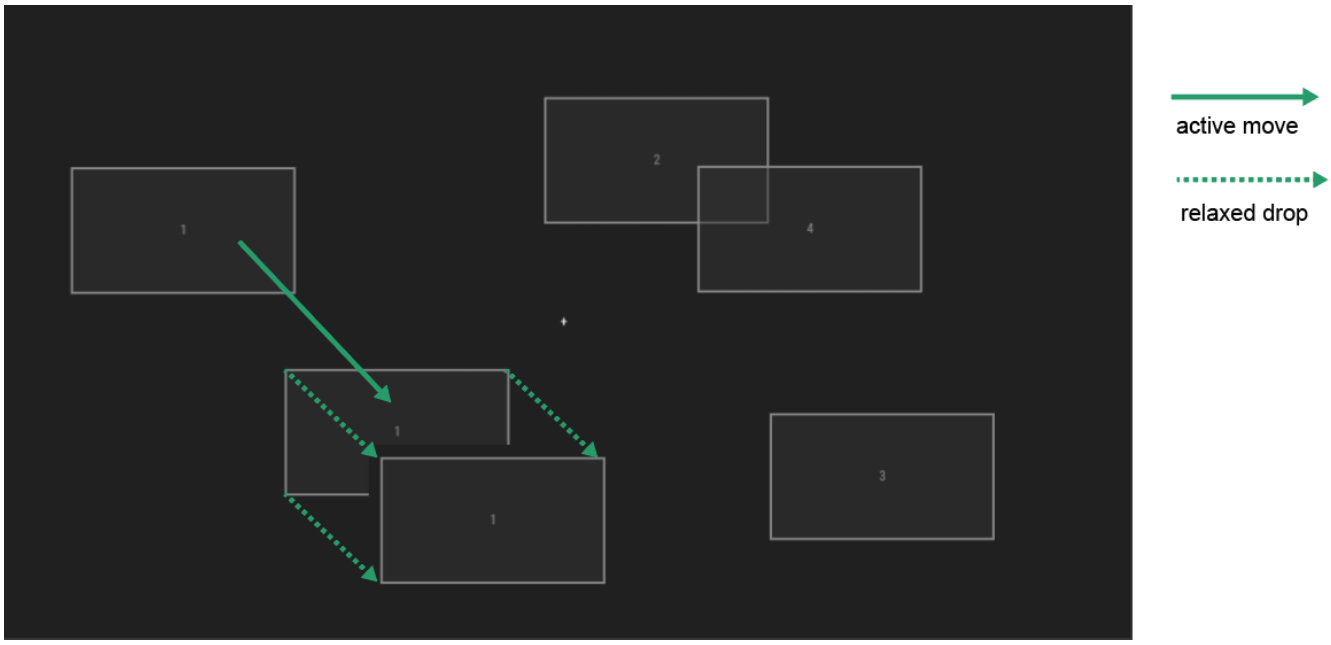
Point Leave on Exit

Multi Site
The Multi Site Mode allows controlling multiple Servers simultaneously. This is useful when virtual sites can not be used. Check this option and enter the amount of Sites the position and scale data of the MultiTouch Items should be routed to.

Example
MultiTouch Item 1 refers to Site1, Layer1. When Multi Site Mode with 2 Sites is enabled, the data of this Item will now refer to Site1, Layer1 and Site2, Layer 1 as well.

Relax Time
When Relaxed Drag is enabled, an item will not stop hard at a position to which it was dragged, but it will run out smoothly depending on the dragging speed. This option is active by default.

Example

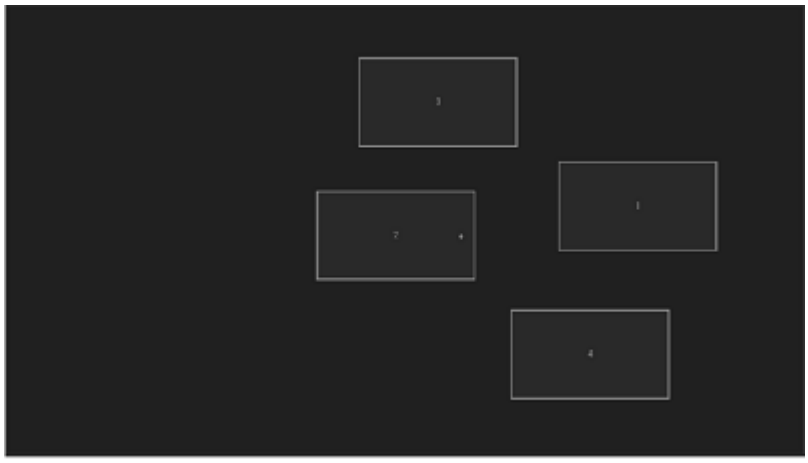


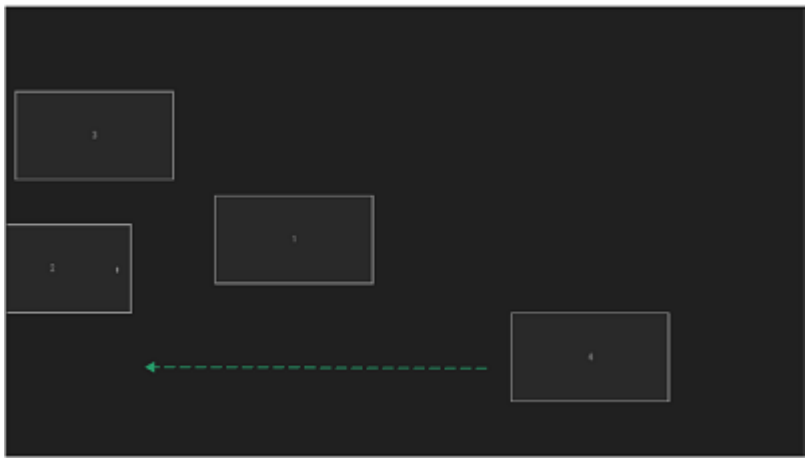
The Relax time (in ms) and the maximum Relax Delta (in px) may be adjusted in the text fields below "Relaxed Drag". The higher the Relax time, the longer the Item will be dragged. The Relax Delta defines the max. distance in px the Item will be dragged.

Pan Mode
The Pan Mode needs to be checked if you want to move items on the MultiTouch area or out of it by clicking in the background and proceed a move there. This will then apply to all items that have the Pan option enabled.

Please note:
Pan is only working when option [Enable Mouse] is checked inside the MultiTouch Panel and when your device (Mouse, Kinect, AirScan, Multitouchdevice etc.) controls the mouse of your WD computer.

Example





Pan-Mode is enabled for the MultiTouch Panel and for the Items 1, 2 and 3, but not for Item4.

The two pictures above show the result after clicking in the panel's background and moving the mouse over to the left side:

Item4 stays at its previous position while the Items 1,2 and 3 are moved to the left side as well.
The little + in the panel's background shows you the PAN position.

Panning Limits Left / Right / Top / Bottom
The values inside the text fields define how many percent the MultiTouch area can be moved in each direction when PAN is enabled.

Example
If you change the value for Top and Bottom to 0 (%) you won't be able to move the MultiTouch area up and down, but only to the left and right side.

Bounce
When the Bounce Mode is enabled, an Item running out smoothly with relaxed drag will bounce back from the display border and reverse its direction towards the display region.

Bounce Limits Left / Right / Top / Bottom

Input Sources

Important notice regarding Multi-Touch Input on Win7:
The Tablet Service must be active in order for Win7 to provide the multi touch data!

Mouse
Enable the mouse input if you want to use the mouse to control the MultiTouch items.

Please note:
The Mouse Input needs to be enabled when you want to use the Pan Mode!

Gestures
Enable this option if you want to use Synaptics Touch Pads and Wacom Bamboo Devices.

Touch
Enable this option if you want to use touch devices like (Multi-)Touch Monitors.

AirScan
Enable this option if you want to use the AirScan [1277].

Kinect
Enable this option if you want to use the Kinect [1283].

Camera Tracker
Enable this option if you want to use the Camera Point Tracker [1291].

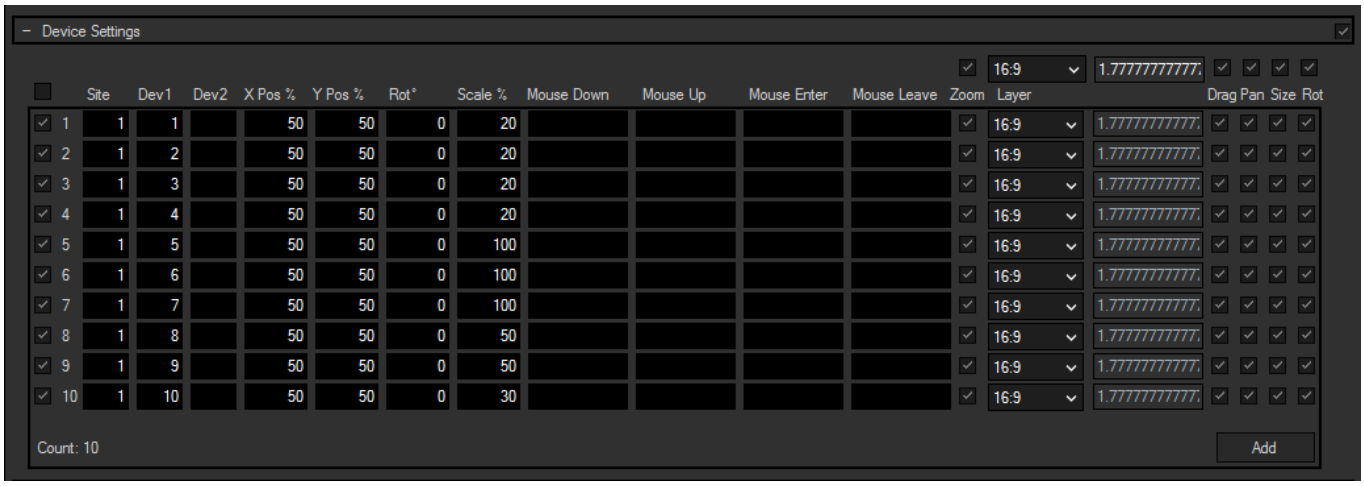
iPhone
Enable this option if you want to use your iPhone with the Widget Designer Remote App to control the MultiTouch items.

Remote Input
Enable this option if you want to use Remoting. [1273]

TUIO Input
Enable this option if you want to send the touch-data via the open source protocol TUIO [1258].

7.3.11.4.1.2 Device Settings

This chapter explains the Device Settings from the Multitouch Panel [899] in Widget Designer.



Check box in front of each line
To enable items in the MultiTouch Panel, that will control a layer in Pandoras Box, you have to check the small box in front of a line. To enable all possible 48 items at once, check the small box that is located above all others.

Device (Site/Dev1/Dev2)
To link a MultiTouch Item to a layer in Pandoras Box, enter the Site ID (Site) and the Layer ID (Dev1).

Example If MultiTouch item 1 should control Layer 2 of site 1 (number in front of the Layer is 1.2), please enter 1 in the first text field and 2 in the second one.

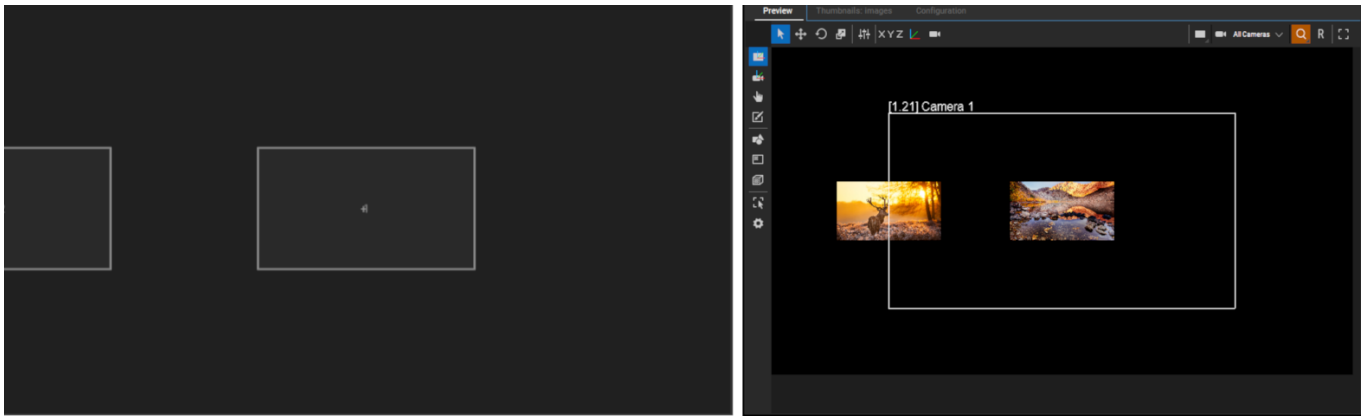
You may link a second layer to the same MultiTouch Item – for example a shadow that is on a separate layer – enter the Layer ID into the Dev2 text field.

Example If MultiTouch item 1 should control Layer 4 and 5 of site 1, please enter 1 | 4 | 5.

X Pos and Y Pos
The X and Y position values are percentage values.

Example
Having an item (1) inside the MultiTouch Panel with the value 50 for X and Y Position means that its center point is located at 50% of the whole Panel's X and Y range. The center point of an item (2) with the values X=0 and Y=25 is, according to this, located leftmost and in the upper quarter of the panel, see image below at the left side.

As the whole MultiTouch Panel (no matter up to which size you scale it) always represents the visible PB output area (if no Cam Z Position is changed) these values will be transferred 1:1 to the PB fullscreen output, see image below at the right side.



Left image: Multi Touch Panel: Item1 is located 50,50(X,Y), Item 2 in 0,50(X,Y).

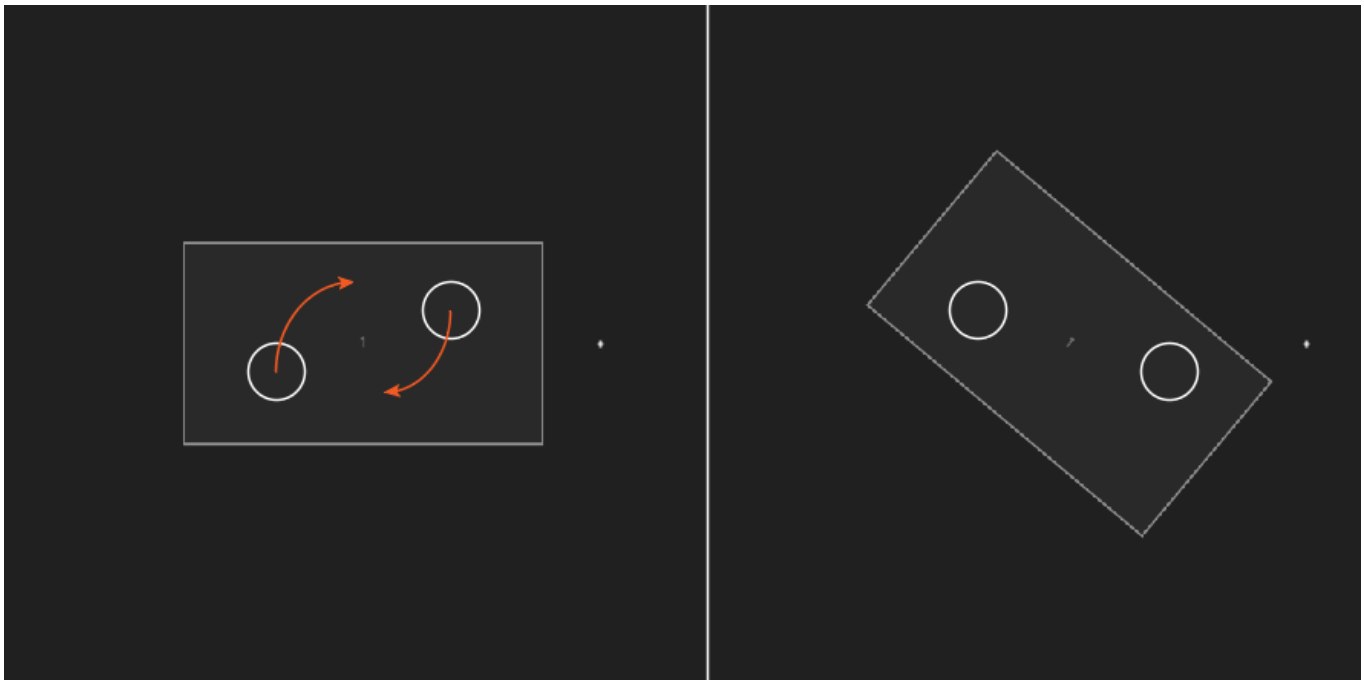
Right image: Pandoras Box Output: The two layers match perfectly to the item's positon and scaling just as in the Multi Touch Panel.

Rot°
The rotation value of an item (in °) can be entered manually in this text field.

The interactive rotation of an item will be achieved when using a multi-point / -touch device.

Enter the Item with two touch points and change the angle between them, see picture below.

Please note: the option [Rotate] has to be enabled for this Item!

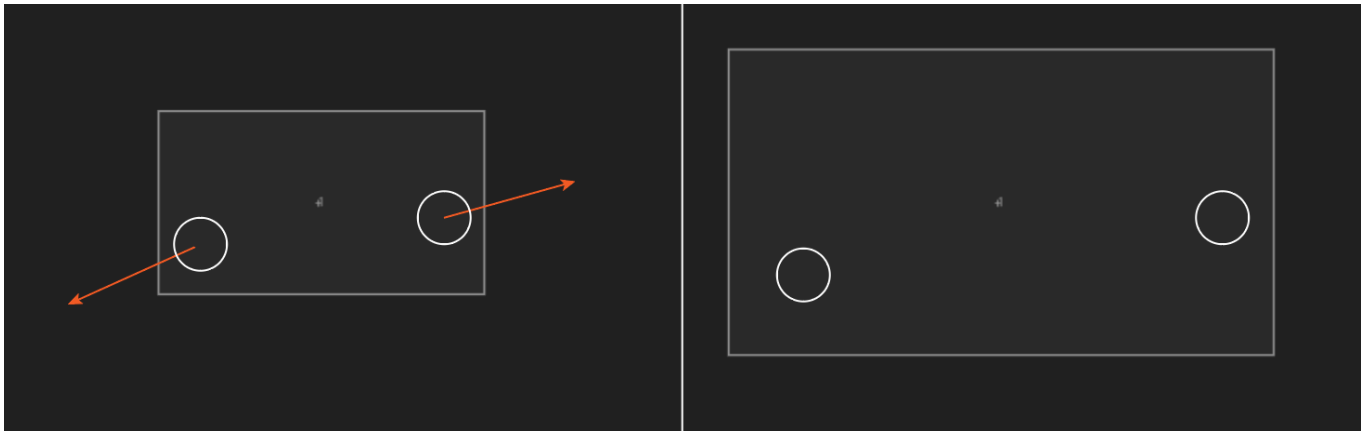


Scale %
The scale of an item (in %) refers, same as the position values, to the whole MultiTouch Panel area and therefore to the PB Output area. The item in the example below (left side) has the scaling value 30, so it covers 30% of the width and the height. An item with the scaling value 100 covers the whole screen.

The interactive scaling of an item will be achieved when using a multi-point / -touch device.

Enter the Item with two touch points and change the distance between them, see picture below.

Please note: the option [Size] has to be enabled for this Item!



Mouse Down / Up / Enter / Leave

The MultiTouch Panel offers script processing for different mouse interactions.

"Mouse Down" is processed when a mouse left button down click on the item is done.
"Mouse Up" is processed when a mouse left button up click on the item is done.
"Mouse Enter" is processed when the mouse cursor enters the item.
"On Mouse Leave" is processed when the mouse cursor leaves the item.

Any command that is available from the command list [1520] may be used, just enter it in the text field. Multiple commands (i.e. one script) can be executed by delimiting every command with a carriage return at the end of the command. As these small text fields do not offer a good overview when using multiple commands, working with

functions or macros [1897] is a good option. The topic Script Language [1511] explains commands and how to use them in more detail.

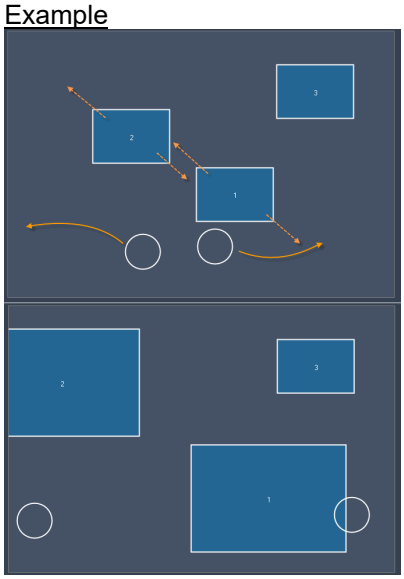
Zoom
The Zoom option applies to the whole MultiTouch Panel and to all Items that have the Zoom option enabled.

The interactive zooming will be achieved when using a multi-point / -touch device.

Enter the MultiTouch Panel with two touch points and change the distance between them.

Please note: this only works for Items that have the option [Zoom] enabled. All other items will keep their original size.

Example



Item 1 and 2 have the Zoom option enabled, Item 3 doesn't.

So after zooming the MultiTouch Area with two touch points, the zoom is applied to Item 1 and Item 2. Item 2 keeps its former size.

The Zoom factor may be changed via commands as well:
WDMultiTouchSetZoom(ID,Zoomfactor) [1777]

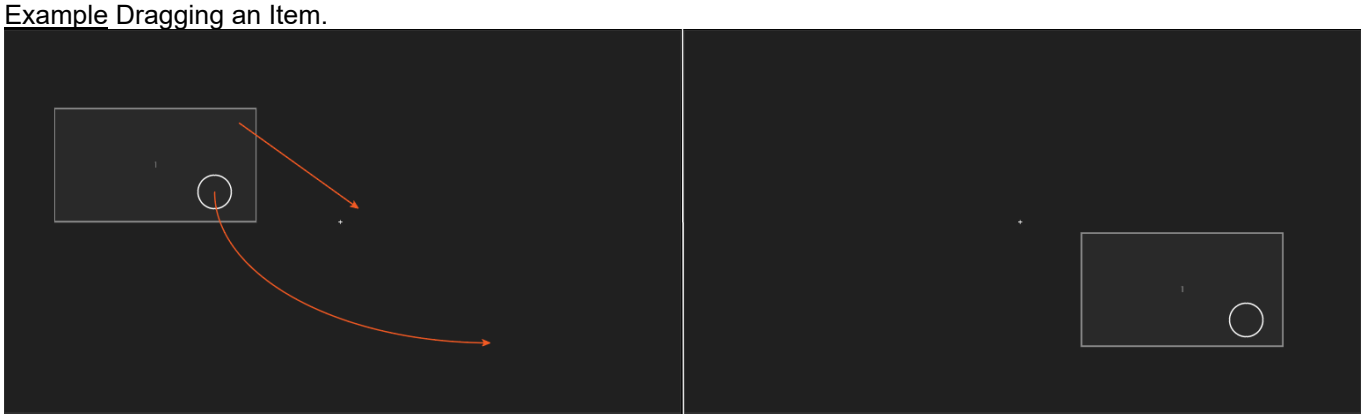
Layer
Please choose the Aspect Ratio for the item (Custom, 16:9, 16:10, 4:3 or 1:1), according to the aspect ratio of the file assigned to the layer. This is needed to display the media file with its correct aspect ratio in Pandoras Box.

If all media files on the layers in Pandoras Box have the same aspect ratio just check the small box on top to apply this aspect ratio to all items.

Drag
To move an item on the MultiTouch Panel you need to have the mouse clicked on this Item or place a touch point on it. Moving the mouse / touch point will drag the Item. See example below.

Please note: The option Drag needs to be enabled for the Item.

Example Dragging an Item.



Pan
Enable Pan if you want to move an item on the MultiTouch area or out of it by clicking in the background and proceeding a move there (instead of moving the item itself directly). This allows moving several items that have the Pan option enabled at once.
The Pan Mode in the Panel Settings as well as [Enable Mouse] needs to be checked!

--> See an example for Pan in the section "A. Panel Settings"!

Size
Enable Size if you want to scale an item with a support device that provides two (mouse-) points (e.g. the AirScan) instead of a single touch (e.g. the mouse). By changing the distance between the two points you will increase or decrease the item's scaling when the item is selected.

--> See an example for Size in the section "B. Device Settings" under the cue "Scale"!

Rotate
Enable Rotate if you want to rotate an item with a support device that provides two (mouse-) points (e.g. the AirScan) instead of a single touch (e.g. the mouse). By changing the angle between the two points you will rotate the item clockwise or anti-clockwise when the item is selected.

--> See an example for Rotate in the section "B. Device Settings"!

7.3.11.4.1.3 PB Settings

This chapter explains the PB Settings from the Multitouch Panel [899] in Widget Designer.

[Apply Defaults]
To recall the default values for the MultiTouch Panel (see [Store Defaults] above) press [Apply Defaults] or use the command WDMultiTouchApplyDefaults(ID) [1767].

[Store Defaults]
Once the MultiTouch Panel is set up, you may want to store the current values as Default Values within this WD Project. To do so, press [Store Defaults] or use the command WDMultiTouchStoreDefaults(ID) [1777].

PB X Offset / PB Y Offset
If you need an offset for the X or Y position of all items in Pandoras Box please enter it here. The values correspond to the position values used in Pandoras Box.

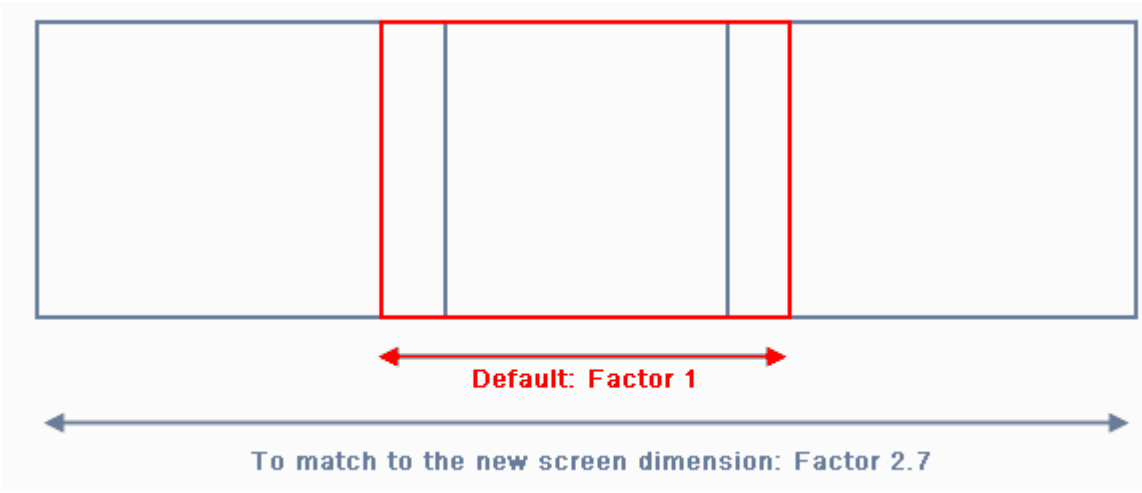
Example
Item 1 is on position 50 (this corresponds to the X Pos value 32768 in PB, when being in non-centered mode).

Entering a PB X Offset of 200 changes this X Pos value from 32768 to 32968 in PB – the item in the MultiTouch Panel stays at the same position.

PB X Factor / PB Y Factor
To use the MultiTouch Panel with several Pandoras Box Outputs (e.g. with an softedge projection with 3 outputs in a line) the X and/or Y Factors have to be adjusted to assign the MultiTouch panel area to the new screen dimension.

By default the PB X/Y Factor is set to 1, so the MultiTouch Panel refers to only one PB fullscreen output.

See here an example how to assign the MultiTouch Panel area to a new screen dimension:



Having a softedge projection with 3 Pandoras Box outputs in a line, the PB X Factor has to be changed to the new screen width (in the image above the new factor has to be 2.7). Otherwise you won't be able to position a MultiTouch Item on the left or right screen.

Mouse Down / Up / Enter / Leave
The MultiTouch Panel offers script processing for different mouse interactions, valid for the whole MultiTouch area:

"Mouse Down" is processed when a mouse left button down click on the panel is done.
"Mouse Up" is processed when a mouse left button up click on the panel is done.
"Mouse Enter" is processed when the mouse cursor enters the panel.
"On Mouse Leave" is processed when the mouse cursor leaves the panel.

Any command that is available from the command list [1520] may be used, just enter it in the text field. Multiple commands (i.e. one script) can be executed by delimiting every command with a carriage return at the end of the command. As these small text fields do not offer a good overview when using multiple commands, working with functions or macros [1897] is a good option. The topic Script Language [1511] explains commands and how to use them in more detail.

Zoom
Here you may change the Zoom factor that will be applied to all items having the Zoom option enabled.
The factor can be changed via the command (WDMultiTouchSetZoom(ID,Zoomfactor [1777]) as well.

Min / Max Zoom
These adjustable values will allow you to limit the minimum and maximum zoom factor (in %).
--> See an example for using the Zoom in the section "B. Device Settings"!

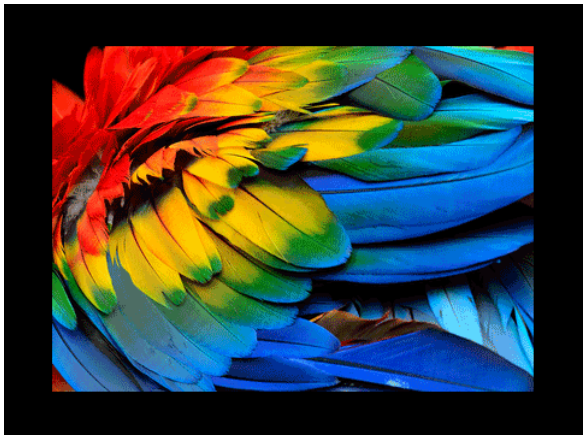
Min / Max Rot
The values (in degree) for minimum and maximum rotation allow you to set the amount of degrees the items are allowed to be turned. This function is limited to maximum +/- 360 degree.
--> See an example for Rotate in the section "B. Device Settings"!

Min / Max Size
Enter new values here (in %) in order to change the minimum and maximum size an item is allowed to have.
--> See an example for Size in the section "B. Device Settings" under the cue "Scale"!

See here [1777] the list of all commands that influence the MultiTouch Panel:

7.3.12 Picture Box

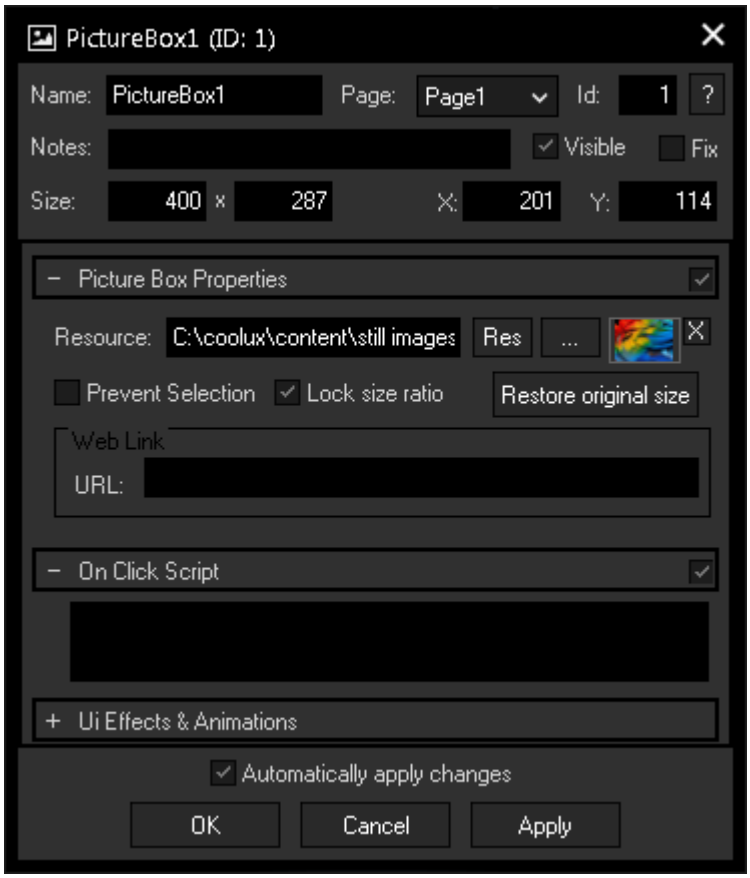
The PictureBox control lets you add images on your user interface.



To create a PictureBox widget choose "Widgets > PictureBox". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the PictureBox properties simply right-click it and choose the first menu entry "PictureBox Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The PictureBox property dialog opens up.

General Widget Settings



Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The PictureBox' ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the PictureBox will be displayed on every page.

Size:
Enter a pixel size for the PictureBox' size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Picture Box Properties

Resource:
Click on the "..." button to open a file dialog where you can choose your path and image. If you like to choose an image from the Widget Designer "library", click on the "Res" button. The Resource Manager [1509] opens where you

can double-click on the image of your choice. You can also save custom images in the Resource Manager to access them faster. The small "x" at the right sets the default image.

The image file will be copied to the project's data folder while saving.

Prevent Selection:
Check this box to avoid selecting the widget. It then can't be selected when in moving mode, except with a right-click directly on the PictureBox. Ticking this box also has the effect that the PictureBox' Z-position is being sent to the back.

Lock Size Ratio:
If this box is checked, the current size ratio will be maintained, even if you change the size manually.

Restore original size:
Click this button restore the image's original size. The original size can be displayed when hovering the mouse over the the image's thumbnail.

Web Link URL:
The Web Link section is of special interest when working with the Web Server [1929] feature, i.e. clicking the PictureBox in a web browser. Enter a URL e.g. "https://www.christiepandorasbox.com" that your browser should call when executing the On Click Script.
It is also possible to achieve a quick page change with the URL, simply enter a hash tag "#" and the page name. E.g.: #Page2

Starting with Widget Designer version 6.0.6 the URL is editable from any version or license.

Script

In the Script section you may enter commands to be executed. You can type directly in the text field, the Script Assistant will help you finding the expression you search for. The topic Script Language [1511] explains this in more detail.

You can put as much text in one of the scripting fields as you like, but for keeping a good overview, using Macros and Functions [1897] is recommended for sophisticated scripts

See here a list of all commands [1520].

On Click Script:
Enter a script that will be executed when clicking on the image. Functions and Macros [1897] are a good option to manage large and sophisticated scripts.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.3.13 Playlist WD

playlisrThis chapter describes the Playlist in Widget Designer, for the Playlist in Pandoras Box [239], please follow the link.

The WD Playlist in General

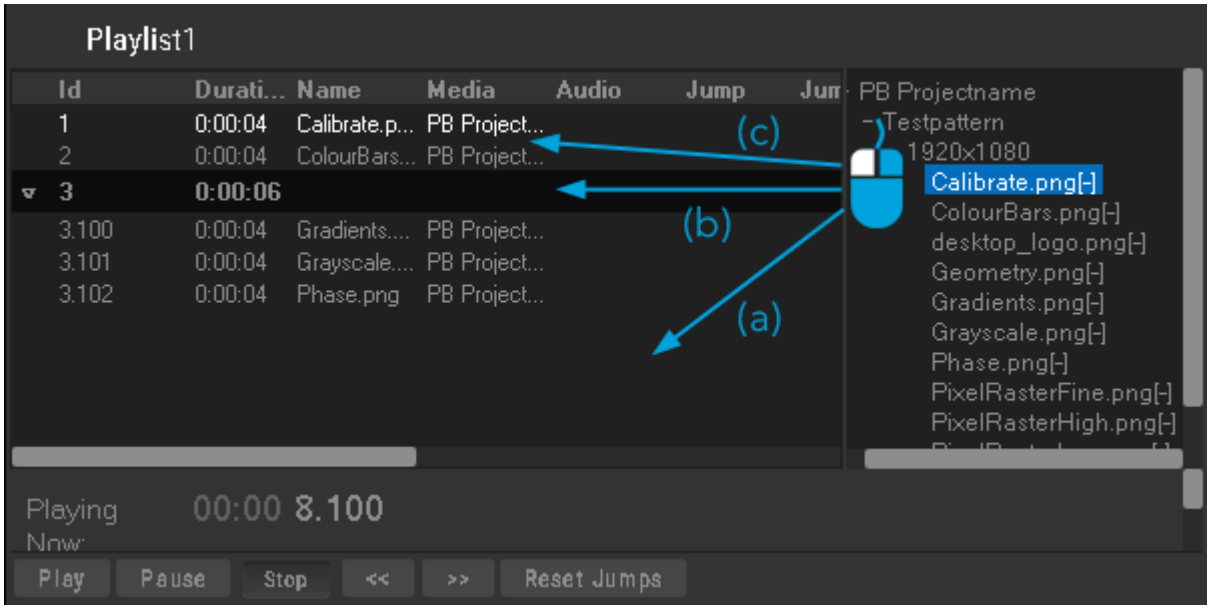
The Playlist widget offers the possibility to create a composition of different media files that playback in a certain order on dedicated Pandoras Box Layers.
You can set up whether the chosen files cross-fade one after another or playback with a hard cut. The playback order can be set to shuffle mode or simply from the beginning to the end whilst in both modes you can set up jumps, e.g. when a certain file should always be played back after another one. The Playlist assigns the media files to two alternating Layers, or rather Layer groups as the main media can be played back with two overlays and an audio file. Last but not least, you can enter individual script commands that should be executed when a file is played back.

As for all widgets, the Widget Designer script language offers commands that change certain widget settings or functions. See here a list of the available WDPlaylist... [1796] commands. All widgets can also be controlled using the member notation [1904].

Adding Files and Folders to the Playlist

To create a Playlist widget choose "Widgets > Playlist". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

When connected to a PB project (see dialog PB Network Configuration [1256]), the tree view section from the Playlist shows the files in the Project tab [278]. If you changed something in PB, right-click in the tree view section and choose "Refresh tree".
However, it is also possible to add, rename and delete files and folders from the PB project by working in the WD Playlist only. Right-click in the tree view to open its menu and choose an according command. Note that the check-box "Show Menu" in the Item Properties must be ticked. Note that you can also add content from a remote computer by using the Web Server feature [153]. This is explained in more detail further down under "Adding and Uploading Files to the Pandoras Box Project" [917].



Now you can drag and drop graphic, video and audio files and entire folders into the Playlist.
(a) If you release the mouse anywhere under the existing list, a new item is generated at its end and it gets the highest ID. Any item in the list can be identified by its ID which starts with 1 and counts up with every new item.
(b) When you release the mouse on an existing folder, the selected item is added to the end of this group. If a Playlist folder has the ID 9, its files will be 9.001, 9.002 etc. When adding a folder from the PB project, WD asks whether you like to keep the items in a folder. Otherwise you can also choose to "Insert a folder" from the right-click menu in the Playlist, or to "Move to folder" when an item is selected.

(c) Lastly, you can release the mouse between two items (note the wide line). All IDs after the new item as well as already created jump IDs will increase automatically.
Alternatively, you can right-click in the Playlist and say "Insert Item" and right-click the item to choose "Edit cue".

Editing Existing Files and Folders

You can edit the Playlist as follows (note that the check-box "Editing Mode" in the Item Properties must be ticked). Double-click on the property of the item that you want to edit, e.g. the cell "Duration". Enter a new value and press [Enter].

Alternatively, you can right-click on the item and choose "Edit cue" which opens a dialog that offers even more properties. All properties are explained below together with some commands from the right-click menu.

ID and Renumber:
As explained above, new files or folders get unique IDs. If you like to change the order from items in most cases it is the fastest way to drag the item to its new place as WD renumbers all items automatically. However you can also enter new IDs manually. Note that the right-click menu offers some commands to "Renumber" items.

Duration, Fade, Pre Roll and Show Total Duration:
The **Duration** is the playback duration of the file including the fade time. Both times are displayed in the hour:minutes:seconds format. The default duration of video and audio files is automatically adopted to the file length. Graphic files are added with the "Default Length [917]" set up in the Playlist Item Properties dialog. If you like to set up a duration of one minute and 5 seconds, enter "1:05" or "65".
The **Fade time** is the duration where the opacity from one file drops from 255 to 0 and where two files are visible at the same time. This is also available as a default option [917] in the Item Properties.
Use the **Pre Roll** time setting to start loading the next media from the hard disk before the Play cue. This ensures a smooth playback and fade. If you notice that media files are loaded to late, e.g. graphics with a large resolution, increase the Pre Roll time. Again, this is also available as a default option [917] in the Item Properties.
If you like to know how long your Playlist will playback, right-click in the Playlist and choose "**Show total duration**".
The duration of a sub folder is displayed automatically.

Name:
The Name is only used internally and can be changed, e.g. for a better overview. Note that you can also enter a Name for a folder.

Media, Audio, Overlay1 and Overlay2:
Enter a project path, e.g. "PB Projectname\File.png" in the Media, Audio, Overlay1 and Overlay2 field.

Inpoint and Loop:
Both options are only available in the "Edit cue" dialog and apply to the respective Media, Audio, Overlay1 and Overlay2 media.
An Inpoint above 0 means that the file does not start at its beginning but at the Inpoint time.
With an activated "Loop" option the media starts to play from the beginning (i.e. the Inpoint time) in case the duration is longer than the media length.

Mode:
The "Edit cue" dialog offers a drop-down-list with three Cue Mode options. This is also available as a default option [917] in the Item Properties.

Continue: In Play Mode, one item plays after another and the layers fade.

Pause and hold: In Play Mode, only one item plays. When the duration of this item is over, the item does not fade down, it stays visible (and loops if not deactivated).
The next item in order is assigned to the layer but the opacity is set to 0. Note that this item is highlighted in dark cyan. The Playlist pauses. Once you hit the "Play" button, the item plays with an immediate opacity value of 255 and is now highlighted in green.

Pause and fade: In Play Mode, only one item plays and fades down before its duration is over.
The next item in order is assigned to the layer but the opacity is set to 0. Note that this item is highlighted in dark cyan. The Playlist pauses. Once you hit the "Play" button, the item plays with an immediate opacity value of 255 and is now highlighted in green.

Jump, JumpCount, JumpsLeft and Reset Jumps
Enter the ID from the file that should be played back after this item into the "Jump" field. If you like to jump to this file twice before going on with the next one in order, enter "2" in the "JumpCount" field, enter nothing for an endless routine. During Playback the field "JumpsLeft" displays the number of jumps that are still going to happen. As soon as all jumps were executed, the number will be reset. If you like to reset it manually, click the button "Reset Jumps" at the bottom of the Playlist. You can of course also double-click the "JumpsLeft" field and enter a new number.

Script:
Enter a command [1520] into the list that should be executed with the playback of the according item. This property is not available in the "Edit cue" dialog.

Start Date, End Date, Verify and Filter by date:
You can limit the playback of an item to a time window between the Start Date and End Date that can be entered in the "Edit Cue" dialog. If the current time is not in the time slot, the item is simply ignored. In that case it can be displayed in italics if the "Verify" option is ticked. With the option "Filter by date" from the Playlist's right-click menu you can toggle the visibility from verified items in order to see only those files that are currently available.

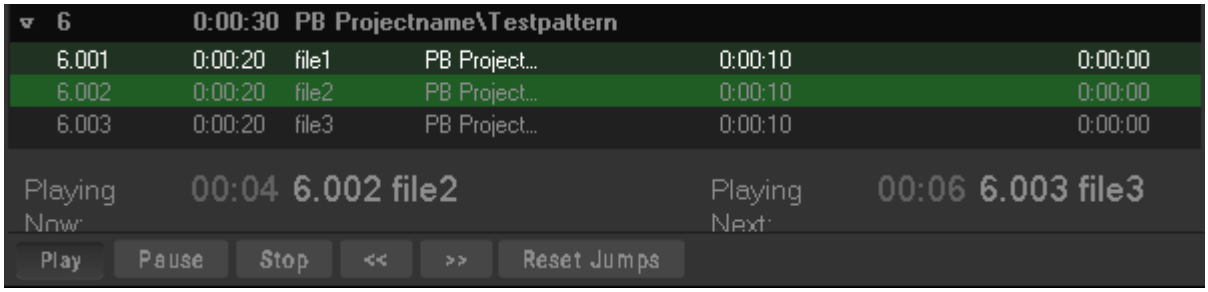
Edit column Title:
This command is offered when you right-click on one of the titles in the bar just above the list section, e.g. Duration. You can rename each column.

Removing Files and Folders

If you like to remove a folder or a single item, right-click it and choose "Delete". Alternatively, a selected item can be deleted with the [Del] key. Click once on a list item to select it, note that the text is displayed in white. You can also multi-select items by holding the [Ctrl] or [Shift] key.

Playing Back the Playlist in WD

The buttons below the Playlist allow to start, pause or stop the playback from the Playlist. If no item is selected, the playback starts with the first available item from the list. If you like to start with a certain one, you can either stop the Playlist first. Select the item with a left-click (or use the buttons "<<" / ">>") so that the item is displayed in white and click "Play". Otherwise you can also right-click the according item and choose "Play cue" which is also possible during playback.
Per default, the items are played back one by one, starting at the top of the list and going down. The right-click menu offers the command to switch to shuffle mode. Uncheck "Loop" in the right-click menu or in the Item Properties if the Playlist should not play continuously.



The currently playing file is highlighted in light green and is depicted next in the "Playing Now" section together with a counting-up timer. During a fade when two items are visible, the fading item is highlighted in dark green. The section "Playing next" shows a counting-down timer with the name in The file Above the buttons you can see the section Use the Play, Click once on a list item to select it, note that the text is displayed in white.

Item Properties dialog

The Item Properties offer general widget settings, style options and other settings for the Playlist including the to layer assignment for Pandoras Box.
Right-click in the Playlist and open the Item Properties. When you are in the Run Mode click the top or bottom part of the Playlist as the list itself or the tree view open their menus with editing commands. Alternatively you can toggle to the Create Mode and either double-click on a widget or use the shortcut [Alt + P] to open the Item Properties dialog.

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The widgets's unique ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID. Note the property member [1904].

Visible:
Uncheck the "Visible" option to hide the widget.

Fix:
Check the option "Fix" to display the widget on every page. If the option is unchecked (again) the widget is only displayed on the page chosen with the Page drop-down list.

Size:
Enter the width and height for the widget in pixels.

X and Y:
Enter the position in pixels of the widget on the page in pixels. 0,0 is the top left corner of the Page and the position refers to the top left corner of the widget.

Layer, Track and Overlays A/B:
Enter the Site and Device IDs, e.g. "2.1" for Layer A and "2.2" for Layer B if the main media files from the Playlist items should alternate between Layers 1 and 2 of Site 2 (e.g. a Server). You can also enter "2.1 2.2 2.3" if you like to share the media with multiple Devices. The "Track" refers to the audio media and the Overlay fields to the according Overlay media files.
Note that the B layers should be visible in front of the A layers, otherwise the "Cross-Fade" option might be of interest.

Default Length:
The default length applies to all media files e.g. graphics that have no own media length. Note that the Fade time is included in the length.

Title:
The Title is displayed above the Playlist and can be toggled on and off and renamed.

Path Filter:
Per default, the tree view section shows the entire PB project path, with the Path Filter you can restrict it to display only certain sub folders. For example, with "PB Projectname\Testpattern" all other folders in the project "PB Projectname" than the one called "Testpattern" are invisible.

Editing Mode:
The Editing Mode allows to edit items that are already part of the Playlist. You can edit an item by either double-clicking a property or choosing "Edit cue" from the right-click menu.

Show Buttons:
Deactivate this option to hide the lower button part of the Playlist (Play, Pause etc.).

Show Project Tree:
Deactivate this option to hide the right tree view part of the Playlist.

Show Menu:
Deactivate this option if a right-click in the list or tree view section of the Playlist should not display their menus with editing commands but the context menu of WD.

Cross-Fade:
The Cross-Fade option is deactivated per default which means that only the B layers fade up and down whilst the A layers alternate their media files but keep a constant opacity of 255. If the A layers should also fade up and down, i.e. if the layers should cross-fade, activate the option. This is especially useful when some media files in the Playlist are partly transparent.

Please note that changes in the Pre Roll Time, Fade Time and the Default Cue Mode do not affect items that are already added to the Playlist!

Pre Roll Time and Fade Time:
Please see above the explanation for "Duration, Fade, Show Total Duration" [914].

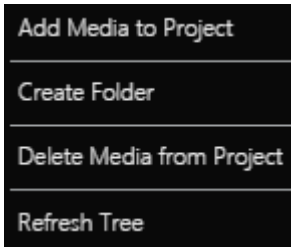
Default Cue Mode:
Please see above the explanation for "Mode" [914].

Loop Cue List:
Uncheck "Loop" in the right-click menu or in the Item Properties if the Playlist should not play continuously.

In the section "UI effects & Animations [814]" you can apply CSS based style effects to all widgets.

Adding and Uploading Files to the Pandoras Box Project

If you perform a right-click in your Pandoras Box tree, a context menu will offer you some additional options:



Add Media to Project:
Opens an explorer dialog where you can choose a file to add to the selected folder in the Pandoras Box project.

Create New Folder:
Creates a new folder in the selected directory. A dialog will ask you to enter the name of the new folder.

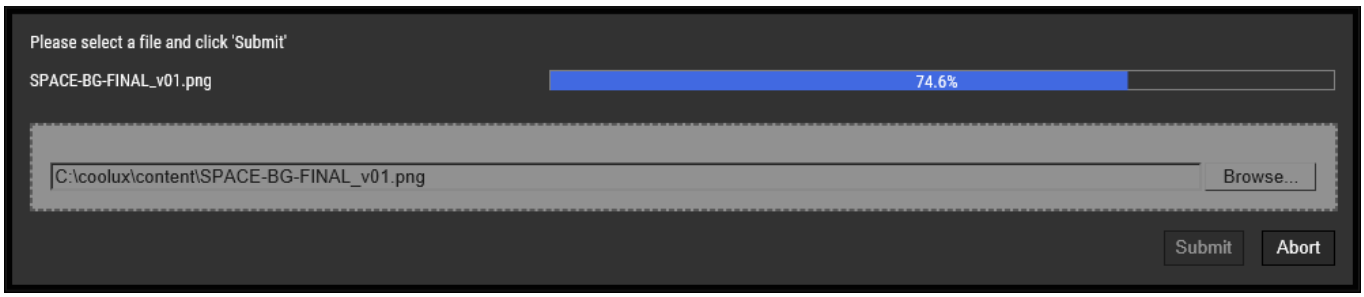
Delete Media from Project:
Removes the selected item, folder or file, from the project.

Refresh Tree:
Refreshes the tree if changes were made from outside

One special feature of the Playlist is an easy way to transfer files from an external computer to the local machine and the direct import to Pandoras Box. To achieve this, simple access the Widget Designer project with an external browser. Here [1929] you can learn more about the Web Server feature.

In the browser, you will see the Playlist still containing the Pandoras Box tree. If you execute a right-click on a folder there, you have the same options as in the main GUI. When you choose to add media, a new dialog will open and offer you a possibility to browse the external PC for files. The progress bar shows how much of the file is already transferred. If an error occurs, a respective message is displayed in the upper left corner of the dialog. Multiple files can be selected and transferred at once.

If you transfer files directly to a Pandoras Box project, they will be physically copied to the Widget Designer project's data folder (Data/Uploads) which will also automatically be created if it didn't exist before. **Both applications have to run on the same PC!**
In PB, the file is added to the selected project folder, but as usual only as a link. This link points to the data folder of Widget Designer. Upon content spread, the file is copied physically to the PB clients.



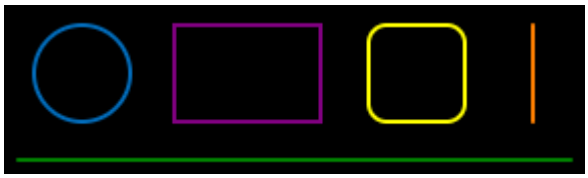
Browse:
Opens a dialog to browse for files to upload.

Submit:
The selected file is transferred to the data folder of Widget Designer and then added (as a link) to the selected PB folder.

Abort:
Aborts the current transfer process.

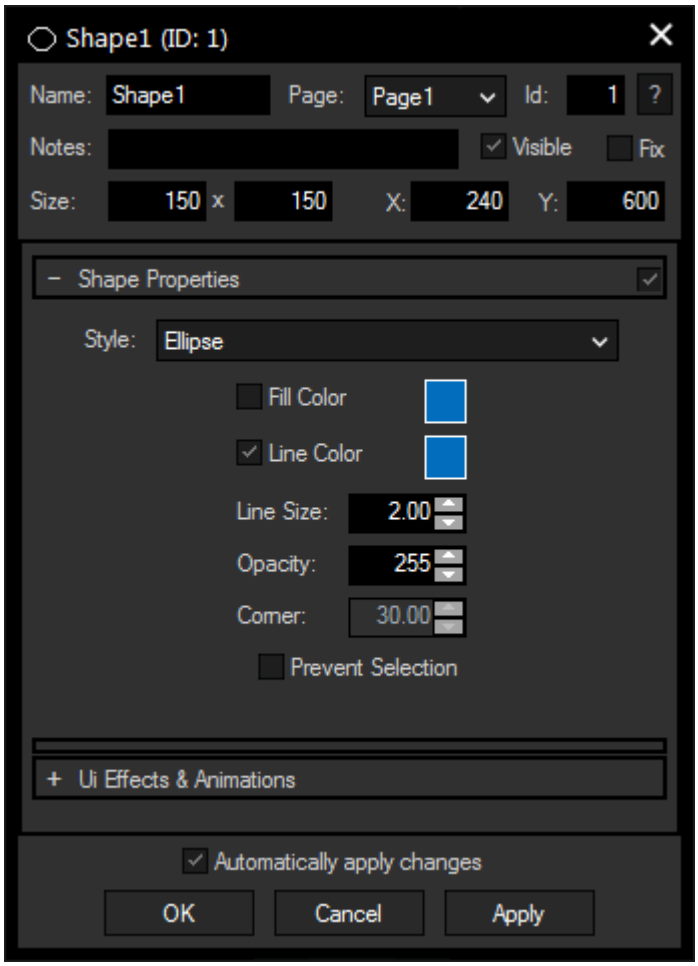
7.3.14 Shape

Shapes can be used for highlighting elements in the user interface, to group widgets, to subdivide areas or simply as a decorative item.



To create a Shape widget choose "Widgets > Shapes > Ellipse or or Line horizontal / vertical or Rectangle or Rounded Rectangle". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the Shape properties simply right-click it and choose the first menu entry "Shape Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Shape property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The Shape's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the Shape will be displayed on every page.

Size:
Enter a pixel size for the Shape's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Shape Properties

Style:
You can choose one out of three designs: ellipse, rectangle and rounded rectangle

Fill Color:
Check this box to activate the fill color, a color picker for changing the hue will open when you click on the small box on the right.

Line Color:
Check this box to activate the line color, a color picker for changing the hue will open when you click on the small box on the right.

Line Size:
Enter here the thickness of the outline in pixels.

Opacity:
Enter the transparency for the widget, "255" equals not transparent at all while "0" means completely transparent.

Corner:
This parameter is only available for rounded rectangle Shapes and specifies the rotundity of the corners. A value of "0" would be a straight rectangle, increasing the value leads to more rounded corners.

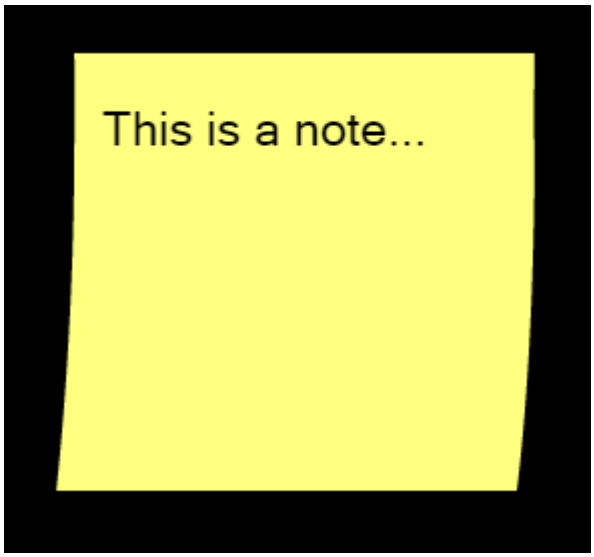
Prevent Selection:
Check this box to avoid selecting the widget. It then can't be selected when in moving mode, except with a right-click directly on the Shape. Ticking this box also has the effect that the Shape's Z-position is being sent to the back.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

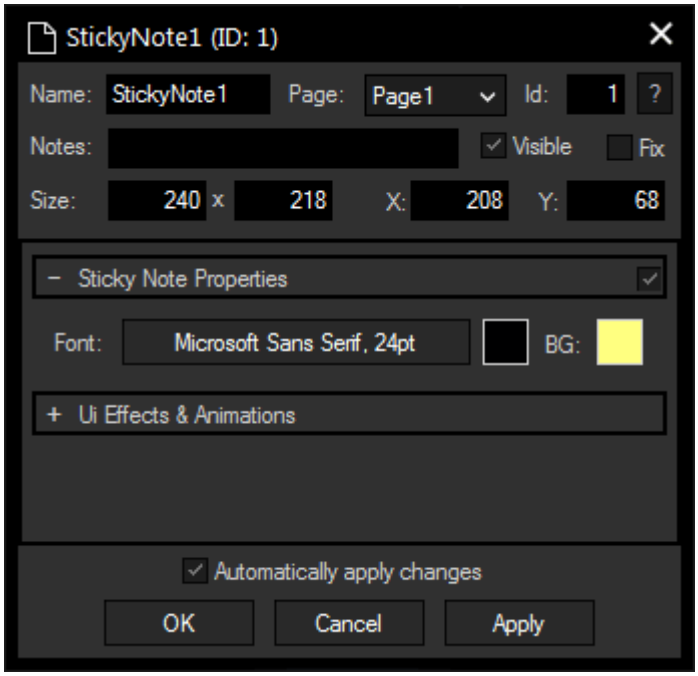
7.3.15 Sticky Note

The StickyNote is a tool for adding short messages or instructions to the user interface. They are only visible if enabled in the View menu [799] or by using [Ctrl + Alt + N].



To create a StickyNote widget choose "Widgets > StickyNote". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the StickyNote properties simply right-click it and choose the first menu entry "StickyNote Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The StickyNote property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The StickyNote's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the StickyNote will be displayed on every page.

Size:
Enter a pixel size for the StickyNote's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Sticky Note Properties

Font:
Edit the StickyNote text font and the font size by clicking on the button with the current font.

To change the text color click in the small box on the right side.

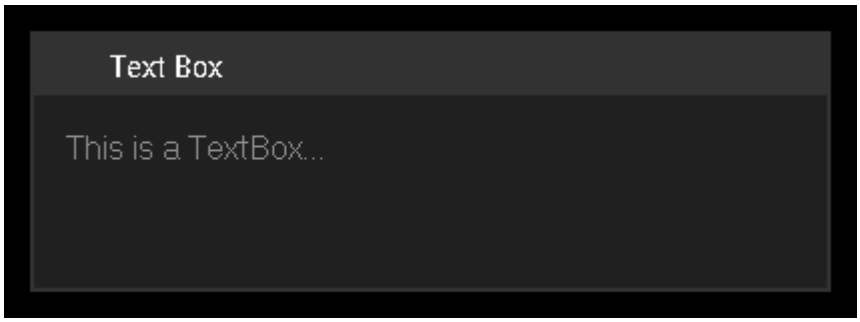
Background Color:
Click the box to open a color picker dialog for the background color.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

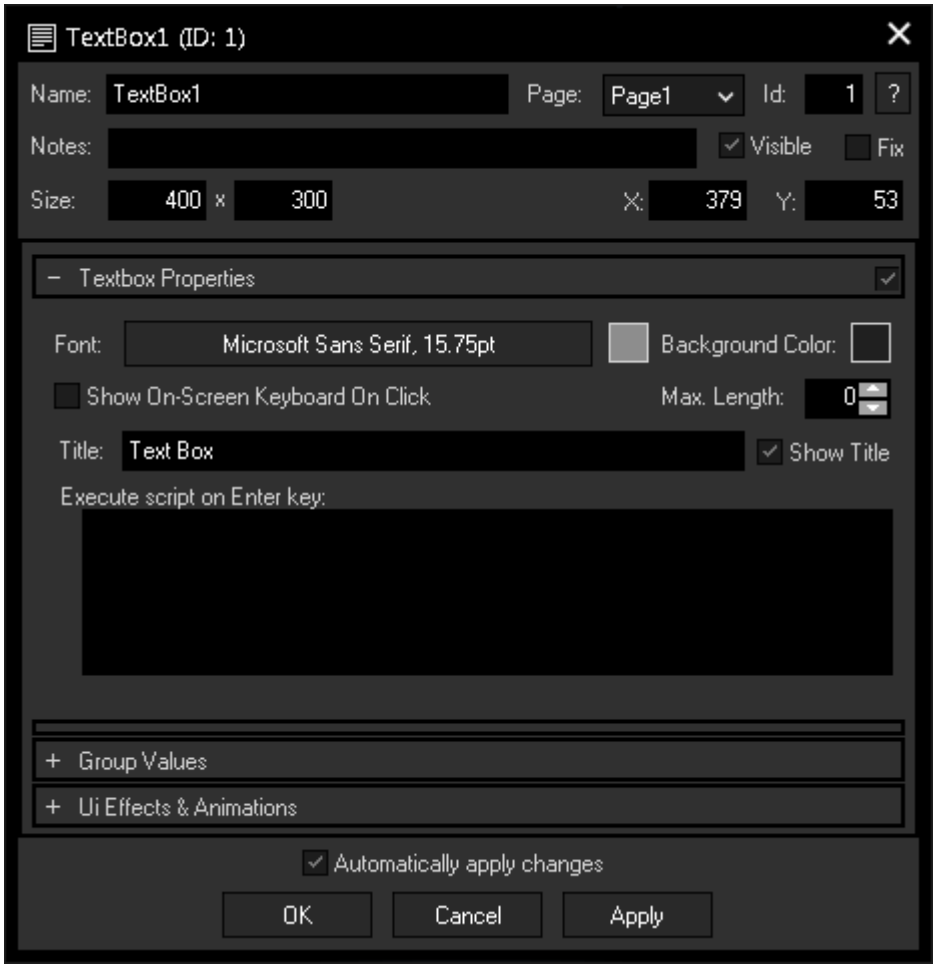
7.3.16 Textbox

The TextBox widget allows you to add an editable text to remote control text assets within Pandoras Box via the Textbox Input Node [1088] and the PB Text Output Node [1225].



To create a TextBox widget choose "Widgets > TextBox". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the TextBox properties simply right-click it and choose the first menu entry "TextBox Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The TextBox property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The TextBox' ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the TextBox will be displayed on every page.

Size:
Enter a pixel size for the TextBox' size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Textbox Properties

Font:
Edit the TextBox text font and the font size by clicking on the button with the current font.

To change the text color click in the small box on the right side.

Background Color:
Click the box to open a color picker dialog for the background color.

Show On-Screen Keyboard on click:
Check this box to open the integrated on-screen keyboard as soon as a click is performed inside the TextBox. This is especially useful for touch applications such as tablet PCs or touch monitors.

Max.Length:
Enter the maximum length of the entered character string. "0" stands for an infinite number of characters.

Title:
Enter here a title that is displayed at the top of the TextBox. Uncheck the box "Show Title" if you want to remove the title bar.

Execute script on Enter key:
Enter a script that will be executed when the [Enter] key is pressed. Functions and Macros [1897] are a good option to manage large and sophisticated scripts.

Group Values

This option is only available for the Unlimited version and offers the possibility to assign a group. Please refer to the chapter Group Values [1933] for more information.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

To link the TextBox to a Text asset in Pandoras Box, please use a Text Box Input Node and connect it to a PB Text output node.

To change the text of a TextBox e.g. via a Custom Script Button, these commands [1833] are available:

Dynamic script setup

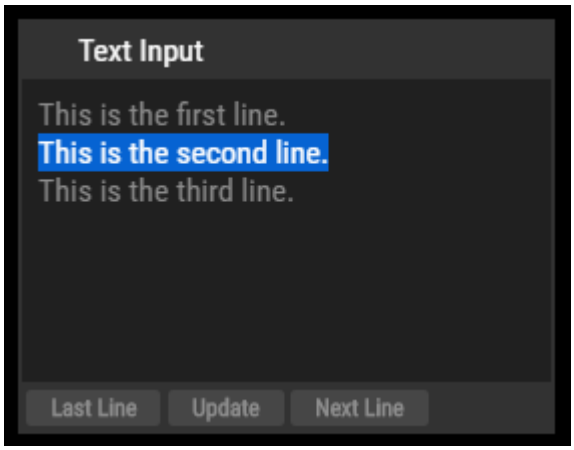
With the following commands you can use a TextBox to hold and collect script snippets that could be executed at any given time:
WDTextboxAppend(ID,Value) [1832]
WDTextboxNewline(ID) [1837]
WDTextboxExecuteAsScript(ID) [1834]

Imagine you want to build a signal router remote application.
You can use buttons to set the source selection and buttons to append the script code for the target selections. A take button would use the entire script that is inside the TextBox and executes it as one big dynamically created script.
This way it is much easier to create and react to certain conditions and logics that would require many more nodes to be set up.

7.3.17 Text Input

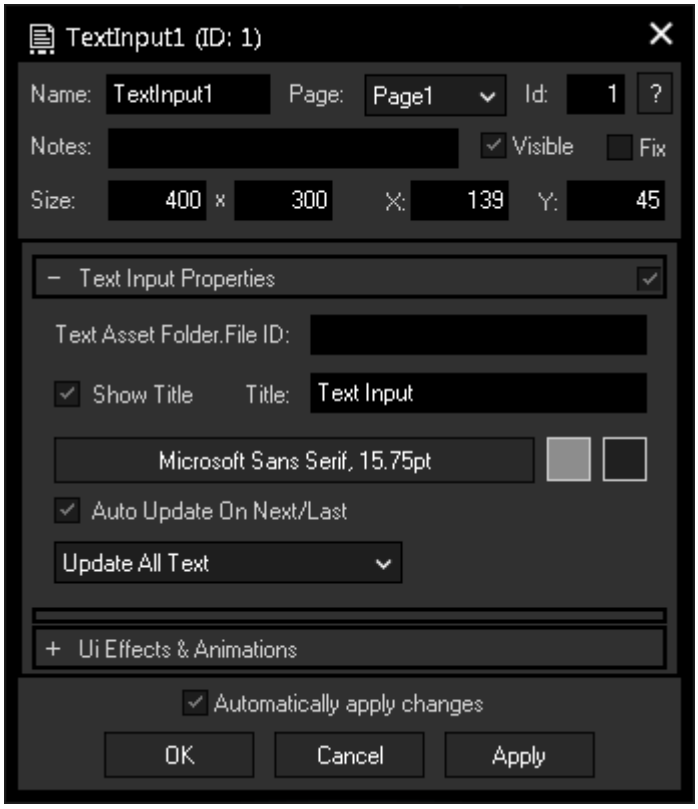
The TextInput control lets you add an editable text to remote control text assets [307] within Pandoras Box.

Depending on the configuration of the TextInput control you may update dedicated text assets by clicking on the Last Line and Next Line button. This way you may continuously update text assets line by line or update the entire TextInput contents to the text asset on all connected Pandoras Box Client systems.



To create a TextInput widget choose "Widgets > TextInput". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the TextInput properties simply right-click it and choose the first menu entry "TextInput Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The TextInput property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The TextInput's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the TextInput will be displayed on every page.

Size:
Enter a pixel size for the TextInput's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Text Input Properties

Text Asset FolderID.FileID:
Define one or multiple Text Assets in PB this Text Input should be linked to. To do this please enter the Folder and File IDs.

Example:
To assign the Text Input to PB Text Asset 1.1 and 1.2, enter: "1.1 1.2".

Title:
Enter here a title that is displayed at the top of the TextInput. Uncheck the box "Show Title" if you want to remove the title bar.

Font:
Edit the TextInput text font and the font size by clicking on the button with the current font.

To change the text color click in the small box on the right, to change the background color click on the far right button.

Auto Update On Next/Last:
This option will send text contents directly upon button press.

Update Selected Line / Update All Text:
Choose between these two options to update only by line or complete text. If the "Update Selected Line" mode is active, only the line highlighted in blue will be sent.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

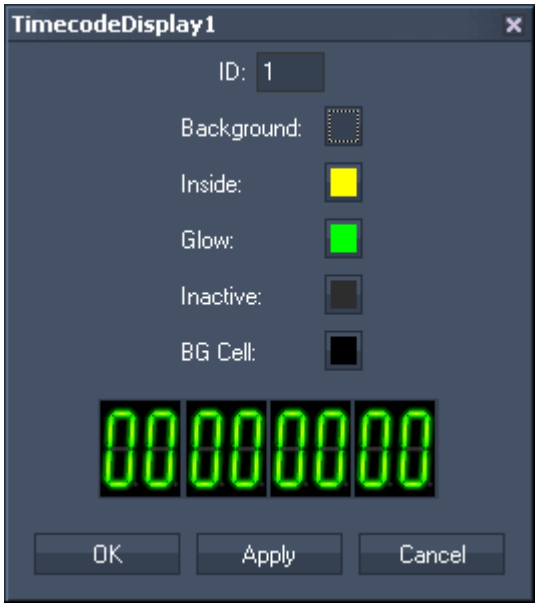
7.3.18 Timecode

The Timecode control displays the timecode sent or received via a connected SMPTE Link device.
Set up the SMPTE Link device in the Connection Manager [1258].

You may also route the sent / received SMPTE Timecode via the SMPTE Input Node [965] within the WD Node System.

To create a TimeCode widget choose "Widgets > TimeCode". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the Timecode properties simply right-click it and choose the first menu entry "Timecode Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The Timecode property dialog opens up.

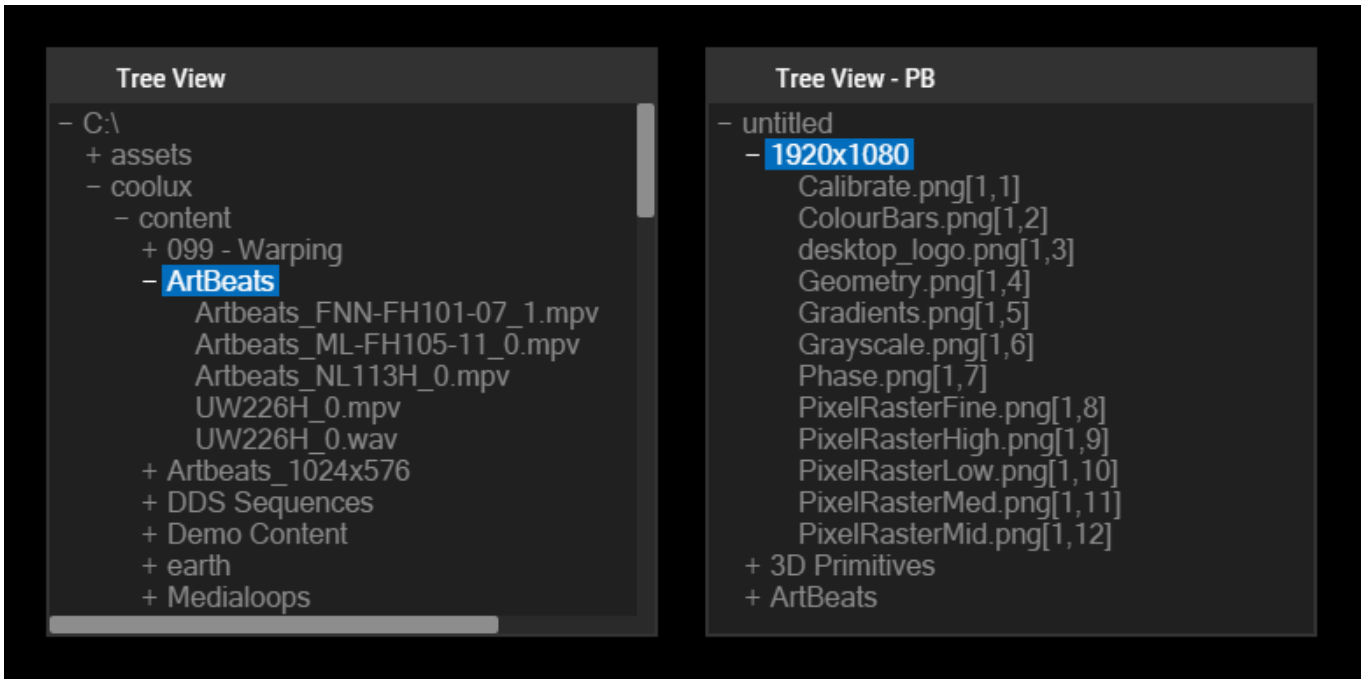


The Timecode's ID may be changed by entering a new one in the text field on the top.

It allows you to choose the colour for the timecodes
- background,
- inside,
- glow,
- Inactive,
- background cell.

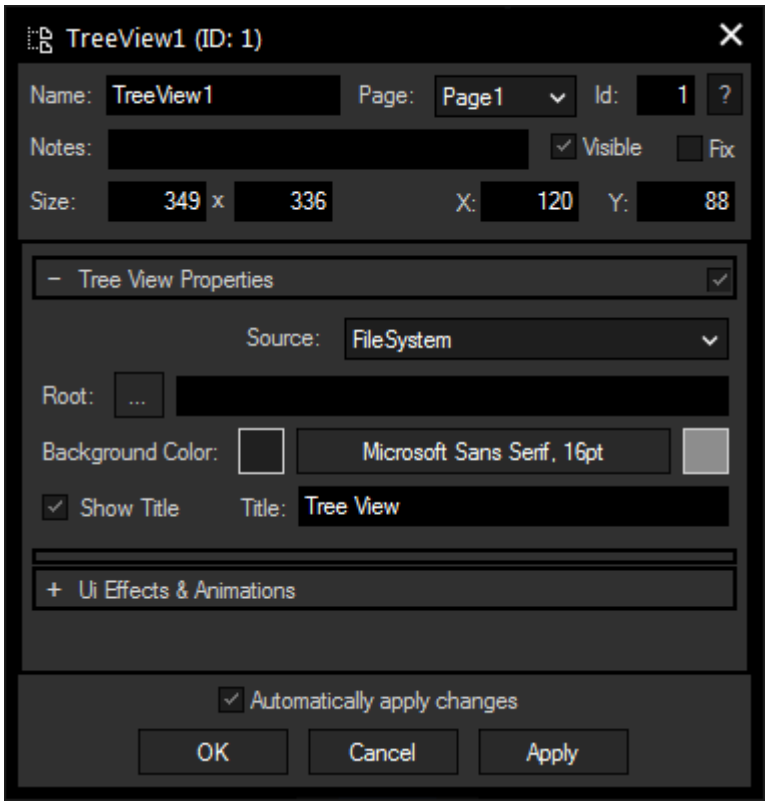
7.3.19 Tree View

The TreeView enables you to browse and manage data on your local PC or in your Pandoras Box project. It is also able to easily transmit files from an external device to the local machine, using the TreeView in a web client.



To create a TreeView widget choose "Widgets > TreeView". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the TreeView properties simply right-click it and choose the first menu entry "TreeView Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The TreeView property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The TreeView's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the TreeView will be displayed on every page.

Size:
Enter a pixel size for the TreeView's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels.

Tree View Properties

Source:
Choose between the computer's file system or the Pandoras Box project the Widget Designer is connected to.

Root:
Specify here a root of the displayed file tree, you can also click on the "..." button to choose a path from the explorer. This parameter is only available if the source is set to "File System".

Background Color and Font:
Click the box to open a color picker dialog for the background color.
Edit the TreeView text font and the font size by clicking on the button with the current font. To change the text color click in the small box on the right side.

Title:
Enter here a title that is displayed at the top of the TreeView. Uncheck the box "Show Title" if you want to remove the title bar.

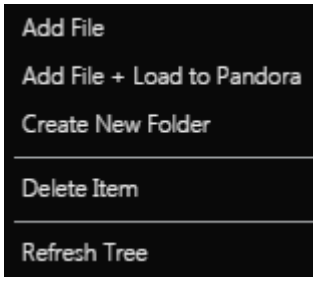
Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

How to use the TreeView

Local file management can be performed by right clicking on a file or folder in the TreeView, the opening context menu depends on the selected TreeView source:

File System



Add File:
Opens an explorer dialog where you can choose a file to copy to the selected folder

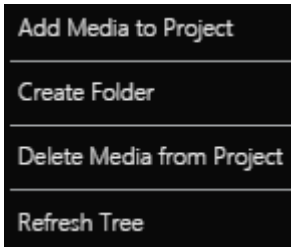
Add File + Load to Pandora:
Opens an explorer dialog where you can choose a file to copy to the selected folder. The copied file will also immediately be added to the Pandoras Box project if the software is connected.

Create New Folder:
Creates a new folder in the selected directory. A dialog will ask you to enter the name of the new folder.

Delete Item:
Deletes the selected item. If a folder is selected, it has to be empty to be deleted.

Refresh Tree:
Refreshes the tree if changes were made from outside

Pandoras Box



Add Media to Project:
Opens an explorer dialog where you can choose a file to add to the selected folder in the Pandoras Box project.

Create New Folder:
Creates a new folder in the selected directory. A dialog will ask you to enter the name of the new folder.

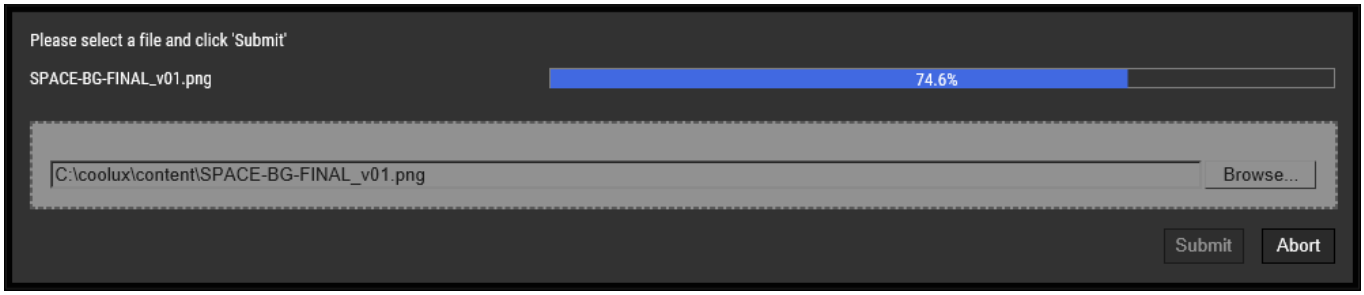
Delete Media from Project:
Removes the selected item, folder or file, from the project.

Refresh Tree:
Refreshes the tree if changes were made from outside

One special feature of the TreeView is an easy way to transfer files from an external computer to the local machine and the direct import to Pandoras Box. To achieve this, simple access the Widget Designer project with an external browser. Follow the link to learn more about the Web Server [1929] feature.

In the browser, you will see the TreeView still containing the file system of the local machine or Pandoras Box tree. If you execute a right-click on a folder there, you have the same options as in the main GUI. When you choose to add a file or a media, a new dialog will open and offer you a possibility to browse the external PC for files. The progress bar shows how much of the file is already transferred. If an error occurs, a respective message is displayed in the upper left corner of the dialog. Multiple files can be selected and transferred at once.

If you transfer files directly to a Pandoras Box project, they will be physically copied to the Widget Designer project's data folder (Data/Uploads) which will also automatically be created if it didn't exist before. **Both applications have to run on the same PC!**
In PB, the file is added to the selected project folder, but as usual only as a link. This link points to the data folder of Widget Designer. Upon content spread, the file is copied physically to the PB clients.



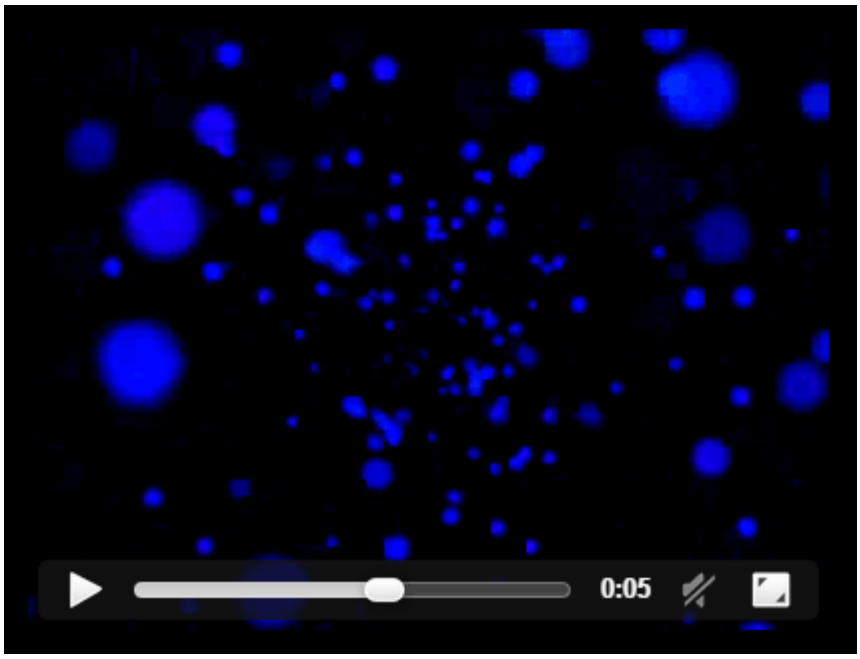
Browse:
Opens a dialog to browse for files to upload.

Submit:
As described above, if the TreeView displays the file system, the selected file is transferred to the selected directory. In case it displays the Pandoras Box folder, the file is transferred to the data folder of Widget Designer and then added (as a link) to the selected PB folder.

Abort:
Aborts the current transfer process.

7.3.20 Video Player

The VideoPlayer widget lets you add a borderless media player widget to your interface.

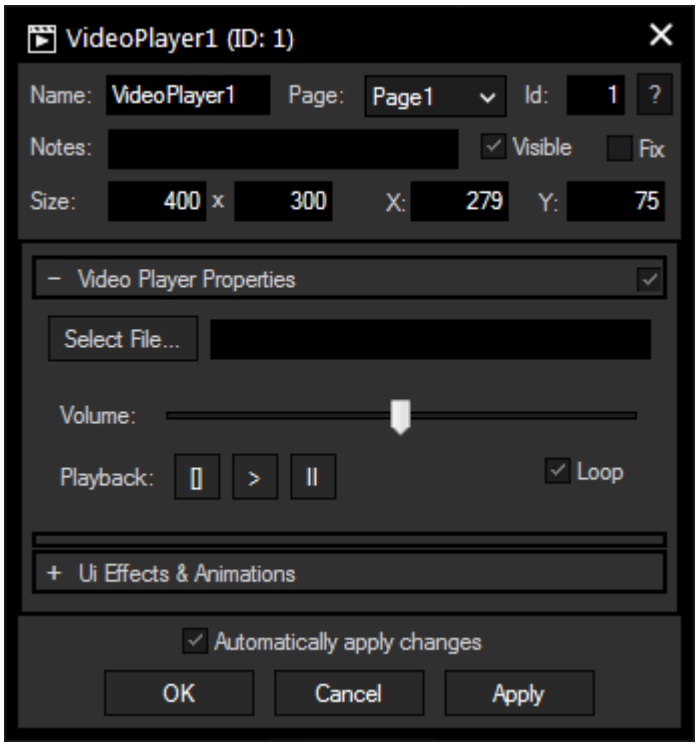


The on-screen controls offer you the possibility to play/pause the video, adjust the volume of audio and toggle full-screen.

Please note:
In order to control the VideoPlayer widget within your interface it is a good practice to create dedicated CustomScript [822] buttons or use commands [1511] in general to control the VideoPlayer widget. The VideoPlayer related commands are these ones [1851].

To create a VideoPlayer widget choose "Widgets > VideoPlayer". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the VideoPlayer properties simply right-click it and choose the first menu entry "VideoPlayer Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The VideoPlayer property dialog opens up.



General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The VideoPlayer's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the VideoPlayer will be displayed on every page.

Size:
Enter a pixel size for the VideoPlayer's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Video Player Properties

File:
Click on the "Select File..." button to open a file dialog where you can choose your path and video file.

In addition to the on-screen controls, you also have the option to adjust the **Volume** in the item properties, as well as **play** (>), **pause** (||) and **stop** ([]) the video.

Tick the **Loop** box to continuously loop the video playback, un-tick it if you only want to have the video played once.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

Supported Video Codecs

As Widget Designer V6 is based on HTML5, it also uses the integrated video playback feature HTML5 offers. As a result, the format of the videos played back with it are restricted to certain, mostly open source codecs. Those codecs also depend on the browser wherein the video is played. While it is not possible to playback the WebM format with a Safari browser, the Internet Explorer for example does not allow the Theora codec (OGG).

The Widget GUI itself consists of an embedded Chromium browser, called CEF ("Chromium Embedded Framework"). Considering this, the main GUI is currently able to playback Theora (**.ogg**), VP8 and VP9 (**.webm**), as well as H.264 via FFmpeg (**.mp4**). Other formats can be played back when accessing the project with an external browser, of course depending on the used browser.

For more information on the HTML5 video player as well as a table with browsers and supported codecs, please refer to the Wikipedia entry concerning this topic.

If you need to re-encode files, the VLC player proves a useful open source tool until we offer a converter.

7.3.21 Web Browser

The WebBrowser control lets you add a borderless WebBrowser control to your interface.

To create a WebBrowser widget choose "Widgets > WebBrowser". The mouse cursor changes to a crosshairs icon, indicating the create mode. With a left-click anywhere on the empty main background window you add the selected widget to the current page. Left-click again if you like to add one more. To quit the create mode, you can switch to the run mode with [F8], where you can use all widgets.

To edit the WebBrowser properties simply right-click it and choose the first menu entry "WebBrowser Properties". Alternatively, you can press [Alt + P] whilst the mouse is hovering above it or switch to the edit / move mode with [F9] and double-click on it. The WebBrowser property dialog opens up.

General Widget Settings

Name:
A unique name can be entered to identify the widget via the Object and Member Notation [1904]. The default name is based on the widget type and ID.

Page:
This drop-down offers all available pages to place the widget on.

ID:
The WebBrowser's ID may be changed by entering a new one in the text field top left. If you change it, you will be asked if you also want to adapt the name to the new ID.

Notes:
A short note can be added here. It is not displayed outside the widget but can be set and retrieved with the WidgetID.Note property member [1904].

Visible:
Uncheck this box to hide the widget.

Fix:
When the option "Fix" is checked, the WebBrowser will be displayed on every page.

Size:
Enter a pixel size for the WebBrowser's size.

X and Y:
Enter the location of the widget (upper left corner) in pixels

Web Browser Properties

URL:
Enter the URL which should be displayed in the WebBrowser.

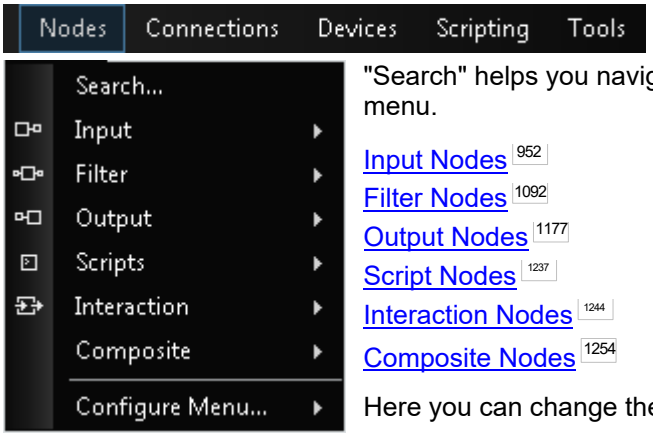
If you want to add buttons to navigate to different URLs or to clear the cache info you may use CustomScript buttons and these commands [1870] to remote control the WebBrowser.

Ui Effects & Animations

The topic Effects & Animations [814] explains how to add and apply CSS based effects and animations.

7.4 Nodes

The Nodes menu lists all node categories including all nodes you may create in Widget Designer Pro and Ult, currently there are about 250 nodes available. This chapter includes general information about the node system.



"Search" helps you navigate to the correct menu and sub menu.

Input Nodes [952]
Filter Nodes [1092]
Output Nodes [1177]
Script Nodes [1237]
Interaction Nodes [1244]
Composite Nodes [1254]

Here you can change the order of the nodes in the menu.

The node system in Widget Designer allows to create custom logic routing of control data. For example you achieve to...

- remote control a Pandoras Box Layer with a standard Joystick
- connect a sensor input to any Pandoras Box Layer Parameter
- receive Art-Net values from another device and map these to Midi output values

etc,... The list of possible scenarios is probably endless.
If you have never worked with a node based system, the way it works within Widget Designer is very easy to learn.

The next chapter [937] informs you about the different types of nodes and how to create and work with them. Then a small tutorial [939] follows.

There are different types of nodes: Inputs [952], Filters [1092], Outputs [1177] and the more advanced Scripts [1237] and Interactions [1244]. Additionally, you can create your own custom node using all other types of nodes with the Composite Node.

More advanced users might need to enter incoming or outgoing values as string, decimal or hexadecimal values, please find here the Syntax TCP- / UDP- / Serial Messages [944].
Another advanced technique is to control nodes via methods, which is explained in the chapter Object and Member Notation [1904].

7.4.1 Creating and Setting up Nodes

To create a node
a) open the node menu from the main menu bar...
b) click on the node symbol in the toolbar...
c) right-click anywhere in the empty main background and open the Nodes menu there...
c) use the keyboard shortcut [Alt + N] whilst you are in the edit / move mode ...

...and choose the desired node. The nodes are attributed to five node types: Inputs [952], Filters [1092], Outputs [1177], the more advanced Scripts [1237] and Interactions [1244] and additionally the customizable Composite Node. Some categories are sub-divided once more in order to make it easier and faster to find the nodes you are looking for. The table on the previous [936] page lists all categories and nodes. If you know how your node is called, it may be faster to click on "Search" and start typing, the drop-down will suggest possible nodes to you.

Node types

Input nodes provide numeric or text based input values.
For example the Mouse or Joystick Input node provides you with an X&Y value.
This value might be related to screen coordinates based on your local display resolution 1920x1080.

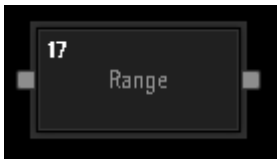
Filter nodes are used between Input and Output nodes, they recalculate the input value.
For example you might want to map the mouse motion to the 16bit value range of a layer in Pandoras Box between -8 and +8.
To resolve this you simply connect a Range Filter node to the Mouse Input and tell the Range Filter to recalculate the range from {0 to 1023} to {-8 to +8}. This is all it takes, this example is described step-by-step in this short tutorial [939].

Output nodes are used to send values out.
They receive an incoming value (from Input nodes directly or Filter nodes) and -according to their category- send them to another external device, or to Pandoras Box, or to a widget used in Widget Designer itself.
For example, you if you connect a Range Filter node with a Layer Output node, you can set up in the Layer Output node that the Range value should be send to the X Position parameter of a given layer.

Script nodes execute your customized script; they are triggered through a certain action.
They cannot be linked to other nodes, they are stand-alone. One could also say they are a fixed combination of an Input node to several Script Output nodes. This makes them very performance saving. In detail, Script nodes execute directly customized scripts as soon as a chosen action happens within Widget Designer. This can be, for example, a certain ASCII or Byte value coming in through the local COM port. One single Script node would combine one ASCII value with the first script and another ASCII value with another script and so on. Another type of (incoming) trigger can be an Action (e.g. a button is clicked, a fader changes the value etc.) or even a Gesture (e.g. Swipe Up) that is recognized in a tool like the AirScan.

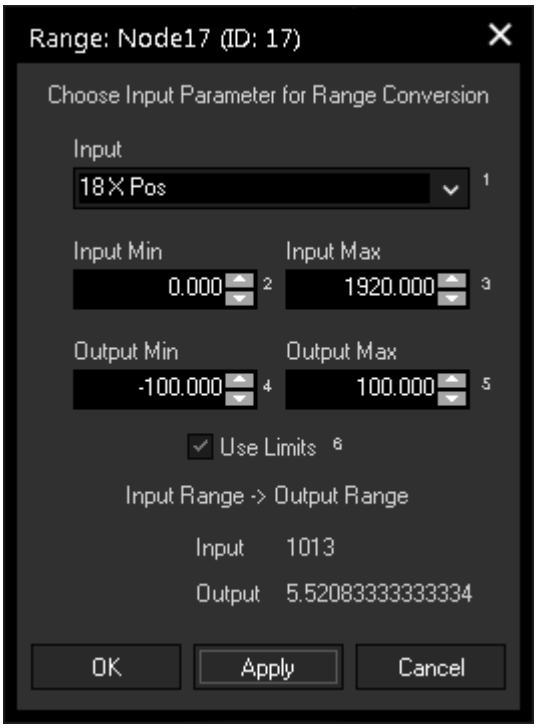
Interaction nodes execute an underlying application that enables to interact with a remote system.
They cannot be linked to other nodes, they are stand-alone. Each Interaction node has a special function that would require a complex node and command combination. For example:
- they send the mouse or touch events happening on the local system to another system, or
- they use the local mouse or touch events to draw on a Canvas Asset [280], or
- they watch a folder on the local hard drive and add new files to a Pandoras Box project plus append it to a Playlist [239]

Composite Nodes are especially useful if certain combinations of nodes are being used multiple times, for reducing complexity or for distribution of node-based solutions of any kind. As you can create your own customized node containing a composition of nodes and widgets, this feature offers high flexibility.



Once you have chosen a node, the mouse cursor will change to a crosshair icon, telling you that you are in the operating mode [789] called "create mode". Wherever you left-click now, the chosen node is generated. You may click again, to generate a second node from the same type.

The small number in the upper left corner is the node`s ID.

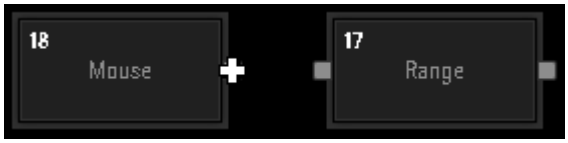


If you want to configure the newly built node, simply double-click it, or right-click on it and choose "Configure Node...".

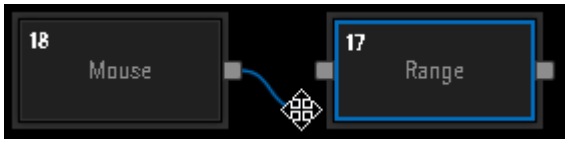
The node's configuration dialog opens up. The left image depicts the configuration of a Range Node [1147]. (It is already set up with example values.)

Here you can set up the static parameters and everywhere you see a drop-down list you may choose to pick a source node value output or enter a static numeric value into the text field of the combo box directly.

In many nodes you will notice little index numbers next to the input fields. These index numbers are important if you wish to remote control a node source or node value via commands from any other element such as a Custom Script Button or external remote or other output script node. This makes the node system very dynamic and lets you design systems that can change their behavior based on specific input parameters and controls.

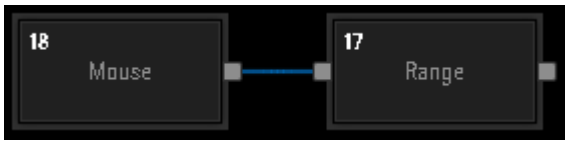


To connect two nodes with each other first switch to the edit mode by pressing [F9] (the mouse cursor differs from the standard mouse icon). Now, left-click and drag the source node output pin to the target node and release the left mouse button.



The connection between the node is now displayed as a gray line.

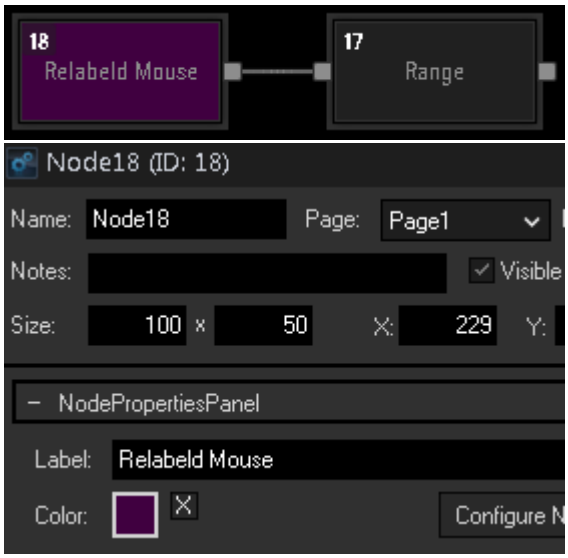
It is possible as well to connect nodes on different pages. To achieve this press [Page-Up] or [Page-Down] on your keyboard when dragging the connect node line to navigate to the desired page.



To delete a single connection between two nodes, switch to the edit / move mode and select the Node Connection by clicking on it. The Node Connection turns blue. Now press [Delete] on your keyboard to delete the Node Connection and confirm the upcoming pop up dialog.

Multiple connections can be selected with a selection frame (left-click and drag).

Removing connections does not reset all node values. Some node parameters are not based on source node values and can be set manually.



To copy and paste nodes, you have to be in the edit / move mode [F9]. To copy a single node, just click on it or drag a window across it.
To select multiple nodes, hold [Ctrl] pressed while clicking on the nodes or drag a window across all the nodes. Once selected, the nodes may be copied ([Ctrl + C]) and pasted ([Ctrl + V]) within the Widget Designer Project.

All Node Properties will also be copied, as long as the Node Sources are copied as well. This means also, that copied Input Nodes will not have any connections attached, in opposition to Filter and Output Nodes which will automatically be connected to the same Input Node(s) as the original.

To set up the properties of the node (= the node's look, i.e. Text and background color as well as Protection Settings),

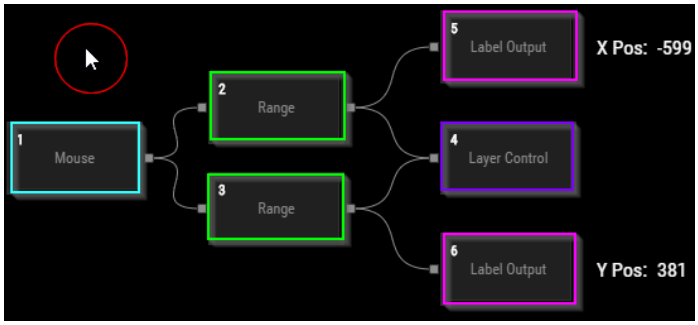
In addition to the Config dialog there is also an Item Properties dialog. Right-click on the node and choose the menu entry "Item Properties" or press [Alt + P] whilst having it selected. The left depicted dialog opens up.

Now you can change various settings like the text of the node or its background color. The Name is of interest when using the member notation [1904].

7.4.2 Tutorial: Nodes

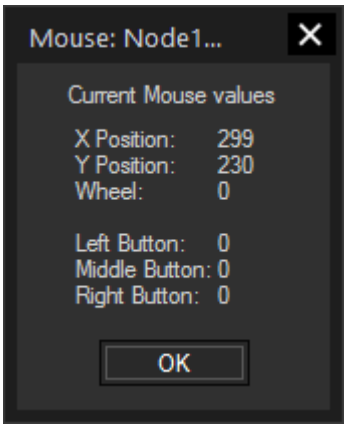
This tutorial shows how to map the X and Y position of a layer in Pandoras Box to a mouse cursor. Moving the mouse cursor to the left /right /upper or lower screen side should result in an according movement of the layer.

At first please make sure that the PB Network Configuration [1256] of your Widget Designer is set up correctly to control Pandoras Box.



First, please create the depicted node chain. You may refer to the last chapter " Creating and Setting up Nodes" [937] for more help with that. You need three different types of nodes:
1. One Mouse Input Node: Input > Devices Mouse.
2. Two Range Filter Nodes: Filter > Range. One for the translation of the X position, one for the translation of the Y position.
3. Two Label Output Nodes: Output > Widgets > Label. One for displaying the converted X position, one for the converted Y position.
4. One Layer Control Output Node: Output > Pandoras Box > Layer Control.

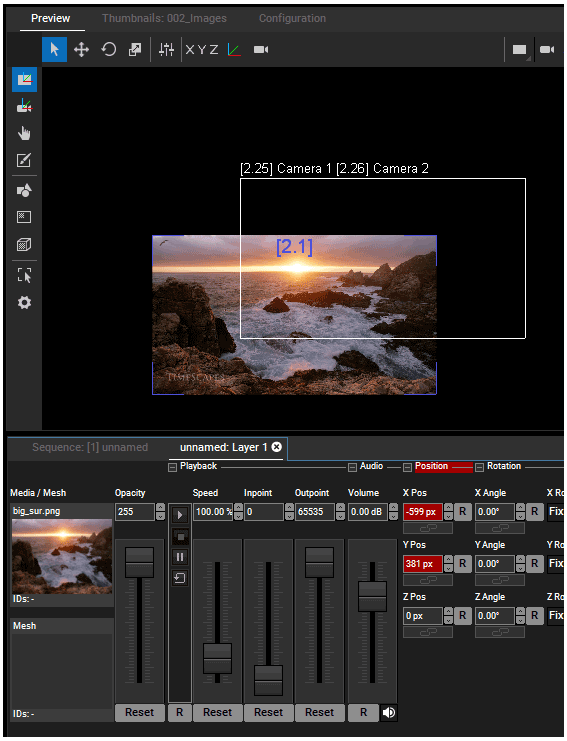
Whilst being in the edit / move mode [F9], connect the nodes as depicted.



The range nodes will be the translators between the mouse cursor X and Y position data and the X and Y Position values in Pandoras Box. Now, we need to find out what input range needs to converted to what output range.

Double-click on the Mouse Node to open the Node Configuration. Now the cursor positions are shown.

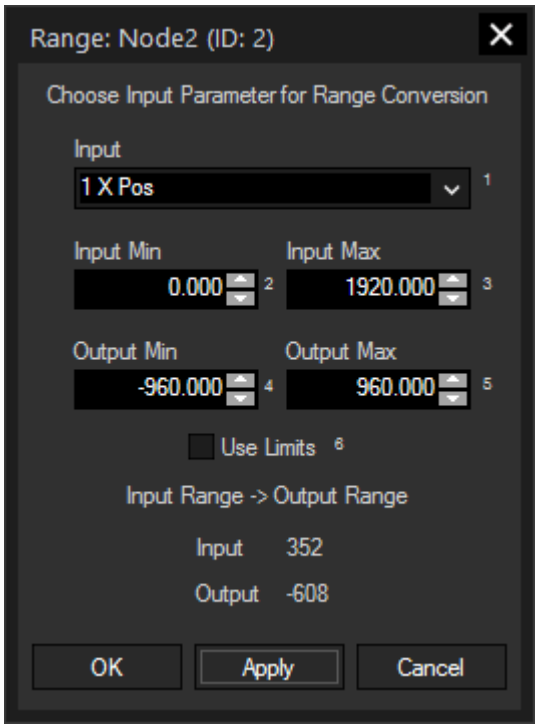
When your mouse cursor is at the left border of your screen, the mouse X Position shows 0. The Y Position depends on your screen resolution. When you are working with HD resolution, the value range is 0 to 1080px.



Now, we have a look at the Preview window of Pandoras Box Software. As the image shows, when Layer 1 of Site 2 is moved to the left side, the X position value for this position is -599Px. According to this, when the layer is moved to the right side, the X Position value is 599Px. If you prefer to have the center of the image at the screens border, use -/+ 960Px.

Before returning to WD, please check one thing in the Configuration tab. Per default Pandoras Box starts in Pixel Mode. If you like to send pixel values, tick the check box "Interpret Automation Param Input as Pixel Values (Input of Type Double only)" under PB Configuration > Unit Management 160. Otherwise sent data will be interpreted as Generic Units as explained in the Unit Management chapter.

Now, we can go back to WD and take care of the Range node. Range Filter Nodes allow to map the mouse position data to Pandoras Box Layer Position data.

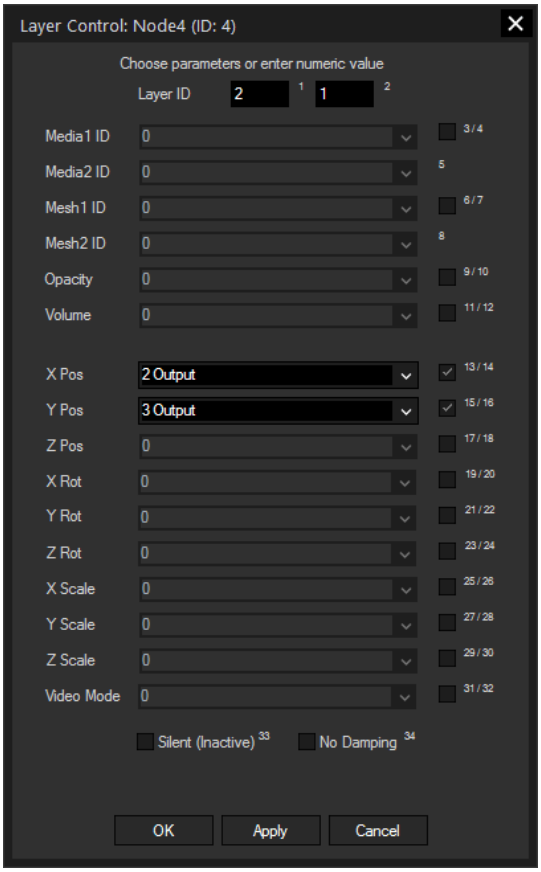


We start with the first Range Node that we set up to map the X position. Please double-click on the first Range Node.

Choose "X Pos" from the list as Input.
The Input Min and Input Max text fields are related to your mouse cursor X position.
For Input Min enter the min mouse X position value (keeps the value 0), assign the max mouse X position value for Input Max (read it from mouse input node properties, when your mouse cursor is moved to the right side of your screen).

Output Min and Max are related to Layer 2.1 in Pandoras Box.
For Output Min enter the value -960 (value is read out of Pandoras Box Layer 2.1 X Position, when the layer is moved to the left side of the screen) and for Output Max enter +960 (layers X position value at the right side of the screen). Press Apply / OK. The filter node properties should look like the left example.

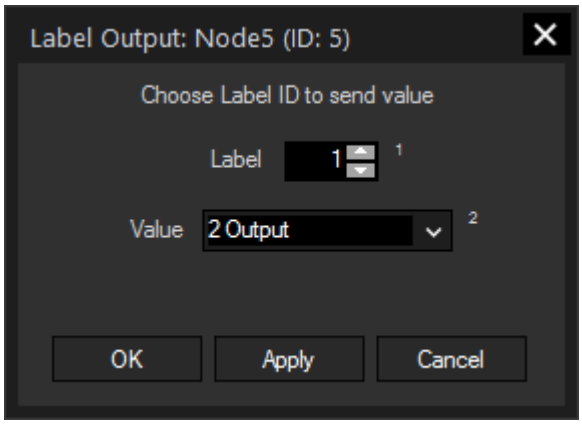
After preparing the translation of X position, the second range filter node needs to be set up for translation of Y position.
The mouse cursor at top of the screen delivers Y position value 0, at the bottom of the screen it delivers Y position value 1080 (according to your screen size).
The Layer in Pandoras Box gets the following values in this example: Y position on top of the screen is -540, Y position on bottom of the screen is 540.



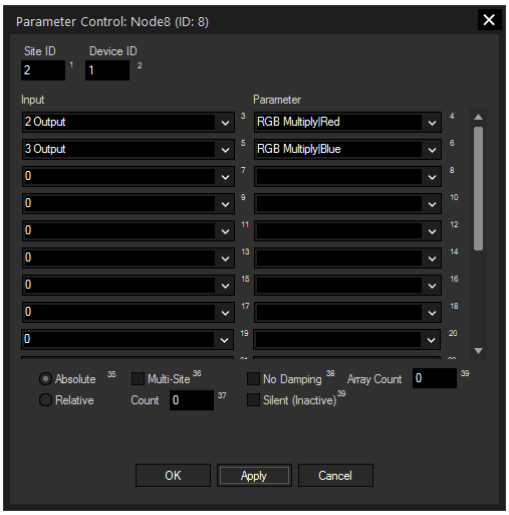
The last thing to do is setting up the Layer Control Output Node. Right-click on it and choose "Item Properties".

Enter the Layer ID on top of the dialog. In the first text field enter the site ID, in the second text field enter the layer ID.
Tick the check box next to X Pos and choose the Range Output from the list, that is set up for the X position.
Tick the check box next to Y Pos and choose the Range Output from the list, that is set up for the Y position.
Press Apply / OK.

Move your mouse cursor and the layer now will be mapped to it (make sure you are in the run mode [F8]).



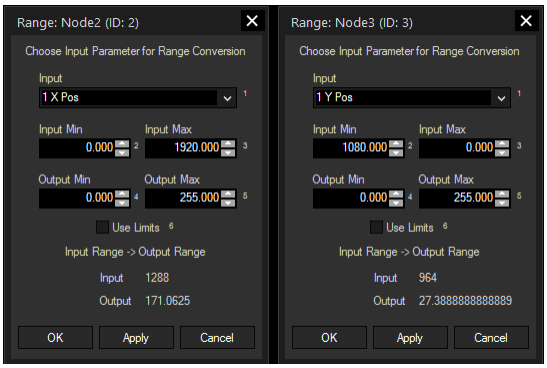
Lastly, to monitor the calculated pixel position, create two Labels next to the Label nodes. Open the Widgets menu and choose "Label". In the move/edit mode, the small ID in the Label shows its ID. If you have not created Labels before, they should have ID 1 and 2. Enter these IDs in the Label output nodes and choose the same values from the Range node as above. The Labels now display the same data that is sent to PB.

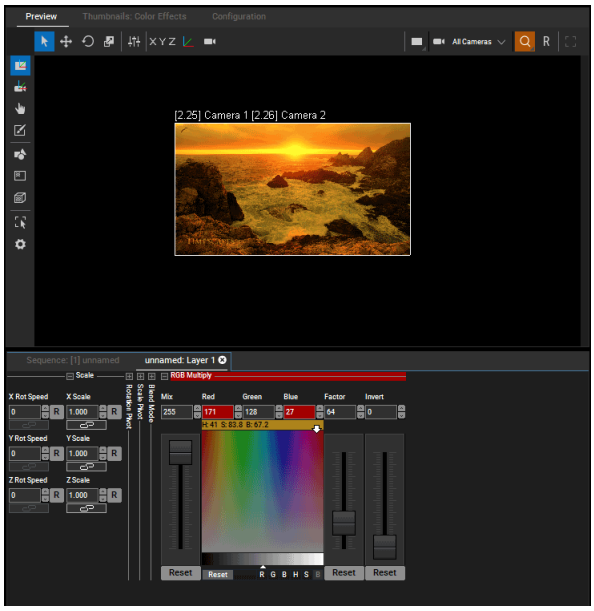


For more advanced programming e.g. to control other parameters in PB like effects, particles, lights, etc. you can use the Parameter Control Output Node, please see, Parameter Control. [1218]

The last part of the example shows the translation from the Mouse cursors position X and Y to Red and Blue value [0-255] of the RGB Multiply [399] effect which was added to Layer 2.1. Please refer to the chapter "Adding [323] FX" for more help with that.

The values for the Range nodes should be as depicted.





Finally, we have a look at the Preview window of Pandoras Box. The mouse cursor is located at the bottom right corner, the X Position is 1288Px and the Y Position is 964Px. The color result is orange, because the converted red value is 171 and blue: 27.

7.4.3 Syntax TCP- / UDP- / Serial Messages

Via TCP-, UDP- and COM Port nodes incoming or outgoing values can be entered as string, decimal or hexadecimal values.

Example:
Let's say, you would like to express a "Carriage Return" at the end of other values. This control character is used in many codes as a command to move to a new line. On a normal keyboard you have the possibility to press the key [Carriage Return], also known as the return key or enter key, the operating system understands this automatically and translates it into the correct code. When writing strings in ASCII or decimal or hexadecimal language you need to translate the command yourself. Please see the DEC HEX ASCII Table below to find the code for the character.

Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

Please note: Mostly, you need a "Line Feed" command as well. For two commands you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

Command in a Custom Script Button that sends a string via a TCP connection	Displayed string
TCPSend,1,Hello world, how are you ?	Hello world, how are you ?
TCPSend,1,Hello world,[CR LF]How are you [d191]	Hello world, How are you ¿

DEC	HEX	ASCII Symbol	Description
0	0	NUL	Null char
1	1	SOH	Start of Heading
2	2	STX	Start of Text
3	3	ETX	End of Text
4	4	EOT	End of Transmission
5	5	ENQ	Enquiry
6	6	ACK	Acknowledgment
7	7	BEL	Bell
8	8	BS	Back Space
9	9	HT	Horizontal Tab
10	0A	LF	Line Feed
11	0B	VT	Vertical Tab
12	0C	FF	Form Feed
13	0D	CR	Carriage Return
14	0E	SO	Shift Out / X-On
15	0F	SI	Shift In / X-Off
16	10	DLE	Data Line Escape
17	11	DC1	Device Control 1 (oft. XON)
18	12	DC2	Device Control 2

DEC	HEX	ASCII Symbol	Description
19	13	DC3	Device Control 3 (oft. XOFF)
20	14	DC4	Device Control 4
21	15	NAK	Negative Acknowledgement
22	16	SYN	Synchronous Idle
23	17	ETB	End of Transmit Block
24	18	CAN	Cancel
25	19	EM	End of Medium
26	1A	SUB	Substitute
27	1B	ESC	Escape
28	1C	FS	File Separator
29	1D	GS	Group Separator
30	1E	RS	Record Separator
31	1F	US	Unit Separator
32	20		Space
33	21	!	Exclamation mark
34	22	"	Double quotes (or speech marks)
35	23	#	Number
36	24	$	Dollar
37	25	%	Procenttecken
38	26	&	Ampersand
39	27	'	Single quote
40	28	(	Open parenthesis (or open bracket)
41	29	)	Close (or close bracket)
42	2A	*	Asterisk
43	2B	+	Plus
44	2C	,	Comma
45	2D	-	Hyphen
46	2E	.	Period, dot or full stop
47	2F	/	Slash or divide
48	30	0	Zero
49	31	1	One
50	32	2	Two
51	33	3	Three
52	34	4	Four
53	35	5	Five

DEC	HEX	ASCII Symbol	Description
54	36	6	Six
55	37	7	Seven
56	38	8	Eight
57	39	9	Nine
58	3A	:	Colon
59	3B	;	Semicolon
60	3C	<	Less than (or open angled bracket)
61	3D	=	Equals
62	3E	>	Greater than (or close angled bracket)
63	3F	?	Question mark
64	40	@	At symbol
65	41	A	Uppercase A
66	42	B	Uppercase B
67	43	C	Uppercase C
68	44	D	Uppercase D
69	45	E	Uppercase E
70	46	F	Uppercase F
71	47	G	Uppercase G
72	48	H	Uppercase H
73	49	I	Uppercase I
74	4A	J	Uppercase J
75	4B	K	Uppercase K
76	4C	L	Uppercase L
77	4D	M	Uppercase M
78	4E	N	Uppercase N
79	4F	O	Uppercase O
80	50	P	Uppercase P
81	51	Q	Uppercase Q
82	52	R	Uppercase R
83	53	S	Uppercase S
84	54	T	Uppercase T
85	55	U	Uppercase U
86	56	V	Uppercase V
87	57	W	Uppercase W
88	58	X	Uppercase X

DEC	HEX	ASCII Symbol	Description
89	59	Y	Uppercase Y
90	5A	Z	Uppercase Z
91	5B	[	Opening bracket
92	5C	\	Backslash
93	5D	]	Closing bracket
94	5E	^	Caret - circumflex
95	5F	_	Underscore
96	60	`	Grave - accent
97	61	a	Lowercase a
98	62	b	Lowercase b
99	63	c	Lowercase c
100	64	d	Lowercase d
101	65	e	Lowercase e
102	66	f	Lowercase f
103	67	g	Lowercase g
104	68	h	Lowercase h
105	69	i	Lowercase i
106	6A	j	Lowercase j
107	6B	k	Lowercase k
108	6C	l	Lowercase l
109	6D	m	Lowercase m
110	6E	n	Lowercase n
111	6F	o	Lowercase o
112	70	p	Lowercase p
113	71	q	Lowercase q
114	72	r	Lowercase r
115	73	s	Lowercase s
116	74	t	Lowercase t
117	75	u	Lowercase u
118	76	v	Lowercase v
119	77	w	Lowercase w
120	78	x	Lowercase x
121	79	y	Lowercase y
122	7A	z	Lowercase z
123	7B	{	Opening brace

DEC	HEX	ASCII Symbol	Description
124	7C	|	Vertical bar
125	7D	}	Closing brace
126	7E	~	Equivalency sign - tilde
127	7F		Delete
128	80	€	Euro sign
129	81		
130	82	‚	Single low-9 quotation mark
131	83	ƒ	Latin small letter f with hook
132	84	„	Double low-9 quotiation mark
133	85	…	Horizontal ellipsis
134	86	†	Dagger
135	87	‡	Double dagger
136	88	ˆ	Modifier letter circumflex accent
137	89	‰	Per mille sign
138	8A	Š	Latin capital letter S with caron
139	8B	‹	Single left-pointing angle quotation
140	8C	OE	Latin capital ligature OE
141	8D		
142	8E	Ž	Latin capital letter Z with caron
143	8F		
144	90		
145	91	‘	Left single quotation mark
146	92	'	Right single quotation mark
147	93	"	Left double quotation mark
148	94	"	Right double quotation mark
149	95	•	Bullet
150	96	–	En dash
151	97	—	Em dash
152	98	˜	Small tilde
153	99	™	Trade mark sign
154	9A	š	Latin small letter S with caron
155	9B	›	Single right-pointing angle quotation mark
156	9C	oe	Latin small ligature oe
157	9D		
158	9E	ž	Latin small letter z with caron

DEC	HEX	ASCII Symbol	Description
159	9F	Ÿ	Latin capital letter Y with diaeresis
160	A0		Non-breaking space
161	A1	¡	Inverted exclamation mark
162	A2	¢	Cent sign
163	A3	£	Pound sign
164	A4	¤	Currency sign
165	A5	¥	Yen sign
166	A6	¦	Pipe, Broken vertical bar
167	A7	§	Section sign
168	A8	¨	Spacing diaeresis - umlaut
169	A9	©	Copyright sign
170	AA	ª	Feminine ordinal indicator
171	AB	«	Left double angle quotes
172	AC	¬	Not sign
173	AD	Soft	hyphen
174	AE	®	Registered trade mark sign
175	AF	¯	Spacing macron - overline
176	B0	°	Degree sign
177	B1	±	Plus-or-minus sign
178	B2	²	Superscript two - squared
179	B3	³	Superscript three - cubed
180	B4	´	Acute accent - spacing acute
181	B5	µ	Micro sign
182	B6	¶	Pilcrow sign - paragraph sign
183	B7	·	Middle dot - Georgian comma
184	B8	¸	Spacing cedilla
185	B9	¹	Superscript one
186	BA	º	Masculine ordinal indicator
187	BB	»	Right double angle quotes
188	BC	¼	Fraction one quarter
189	BD	½	Fraction one half
190	BE	¾	Fraction the quarters
191	BF	¿	Inverted question mark
192	C0		Latin capital letter A with grave
193	C1	Á	Latin capital letter A with acute

DEC	HEX	ASCII Symbol	Description
194	C2	Â	Latin capital letter A with circumflex
195	C3	Ã	Latin capital letter A with tilde
196	C4	Ä	Latin capital letter A with diaeresis
197	C5	Å	Latin capital letter A with ring above
198	C6	Æ	Latin capital letter AE
199	C7	Ç	Latin capital letter C with cedilla
200	C8	È	Latin capital letter E with grave
201	C9	É	Latin capital letter E with acute
202	CA	Ê	Latin capital letter E with circumflex
203	CB	Ë	Latin capital letter E with diaeresis
204	CC	Ì	Latin capital letter I with grave
205	CD	Í	Latin capital letter I with acute
206	CE	Î	Latin capital letter I with circumflex
207	CF	Ï	Latin capital letter I with diaeresis
208	D0	Ð	Latin capital letter ETH
209	D1	Ñ	Latin capital letter N with tilde
210	D2	Ò	Latin capital letter O with grave
211	D3	Ó	Latin capital letter O with acute
212	D4	Ô	Latin capital letter O with circumflex
213	D5	Õ	Latin capital letter O with tilde
214	D6	Ö	Latin capital letter O with diaeresis
215	D7	×	Multiplication sign
216	D8	Ø	Latin capital letter O with slash
217	D9	Ù	Latin capital letter U with grave
218	DA	Ú	Latin capital letter U with acute
219	DB	Û	Latin capital letter U with circumflex
220	DC	Ü	Latin capital letter U with diaeresis
221	DD	Ý	Latin capital letter Y with acute
222	DE	Þ	Latin capital letter THORN
223	DF	ß	Latin small letter sharp s - ess-zed
224	E0	>	Latin small letter a with grave
225	E1	á	Latin small letter a with acute
226	E2	â	Latin small letter a with circumflex
227	E3	ã	Latin small letter a with tilde
228	E4	ä	Latin small letter a with diaeresis

DEC	HEX	ASCII Symbol	Description
229	E5	å	Latin small letter a with ring above
230	E6	æ	Latin small letter ae
231	E7	ç	Latin small letter c with cedilla
232	E8	è	Latin small letter e with grave
233	E9	é	Latin small letter e with acute
234	EA	ê	Latin small letter e with circumflex
235	EB	ë	Latin small letter e with diaeresis
236	EC	ì	Latin small letter i with grave
237	ED	í	Latin small letter i with acute
238	EE	î	Latin small letter i with circumflex
239	EF	ï	Latin small letter i with diaeresis
240	F0	ð	Latin small letter eth
241	F1	ñ	Latin small letter n with tilde
242	F2	ò	Latin small letter o with grave
243	F3	ó	Latin small letter o with acute
244	F4	ô	Latin small letter o with circumflex
245	F5	õ	Latin small letter o with tilde
246	F6	ö	Latin small letter o with diaeresis
247	F7	÷	Division sign
248	F8	ø	Latin small letter o with slash
249	F9	ù	Latin small letter u with grave
250	FA	ú	Latin small letter u with acute
251	FB	û	Latin small letter u with circumflex
252	FC	ü	Latin small letter u with diaeresis
253	FD	ý	Latin small letter y with acute
254	FE	þ	Latin small letter thorn
255	FF	ÿ	Latin small letter y with diaeresis

7.4.4 Input Nodes

Input nodes provide numeric or text based input values. The data from an Input node can then be used in another Filter or Output node. Please see the introductory chapter if you like to learn more about other node types or how to create and work with nodes [937] in general.

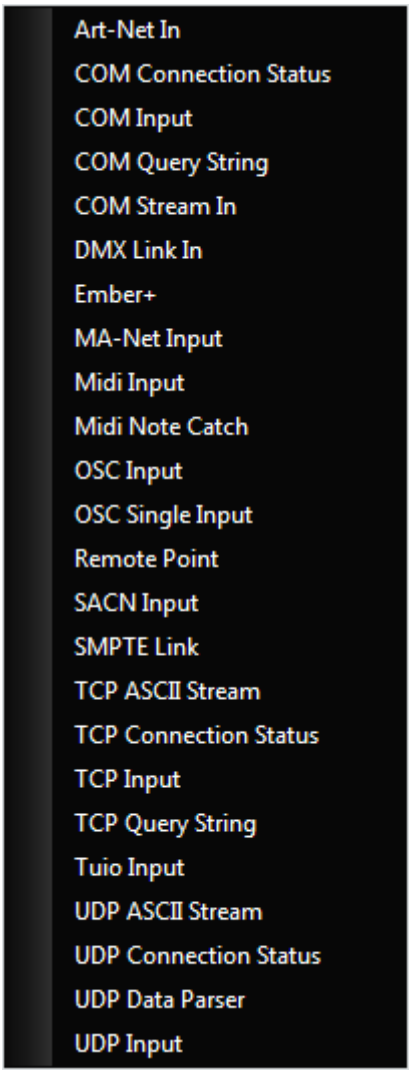
The following sub chapters describe the various Input nodes sorted in different categories.

7.4.4.1 Connections Input

The Connection input nodes receive data from listening to different kinds of input protocols, mostly over network. All of the here listed connections can be set in the Connection Manager [1258], except Ember+ which only needs a valid IP connection that is set up in the node configuration itself.

Please note that OSC and SACN are a UDP based protocols and thus need a UDP connection [1267] set up.

The available nodes are:

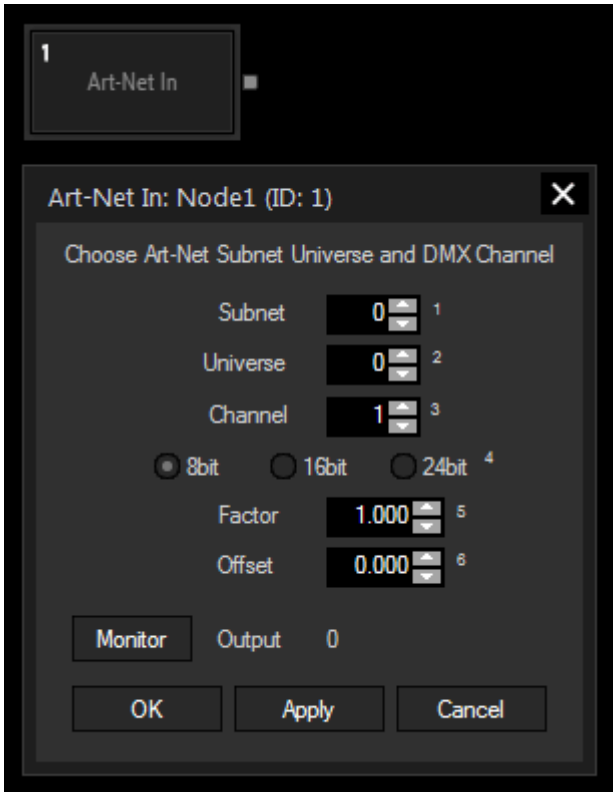


Art-Net In [953]
COM Connection Status
COM Input [954]
COM Query String [957]
COM Stream In [956]
DMX Link In [958]
Ember+
MA-Net Input
Midi Input [959]
Midi Note Catch [960]
OSC Input [961]
OSC Single Input [962]
Remote Point [963]
SACN Input [963]
SMPTE Link [965]
TCP ASCII Stream [965]
TCP Connection Status [967]
TCP Input [968]
TCP Query String [969]
Tuio Input [971]
UDP ASCII String [972]
UDP Connection Status [973]
UDP Data Parser [974]
UDP Input [975]

7.4.4.1.1 Art-Net Input

The Art-Net input node lets you choose an 8,16 or 24 bit DMX value by choosing the desired source Subnet and Universe.

This node can be found under Nodes > Input > Connections > Art-Net Input



Node Properties

Subnet:
Enter subnet ID.

Universe:
Enter the Universe ID.

Channel:
Enter the Channel ID.

8bit / 16 bit / 24bit:
Choose if you want to get a eight, sixteen or twenty-four bit DMX value.

Factor:
Multiplies the value received with this factor.

Offset:
Adds the specified offset to the values received.

Monitor:
Click here to open the Art-Net Monitor 2051.
Confirm your settings by pressing OK or Apply.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

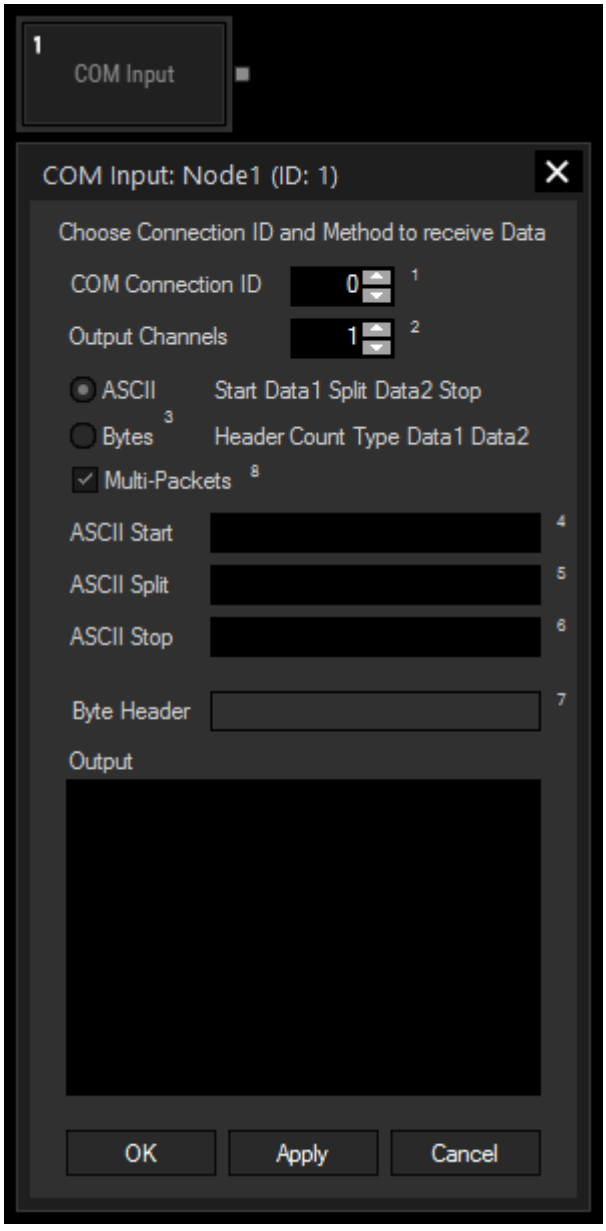
The Node generates the following output:

- DMX Channel Value

7.4.4.1.2 COM Input

The Com port input node allows receiving either ASCII or Byte values over a local COM Port connection. The COM Port connection needs to be enabled in the Connection Manager. [1258]
This input node can output by default any incoming packet as text if no ASCII filter is applied.

This node can be found under Nodes > Input > Connections > Com Input



Node Properties

COM Connection ID:
Enter the ID of the COM Port connection [1258].

Output Channels:
Set the Amount of Output Channels of this Input Node.
Input Value 1 will output Data 1, Input Value 2 will output Data 2, etc.

ASCII Mode:
The ASCII Mode allows creating a custom ASCII protocol for data transfer.
Multiple data can be separated by the split string.
To determine the Start and End of the message, use the Start and Stop strings.

ASCII Start, Split and Stop:
Enter values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages 944.

Bytes Mode:
The Byte protocol allows data transmission based on a byte stream.
First send the **header** with any amount of Bytes.
Followed by the **data count** as 4 Byte Integer. Each piece of data, i.e each 4 Byte Integer or each 8 Byte double, counts as one. Please take into account that the processing uses a little-endian Byte order, meaning that the lowest byte comes first.
Followed by a single Byte to describe the **data type** used (1 = 4byte integer, 2 = 8byte double).
Followed by one or multiple **data** sets of the specified type, also in little-endian Byte order.

Example: This byte stream transmits one integer with value 5000, the header is "WD!":

[87 68 33 1 0 0 0 1 136 19 0 0]

- Header: 87 68 33
- data count: 1 0 0 0
- data type: 1
- data: 136 19 0 0

Byte Header:
Enter values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages 944.

Multi-Packets:
If the COM packets are being received in parts, this option allows reading all part packets as one.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

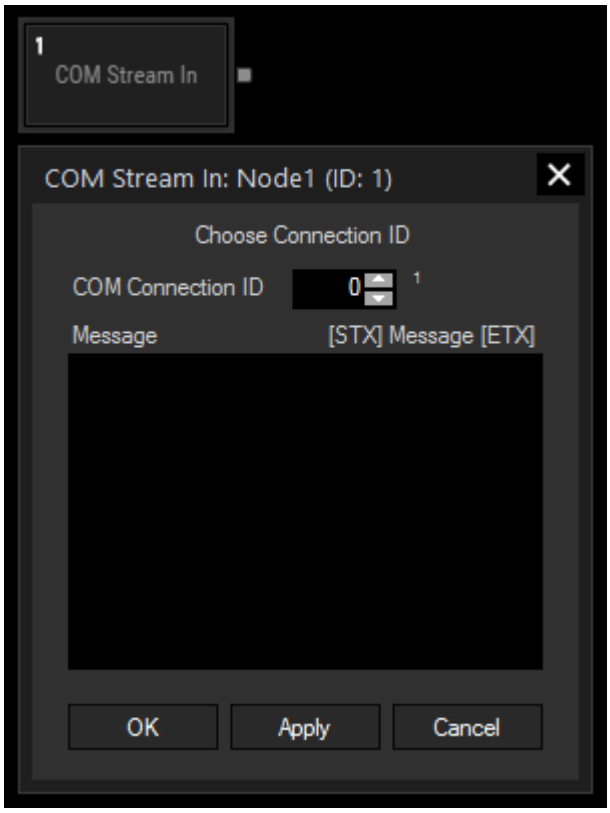
The Node generates the following output:

- Input Value 1 up to Input Value X (X = number of entered output channels).

7.4.4.1.3 COM Stream Input

The Com ASCII Stream input node allows receiving ASCII values over a local COM Port connection. The COM Port connection needs to be enabled in the Connection Manager [1258].
This input node can output by default any incoming packet as text if no ASCII filter is applied.

This node can be found under Nodes > Input > Connections > COM Stream Input



Node Properties

COM Connection ID:
Enter the ID of the COM Port connection [1258].

Message:
The received message between Start [STX] and End [ETX] symbol is displayed here.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

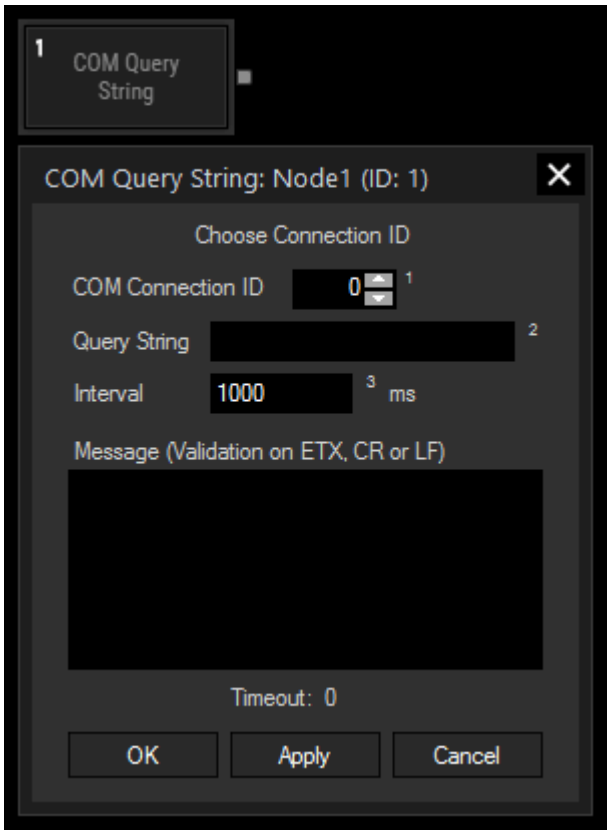
The Node generates the following output:

- Stream Message as String

7.4.4.1.4 COM Query String Input

The Com Query String input node allows receiving ASCII feedback messages from COM devices over a local COM Port connection. The COM Port connection needs to be enabled in the Connection Manager [1258].
This input node will output any incoming packet as Text Only, Numeric Only or mixed string.

This node can be found under Nodes > Input > Connections > COM Query String Input



Node Properties

COM Connection ID:
Enter the ID of the COM Port connection [1258].

Query String:
Enter the query string that you want to send to the specified COM Device.

Interval:
Enter the time interval in ms in which the query string will be sent to the COM Device.
If you want to query the device manually (e.g. by pressing a Custom Script Button using the command "WDNodeSetParam(NodeID,ParamID,Value) [1782]"), please set the Interval Time to 0 ms.

Message:
The response of the COM Device will be displayed here. Please note that the responding device needs to send a "carriage return", a "line feed" or an "end of text" at the end of its message. This can be done in ASCII, decimal or hexadecimal language [944].
- carriage return is either [CR] or [d13] or [h0D]
- line feed is either [LF] or [d3] or [h3]
- end of text is either [ETX] or [d10] or [h0A]

Timeout:
The Timeout gives you feedback about the connection status. If Timeout is "0", there is no communication with the specified COM Device. If Timeout is "1" the communication works.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

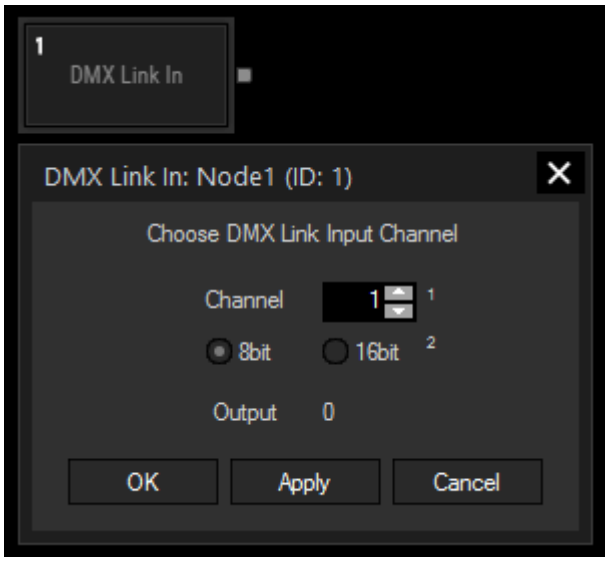
Node output values

The Node generates the following output:

- Response
- Text Only
- Numeric Only
- Timeout (0 for "no communication", 1 for "successful communication")

7.4.4.1.5 DMX Link Input

The DMX Link input node provides all incoming DMX data via the Pandoras Box DMX Link interface.



Node Properties

Channel:
Choose the DMX channel and if it is a 8bit or a 16bit value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

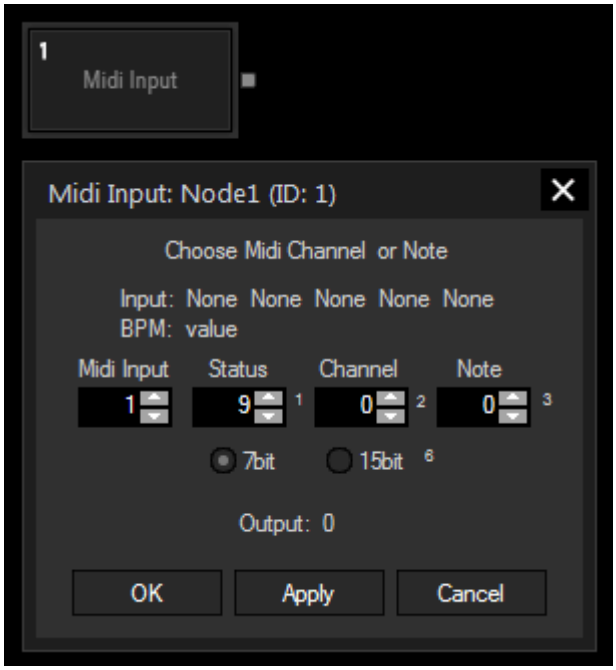
The Node generates the following output:

- DMX Value

7.4.4.1.6 Midi Input

The Midi input node allows receiving 7bit or 15bit input values as well as Note On/Off and raw byte messages.
The Midi connection needs to be enabled in the Connection Manager [1258].

This node can be found under Nodes > Input > Connections > Midi Input



Node Properties

Midi Input:
Enter here the ID of your Midi Input device as assigned at the Midi Connection Manager.

Add the **Status**, **Channel** and **Note** value.

Choose between **7bit** or **15bit** input values.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

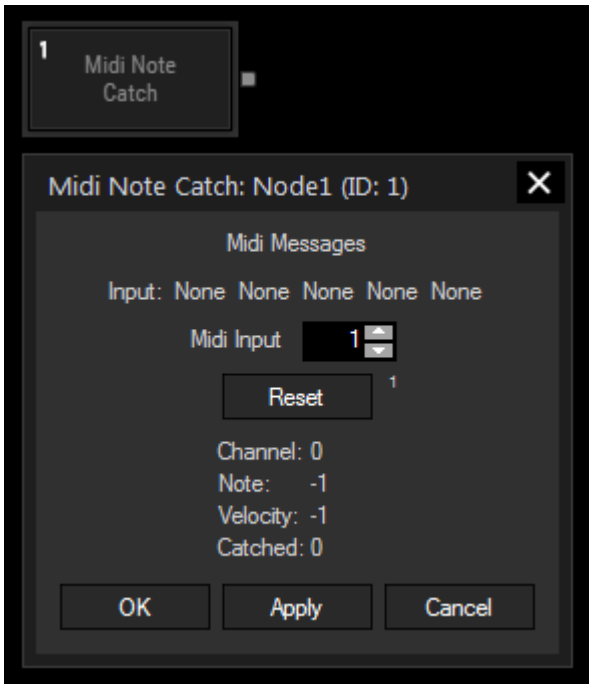
The Node generates the following output:

- Current Value
- Trigger
- Status Byte
- Channel Byte
- DataByte 1 to DataByte 3
- BPM

7.4.4.1.7 Midi Note Catch Input

The node catches and outputs the first incoming Midi Note Message.
This is in particular useful for game scenarios where WD needs to identify the first incoming Midi Note event.

This node can be found under Nodes > Input > Connections > Midi Input



Node Properties

Midi Input:
Enter here the ID of your Midi Input as assigned at the Midi Connection Manager.

This node can be reset by setting Parameter 1 to 0 via the command "WDNodeSetParam".

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

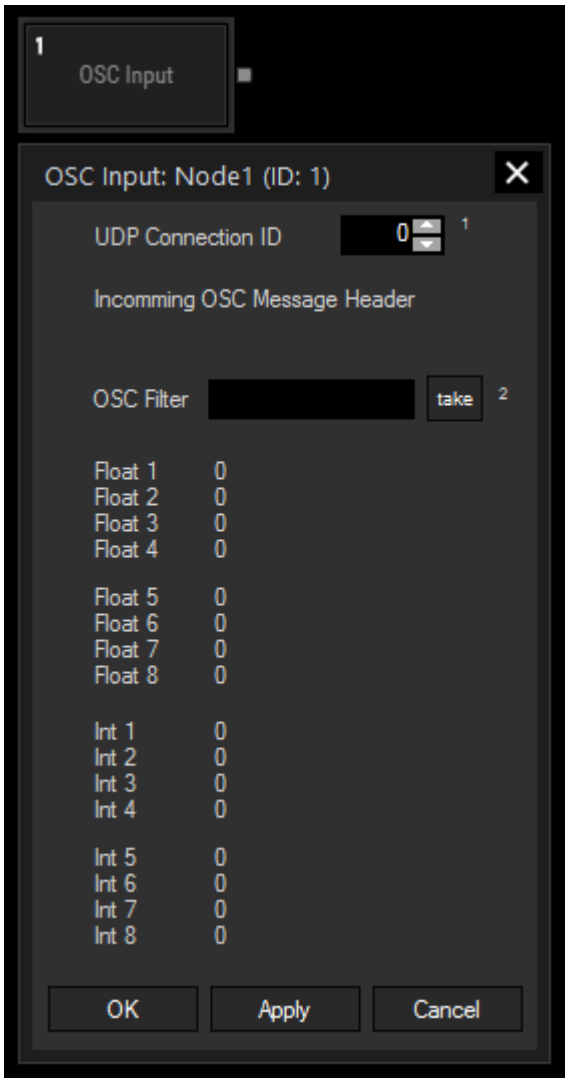
The Node generates the following output:

- Channel
- Note
- Velocity
- isCatched

7.4.4.1.8 OSC Input

The OSC Input node lets you receive data as bundled messages via the Open Sound Control Protocol, e.g. from iPhone, iPad or iPod (tested with Touch OSC).
A UDP Broadcast Input Connection is required, see Connection Manager[1258].

This node can be found under Nodes > Input > Connections > OSC Input



Node Properties

UDP Connection ID:
Choose the UDP Connection to listen to. The UDP Connections can be found and setup in the Connection Manager[1258].

Incomming OSC Message Header:
The header of the incoming OSC message on which the filter can be applied is displayed here.

OSC Filter:
To assign messages with the displayed header to this OSC node (e.g. sent from a fader or button in OSC Touch), enter complete header or the part of the header you want to filter your messages with and press **take**.
Float1 to Float8 and Int1 to Int8 will now display the message details.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

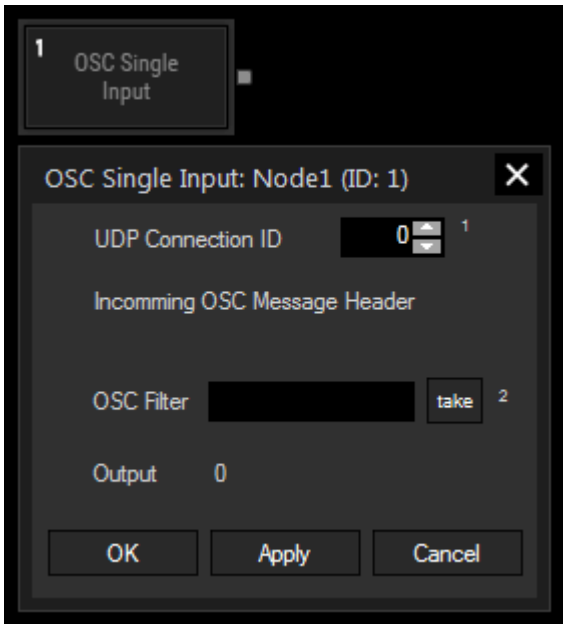
- Float 1-8
- Int 1-8
- Msg (the header of the received package if it does not comply with the set OSC Filter)

7.4.4.1.9 OSC Single Input

The OSC Single Input node lets you receive data via the Open Sound Control Protocol, e.g. from tablet devices, smartphones or touch monitors.
Unlike the OSC Input node [961] which allows receiving bundle messages, this node only processes single OSC messages.

A UDP Broadcast Input Connection is required, see Connection Manager [1258].

This node can be found under Nodes > Input > Connections > OSC Single Input



Node Properties

UDP Connection ID:
Enter here the ID of your UDP connection.

OSC Filter:
Enter here the address of the OSC message you want to filter, e.g. /synthesizer/filter/abc

Please note that the OSC Single Input node is currently not able to process string values (except the string consists of numbers).

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

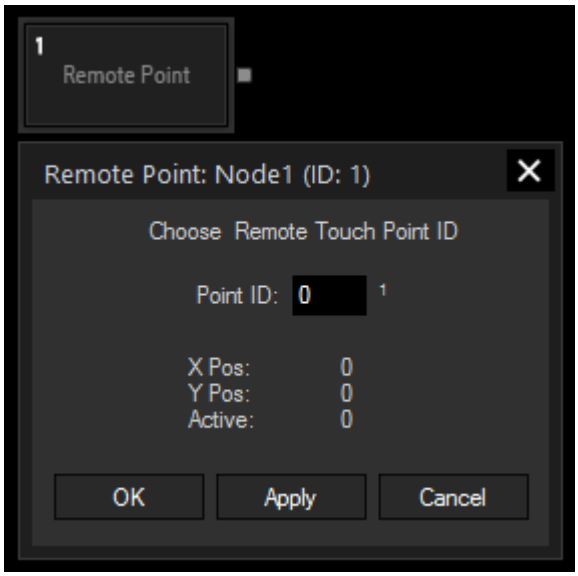
The Node generates the following output:

- Value

7.4.4.1.10 Remote Touch Input

The Remote Touch input node allows processing any incoming remote touch point [1260].

The remote input needs to be set up in the Connection Manager [1258].



Node Properties

Point ID:
Enter here the ID of the incoming remote touch point you want to use for further processing.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

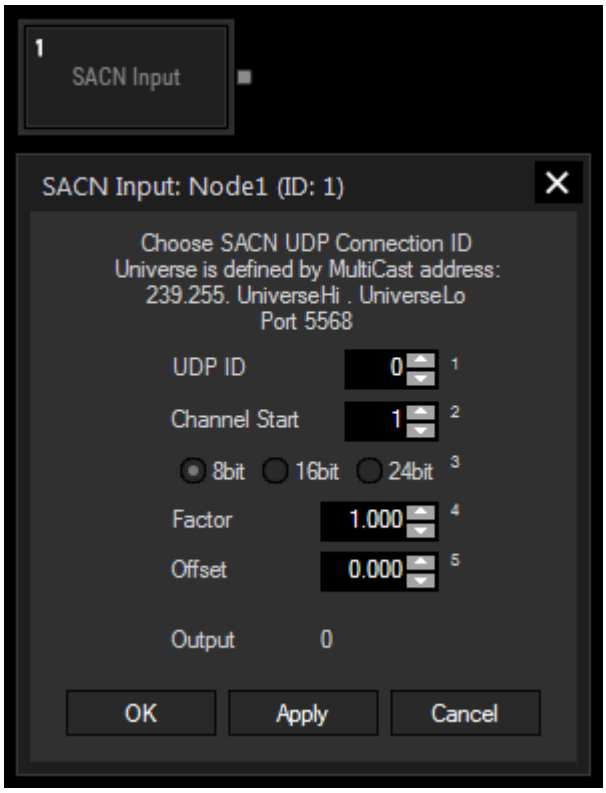
- X Pos
- Y Pos
- Active ("1" for active, "0" for non-active)

7.4.4.1.11 SACN Input

The SACN input node allows receiving packets in sACN format. sACN or Screaming ACN is a protocol for transporting the data of DMX data over IP networks.
A valid UDP connection [1267] must be set up to the device sending SACN packets. The IP address this connection needs to listen to correlates with the universe number transmitted. The first two numbers of the address need to be the MultiCast address "239.255.", the last two numbers form as HighByte and LowByte the universe number. E.g. The equivalent of Universe "1" would then be the address "239.255.0.1".

The Port used is 5568.

This node can be found under Nodes > Input > Connections > SACN Input



Node Properties

UDP ID:
Choose the UDP Connection to listen to. The UDP Connections can be found and setup in the Connection Manager [1258].

Channel Start:
Set Channel number you want to listen to. If you select 16bit or 24bit values, this is the number of the first Channel.

8bit / 16bit / 24 bit:
Choose if you want to get a eight, sixteen or twenty-four bit DMX value.

Factor:
Enter a factor with which the output value should be multiplied.

Offset:
Enter an offset that should be added to the output value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

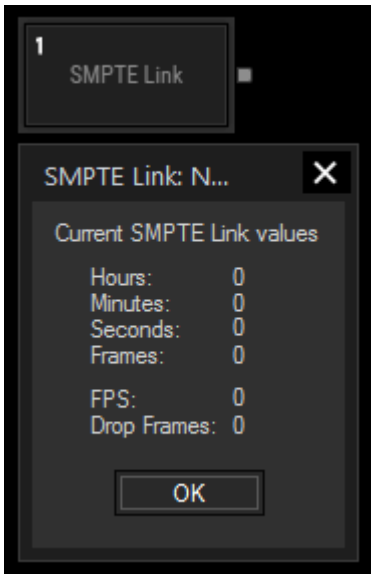
The Node generates the following output:

- Value (received DMX value, including Factor and Offset)

7.4.4.1.12 SMPTE Link Input

The SMPTE Link input node allows reading and using timecode within Widget Designer.
The SMPTE connection needs to be enabled in the Connection Manager [1258].

This node can be found under Nodes > Input > Connections > SMPTE Link



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

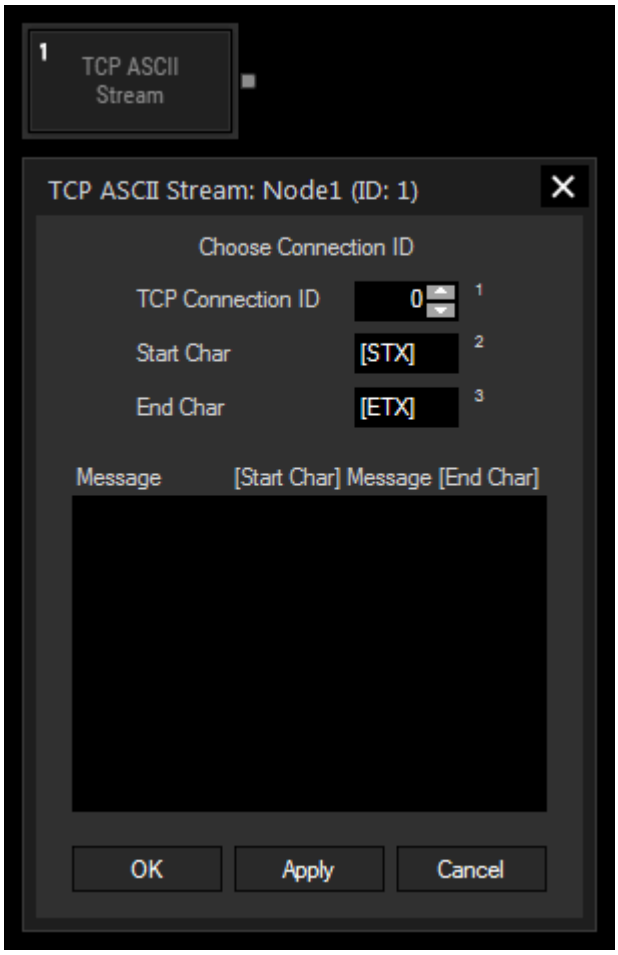
The Node generates the following output:

- Hours
- Minutes
- Seconds
- Frames
- FPS
- Drop Frames

7.4.4.1.13 TCP ASCII Stream Input

The TCP ASCII Stream input node allows receiving either ASCII values based on the selected TCP connection. It can output by default any incoming packet as text if no ASCII filter is applied.
The TCP Connections can be found and setup in the Connection Manager [1258].

This node can be found under Nodes > Input > Connections > TCP ASCII Input



Node Properties

TCP Connection ID:
Choose the TCP Connection to listen to. The TCP Connections can be found and setup in the Connection Manager [1258].

Start / End Char:
Enter the characters indicating start and end of the TCP message. You can enter the values as string, dec or hex value Please refer to Syntax TCP- / UDP- / Serial messages [944].

Message:
The received message is displayed here.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

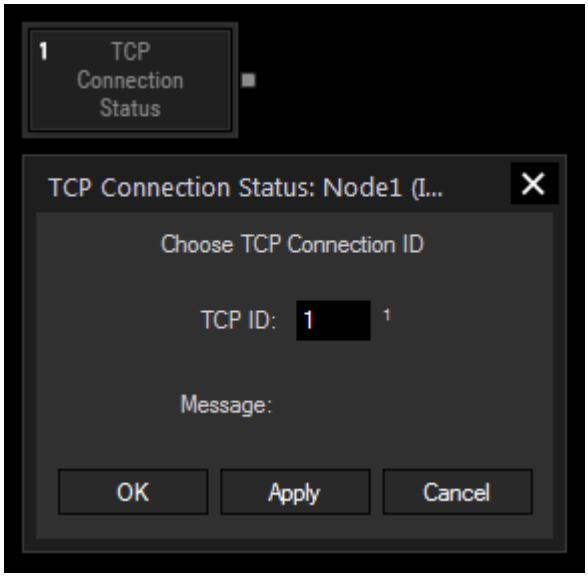
The Node generates the following output:

- TCP Stream as String.

7.4.4.1.14 TCP Connection Status

The TCP Connection Status Node provides the connection status of the selected TCP connection.

This node can be found under Nodes > Input > Connections > TCP Connection Status



Node properties

TCP ID:
Enter the ID of the TCP connection to be watched.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

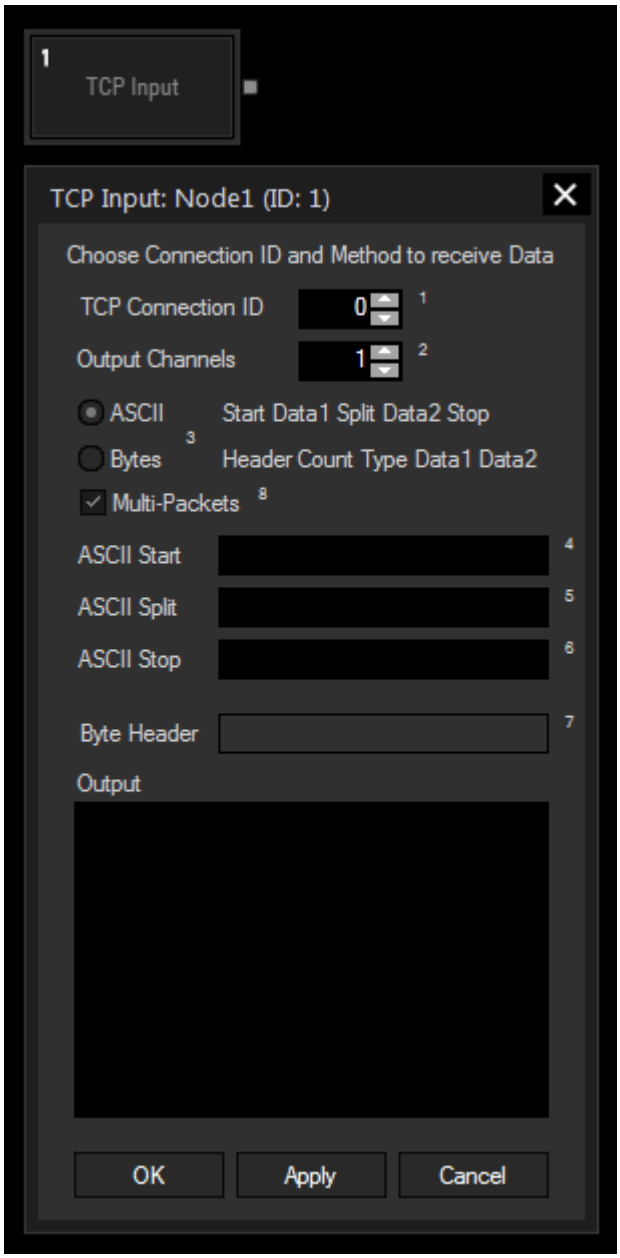
The Input node generates the following output:

- Message: Waiting for Connection, Connected, Client stopped

7.4.4.1.15 TCP Input

The TCP input node allows receiving either ASCII or Byte values based on the selected TCP connection. It can output by default any incoming packet as text if no ASCII filter is applied.
The TCP Connections can be found and setup in the Connection Manager [1258].

This node can be found under Nodes > Input > Connections > TCP Input



Node Properties

TCP Connection ID:
Choose the TCP Connection to listen to. The TCP Connections can be found and setup in the Connection Manager [1258].

Output Channels:
Set the Amount of Output Channels of this Input Node.
Input Value 1 will output Data 1, Input Value 2 will output Data 2, etc.

ASCII Mode:
The ASCII Mode allows creating a custom ASCII protocol for data transfer.
Multiple data can be separated by the split string.
To determine the Start and End of the message, use the Start and Stop strings.

ASCII Start, Split and Stop:
Enter values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages [944].

Bytes Mode:
The Byte protocol allows data transmission based on a byte stream.
First send the **header** with any amount of Bytes.
Followed by the **data count** as 4 Byte Integer. Each piece of data, i.e each 4 Byte Integer or each 8 Byte double, counts as one. Please take into account that the processing uses a little-endian Byte order, meaning that the lowest byte comes first.
Followed by a single Byte to describe the **data type** used (1 = 4byte integer, 2 = 8byte double).
Followed by one or multiple **data** sets of the specified type, also in little-endian Byte order.

Example: This byte stream transmits one integer with value 5000, the header is "WD!":

[87 68 33 1 0 0 0 1 136 19 0 0]

- Header: 87 68 33
- data count: 1 0 0 0
- data type: 1
- data: 136 19 0 0

Byte Header:
Enter values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages [944].

Multi-Packets:
If the TCP packets are being received in parts, this option allows reading all part packets as one.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

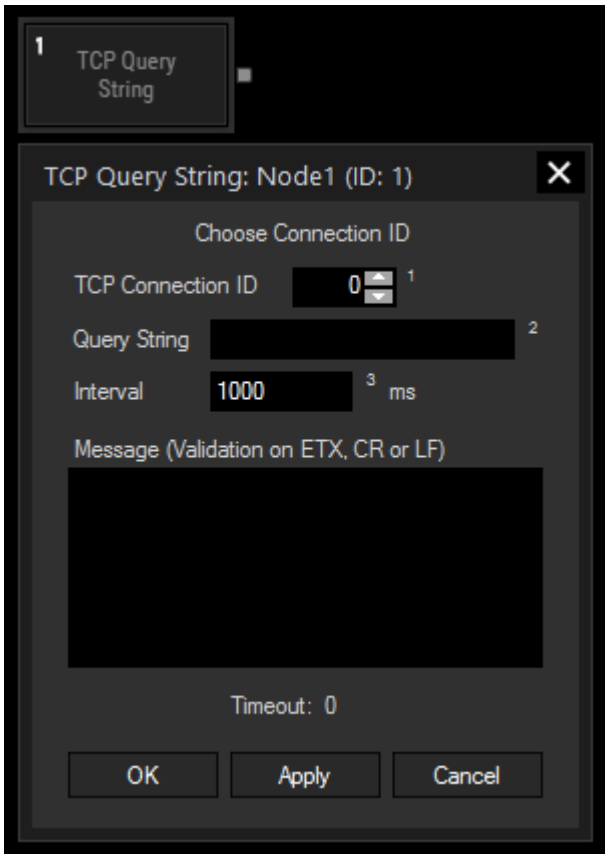
- Value 1 - Value X (X = Amount of Output Channels).

7.4.4.1.16 TCP Query String Input

The TCP Query String input node allows receiving ASCII feedback messages from devices over a TCP connection. The TCP connection needs to be enabled in the Connection Manager [1258].

This input node will output any incoming packet as Text Only, Numeric Only or mixed string.

This node can be found under Nodes > Input > Connections > TCP Query String



Node Properties

COM Connection ID:
Enter the ID of the TCP Connection [1265].

Query String:
Enter the query string that you want to send to the specified TCP device.

Interval:
Enter the time interval in ms in which the query string will be sent to the TCP device.
If you want to query the device manually (e.g. by pressing a Custom Script Button using the command "WDNodeSetParam(NodeID,ParamID,Value) [1782]"), please set the Interval Time to 0 ms.

Message:
The response of the TCP device will be displayed here. Please note that the responding device needs to send a "carriage return", a "line feed" or an "end of text" at the end of its message. This can be done in ASCII, decimal or hexadecimal language [944].
- carriage return is either [CR] or [d13] or [h0D]
- line feed is either [LF] or [d3] or [h3]
- end of text is either [ETX] or [d10] or [h0A]

Timeout:
The Timeout gives you feedback about the connection status. If Timeout is "0", there is no communication with the specified TCP device. If Timeout is "1" the communication works.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

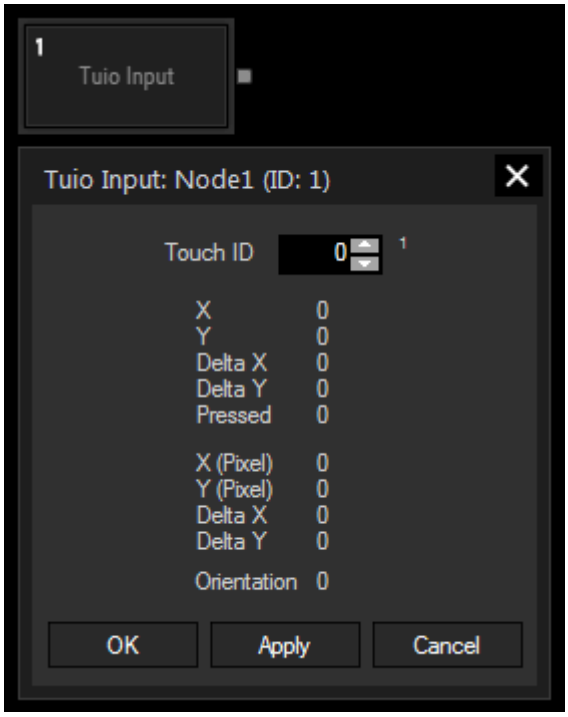
- Response
- Text Only
- Numeric Only
- Timeout (0 for "no communication", 1 for "successful communication")

7.4.4.1.17 Tuio Input

The Tuio Input node enables you to receive various touch point information transmitted by a TUIO touch device.

You need to set up the TUIO Receiver [1261] in the Connection Manager [1258] before using this node.

This node can be found under Nodes > Input > Connections > Tuio Input



Node Properties

Touch ID:
Enter here the ID of the touch point you want to track.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

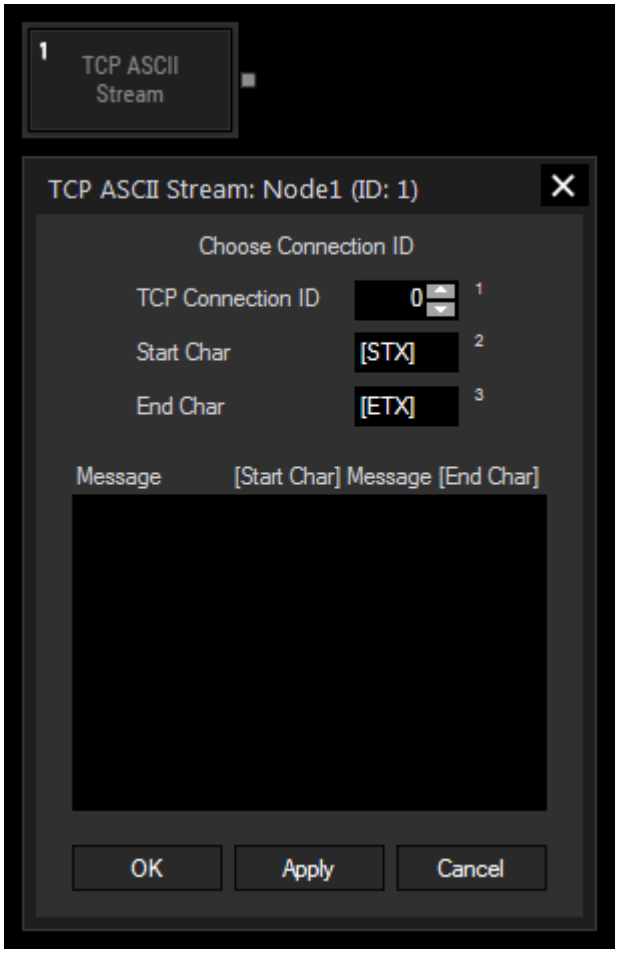
- X
- Y
- Delta X
- Delta Y
- Pressed (yes = 1 / no = 0)

- X (Pixel)
- Y (Pixel)
- Delta X (Pixel)
- Delta Y (Pixel)
- Orientation

7.4.4.1.18 UDP ASCII Stream Input

The UDP ASCII Stream input node allows receiving ASCII values based on the selected UDP connection. It can output by default any incoming packet as text if no ASCII filter is applied.
The UDP connection can be found and set up in the Connection Manager [1258].

This node can be found under Nodes > Input > Connections > UDP ASCII Input



Node Properties

UDP Connection ID:
Choose the UDP Connection to listen to. The UDP Connections can be found and setup in the Connection Manager [1258].

Start / End Char:
Enter the characters indicating start and end of the UDP message. You can enter the values as string, dec or hex value Please refer to Syntax TCP- / UDP- / Serial messages [944].

Message:
The received message is displayed here.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

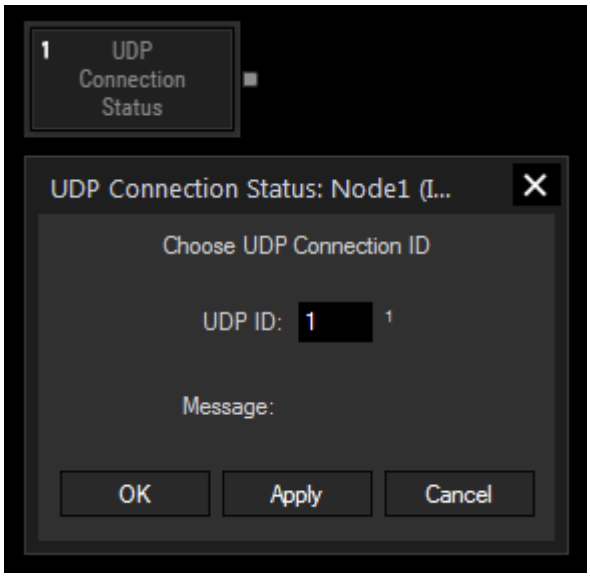
The Node generates the following output:

- UDP Stream as String

7.4.4.1.19 UDP Connection Status

The UDP Connection Status Node provides the connection status of the current UDP connection.

This node can be found under Nodes > Input > Connections > UDP Connection Status



Node properties

UDP ID:
Enter the ID of the UDP connection to be watched.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

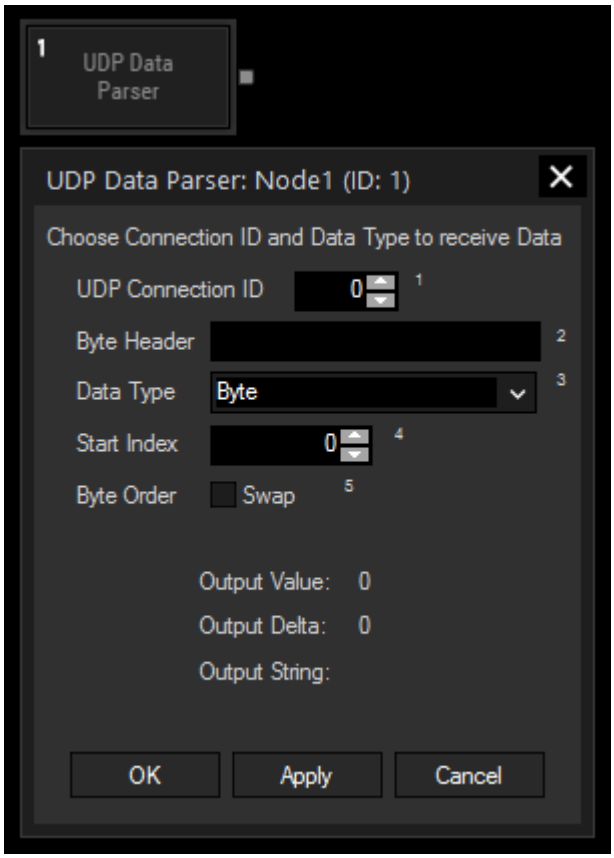
The Input node generates the following output:

- Message: Started, Stopped

7.4.4.1.20 UDP Data Parser

The UDP Data Parser node enables you to interpret incoming byte streams as values of different data types. While the UDP Input node [975] can only interpret incoming packets and streams as ASCII characters or bytes, this node additionally offers numeric data types such as integer, double and single/float.

This node can be found under Nodes > Input > Connections > UDP Data Parser



Node properties

UDP ID:
Enter the ID of the UDP connection to be watched.

Byte Header:
Enter here the header as ASCII, Hex or Dec value. Please refer to the chapter Syntax TCP- / UDP- / Serial Messages [944] for more information about the syntax to be used or hover your mouse over the field.

Data Type:
Select here the data type your message is supposed to be interpreted with. Please keep in mind that **Integer and Single/Float require four bytes** (32bit) each, **Double eight bytes** (64bit) and **Byte exactly one byte**. If the message contains a String, the character string needs to be **terminated by a Null byte**.

Start Index:
Enter here the offset of bytes that should be applied to extract the needed value. The counting starts with 0 for the first byte of the header. So if your header is e.g. "!WD", followed by the value, your Start Index needs to be set to 3.

Byte Order:
Check this box to swap the byte order of the received message for parsing.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

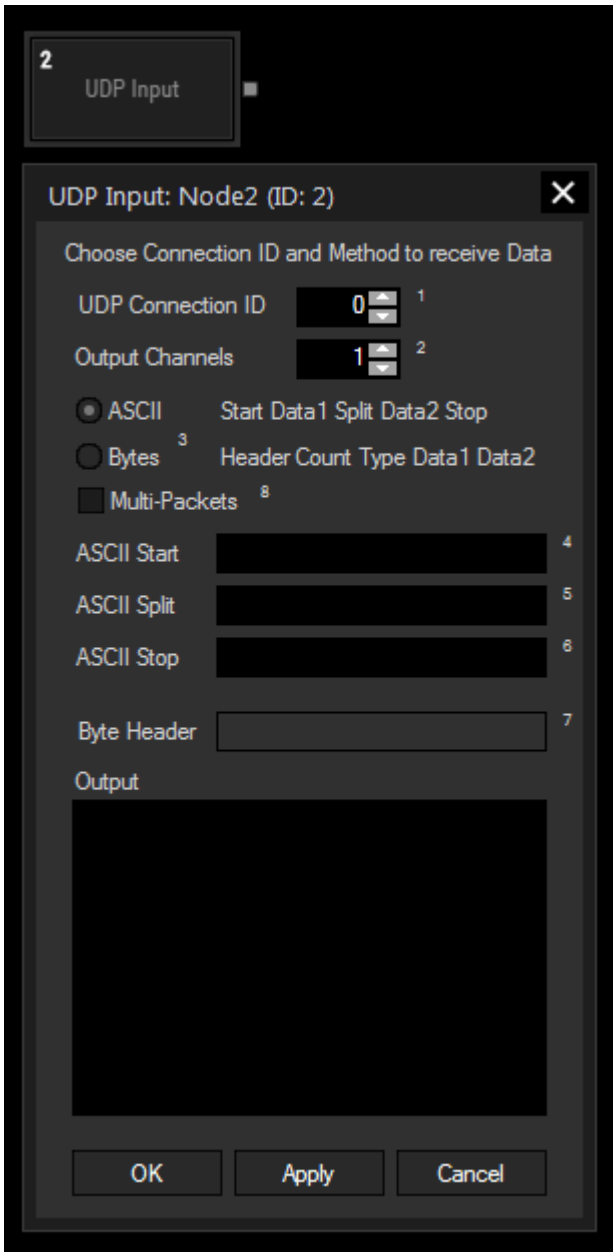
The Input node generates the following output:

- Value: outputs the message as numeric value
- Delta: outputs the delta value between the last received numeric value and the current one
- String: outputs the string message

7.4.4.1.21 UDP Input

The UDP input node allows receiving either ASCII or Byte values based on the selected UDP connection. It can output by default any incoming packet as text if no ASCII filter is applied.
The UDP connection can be found and set up in the Connection Manager [1258].

This node can be found under Nodes > Input > Connections > UDP Input



Node Properties

UDP Connection ID:
Choose the UDP Connection to listen to. The UDP Connections can be found and setup in the Connection Manager [1258].

Output Channels:
Set the Amount of Output Channels of this Input Node.
Input Value 1 will output Data 1, Input Value 2 will output Data 2, etc.

ASCII Mode:
The ASCII Mode allows creating a custom ASCII protocol for data transfer.
Multiple data can be separated by the split string.
To determine the Start and End of the message, use the Start and Stop strings.

ASCII Start, Split and Stop:
Enter values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages [944].

Bytes Mode:
The Byte protocol allows data transmission based on a byte stream.
First send the **header** with any amount of Bytes.
Followed by the **data count** as 4 Byte Integer. Each piece of data, i.e each 4 Byte Integer or each 8 Byte double, counts as one. Please take into account that the processing uses a little-endian Byte order, meaning that the lowest byte comes first.
Followed by a single Byte to describe the **data type** used (1 = 4byte integer, 2 = 8byte double).
Followed by one or multiple **data** sets of the specified type, also in little-endian Byte order.

Example: This byte stream transmits one integer with value 5000, the header is "WD!":

[87 68 33 1 0 0 0 1 136 19 0 0]

- Header: 87 68 33
- data count: 1 0 0 0
- data type: 1
- data: 136 19 0 0

Byte Header:
Enter values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages [944].

Multi-Packets:
If the UDP packets are being received in parts, this option allows reading all part packets as one.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

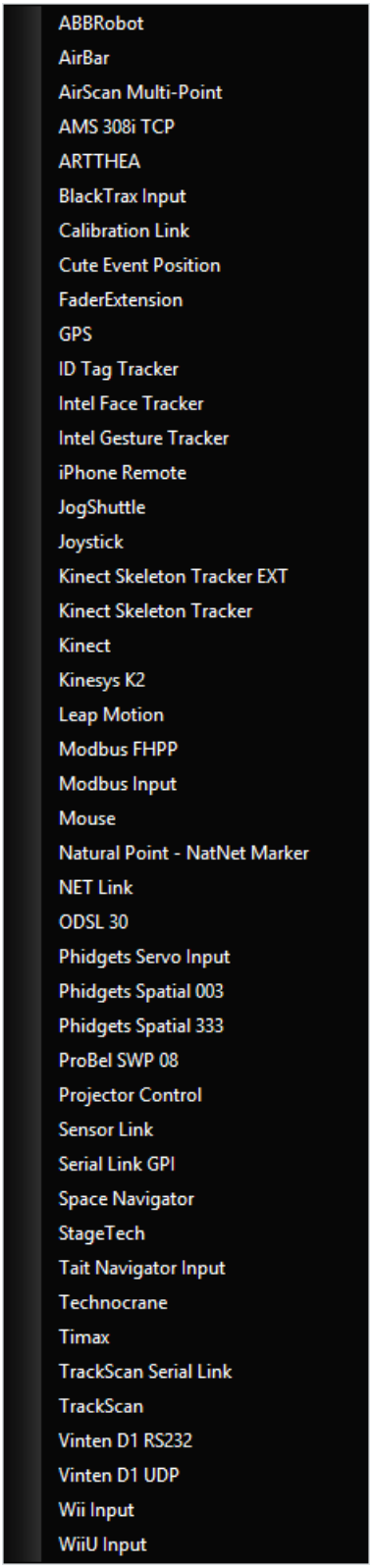
The Node generates the following output:

- UDP Input Value 1 - Value X (X = Amount of Output Channels).

7.4.4.2 Devices Input

The Devices input nodes receive data from dedicated hardware devices, such as the mouse, the Microsoft Kinect [1283] sensor, Phidgets [1383] or ID Tag Tracking [985].

The communication with the here listed devices is implemented in the Widget Designer software. For most devices, it is necessary to establish a connection first, e.g. if it is connected via network, and configuration. The devices listed in the Devices Menu can be set up with the respective entry there. All others only need the configuration the node provides.



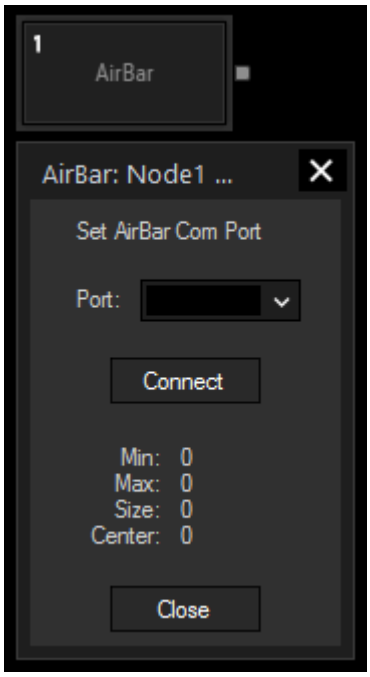
ABBRobot
AirBar 979
AirScan Multi-Point 980
AMS 308i TCP
ARTTHEA
BlackTrax Input
Calibration Link 981
Cute Event Position
FaderExtension 983
GPS 984
ID Tag Tracker 985
Intel Face Tracker
Intel Gesture Tracker
iPhone Remote 1008
JogShuttle 1009
Joystick 1010
Kinect Skeleton Tracker EXT
Kinect Skeleton Tracker
Kinect 1011
Kinesys K2 1012
Leap Motion
Modbus FHPP
Modbus Input 1015
Mouse 1016
Natural Point - NatNet Marker
NET Link 1016

ODSL 30 1019
Phidgets Servo Input 1020
Phidgets Spatial 003 1021
Phidgets Spatial 333 1022
ProBel SWP 08
Projector Control 1024
Sensor Link 1026
Serial Link GPI 1028
Space Navigator 1029
StageTech
Tait Navigator Input
Technocrane
Timax 1030
TrackScan Serial Link 1031
TrackScan 1032
Vinten D1 RS232
Vinten D1 UDP
Wii Input 1034
WiiU Input

7.4.4.2.1 Airbar Input

The Airbar Input node provides the following output values: Min, Max, Size and Center. Set up the Airbar via Com Port [1269].

This node can be found under Nodes > Input > Devices > AirBar



Node Properties

Port:
Enter the number of the COM Port the Airbar is connected to. Press Connect.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.

Node output values

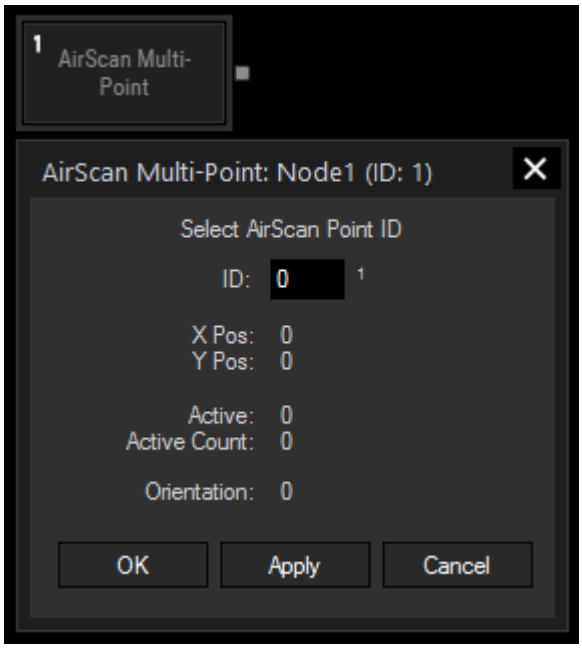
The node generates the following output:

- Min Output = Minimum Position from left
- Max Output = Maximum Position from left
- Size Output = distance between two points
- Center Output = center Position between Min and Max

7.4.4.2.2 AirScan Multi-Point Input

The AirScan Multi-Point Input node provides the positions of each detected point, as well as its status, the detected active point count and its orientation.
The AirScan has to be set up in Multi-Point Mode, see AirScan Tools Menu [1277].

This node can be found under Nodes > Input > Devices > AirScan Multi-Point



Node Properties

ID:
Enter here the ID of the point you want to use the data of (1-24).

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

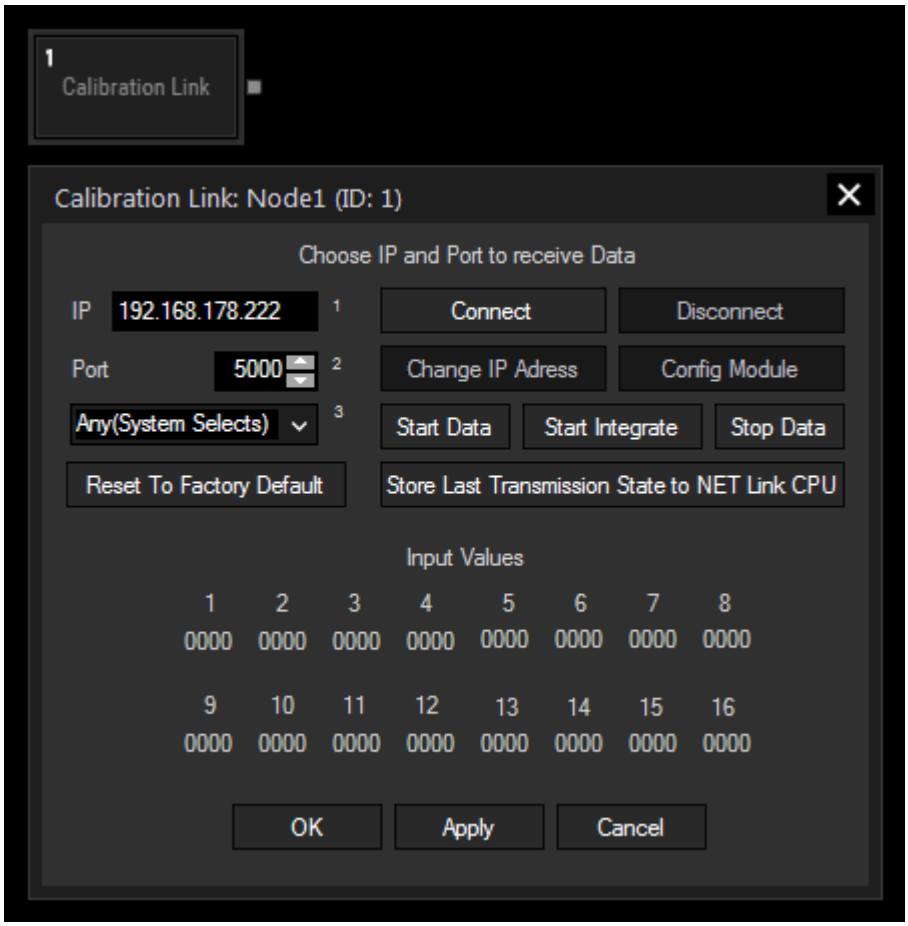
The node generates the following output:
- X Pos
- Y Pos
- Active (true or false)
- Active Point Count
- Orientation

7.4.4.2.3 Calibration Link Input

The Calibration Link Input node allows to communicate with the Calibration Link or a NET Link equipped with fiber input boards. Please see more information in the chapter covering the NET Link [2000].
On the one hand the node receives information send from the 16 fibre inputs, hence, it provides this data to other nodes in Widget Designer. On the other hand, it allows to configure the device and send commands to it.

The NET Link Input node [1016], the Relay Output node [1202] and the tool "Projector Calibration Manager" [1496] might also be of interest for you.

This node can be found under Nodes > Input > Devices > Calibration Link



Node properties

IP and Port:
Enter the correct IP address and port from the NET Link's processor or from the Calibration Link.

Select Adapter:
Specify here the NIC you want to use for communication with the NET Link.

Reset To Factory Default:
This buttons resets the above settings in the Widget Designer interface. To reset the device itself (to the IP address 192.168.178.222 and the port to 5000), hold the "Reset" button down whilst plugging the power into the device. Release the button again.

Connect / Disconnect:
Before starting to communicate with the device, for instance receive data, the node must be connected to the device. The connection itself consumes no performance.

Change IP Address:
This button opens a new dialog whereto you may enter another IP address and port for the device. Power-cycle the device to apply the changes.

Config Module:
A NET Link / Calibration Link is configured by us as you have ordered it. However, if you have changed some input / output boards, the processor must be configured in terms of giving him the information which boards are connected to it. Click the "Config Module" button to open a new dialog where you may choose the according boards per processor connection. Find more details in the NET Link hardware chapter [2002].

Start Data, Start Integrate and Stop Data:
As soon as data is processed via the network from the device to Widget Designer, performance is drawn. For normal use, click "Start Data" (sets a high data rate). For the use of measuring the light distribution over a longer time period, activate the slower data rate with "Start Integrate".

Store Last Transmission State to NET Link CPU:
Click this button and power-cycle the device if you wish that it remembers whether it should (not) send data as soon as it is powered up.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

In addition to the general commands, this nodes provides specific function commands that can replace pressing the buttons in the Configuration manually. E.g. `Node1.StartDate` starts data transmission without opening the dialog.

The following additional function commands are available for the Calibration Link node:

- ChangeIP
- Connect
- Disconnect
- getconfig
- SetConfig
- StartData
- StartIntegrate
- StopData
- StoreLastTxMode
- Tx3

Node output values

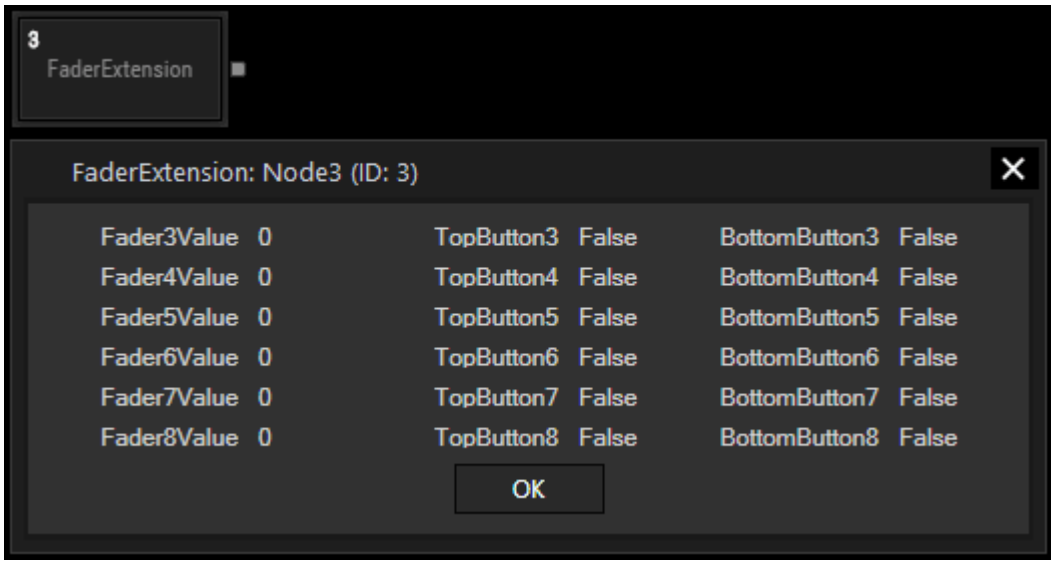
The node generates the following output:

- Value 1 to 16 (0 - 1023)
- Connected

7.4.4.2.4 FaderExtension Input

The FaderExtension input node allows you to register fader values and button clicks of a Fader Extension hardware [1993] device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Fader Extension" [1354] which also shows how to work with the device using the scripting language.

This node can be found under Nodes > Input > Devices > FaderExtension



If you like to connect to a Jog/Shuttle device [1991], please use the JogShuttle input node [1009].

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.

Node output values

The Node generates the following output:

- Fader Value 3 - 8
- TopButton State 3 - 8 (True = Pressed down / Clicked)
- BottomButton State 3 - 8 (True = Pressed down / Clicked)

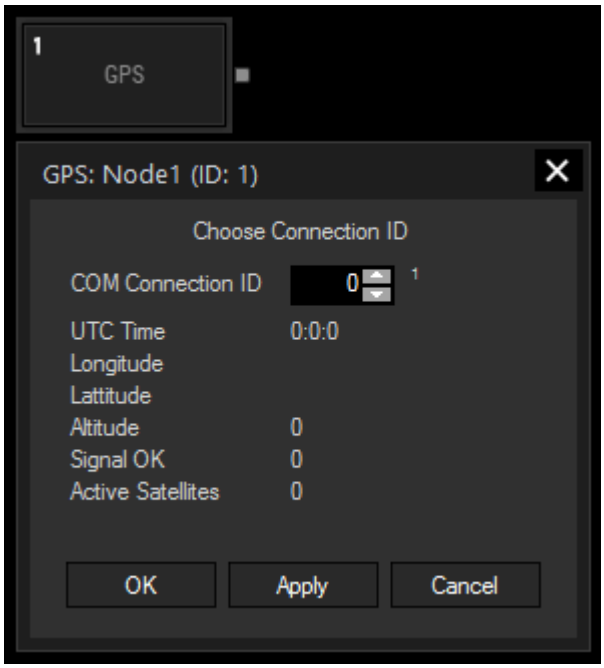
Note that the ID 1 and 2 for the Fader and Buttons belong to the Jog/Shuttle.

- IsConnected State

7.4.4.2.5 GPS Input

The GPS COMPort input node supports the GPGGA data format and is compatible with NMEA 0183.
This node was tested with Navilock USB GPS Input device. To use this node, please set up a COM connection [1269] with your GPS device.

This node can be found under Nodes > Input > Devices > GPS



Node Properties

COM Connection ID:
Choose the COM Port your GPS device is connected to.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

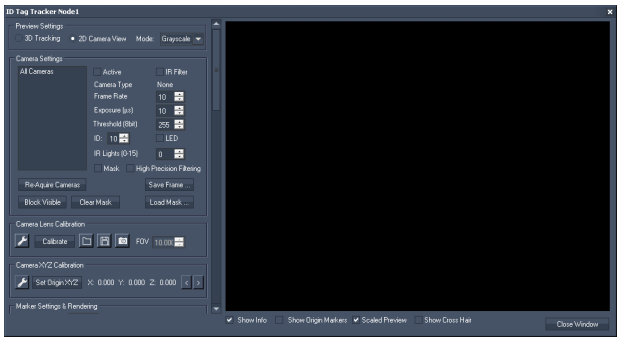
Node output values

The node generates the following output:

- Hours
- Minutes
- Seconds
- Longitudes Text
- Longitudes Degrees
- Longitudes Minutes
- Longitudes Seconds
- LattitudeText
- Lattitude Degrees
- Lattitude Minutes
- Lattitude Seconds
- Altitude
- SignalOK
- Active Satellites

7.4.4.2.6 ID Tag Tracker Input

The ID Tag Tracker Input node consists of all settings related to tracking IR ID tags from Christie Digital Systems Germany. The node requires the enabled option "OptiTrack Camera Manager" in the Connection Manager [1258]. It is available in Widget Designer PRO and ULT edition. The tag configuration is included in the Widget Designer STD too.



The ID Tag Tracker node includes a lot of settings to set up, adjust and optimize the tracking. It includes the following sections:
- Preview [991]
- Preview Settings [992]
- Camera Settings [992]
- Camera Lens Calibration [994]
- Camera XYZ Calibration [996]
- Marker Settings & Rendering [999]
- Global Tracking Settings [1000]
- 2D Camera Tracking Settings [1001]
- 3D Tracking Settings [1002]
- Performance Options [1002]
- Motion Prediction [1003]
- MA PSN Output [1004]
- ID Tag Settings [1004] and Config ID Tag [989]

Below the principle of ID camera tracking is explained whilst the next chapter covers the hardware and environment requirements [986]. The very last chapter approaches tracking with moving lights [1005] (automatic follow spot) and how to set up your GrandMA console for it.

The principle of optical camera tracking and IR ID tags

The general idea behind the optical tracking method with infrared cameras and infrared identification tags -short IR ID tags - is to locate a person or any object, identify it and track its movement. This can be done without any cables attached to the person / object as the tag is pinned on the tracked object and is seen by one or several cameras.

The tag is an active marker with an internal power source: in contrary to a passive marker it does not reflect but emits light itself which allows a higher luminosity (and hereby a maximum range) as well as more flexibility in your stage setup.
In addition to that it can be configured with an identifiable number. Thus it is possible to distinguish two objects from each other, even if they overlap and separate again. This is a huge advantage compared to other camera tracking methods. As well, it allows playful interactive scenarios. For instance it is possible to assign dedicated properties (e.g. content media or color) to dedicated persons or objects. As well, these properties can be shared or swapped if two objects meet.
Currently, you may track up to 256 different IDs. Several tags can be set up with the same ID. The maximum total number of trackable tags depends on the hardware system.
Another requirement in the tag development was allowing it to be hidden in scene pieces, costumes etc. whilst providing good handling on stage. We met this need by reducing the tag to the minimum size possible, the pure printed circuit board with an onboard power source. Furthermore, the tag can be used out of the box or it can be modified to suit individual needs. That includes different power and / or light sources, e.g. spatially divided from the tag board.

The Widget Designer offers a unique,self-developed tracking method that processes the camera data, allows user adjustments and provides the object's position. With a minimum of three cameras, it is possible to track the 3D

position, that is the exact location in a 3D space, expressed through the X-, Y- and Z-position. It is also possible track only in two dimension (i.e. a plane), for instance, if one axis does not change. This reduces the camera count to one and other hardware too.
The 2D or 3D data can then be send (through other Output nodes) to a Pandoras Box playback system, e.g a Video Layer. Alternatively or in addition, the data can be directly send to the MA PSN protocol and auxiliary tracking servers.
Besides the position tracking itself the Widget Designer offers the management of the tags properties mentioned above. To fulfill creative tracking scenarios you may adjust settings in the tracking node itself or combine it with everything else, the Widget Designer offers: from the command language and the entire nodes system to other tools and user-controls through to variables and functions. The combination of all this allow to program even the most complex logic.

Hardware-wise the IR ID tracking method requires tags, cameras, (a switch), and a PC running Widget Designer. The surrounding environment must be suitable. The next page explains more about requirements regarding hardware and the environment [986].

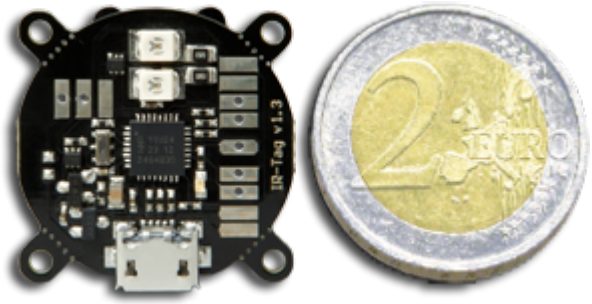
7.4.4.2.6.1 Required Hardware and Tracking Environment

This topic describes the requirements for the IR ID tracking method regarding hardware and environment. For general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

Hardware-wise the IR ID tracking method requires tags, cameras, (a switch), and a computer running Widget Designer. The surrounding environment must be suitable.

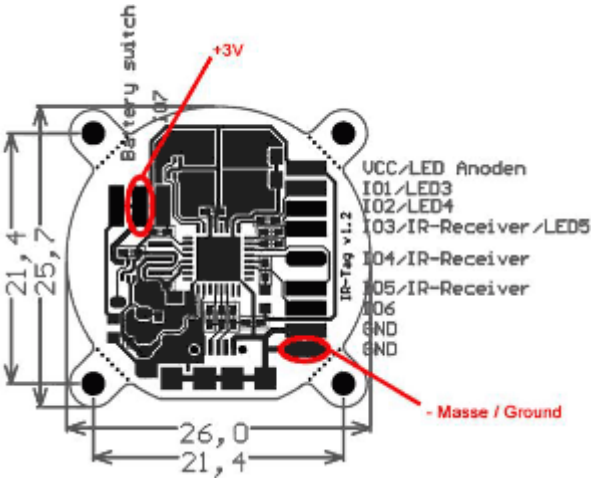
The tags can be purchased through your local Christie distributor. You can choose between an "ID Tag" that provides one configurable ID and the "Quad ID Tag" that provides up to four configurable IDs.

(Single) ID Tag



The tag is a printed circuit board with two onboard infrared LEDs, one onboard control LED and a power source. It has a size of 26x26mm and a weight of 10g. The board offers pads for external interfaces, thus it can be modified to suit individual needs, e.g. a spatially divided LED or another power source. Please note that the warranty is not voided by using these pads but malfunctions due to faulty use (too hot soldering, too much power etc.) are not covered!

The onboard or added LEDs blink in a certain pattern according to the set identification number. The LEDs' luminosity can be adjusted whilst only the onboard ones can be powered by 100%. Additional LEDs can be driven with max. 70%, independent from the power source. You may set up to use one or both onboard LEDs and up to three possible additional ones. A brighter tag allows a more distant camera, a more lightened surrounding and a more robust tracking. On the other hand it consumes more power.



The power source can be one of the following three:
- onboard hearing cell holder: holds two button cells; smallest space; feeds one LED with 70%; battery life span approx. 10 hours (as soon as the batteries are activated this span cannot be stopped!)
- onboard Micro-USB interface: allows to plug the delivered battery (with four 1.5 AA batteries); a battery pack lasts up to 24 hours; auxiliary battery packs should deliver 4.2 to 6.5V
- custom power source: the pads allow to add another power source with max. 3V

The tag can be configured [989] via the onboard Micro-USB interface. Connect the tag via the delivered USB cable (Micro-USB to Standard Type A plug) to your computer running Widget Designer. The configuration involves the number and power of the IR LEDs as well as the brightness of the blue control LED. Further, the blinking mode including the ID can be set up. See below for more details regarding the

software settings. The settings are stored in the tag, allowing to unplug it from the computer to change back to the delivered battery pack.

Quad ID Tag

The quad tag's printed circuit board is covered with a housing. It provides four twist-lock connectors to which you can connect the provided external infrared LEDs. The tags have a cable length of 1,5m. The housing also includes the power source: two AAA-batteries. A power switch and an onboard control LED are also included. It has a size of 110x70x20mm and a weight of 50g.

The connected LEDs blink in a certain pattern according to the set identification number. Each port can be numbered individually. The LEDs' luminosity can also be adjusted. A brighter tag allows a more distant camera, a more lightened surrounding and a more robust tracking. On the other hand it consumes more power.

The tag can be configured [989] via the onboard Mini-USB interface. Connect the tag via the delivered USB cable (Mini-USB to Standard Type A plug) to your computer running Widget Designer. The configuration involves the options for the brightness of the IR LEDs as well as the brightness of the blue control LED. Further, the blinking mode including the four IDs can be set up. See below for more details regarding the software settings. The settings are stored in the tag.

The cameras supported are third-party motion capture cameras from the OptiTrack system developed by the company NaturalPoint. Please contact an OptiTrack distributor if you are interested as they cannot be purchased through your Christie distributor.
The implemented and supported OptiTrack models are called:
V100R2
Flex13
Prime13
Prime17
Prime17W
Prime41
s250

The camera models differ in resolution, frame rate, field of view (opening angle), max. tracking distance, latency, etc. Please refer to the "Compare Cameras" web site offered by NaturalPoint.

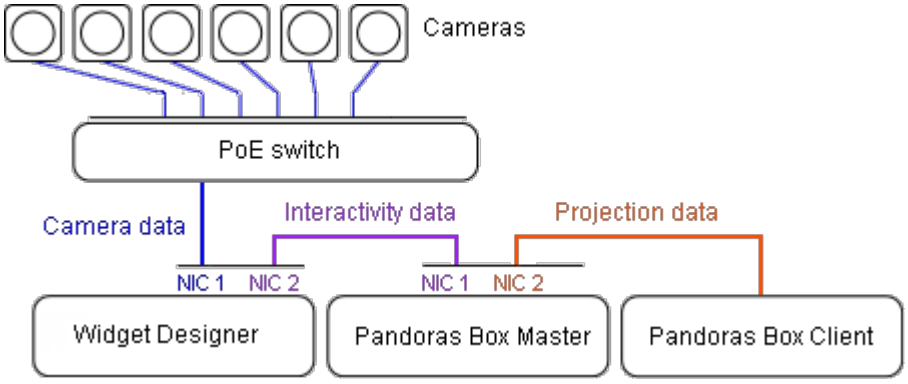
The question how many cameras are needed and where they need to be positioned depends first of all on the choice to track 2D or 3D. As mentioned above, to capture the 3D position of a tag, a minimum of three cameras must "see" it at any time. For the 2D position, only one camera is needed. Secondly, the maximum distance from a camera to a tag must be defined by combining the camera model's specifications in theory with the stage environment in field. This is influenced by light, the stage setup itself, object movement etc. Please see the below paragraph about "Environment". Afterwards, you may position the cameras in such a way that they cover the area of interest so that one tag is seen by either one or three cameras at any time.
Please keep in mind that the cameras' position should be as static as possible. If they are mounted with other devices such as moving lights on a truss, we strongly advice to check whether any light cue state moves the cameras so much that it worsens the tracking too much.

An PoE Ethernet switch is needed when working with cameras with an Ethernet interface, e.g. the s250e or the Prime models. We recommends to consider the product recommendation from NaturalPoint. They offer accessories including switches. As well they provide detailed information about the specification a switch must meet as not all PoE switches are suitable for use with OptiTrack PoE Ethernet cameras. Quoting their FAQ page: " Ethernet cameras require PoE or PoE+ Gigabit (1000 Mbit/s) Ethernet switches. Standard PoE switches must provide a full 15.4 watts to every port simultaneously and PoE+ switches must provide a full 25 watts to every port simultaneously." Please consider that the required power is provided on all ports simultaneously and not only to a subset of ports whilst downgrading other ones. The cameras will not function properly when insufficient power is available to them.
Same applies to the bandwidth per port. "(...) the switch must support Gigabit (1000 Mbit/s) for every port. Connecting multiple PoE switches in a star topology to a non-PoE Gigabit uplink switch is recommended for larger camera counts." It is not recommended to connect more than 4-8 cameras on one switch.

As the Ethernet cameras assign themselves an **IP address**, the switch must not provide a DHCP Server!

The cabling between Ethernet cameras and the switch must be done with Cat6 or Cat7 cables; Cat5/Cat5e/etc not supported. For the USB cameras (e.g. V100:R2) the USB cable length must match the NaturalPoint recommendations. Both camera types, Ethernet and USB, draw their power from the connected "data" cables, an additional power supply and connection is not needed.

Lastly, the **Widget Designer** communicates with the cameras, processes all the (image) data and extracts the object's position in real-time. Further this data is processed according to your needs and output via Art-Net to a lighting desk or via network to Pandoras Box for example. Hardware-wise it is recommended to meet at least the specifications from a Pandoras Box Workstation. Please contact your local distributor for more detailed information. Next to the network hardware (cables, switches, etc.), the type of Widget Designer hardware (network card, processor...) determine the amount of cameras and tags per system. As a very rough rule of thumb, one Widget Designer hardware and one LAN (local area network) may process the data of six cameras and 32 tags. Several options may increase or decrease this number.
The more tracking data is processed (e.g. from the cameras to WD) the more it is recommended to separate data flows from each other. This can be either done by using manageable switches or with network cards that offer multiple network adapters / NICs (Network Interface Card, physical RJ-45 connector). In the depicted example, the camera tracking data, interactivity data and show data from Pandoras Box are separated.



Widget Designer and Widget Designer Unlimited edition support the tracking node itself and the tag configuration. Widget Designer Free Version supports only the tag configuration.

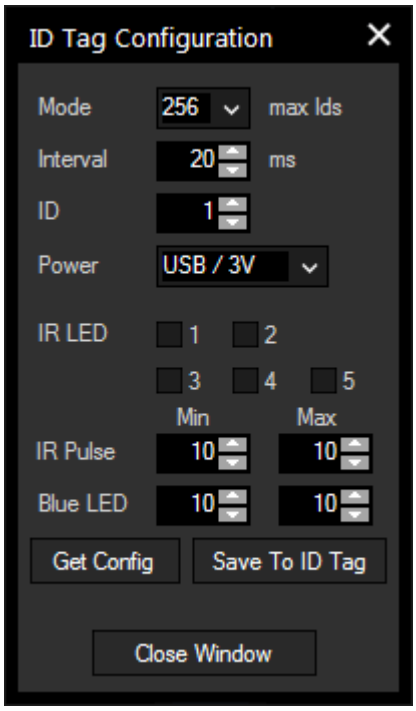
The setup for Widget Designer is described in the "Installation" chapter [787]. If you like to configure tags, connect one via the delivered USB cable (Micro-USB to Standard Type A plug) to your computer. Use the USB-driver "cooUSB-driver v6.0.13.0" (or a newer one) to finish the hardware installation of the device. Further steps are described in the chapter "Config ID Tag" [989].
If working with Ethernet cameras, the computer must be set up with a static IP address in a range and subnet mask freely chosen.
If working with USB cameras, please make sure to install the driver NaturalPoint provides.

The user interface offers several settings including the configuration of the tags, the cameras, the tracking itself including the management of objects' properties and finally the system to which all this data is send. The following pages will explain all option in the user interface.

The surrounding **environment** must match the needs of an optical tracking method based on infrared light. In the nature of things, any other infrared light source influences the stability of the tracking system. Please note, that many materials (glass, metal, etc.) reflect infrared light.
To reduce light noise, it is possible to adjust the cameras' sensitivity and tags' luminosity. These settings help to get rid of most unwanted light sources but still see the tag's light. In addition, masking is available. A mask can be used to exclude certain areas from the image a camera sends to Widget Designer, for instance areas outside the stage or too bright areas on stage. However, these areas need to be static and will also absorb the tag's information.
It is highly recommended to schedule sufficient time on site to cope with these interferences. Keep in mind that any changes done to the stage setup and scenery could influence the tracking behavior.

7.4.4.2.6.2 Config ID Tag

This topic describes how to configure a tag software-wise. For general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

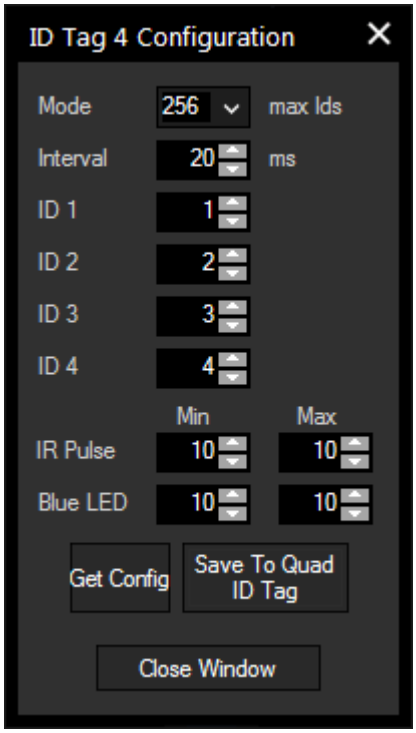


(Single) ID Tag

The software configuration of the tag involves to set up the number and power of the IR LEDs as well as the brightness of the blue control LED. Further, the blinking mode including the ID can be set up.

The tag can be configured via the onboard Micro-USB interface. Connect the tag via the delivered USB cable (Micro-USB to Standard Type A plug) to your computer running Widget Designer. If you connect it the first time, and the driver cannot be found automatically, use the USB-driver "coousb-driver v6.0.13.0" (or a newer one) to finish the hardware installation of the device.

Now, please open the configuration dialog in Widget Designer:
a) Tools > Connection Manager > section ID Tag Configuration
b) Nodes > Input > Devices > ID Tag Tracker > section ID Tag Settings



Quad ID Tag

The software configuration of the tag involves to set up the brightness of the IR LEDs and the brightness of the blue control LED. Further, the blinking mode including the ID for each port can be set up. Note that it is possible to set each port to an individual ID or to share the same ID.

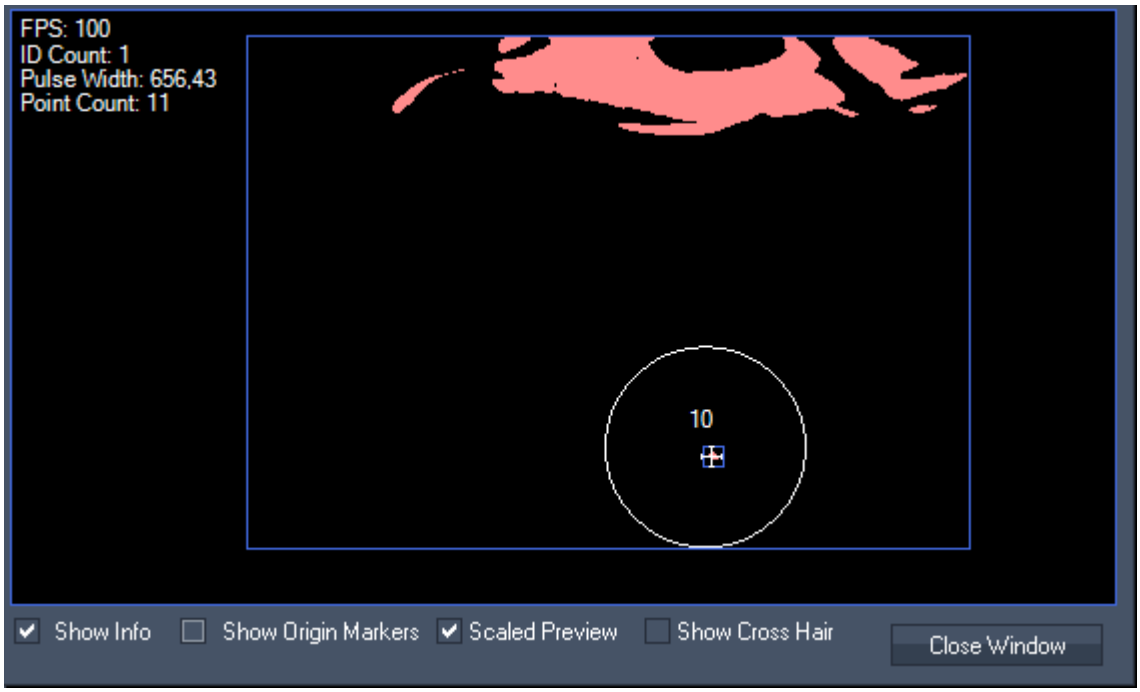
The tag can be configured via the onboard Mini-USB interface. Connect the tag via the delivered USB cable (Mini-USB to Standard Type A plug) to your computer running Widget Designer. If you connect it the first time, and the driver cannot be found automatically, use the USB-driver "coousb-driver v6.0.14.0" (or a newer one) to finish the hardware installation of the device.

Now, please open the configuration dialog in Widget Designer:
Tools > Connection Manager > section ID Tag Configuration

Mode	This influences the maximum ID a tag can hold. It is recommend to choose the smallest mode available that covers the number of tags you use. "4" allows four tags holding the ID 1,2,3 and 4. If you want to use ID 5, you need to choose a higher mode, e.g. 8. The tags transmit their ID with a certain blinking pattern that needs to be seen by cameras. The higher the mode is, the longer this pattern gets. Hence, the more time needs to pass in order to recognize a tag if it (re-)appears in the tracking area. Please note that it is possible to set up several ID tags with the same ID. This does not mean to increase the mode as long as the ID fits into the range. In the section "3D Tracking Settings" 1002 you can set up that "same" IDs should result in one averaged position.
Interval	This influences how fast the above described blinking pattern is transmitted. Per default it is set to 20ms. The lower the frame rate of the used cameras is, the longer an interval needs to be, the slower a tag is recognized. For a frame rate below of 100-120fps an interval of 40 ms is recommended. That applies for most USB cameras. The higher the frame rate of the used cameras is, the shorter an interval can be, the faster a tag is recognized. A frame rate of 120fps allows to lower the interval to 20ms; 150fps allows even 10ms. However, a longer interval still works with fast cameras and is more robust. In theory, a 10ms interval also works for 120fps but in practice this is very prone to error, hence an 20ms interval is better.
ID	The "ID" depends on the "Mode" chosen above and means the identification number of the connected tag.
Power	Choose the power source of your tag. Choose "USB / 3V" if you use the onboard Micro-USB interface with the delivered battery pack or your own 3V power source Choose "A675 Cell" if you use two onboard hearing cells. See more information regarding the power options in the previous chapter "Required Hardware" 986.
IR LED	Choose which LEDs should blink. 1 and 2 are the onboard ones. Only the "USB / 3V" power source allows to use all LEDs at the same time.
IR Pulse	Set the minimum and maximum brightness used in the blinking pattern when transmitting the tag's ID. It is recommended to set the minimum to 0. The brighter the maximum is, the higher the tracking distance gets and the more disturbing light is possible. Only the "USB / 3V" power source allows to power the LEDs with 100%.
Blue LED	Set the minimum and maximum brightness used for the blue control LED. If you do not want a blinking LED, set an equal value. The blue LED serves no other purpose than being able to see at first glance whether a tag is on or off.
Save Config	This stores the above settings to the connected ID tag. Now you can unplug it and power it (later). The settings in the dialog are not reset, hence you can plug a new tag to the PC, change only the ID and save all settings to it.

7.4.4.2.6.3 Preview

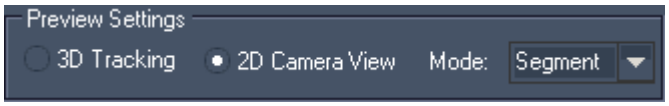
This topic explains the section "Preview" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].



Preview Window	Depending on the options in the section "Preview Settings" [992] this window shows either a camera's image (in grayscale, segment or tracking mode) or the virtual 3D space with all cameras and markers inside. To navigate in the 3D space use the middle mouse button (wheel): Scroll = Zoom, Drag = Pan, [Alt] + Drag = Rotate. More 3D rendering settings... [999] If a mask is set up, it is displayed as an blue overlay. If a tag is found, it is displayed with... - its ID if possible (here: 10) - a white circle indicating the maximum speed (see "2D Tracking Settings") - another circle underneath the white one, if you change the default minimum distance to another tag (see "2D Tracking Settings") - a white cross indicating the tag`s position with the added damping time (see "Prediction Settings"). Please see the below table for detailed information about "Grayscale / Segment / Tracking".
Show Info	This displays the Information in the left upper corner. The frame rate (FPS) depends on the mode chosen: the maximum rate is reached only in tracking mode and is lowered for the other modes. The Point Count informs about the number of valid areas. All valid areas are analyzed whether they hold an ID. If so, their tag ID and circles are displayed.
Show Origin Markers	This displays a red circle at the location from a origin marker when the "Mouse Mode" of the "Camera XYZ Calibration" [996] was used.
Scaled Preview	This toggles between a scaled view of 640x480px and the camera image in its original size (e.g. for the s250e model: 832x832px). A non-scaled view will be shown 1:1 and with scrollbars.
Show Cross Hair	This displays a horizontal and vertical line at the mouse pointer's position. Under certain circumstances it might be useful to see a crosshair overlay in the preview. The center of the image is marked as well as the size of a quarter of a frame. For instance, it is useful for the "Camera Lens Calibration" [994] process.

7.4.4.2.6.4 Preview Settings

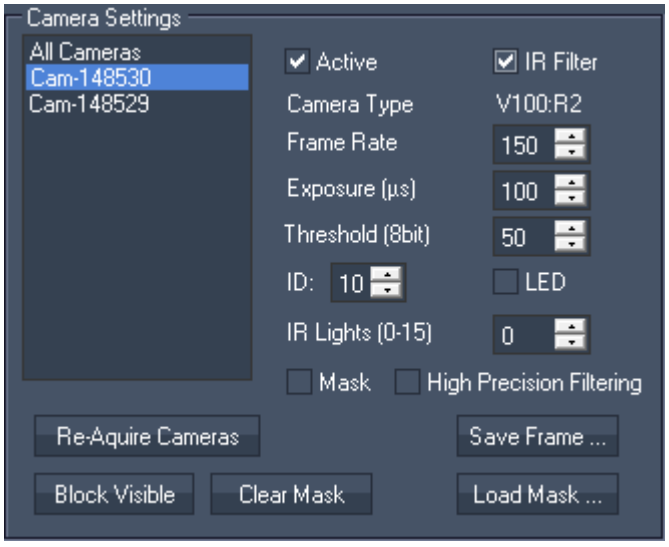
This topic explains the section "Preview Settings" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].



2D Camera View 3D Tracking	Choose what you like to see in the preview [991]. "3D Tracking" shows a virtual 3D space with visualized camera icons, all tags and a 3D model of your stage if set up in the section "Marker Settings & Rendering" [999]. To navigate in the 3D space use the middle mouse button (wheel): Scroll = Zoom, Drag = Pan, [Alt] + Drag = Rotate The "2D Camera View" shows the real camera image. Here, you may choose between grayscale, segment and tracking mode...
Grayscale Segment Tracking	If the preview is in "2D Camera View" mode, the camera's image is seen as: - a grayscale image with detailed information (draws most performance) - a segment image with information reduced to pixels above the setup threshold - all these pixels are examined to wear a tag ID, if they do, a tag is displayed as described in "Preview Window" - a tracking image with information reduced to pixels found to belong to a tag (draws less performance, each time the node properties are closed, this mode is entered) The frame rate of a camera (depicted in the corner of the preview) depends on the selected mode. The maximum available rate is only achieved in the tracking mode.

7.4.4.2.6.5 Camera Settings

This topic explains the section "Camera Settings" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].



Camera List	All cameras found within the network are listed with their serial numbers. If a camera is missing, try to re-initialize the list (see below). Secondly, check the connection to it, e.g. restart your switch. NaturalPoint offers a tool called "Camera Tester", you can also find it (under the name "visualtest.exe") in the installation folder of Widget Designer. If you have problems connecting the Widget Designer to the cameras, try to connect to them using the tool to see whether the problem lies in the physical connection or not. The selection of a single camera or "All Cameras" affects many settings, e.g. the "Frame Rate". Keep in mind, that "All Cameras" overwrites a change to all cameras, even if they were set up individually per camera before"
Re-Acquire Cameras	This button tries to (re-)connect to all available cameras (and removes obsolete cameras). All settings like "Exposure" are kept.

Active	This check box toggles the camera active or inactive. An inactive camera does not contribute its view to the calculation of the tag's position anymore. This can be helpful during setup to find out how "good" a camera matches with other cameras. If a tag jumps far when toggling a camera, it is not setup well. Another necessity for deactivating a camera might be when the connection suddenly became bad or if it's position has changed by accident or something else happened that disturbs the tracking calculation.
IR Filter	This check box influences what light is seen by the camera - only infrared light with a ticked check box or additional visible light if not ticked. You might want to deactivate the filter during setup (e.g. camera position, focus,...) in order to see a camera image containing more information.
Camera Type	This informs about the camera type when a camera is selected from the list, e.g. "V100:R2".
Frame Rate	The frame rate determines how many images are takes per second. For instance, "250" means that 250 images are taken and send per second. In other words one image has a time frame of 1/250 of a second = 0.004 sec = 4 ms = 4000 µs You may lower the frame rate to save performance. You must not exceed the highest frame rate supported by your camera.
Exposure	The Exposure determines how many microseconds (1µs = 1/1 000 000 of a second = 0.000001 sec) the lens shutter is open for light to pass through it and expose the image. The longer that is, the brighter the image gets, meaning that dark areas will be not black any more but dark-gray whilst light-gray areas will become white. Every pixel that is not black is later on examined to be a potential tag. Hence, adjust the time as short as possible to still see the tag's light but nothing else. Before increasing it, lower the threshold first. Exposure and threshold are engaged to each other! Also, you could increase the brightness of a tag or set up a mask. Note that the exposure time depends on the frame rate and should not be longer than one frame itself.
Threshold	The effect of the threshold is only seen in camera mode "Segment" or "Tracking" as it works "after" the grayscale image. It sets the minimum brightness a pixel from the grayscale image must have in order to be considered as a valid pixel. Valid pixels are displayed in the segment mode as pink pixels and are potential tags. The value range is 0-255. In contrary to the exposure, which should be set to a minimum level, the threshold should be set to a maximum level. The higher the threshold is, the brighter a pixel in the grayscale needs to be in order to be handed over into the segment mode, hence to be examined to be a tag. The less pink pixels you see, the less calculation needs to be done. In short, adjust the exposure as low as possible and the threshold as high as possible to still see the tag's light but nothing else. If you need a brighter image, it is better in regards to the performance, to adjust the threshold than the exposure.
ID and LED	If the LED check box is activated, the camera will show the set up ID. This is useful during setup. For show use, you might not want to see the ID. If you untick the check box, the ID itself stays unchanged for later use You may setup a maximum ID of 99.
IR Lights	The cameras are equipped with infrared LEDs themself as they can be used in tracking scenarios with passive (i.e. reflecting) markers. As our tag is an active marker, you won't need to illuminate it. However during setup (with an activated "IR Filter") it might be useful from time to time to illuminate the scene. A value of 0 turns the LEDs completely off whilst 15 results in the maximum brightness.
Mask	With an activated mask, certain areas of the camera's image are blocked. A mask can be set up from the user interface itself (see "Block Visible") or prepared with another image software (see "Load Mask" and "Save Frame") This way you can eliminate areas with reflections or other light noise from the tracking calculation. You may see a mask overlay in the preview mode "Grayscale" and "Segment", not in "Tracking".
Show Crosshair	
Block Visible	This option works only in the camera mode "Segment". It creates a mask. All visible pixels seen in the segment preview are used for the mask and are blocked in consequence. For instance, turn off all tags so that you are sure to block only external light and then press the button. Tick the "Mask" check box to see the blocked pixels as a blue overlay.
Clear Mask	This option clears a mask if you have used the "Block Visible" or "Load Mask" before. If you do not need a mask anymore it is recommended to clear it as well as deactivate it.
Load Mask...	Click this button if you wish to load a mask that you have prepared with a third-party image software. All black areas in the mask, hide the according pixels in the camera's image whilst visible pixels let them pass through. It does not matter whether you use white pixels, or any color,

	as long as it is not pure black. Supported file formats are: BMP, PNG, JPG, GIF Please make sure, that the resolution of the mask matches the full resolution of the camera you are using, e.g. 1280 x 1024 for the Prime13.
Save Frame...	This option saves the current frame of the selected camera that is seen in the preview. It is possible to save a frame from the Camera modes: Grayscale, Segment and Tracking. If you have ticked the "Mask" check box, the mask will not be shown as black pixels. Saving a frame is useful when you like to prepare a mask or if you like to document light ratio, e.g. how much light there was during rehearsals.
Highprecision Filtering	This option is needed in order to enable camera pre-filtering of detected segments above threshold, a higher threshold might be chosen for this mode.

7.4.4.2.6.6 Camera Lens Calibration

This topic explains the section "Camera Lens Calibration" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

This section includes all settings for calibrating the lens.
Each lens deforms its image in a different way and it increases the accuracy of the tracking if this deformation is compensated. The lens deformation usually needs to be done only once as long as the lens is not removed. For the very best accuracy it is recommended to redo the calibration if the lens focus is altered.
As long as the cameras are not calibrated by you, Widget Designer calls the default calibration files from its installation path:
`C:\Program Files\Christie\Widget Designer...\Wd\Data\Resources\OptiTrackNode\camera`
If the default or your custom calibration files were able to be loaded, the section will be colored green, otherwise red.
For calibrating a lens you need a special image that depicts a checkerboard (details below...). You can either print on a hard board or glue a printed paper to it. Note, that the board needs to be as straight and plane as possible.

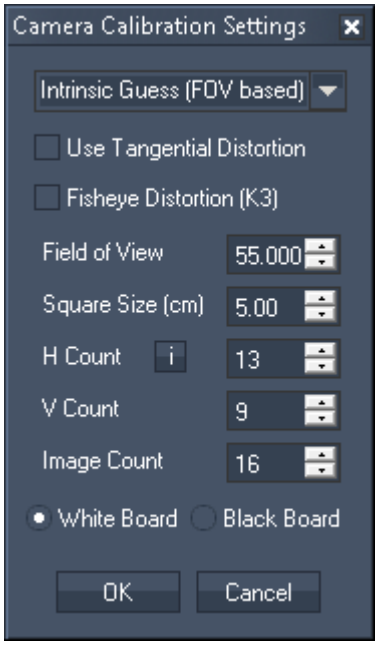
Wrench Icon	The button opens the "Camera Calibration Settings" dialog which is described in more detail below... In short, here you set up how the image looks like that you use for the calibration. The image is a checkerboard and is hold in different poses in front of the camera whilst snap shots are saved.
Calibrate	This calibrates the selected camera with the settings taken in the "Camera Calibration Settings" dialog.

Folder / Load Icon	Click the Folder Icon if you have calibrated the selected camera before and wish to load its data from a different path than the default one:
	`C:\Program Files\Christie\Widget Designer...`
	`\Wd\Data\Resources\OptiTrackNode\camera`

Disk / Save Icon	Click this button if you wish to save the calibration data for the selected camera to a different folder than the default one:
	`C:\Program Files\Christie\Widget Designer...`
	`\Wd\Data\Resources\OptiTrackNode\camera`

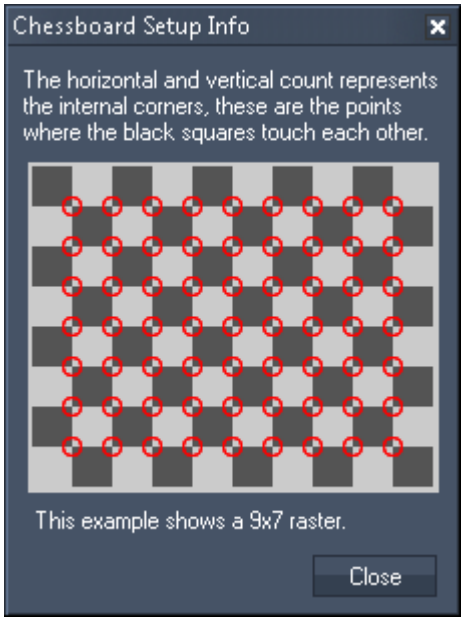
Camera / Precision Icon	Click this button if you wish to see the result of the Calibration (again).
FOV	This informs you about the field of view of the camera lens. The FOV is calculated during the Calibration and cannot be changed manually.

Camera Lens Calibration Steps

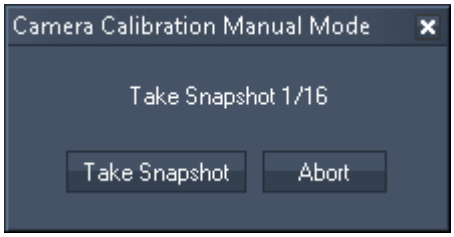


First, click the Wrench Icon to open the left depicted dialog.

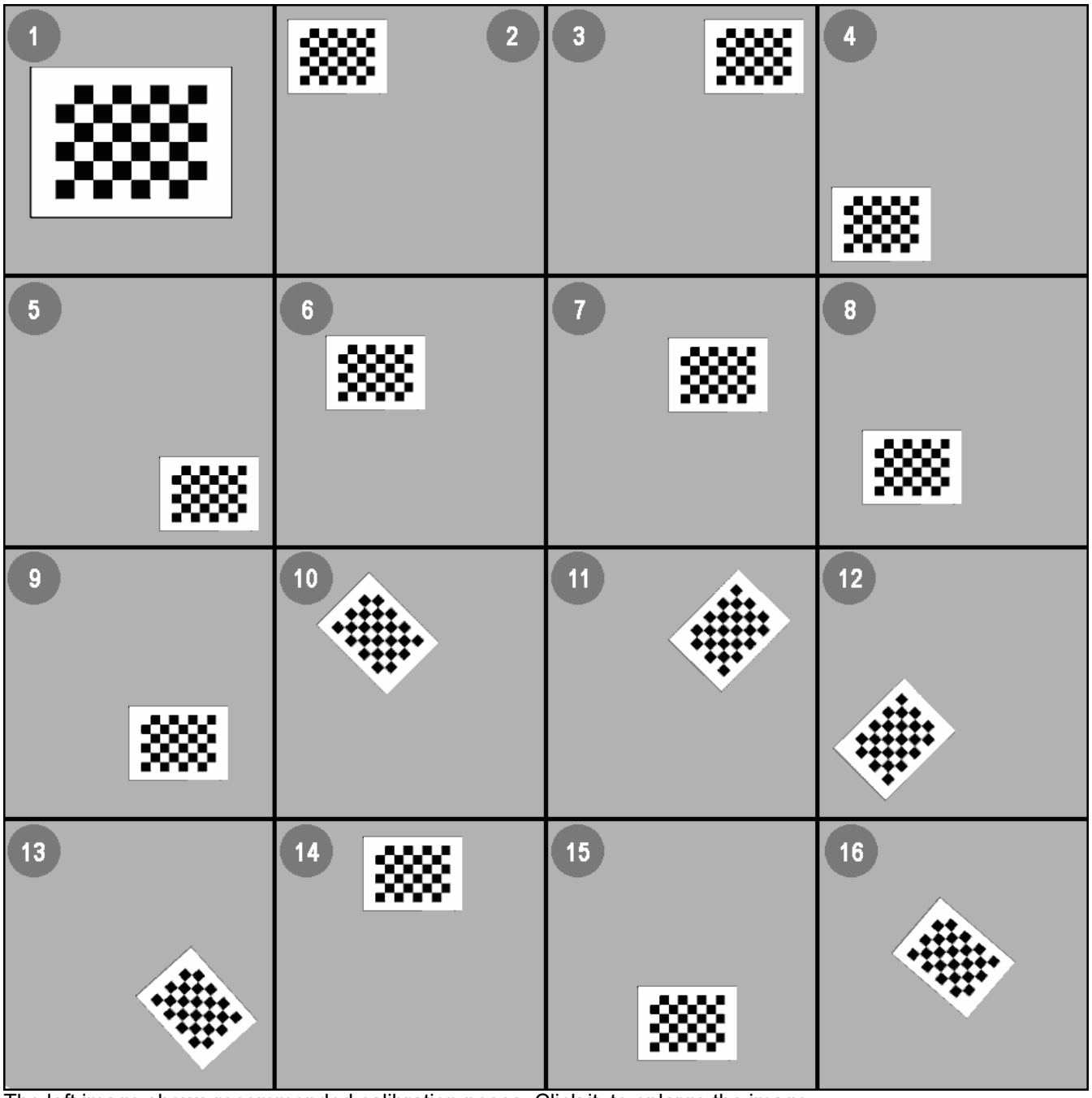
The first drop-down and two check-boxes let you choose the calibration method. Afterwards, enter the FOV of the camera lens. Now, all settings relate to the checkerboard:
Square Size: how wide and high is one square, measure in centimeters!
H Count and V Count: how many horizontal internal corners are there and how many vertical ones (click the info button for more information)
Image Count: how many images will be taken (16 is the recommended minimum)
White / Black Board: is the border around the checkerboard white or black



This dialog explains the horizontal and vertical square count.



You then hold the checkerboard in front of the camera and take pictures whilst moving either the board or the camera in such a manner that the board covers different parts of the taken image.



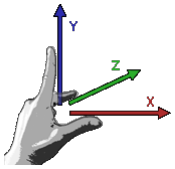
The left image shows recommended calibration poses. Click it, to enlarge the image.

7.4.4.2.6.7 Camera XYZ Calibration

This topic explains the section "Camera XYZ Calibration" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

This section includes all settings for the setup of the fundamental XYZ World description needed when working with multiple cameras in 3D and 2.5D mode. If you work with one camera only or only in 2D mode, you have the choice to skip this step and work with a Range filter node [1147] instead.

In general, if one determines a XYZ coordinate, first, an origin of the 3D space is needed, meaning the coordinate system's origin, the orientation of the X-, Y- and Z-axis as well as their scaling.
Only when the cameras share the same 3D space information, they can agree on correct XYZ position values for a tag. In other words, only if the origin for each camera can be determined accurately, their reported tag position coincides. If the origin from one camera does not correspondent with the one from another camera, they position the tag differently. As long as both cameras see the tag, its final position is averaged but as soon as it is covered for one camera, it can not be averaged anymore and could jump significantly.
The origin calibration also means that the cameras themself are positioned in the 3D space, this is why this section is called "Camera XYZ Calibration". This can be observed in the 3D mode in the Preview section.

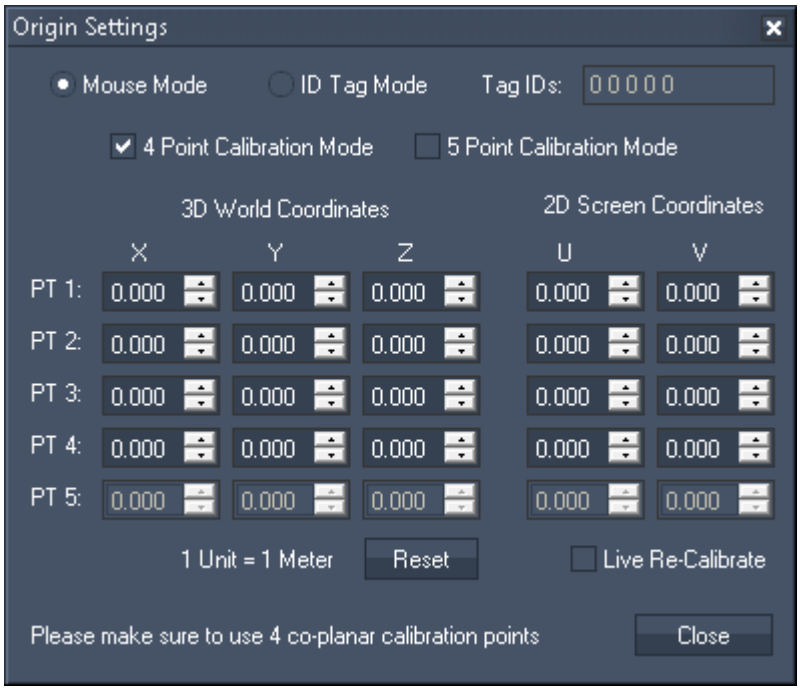


For calibrating the origin for one camera or all of them at the same time, you need to measure and mark a rectangle somewhere on your stage (i.e. four co-planar points). For most applications a 1-2m² spot that is seen by all cameras is ok, but this depends on the distance and orientation of the camera(s). For example, if a camera sees a very large area, a 1m² spot could be too small to achieve accurate results. Enlarge the rectangle until it covers 1/4 - 1/3 of the camera image seen in the Preview. Ideally it should be positioned in the center of the camera image. For complex applications, you might need to constitute several rectangles for different cameras.
For the sake of simplicity, you could mark four points on the floor. To achieve a better accuracy, the rectangle should include the origin itself, the coordinate 0,0,0. Now, you may decide how the orientation of your 3D space should be. For most scenarios, it makes sense to place the XZ plane onto the ground. A positive X axis means "move right" in Pandoras Box; a positive Z axis "move backwards"; a positive Y axis "move to ceiling". Having these axises in mind measure the XYZ position for each point of the rectangle (in meters!) and copy the data to Widget Designer into the "Origin Settings" dialog as described below. It's a good idea to number the corners 1-4 to remember the order. If you need to work with more than one rectangle, measured these points starting from the 0,0,0 point!

Note: Widget Designer now also offers a "5 Point Calibration Mode". Instead of calibrating the cameras with four points forming a rectangle, you can now also calibrate with five points. This method is of interest when you need to achieve a very precise tracking or if you are not satisfied with your results using a rectangle. In order to make text more readable, the manual will talk about four points or a rectangle, but all information applies comparably to the "5 Point Calibration Mode".

In the next step, the four points belonging to one rectangle will be found in the image of each camera. There are two modes available, an automatic and a manual one. Depending on the camera's position, the rectangle's shape looks different and this is how the camera's position is calculated.

< > buttons	With the help of these buttons you can select the preceding or following camera without using the camera list.
Set Origin XYZ	This button calibrates the selected camera based on the settings in the origin dialog (and the current camera image).
Wrench icon	This opens the below described dialog for the origin settings for the selected camera. For most applications, you calibrate the cameras using one rectangle. Choose "All Cameras" and click the wrench icon. Only if different rectangles are used for different cameras, select the specific camera in the camera list and enter its rectangle corners.
XYZ	This informs you about the calibrated XYZ position in meters for each camera and helps you to determine whether the calculation ran without errors. It is a good workflow to (roughly) measure the camera positions from the origin and compare these figures with the WD values for the assumed position. If the calculation is wrong, the WD position does not correspondent with the real one. If this is the case you can simply click on the "Set Origin XYZ" button again to run the calculation again and achieve a better result. If no result satisfies you, check the origin settings, the camera, the tags and network.



As said above, there are two modes available to find the four points of the drawn rectangle in each image of a camera: the "Mouse Mode" and the "ID Tag Mode" which is easier and faster but requires four different tags.

ID Tag Mode	
1	Position four tags (holding four different IDs) on the rectangle's points. You might want to cordon off the area so that the cameras keep a free view during the Calibration.
2a	Select "All Cameras" and open the "Origin Settings" dialog. Enter the XYZ location that you measured in the rectangle into the fields of PT1, 2, 3 and 4.
2b	Select "ID Tag Mode".
2c	Enter the corresponding IDs into the text field. Separate the ID numbers with a space, no comma is needed. If you, for example, got tags with ID 11-14 and positioned them onto the rectangle points as follows: 11 on 1, 12 on 2, 13 on 3 and 14 on 4, then you would enter into the text field: 11 12 13 14
2d	Close the dialog.
3a	Select one camera in the Camera List or with the < > buttons.
3b	Adjust the "Camera Settings [992]" and ... so that you see all four IDs in the Preview [991].
3c	Click the "Set Origin" button.
4a	Now, the camera looks for the four IDs, the camera's position is calculated and displayed in the XYZ field.
4b	Check (roughly) whether the calculated position is correct. If you don't like the result, check whether all tags are positioned accurately, check the camera itself (focus etc.), the origin settings, lens settings and the network. You may click "Set Origin" again.
	Start with step 3a for the remaining cameras.

If some cameras do not see the rectangle, a second one needs to be drawn, measured and equipped with ID tags. Redo all above steps, only for step 2a, do not select "All Cameras" but a single one.

Mouse Mode	
1	You might want to cordon off the rectangle area so that the cameras keep a free view during the Calibration.
2a	Select "All Cameras" and open the "Origin Settings" dialog. Enter the XYZ location that you measured in the rectangle into the fields of PT1, 2, 3 and 4.
2b	Select "Mouse Mode".
2c	Close the dialog.
3a	Select one camera in the Camera List or with the < > buttons.
3b	Adjust the "Camera Settings [992]" so that you see all four IDs in the Preview [991]. If you like you can disable the IR Filter, or increase the exposure. If you have at least one tag, you can use it to mark one corner after another. This helps you as the tag's light is seen more clear and brighter .
3c	Click the "Set Origin" button.
4	Now, a crosshair and magnifying glass is shown in the preview section. Click on the spots that mark PT 1, then PT 2... As soon as all 4 points are marked, the camera's position is calculated automatically and displayed in the XYZ field.
4b	Check (roughly) whether the calculated position is correct. If you don't like the result, check whether all marked points are positioned accurately ("Show Marker" [992] check box, check the camera itself (focus etc.), the origin settings, and the network. You may click "Set Origin" again to mark the points again, for example after adjusting the exposure or light on stage.
	Start with step 3a for the remaining cameras.

7.4.4.2.6.8 Marker Settings & Rendering

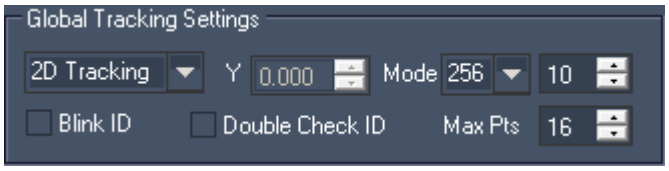
This topic explains the section "Marker Settings & Rendering" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

This section includes an information field for markers and further settings for the 3D view in the Preview [991]. To toggle this on, choose "3D Tracking" in the section "Preview Settings" [992].

Marker Size	This changes the size of the circle that depicts an ID marker. Per default it is rendered with 25cm.
Show Marker ID	This displays a number on top of the ID marker.
Marker Test Info	This is a helpful tool. Enter an ID marker, e.g. 1 and next to the text field 4 numbers will appear, e.g. 1.2 3.0 0.5|4, meaning that the position of the marker is X=1.2, Y=3.0 and Z=0.5 whilst the camera count that see the tag currently is 4. Using this tool you can control whether the calculated position of a stage marker equals the measured one, verifying that the cameras are well calibrated. Secondly, with the camera count, you can control whether all cameras work. Note that the IDs need to be listed in the table in the section "ID Tag Settings" [1004].
Stage (.3ds)	Click on the [...] button to load a 3DS file as a 3D model of your stage into the preview. Make sure that it is saved in meter units and with a pivot point on top the XZ plane [996].
Grid With/Depth	This changes the amount of the rendered grid lines. One square is always 1m x 1m.

7.4.4.2.6.9 Global Tracking Settings

This topic explains the section "Global Tracking Settings" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].



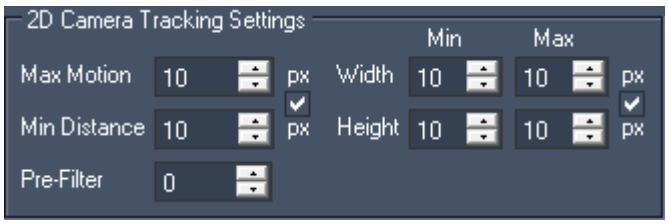
This section is includes important settings that influence whether tags are seen and tracked.

2D 2.5D 3D Tracking Y	Choose how you like to track the ID tags. "2D Tracking" works only with one active camera (that does not need to be calibrated in XYZ-space). The result for the tag(s) is a position described with X and Y, there is no Z-coordinate! The X- and Y-values depend on the resolution of the used camera, hence their unit is pixel. To scale the values down, for example if you like to follow with a Video Layer, you can use a Range Filter [1147] node. If you calibrate the XYZ-position [996] of the camera, the output values are in meters and independent from the camera resolution. "2.5D Tracking" works with one or several active cameras. They need to be calibrated in XYZ-space [996]. The result for the tag(s) is a position described with X and Z, both represented in meter values and depending on the XZ-movement of the tag. The Y-coordinate does exist, but is independent from the real up-down-movement. Set a constant height with the "Y" number field. "3D Tracking" works with several active cameras. At least three cameras must see a tag at every time. The cameras need to be calibrated in XYZ-space [996]. The result for the tag(s) is a position described with X, Y and Z, all represented in meter values and depending on the real XYZ-movement of the tag.
Mode	The mode (consisting of two figures) must match the "Mode" and "Interval" used when configuring the tags [989].
Blink ID	This option allows high tracking distances. Per default, the option "Blink ID" is on. As soon as you exceed tracking distances of 10m, this option must be ticked. If you track with lower distances and you have problems detecting a tag ID it might be that the exposure is too high. In that case the off state in the blinking still produces a little bit of light and disturbs the detection. For distances below 1m, reducing the exposure might not be sufficient. In that case, deactivate the "Blink ID" option. An deactivated "Blink ID" option detects the on and off states of the blinking by comparing brighter and darker pixels. This method allows maximum tracking distances of 5m and requires slow movement towards the camera lens.
Double Check ID	This influences the behavior of keeping the tag ID in case of an error. This does not apply for the initial recognition of a tag which is always a double check, i.e. an ID is initialized as soon as it is recognized twice. During and after recognition, tags transfer their ID with a blinking pattern. As an exception, it might happen that the pattern is not identified as the same ID as before, it can either be lost or identified as a different ID. This can happen when the tag is not seen by the cameras, e.g. when it disappears behind a set piece or is hidden by another object for a moment. Per default, the option "Double Check ID" is off. In case the ID is stable for a long time and then changes or disappears for a short time, this error weighs comparatively little. The "Off" option allows to soften or damp errors. As long as a blinking light is seen within the diameter set up with "Max Motion" (in "2D Camera Tracking Settings") [1001] it is assumed that it is the ID it was in the last frames. If "Double Check ID" is ticked, errors are not seen relatively but count absolutely. Even if a tag ID changes or disappears for one moment, this counts. The old ID is removed and deactivated if the error occurred for at least the number of the "Hold" frame count set up in "2D Camera Tracking Settings" [1001]. If you activate "Double Check ID" it is very important that the tracking recognition is 100% stable as it is demanded that the ID stays constantly the same.

Max Pts	This option defines how many points are examined to "wear" an ID. The information in the left upper corner of the preview [991] depicts how many "points" a camera sees. This is influenced by the exposure and threshold set up in the section "Camera Settings" [992]. In best case, the point count matches the ID tag count. Realistically, it is a little higher due to disturbing light. The higher "Max Pts" is, the more performance is needed. It is recommended to increase this option only if necessary. If you decrease it to save performance make sure that the real point count does not exceed it.

7.4.4.2.6.10 2D Camera Tracking Settings

This topic explains the section "2D Camera Tracking Settings" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

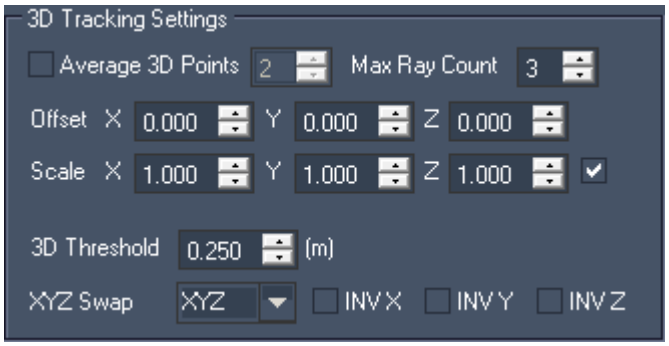


This section includes settings that influence when a tag is recognized as an ID and when the ID is blocked. Before changing these options, make sure that you see the tag's light in the preview [991] by setting up the "Camera Settings" [992] and that the "Mode" in the section "Global Tracking Settings" [1000] matches the one the IDs were configured with [989].

Max Motion	This defines the maximum distance an ID can move from one frame to the next. If it is faster, the ID will be removed from the tag. The value is depicted as an outer red circle in the preview [991]. The higher the value is, the more performance is needed. See also the below description of "Min Distance".
Min Distance	This defines the minimum distance two IDs can be next to each other. If they get closer, their IDs will be removed. The value is depicted as an inner red circle in the preview [991]. The smaller the value is, the less stable the ID detection might become as it is more likely that the two blinking patterns or physical tags overlap each other. For a scenario where several tags are attached to one object (e.g. on a panel) it is likely that they are quite close to each other (from the view of a camera or a certain rotation angle). The minimum distance should be set up in such a way that the tags do not disappear when rotating and moving the object. At the same time the max. motion should not be larger than the min. distance as this could allow that an ID "jumps" over to another tag in case of a short fault detection. This is why the two parameters are linked to each other.
Min /Max Height Width	This defines how small or big a tag can be seen from a camera before its ID is blocked.
Pre-Filter	This can be called a pre-damping as it defines the number of passed frames from which the position of a processed 2D point is averaged.

7.4.4.2.6.11 3D Tracking Settings

This topic explains the section "3D Tracking Settings" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

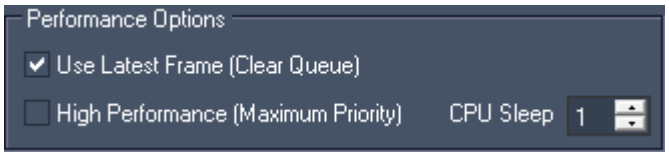


This section includes settings when the 3D Tracking Mode is used (see section "Global Tracking Settings" [1000]).

Average 3D Points	Enable the "Average 3D Points" option if you have set up several ID tags with the same ID and like to average their position. With the number field you define how many tags with the same ID should be taken into account.
Max Ray Count	This defines how many cameras are taken to determine the 3D position of a tag. The higher the value is, the more performance is needed.
Offset XYZ	With the offset options you can shift the origin of the 3D space, and thus all positions of the tags, without the need of re-calibrating all cameras. Read more... [996]
Scale XYZ	With the scale options you can increase or decrease the scaling of the coordinate-system of the 3D space, and thus all positions of the tags, without the need of re-calibrating all cameras. Read more... [996]
3D Threshold	The better the cameras are calibrated and the better the tag light detection and environment is, the more the position from a tag for one cameras consents to the position for another camera. Realistically, the tag's position is a little bit different for each camera. If the estimated position from one camera is further away then the "3D threshold", which is 25cm per default, then this camera is not taken into account and the next available one is taken.
XYZ Swap and Invert	With the swap and invert options you can rotate the X-, Y- and Z axises of the 3D space to change its orientation, and thus all positions of the tags, without the need of re-calibrating all cameras. Read more... [996]

7.4.4.2.6.12 Performance Options

This topic explains the section "Performance Options" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

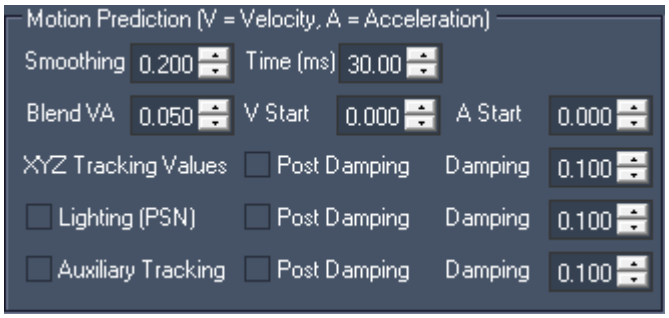


This section includes settings influencing the performance of the tracking calculation.

Use Latest Frame	The default option is to always use the latest frame transmitted from the cameras for the next processing interval. If the option is deactivated, the process waits for all cameras to send a newer frame.
High Performance Mode	This increases the prioritization of the thread for the operating system and allows ultimate CPU usage. Per default the prioritization is set to a normal level.
CPU Sleep	This is the time in milliseconds that is waited before the next frame from the cameras is read and processed.

7.4.4.2.6.13 Motion Prediction

This topic explains the section "Motion Prediction" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].



This section includes settings for smoothing, predicting and damping a tag's position. For predicting where the tag will be in the next frame, a complex formula is needed. Depending on the specific movement of your tag, some factors need to be adjusted. A fast movement without abrupt stops is very different to a slow continuous one. The factors include adjustments for velocity and acceleration. However, the most important settings are "Smoothing", "Time" and "Damping". For most tracking scenarios, the Velocity and Acceleration settings can stay on there default values.

Smoothing	The prediction calculation includes an internal velocity and acceleration part. The "Smoothing" value smoothes this data. It can be understood as a multiplying factor. The smaller Smoothing is, the more the data is damped and modified to reduce noise. Recommended values of 0.1 - 0.2 result in a lot of smoothing as commonly there is much data noise that needs to be reduced. The higher it is, the more the VA-part influences the prediction calculation. A factor of 1 means no data modification, no smoothing.
Time (ms)	"Time" should be adjusted to meet the real duration that exists between measuring the tag's position and seeing or hearing the result, e.g. projecting an image onto this position. This delay is measured in milliseconds. Delay times up to 80-120ms are common.
Blend VA V Start A Start	The velocity and acceleration part contribute to the calculation as soon as a "Start" value is met. In other words, if the velocity (and acceleration respectively) is below that start value, the VA-part is not taken into account. Velocity is measured in mm/sec and Acceleration in mm/sec². If your tag jitters slightly eventhough the tag does not move at all, you can raise the VA Start to 0.3 mm for instance to discard the VA-prediction for these small movements Blend VA blends between these two formulas with and without the VA-part.
Lighting and Aux. Tracking	This applies when you have a node attached to the ID Tag Tracker node. Per default you see in the node's input drop-down list, that every tag is expressed with a position (X, Y, Z). In case you have more than one device following the tracking it might be needed to add different damping times to the XYZ-position to route them to different output nodes. When enabling "Lighting (PSN)" you will see that each tag has an additional X2, Y2 and Z2 value; when enabling "Auxiliary Tracking" X3, Y3 and Z3 are available. As soon as you add different damping times (see below) to these values, they will not coincide any more. Please note that the two check boxes are not of interest when outputting PSN data directly from the node!
Post Damping	This activates an additional damping calculation. "Damping" is similar to the "Smoothing" as it smoothes the data. But as "Damping" happens at the very end of the calculation, after prediction, it can be understood as a post damping that smoothes the predicted curve to reduce jittering. "Post Damping" can be applied to three data sets introduced above. You can setup different damping times to...: - "XYZ Tracking Values" = X1 Y1 Z1 (default values of the node) - "Lighting (PSN)" = PSN output [1004] send out directly from the node as well as X2 Y2 Z2 if the left check box is ticked - "Auxiliary Tracking" = X3 Y3 Z3 if the left check box is ticked You can set up damping times to adjust the tracking behavior for different devices. A factor of 1 means no data modification, no damping whilst values between 0 and 0.1 mean a lot damping.

7.4.4.2.6.14 MA PSN Output

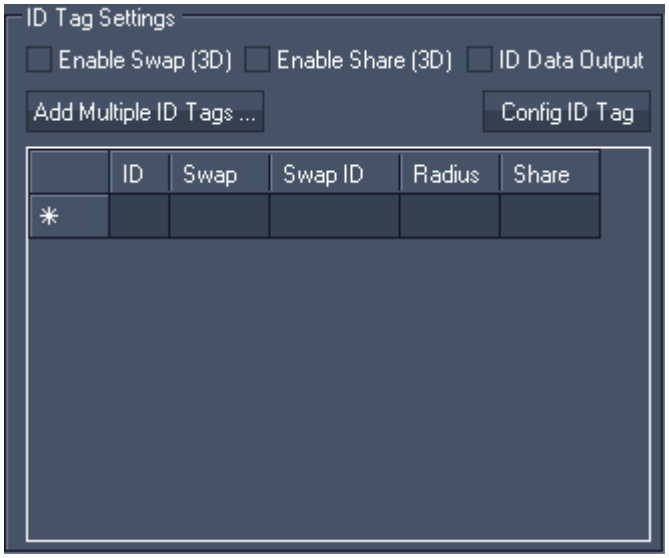
This topic explains the section "MA PSN Output" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

This section includes settings for the PSN data output. The PosiStageNet protocol is for example supported by MA Lighting consoles; more information can be found on www.posistage.net.
In general PSN needs to be activated in the Connection Manager [1262] where you can also find other settings like the IP and port for the PSN MA Tracking Server.

Enable	This outputs the X-,Y- Z-coordinate of each tag.
Inv XYZ and Swap	The coordinate system on which the tag positioning is based can be set up in two steps. First is when calibrating the cameras, second is in the section "3D Tracking Settings". Both influence the values for a tag in Widget Designer. However, if the lighting system uses a different coordinate system, you can change the position for the tags only for the PSN output without the need to adjust Widget Designer or Pandoras Box. You may invert the orientation of each axis, or even change the XYZ order into a YXZ one, for example.

7.4.4.2.6.15 ID Tag Settings

This topic explains the section "ID Tag Settings" of the ID Tag node, for general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].

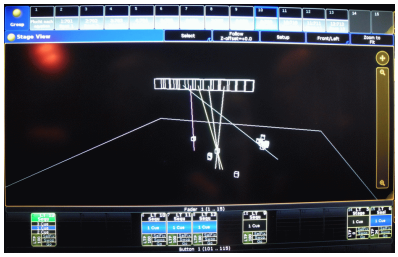


This section includes tag settings, most of them apply when the 3D Tracking Mode is used (see section "Global Tracking Settings" [1000]).
If you like to use data of a tag as an output value of the ID Tag Tracker node or if you like to see the tag in the preview [991], the tag needs to be listed in the table. To add a tag, click into the row and enter its ID or use the "Add Multiple Tags" button. If you need to change an entry, simply (right-)click into the cell. To delete a tag ID, you need to delete the entire row. To do so, click into the first cell (*) to select the row and press the "Delete" key on your keyboard.
To see the data from the node activate "ID Data Output".

Enable Swap	This enables the "Swap" function. After enabling "Swap", two listed tags need to wear a "Swap ID". Then, a "Radius" needs to be set up. Now, as soon as two tags meet each, meaning that there are as close as the "Radius" says, they will exchange their "Swap ID".
Enable Share	This enables the "Share" function. After enabling "Share", at least two listed tags need to wear a "Share ID". Then, a "Radius" needs to be set up. Now, as soon as two tags meet each, meaning that there are as close as the "Radius" says, one tag will overwrite the "Share ID" of the other one.

ID Data Output	This enables the data output meaning that the ID Tag Tracker node outputs data that can be used in other nodes.
Add Multiple Tags	This opens a dialog that fastens the process of entering multiple tags to the list that is described above.
Config ID Tag	Please see the chapter "Config ID Tag" 989.

7.4.4.2.6.16 Lighting Console Settings



This topic explains how to setup tracking with lights, e.g. a moving light (or follow spot) follows the position of a person wearing a tag. The goal is to give you a short to do list with settings for Widget Designer and for the lighting console. For general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker 985.

As an exemplary console the GrandMA 2 was chosen. Other consoles supporting PSN work also but their settings might differ.

Requirements

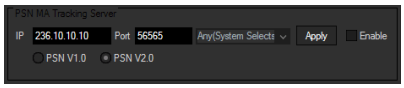
Network connection between console and Widget Designer PC
Widget Designer and the lighting console are connected via network and both are setup with a static IP address. You can use any IP range but if you additionally like to send Art-Net data you should use a 2.x.x.x IP address and 255.255.255.0 as subnet mask. Make sure that this connection works by pinging the console from the WD PC! When using the onPC-version, double-check also that your anti-virus software and Firewall are not blocking incoming data.

Same origin
The 3D world setup in WD for the tracking and the 3D space setup in the console for the (moving) lights should be based on the same origin position, i.e. 0,0,0 point and scaling. It is possible to offset and rescale data, but it makes life much easier to work with the same reference point from the beginning. You may choose to use different orientations (rotation of axises) but you must then swap or invert the axises 1004 for the PSN data. Most likely you need to choose XZY.

Lights position and orientation are measured and the light fixtures are setup accordingly
Based on the mentioned origin, the moving lights' position and orientation needs to be measured and in Stage View all light fixtures need to be placed according to these figures.
1st tip: Level the moving lights (e.g. with a water level). 0.5° rotation can have a huge impact on the position of the follow spot and it is far easier to level a device than measure its exact rotation.
2nd tip: Using a laser distometer saves time especially for many lights and large distances. There are many 3D laser tools available, we offer an interface for the 3D Disto from Leica 2076.
Before mounting the lights it makes sense to mark the point whereto you like to measure e.g with tape. Check whether the console knows your light and the distance (offset) between the pivot in the yoke and the lens. Measure then the distance between your own marker and the lens and add / subtract this when entering the height for the light
3rd tip: Mark 4-8 positions on your stage, measure them and tell the lights to go to this point. Depending on a light's offset to these various stage positions, you can modify its position and orientation until it is accurate enough.
4th tip: Check whether the movement of the lights in Stage View accords with the real movement, if not swap the Pan / Tilt parameter. For doing so, you might need to deactivate "XYZ Positioning" in "Fixture Types" temporarily.

- Tracking in WD is setup
All cameras are calibrated and when a tag moves around, it does the same in the preview without disappearing and jittering. Mark 4-8 positions on your stage, measure them and place a tag onto them. Check whether the value from the ID Tag Tracker node coincides with your measurement.

Settings to control a Moving Light / Stage Marker with a tag



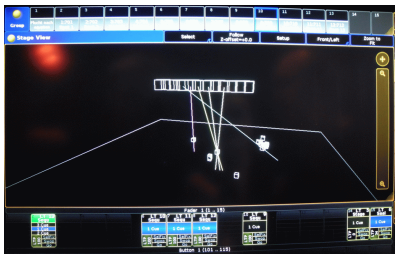
In WD...
- Connection Manager > Enable PSN 1262 protocol (do not change the IP from the PSN Server)
- ID Tag Tracker node > Enable PSN 1004 data
- ID Tag Tracker node > ID Tag Settings 1004 > add all tag IDs you want to use

In the lighting console...
- Setup > Network > PSN Network Configuration > Enabled => button is highlighted in yellow
- Add > Tracking System
  - the column "Enabled" should say "Yes"
  - the column IP" should display the IP address from the WD PC
  - the columns "Multicast" and "Port" should coincide with the (default) settings found in the WD Connection Manager

- as explained above, your patch should already include the light-fixtures; check in "Fixture Types" that "XYZ Positioning" is turn on
- add a MA-Stage Marker-fixture to the patch, that is one fixture per ID tag

- Setup > Network > PSN Network Configuration > View Tracker => displays a list of all PSN IDs, their Fixture ID and incoming XYZ-position
- assign a fixture to each PSN ID (i.e. tag ID)

- the below images depict 14 moving light fixtures hanging site by site in a truss whilst 2 stage markers (IR tags) lay on stage
- as the last step, select a fixture and go to "Position" to attach a marker to it



7.4.4.2.6.17 ID Tag Tracker Node Control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

In addition to the general commands, this nodes provides specific function commands that can replace pressing the buttons or entering values in the Configuration dialog manually. E.g.
`Node1.CameraExposure(148530,500)` sets the exposure of the camera with serial number "148530" to 500µs without opening the dialog.

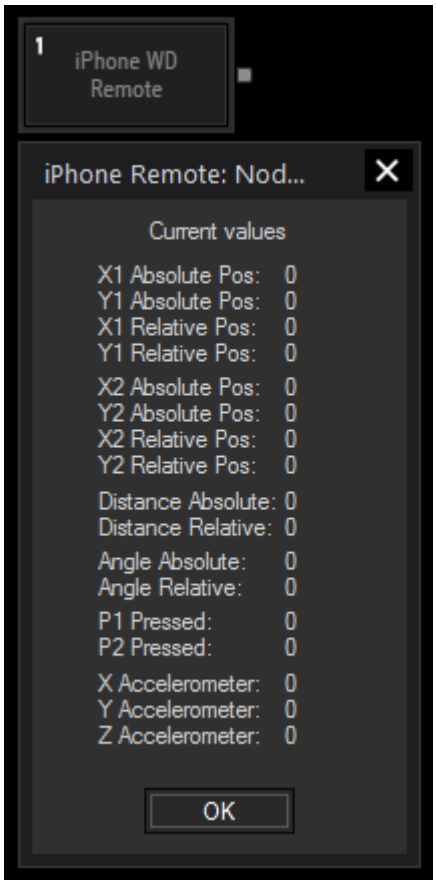
The following additional function commands are available for the ID Tag Tracker node:

- CameraActive
- CameraExposure
- CameraFPS
- CameraIR
- CameraIRLights
- CameraSelect
- CameraThreshold
- DisableDebugInfo
- EnableDebugInfo
- MaxHeight
- MaxMotion
- MaxWidth
- MinDistance
- MinHeight
- MinWidth
- PreFilter
- Reinitialize
- ResetAliasId
- ResetAllAliasIds
- SetAliasId
- SetShareModeOff
- SetShareModeOn
- SetSwapModeOff
- SetSwapModeOn
- SetSwapRadius
- TrackingHeight
- WriteDebugInfo

7.4.4.2.7 iPhone Remote Input

The iPhone Remote input node allows receiving the current iPhone remote single and multi-touch values.
Set up your iPhone Remote Control in the Remoting Tool [1273].

This node can be found under Nodes > Input > Devices > iPhone Remote



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.

Node output values

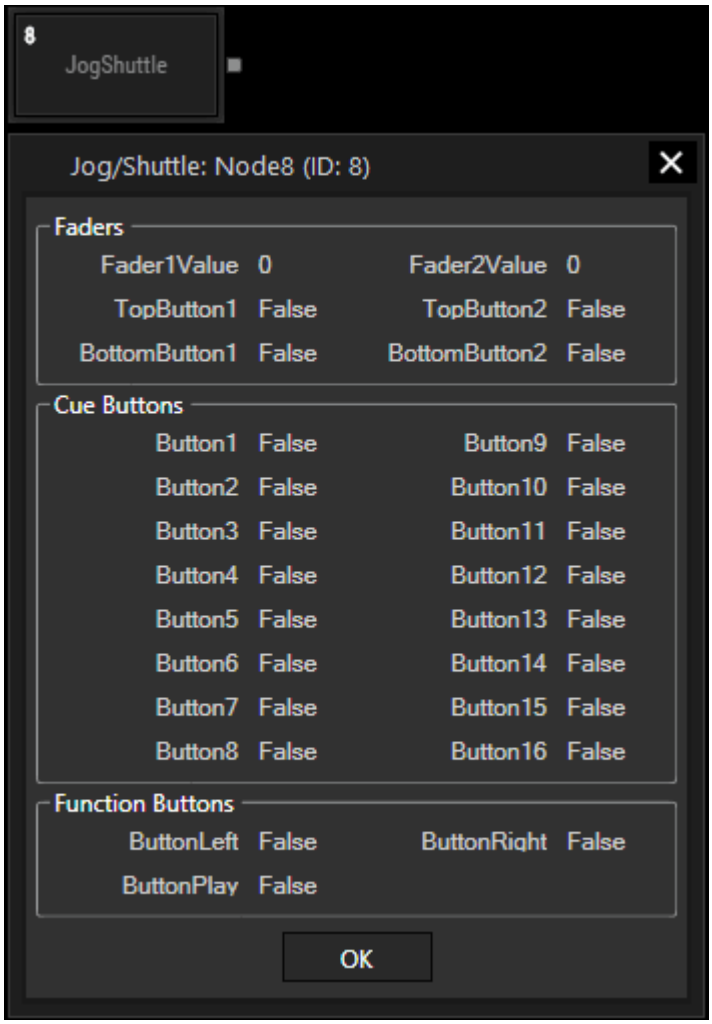
The node generates the following output:

- Absolute and Relative Positions of two points (X1/Y1 and X1/X2)
- Distance and Angle Absolute / Relative
- P1 or P2 pressed (True=1 and False=0)
- X,Y and Z Accelerometer

7.4.4.2.8 JogShuttle Input

The JogShuttle input node allows you to register fader values and button clicks of a Jog/Shuttle hardware [1991] device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Jog/Shuttle" [1364] which also shows how to work with the device using the scripting language.

This node can be found under Nodes > Input > Devices > JogShuttle



If you like to connect to a Fader Extension device [1993], please use the FaderExtension input node [983].

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.

Node output values

The Node generates the following output:

- Fader Value 1 and 2
- TopButton State 1 and 2 (True = Pressed down / Clicked)
- BottomButton State 1 and 2 (True = Pressed down / Clicked)

Note that the ID 3 to 8 for the Fader and Buttons belong to the Fader Extension.

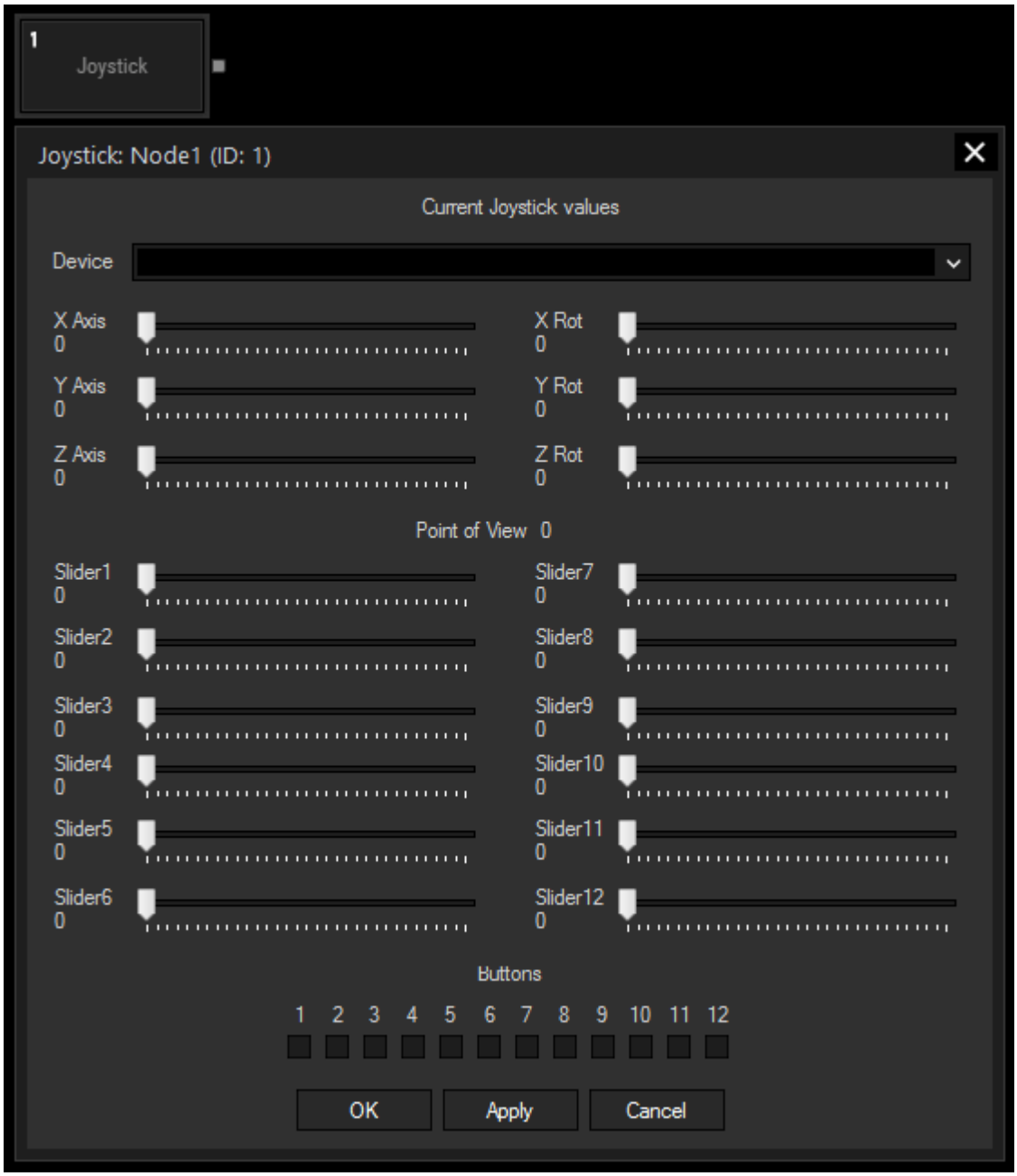
- CueButton State 1 - 16 (True = Pressed down / Clicked)
- LeftButton State (True = Pressed down / Clicked)

- PlayButton State (True = Pressed down / Clicked)
- RightButton State (True = Pressed down / Clicked)
- IsConnected State

7.4.4.2.9 Joystick Input

The Joystick input node allows intercepting standard Windows gaming device input values for axis slider and button input.

This node can be found under Nodes > Input > Devices > Joystick



Node Properties

Device:
Choose your gaming device from the list.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, Node1.TintColor.SetRGB(125,0,255), colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command Node1.SetParam(ID,new Value) or WDNodeSetParam(NodeID,ParamID,Value).

Node output values

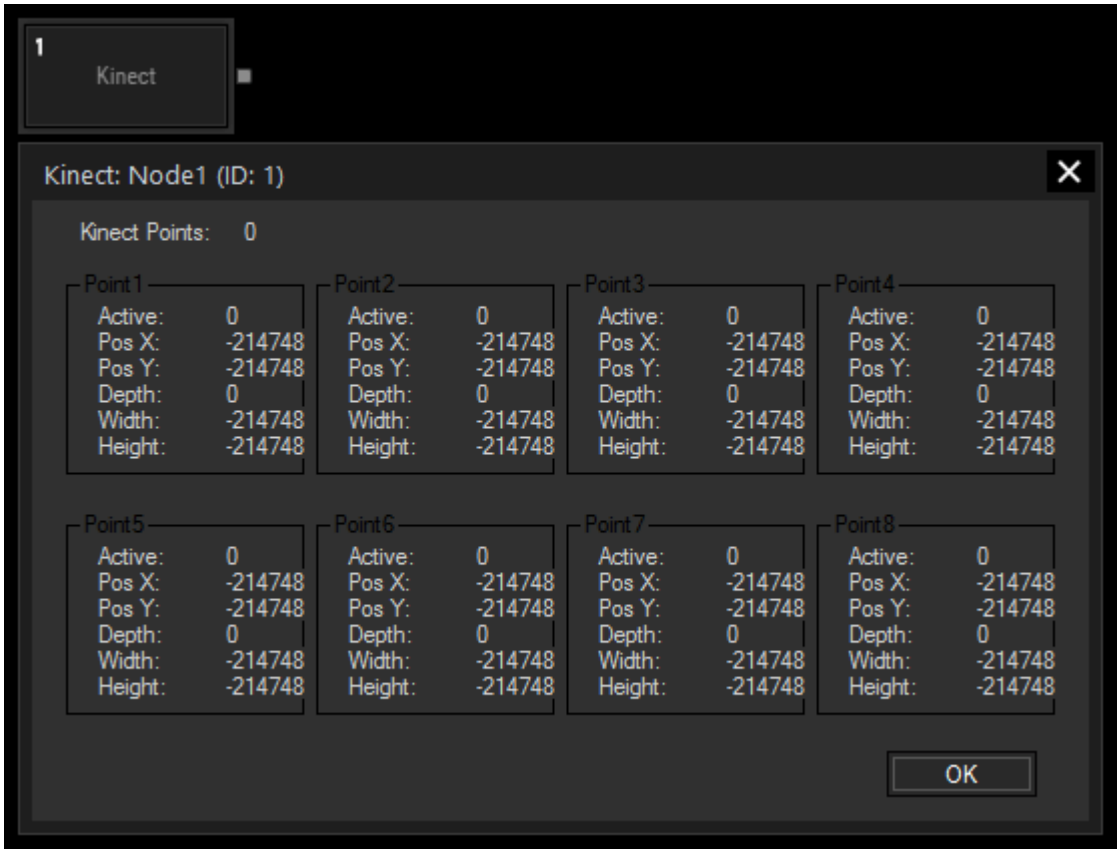
The node generates the following output:

- X Axis
- Y Axis
- Z Axis
- X Rot Axis
- Y Rot Axis
- Z Rot Axis
- Point of View
- Slider 1 to 12
- Button 1 to 12

7.4.4.2.10 Kinect Input

The Kinect input node allows capturing up to 8 filtered and tracked points/regions of interest via the Kinect device. Setup your device in the Kinect Tool [1283] first.

This node can be found under Nodes > Input > Devices > Kinect



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.

Node output values

The node generates the following output for each of the 8 points:

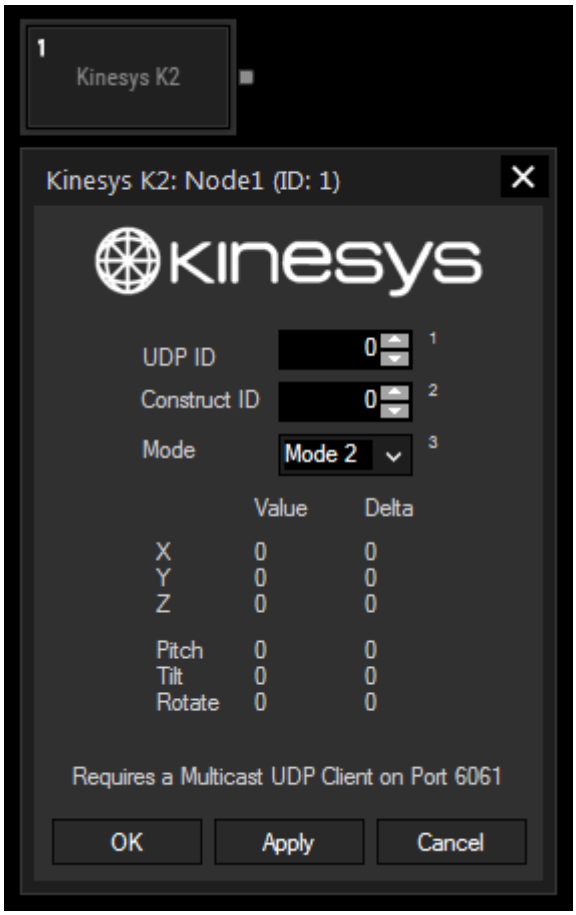
- Active
- Pos X
- Pos Y
- Depth
- Delta Pos X
- Delta Pos Y
- Delta Depth

It outputs the overall active point count as well:
- Point Count

7.4.4.2.11 Kinesys Input

The Kinesys K2 Input Node allows you to grab X,Y,Z, Pitch, Tilt & Roll data from a Kinesys K2
Please note, that a Multicast UDP Client on Port 6061 is required. Set it up in the Connection Manager [1258].

This node can be found under Nodes > Input > Devices > Kinesys K2



Node Properties

UDP ID:
Choose the ID of your UDP Multicast Connection (see Connection Manager [1258]).

Construct ID:
Choose the ID of the Construct as it is named in your Kinesys K2.

Mode:
Select the Mode according to the protocol version of Kinesys K2. Mode 2 is the currently used standard, select Mode 1 for older versions.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

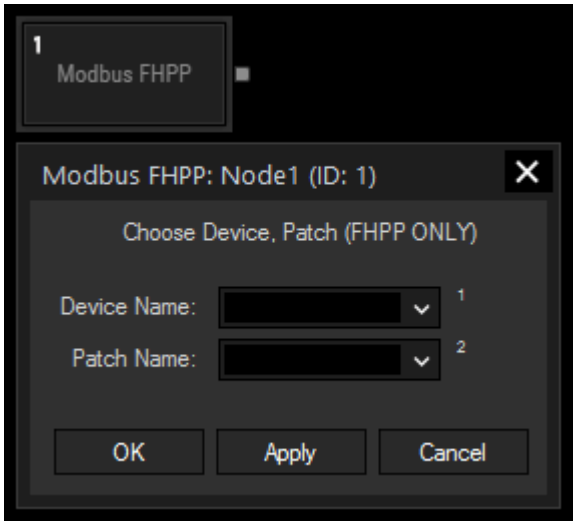
- Kinesys K2 X,
- Kinesys K2 X Delta,
- Kinesys K2 Y,
- Kinesys K2 Y Delta,
- Kinesys K2 Z,
- Kinesys K2 Z Delta,
- Kinesys K2 Pitch,
- Kinesys K2 Pitch Delta,
- Kinesys K2 Tilt,
- Kinesys K2 Tilt Delta,
- Kinesys K2 Rotate,
- Kinesys K2 Rotate Delta.

7.4.4.2.12 Modbus FHPP

The Modbus FHPP input node allows you to retrieve the current value of a Modbus element that contains data in the FHPP format. For other data types, please use the other input node Modbus Input [1015] instead.
Both elements configured as input or output type can be addressed. The respective Master and patch need to be set up before using this node, for more information please refer to the chapters Modbus Master Configuration [1299] and Modbus Patch Configuration [1300].

Modbus FHPP values can also be retrieved via scripting, as described in the chapter Using Modbus [1302].

This node can be found under Nodes > Input > Devices > Modbus FHPP



Node Properties

Device Name:
Select the Master device from the drop-down.

Patch Name:
Select the patch from the drop-down.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

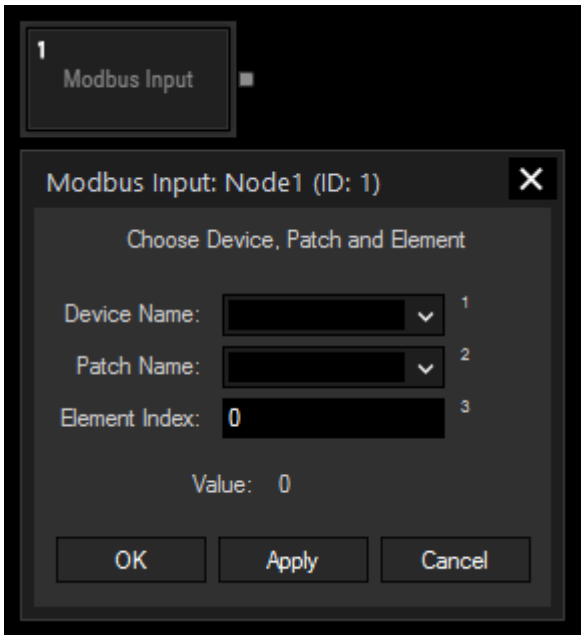
- ccon_enable (Drive enable)
- ccon_release_break (Open break)
- ccon_reset (Reset fault)
- ccon_stop (Stop)
- cpos_clear (Clear remaining position)
- cpos_halt (Halt)
- cpos_jogn (Jog negative)
- cpos_jogp (Jog positive)
- cpos_start (Start positioning task)
- cpos_teach (Teach actual value)
- in_recordset (Input data Record no.)
- out_recordset (Output data Record no.)
- position (Input data Actual position)
- scon_enabled (Drive enabled)
- scon_fault (Fault)
- scon_open (Operation enabled)
- scon_vload (Supply voltage is applied)
- scon_warn (Warning)
- spos_ack (Acknowledge start)
- spos_dev (Drag (deviation) error)
- spos_halt (Halt)
- spos_mc (Motion complete)
- spos_mov (Axis is moving)
- spos_ref (Axis is referenced)
- spos_still (Standstill control)
- spos_teach (Acknowledge teach / sampling)
- scon_fct (Drive control by software)

7.4.4.2.13 Modbus Input

The Modbus input node allows you to retrieve the current value of a Modbus element. Both elements configured as input or output type can be addressed. The respective Master and patch need to be set up before using this node. For more information please refer to the chapters Modbus Master Configuration [1299] and Modbus Patch Configuration [1300]. In case, you use elements that contain data in the FHPP format, please use the other input node Modbus FHPP [1013] instead.

Modbus values can also be retrieved via scripting, as described in the chapter Using Modbus [1302].

This node can be found under Nodes > Input > Devices > Modbus Input



Node Properties

Device Name:
Select the Master device from the drop-down.

Patch Name:
Select the patch from the drop-down.

Element Index:
Enter the index of the element you would like to retrieve the current value from. Please note that the index starts with "0" for the first element. If you have specified names for your elements in the patch configuration [1300], you can as well enter the element's name instead of the index.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

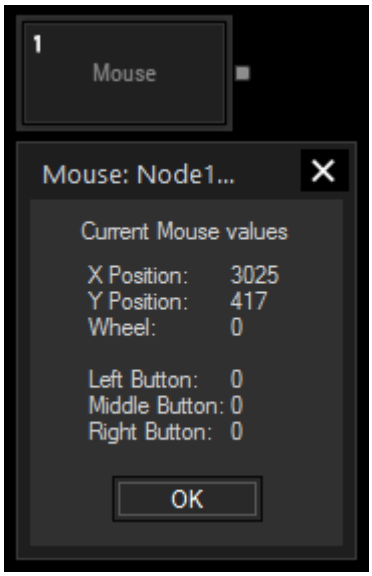
The Node generates the following output:

- Value (as integer)

7.4.4.2.14 Mouse Input

The Mouse input node provides the Mouse X and Y position as well as the three button press states Left, Middle and Right, additionally a relative value for the mouse wheel.

This node can be found under Nodes > Input > Devices > Mouse



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.

Node output values

The node generates the following output:

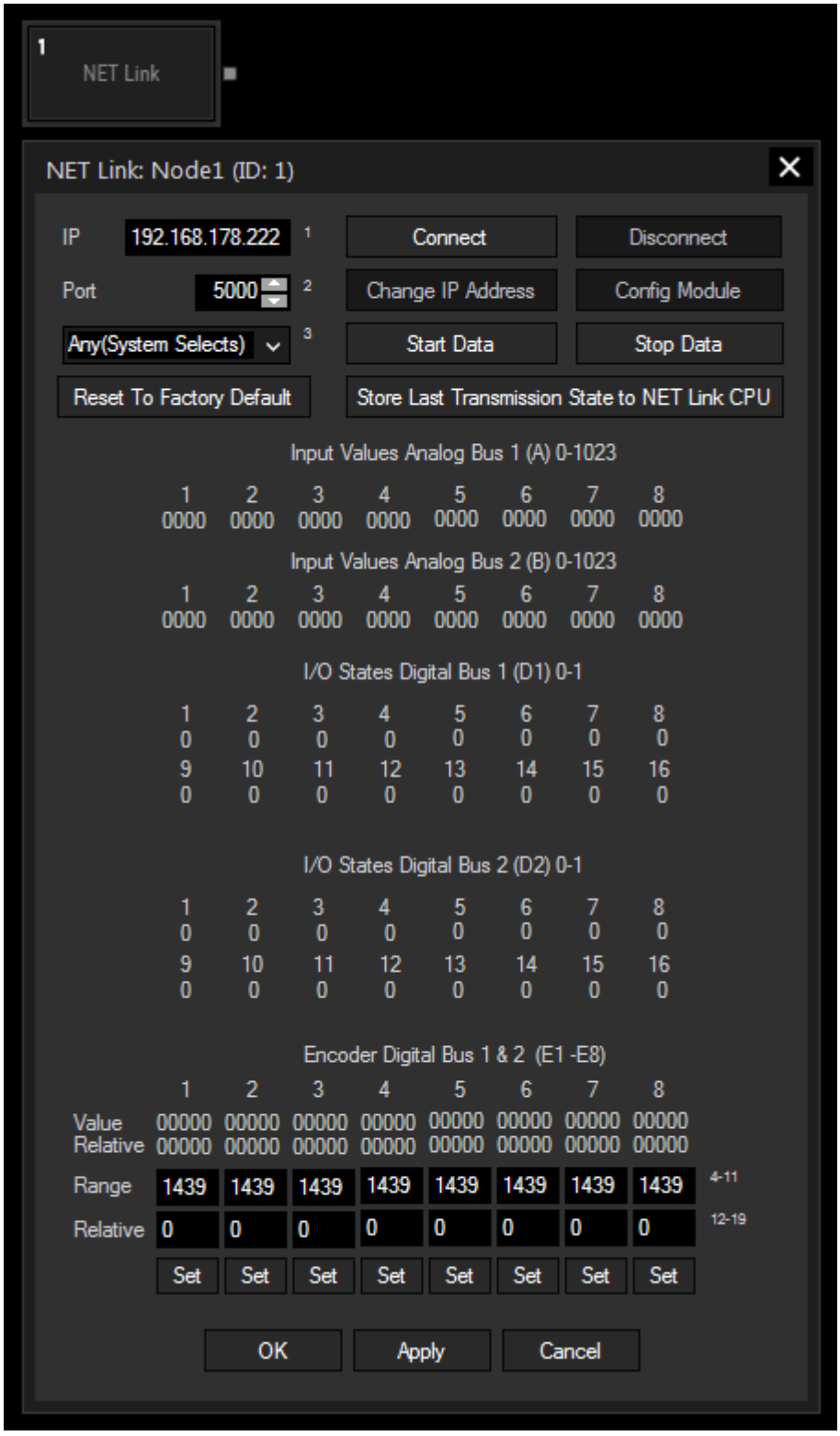
- X Pos
- Y Pos
- Wheel ("-1" for down, "1" for up)
- Left Click
- Middle Click
- Right Click

7.4.4.2.15 NET Link - Generic I/O

The NET Link Input node allows to communicate with the Calibration Link or a NET Link equipped with input and / or output boards. Please see more information in the chapter covering the NET Link [2000].
On the one hand the node receives information send from available inputs, hence, it provides this data to other nodes in Widget Designer. On the other hand, it allows to configure the device and send commands to it. In case of output boards, a contact closure can be commanded through this node.

The Calibration Input node [981], the Relay Output node [1202] and the tool "Projector Calibration Manager [1496]" might also be of interest for you.

This node can be found under Nodes > Input > Devices > NET Link - Generic I/O



Node properties

IP and Port:
Enter the correct IP address and port from the NET Link's CPU or from the Calibration Link.

Select Adapter:
Specify here the NIC you want to use for communication with the NET Link.

Reset To Factory Default:
This buttons resets the above settings in the Widget Designer interface. To reset the device itself (to the IP

address 192.168.178.222 and the port to 5000), hold the "Reset" button down whilst plugging the power into the device. Release the button again.

Connect / Disconnect:
Before starting to communicate with the device, for instance receive data, the node must be connected to the device. The connection itself consumes no performance.

Change IP Address:
This button opens a new dialog whereto you may enter another IP address and port for the device. Power-cycle the device to apply the changes.

Config Module:
A NET Link / Calibration Link is configured by us as you have ordered it. However, if you have changed some input / output boards, the processor must be configured in terms of giving him the information which boards are connected to it. Click the "Config Module" button to open a new dialog where you may choose the according boards per processor connection. Find more details in the NET Link hardware chapter [2002].

Start Data / Stop Data:
As soon as data is processed via the network from the device to Widget Designer, performance is drawn.

Store Last Transmission State to NET Link CPU:
Click this button and power-cycle the device if you wish that it remembers whether it should (not) send data as soon as it is powered up.

Encoder Digital Bus - Range:
Enter the encoder's amount of steps. If the encoder provides 1440 steps, enter 1439 because the counting starts with zero.

Encoder Digital Bus - Relative:
Enter zero to reset the relative step count for the current encoder position or enter an offset value.

Set:
Confirm your Range and Relative Values with "Set".

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

In addition to the general commands, this nodes provides specific function commands that can replace pressing the buttons in the Configuration manually. E.g. `Node1.StartDate` starts data transmission without opening the dialog.

The following additional function commands are available for the Net Link Generic I/O node:

- ChangeIp
- Connect
- Disconnect
- GetConfig
- SetConfig
- SetRelayDigitalBus1
- SetRelayDigitalBus2
- StartData
- StopData
- StoreLastTxMode
- Tx1 - Tx8

Node output values

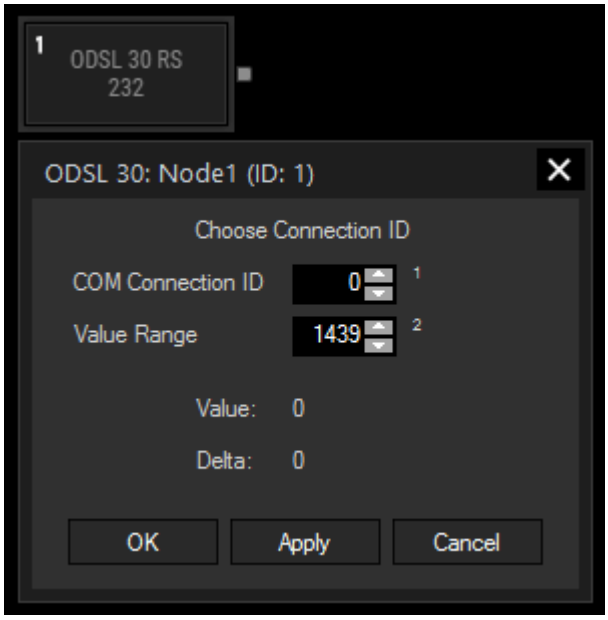
The node generates the following output:

- A1 to A8 (0 - 1023) for the first analog bus
- B1 to B8 (0 - 1023) for the second analog bus
- D1.1 to D1.128 (Open = 0, Closed = 1) for the first digital bus
- D2.1 to D2.128 (Open = 0, Closed = 1) for the second digital bus
- Connected

7.4.4.2.16 ODSL 30 Input

The ODSL 30 Input node lets you read out distance data from the optical distance measuring sensor ODSL 30 via a local COMPort connection (RS232). Please set up a respective Com connection [1269] before using this node.

This node can be found under Nodes > Input > Devices > ODSL 30



Node Properties

COM Connection ID:
Choose the COM Port Connection ID to listen to.

The following settings need to be used:
600/8n1 (Baud Rate: 9600, Parity: None, Data Bits: 8,Stop Bits:1).

Value Range:
The range defines the jump point of a maximum value back to zero.
Enter the range in mm. This is only important for correct calculation of delta values when using as incremental length count.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

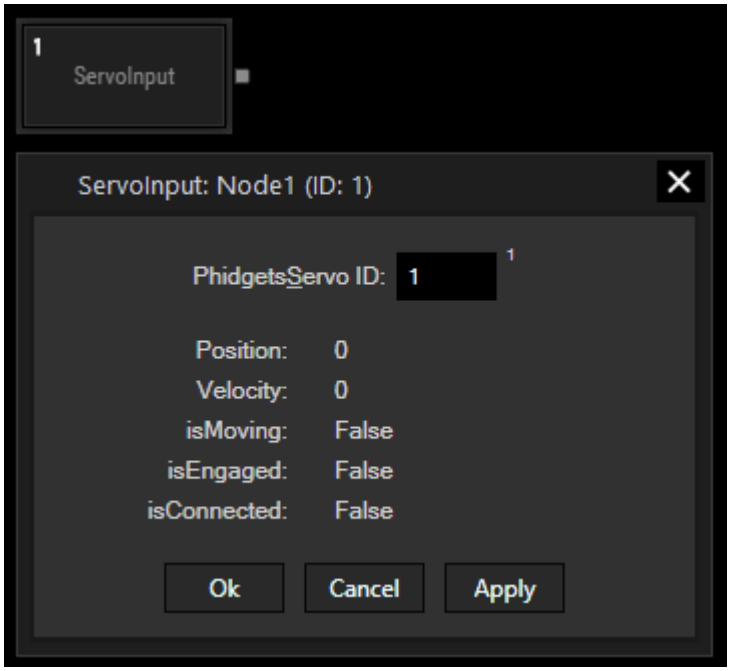
The node generates the following output:

- Value
- Delta

7.4.4.2.17 Phidgets Servo Input

The Phidgets Servo Input node lets you read out data from Servo Phidget hardware device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Phidgets Servo" [1428] which also shows how to work with the device using the scripting language.

This node can be found under Nodes > Input > Devices > Phidgets Servo Input



Node Properties

PhidgetsServo ID:
Enter the ID you used in the Configuration dialog [1428] to connect to the according Phidget.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

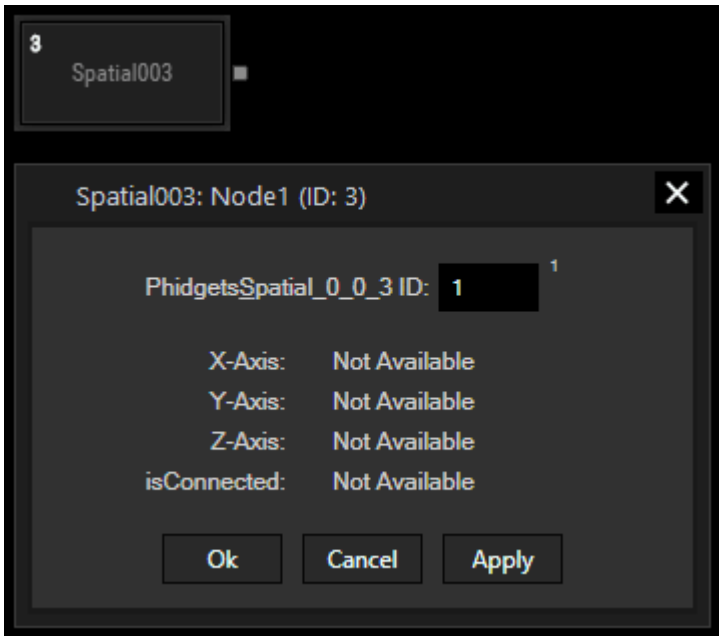
The node generates the following output:

- Position
- Velocity
- isMoving (True/False)
- isEngaged (True/False)
- isConnected (True/False)

7.4.4.2.18 Phidgets Spatial 003

The Phidgets Spatial 003 Input node lets you read out data from Spatial 003 Phidget hardware device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Phidgets Spatial 0/0/3" [1434] which also shows how to work with the device using the scripting language.

This node can be found under Nodes > Input > Devices > Phidgets Spatial 003



Node Properties

PhidgetsSpatial003 ID:
Enter the ID you used in the Configuration dialog [1434] to connect to the according Phidget.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

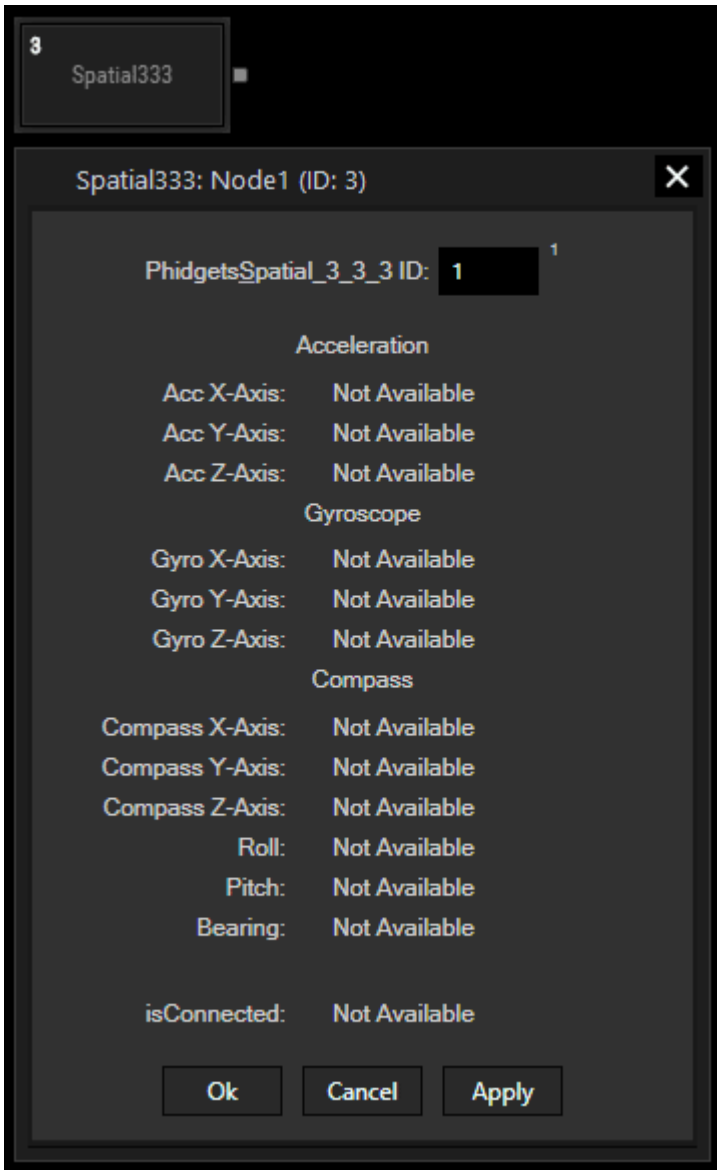
The node generates the following output:

- X-Axis
- Y-Axis
- Z-Axis
- isConnected (True/False)

7.4.4.2.19 Phidgets Spatial 333

The Phidgets Spatial 333 Input node lets you read out data from Spatial 333 Phidget hardware device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Phidgets Spatial 3/3/3" [1364] which also shows how to work with the device using the scripting language.

This node can be found under Nodes > Input > Devices > Phidgets Spatial 333



Node Properties

PhidgetsSpatial333 ID:
Enter the ID you used in the Configuration dialog [1436] to connect to the according Phidget.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

- Acc X-Axis
- Acc Y-Axis
- Acc Z-Axis
- Gyro X-Axis
- Gyro Y-Axis
- Gyro Z-Axis
- Compass X-Axis
- Compass Y-Axis
- Compass Z-Axis
- Roll
- Pitch
- Bearing
- isConnected (True/False)

7.4.4.2.20 Projector Control Input

The Projector Control Input node allows to communicate with projectors via a TCP connection. The communication includes controlling the projector, e.g. closing the shutter or powering it, as well as receiving feedback information, e.g. regarding the shutter or power status. The node itself has three status "lights" that change color depending on the projector's response. Hence, the node provides a good overview without the need to open its properties or connect the node with other controls like labels [888]. In order to send a message to a projector you may either press the according button in the node or execute a node command as described below.

This node can be found under Nodes > Input > Devices > Projector Control

Node properties

IP and Port:
Enter the correct IP address and port from the projector.

Connect / Disconnect:
If the connection can be enabled, the third "light" from the node changes to green; yellow indicates that the connection attempt failed whilst red means that the projector is definitely not connected, e.g. because the connection has not been initialized yet or the Disconnect button has been pressed.

Command Messages:
The Command Messages allow you to control the projector. Enter the messages for each according action. You may find the syntax in the documentation of your projector, mostly referred to as TCP or serial commands. Please note, that a carriage return or line feed could be demanded. Please see the chapter Syntax TCP, UDP, serial

messages 944 for more information.
Click the according Send button to forward the message to the projector. Alternatively you may execute a node control as described below.

If your projector supports PJLink and the password is disabled (by using the browser control window) the syntax to turn it on would be: `%1POWR 1[CR]`
Projectors from Barco usually request hexadecimal commands. Please find an example in our forum. To send "Lamp On" for example: [hfe h00 h00 h03 h02 h76 h1a h01 h96 hff]

Request Messages:
The Request Messages allow you to get feedback from the projector. First, enter the messages for the according request. You may find the syntax in the documentation of your projector, mostly referred to as TCP or serial commands. Please note, that a carriage return or line feed could be demanded. Please see the chapter Syntax TCP, UDP, serial messages 944 for more information.
Enter the time interval in which the request should be forwarded to the projector.
Lastly, enter the expected message what the projector responds, one for a positive answer and one for a negative one. As long as the request is answered positively, the according "light" of the node stays green. A negative answer is indicated by a red color.
The first light informs about the shutter state, the second about the lamp status and the third about the connection status.

The ASCII Request and Response offer you the possibility to check for any parameter besides Lamp and Shutter. All you have to do is enter the symbols that indicate the response.
E.g. For Christie projectors:
Request Message: (CON?) asks for the current contrast value
Response Message: (CON!50) where "(CON!" marks the beginning of the response string

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
The node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

In addition to the general commands, this nodes provides specific function commands that can replace pressing the buttons in the Configuration manually. E.g. `Node1.PowerOn` sends the respective command to the projector.

The following additional function commands are available for the Projector Control node:

- Connect
- Disconnect
- Input1 - Input4
- LampOff
- LampOn
- PowerOff
- PowerOn
- ShutterOff
- ShutterOn

Node output values

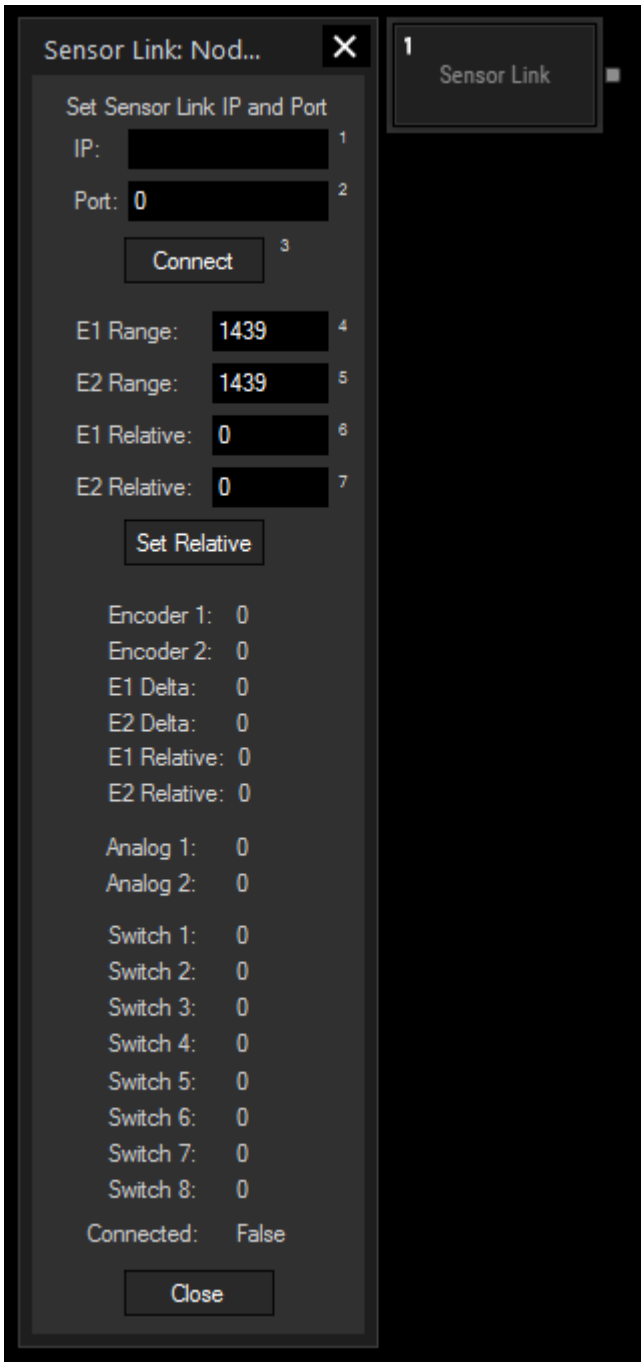
The node generates the following output:

- Connected 0,1
- Lamp State 0,1
- Shutter State 0,1
- ASCII Response

7.4.4.2.21 Sensor Link Input

The Sensor Link input node provides the absolute, relative and delta values of the Sensor Link [2033] device - the encoder inputs, the analog 0-10V inputs and the switch states 1 to 8.

This node can be found under Nodes > Input > Devices > Sensor Link



Node Properties

Please enter the Sensor Link's **IP** as well as **Port** number and press **Connect**.

E1 Range:
Enter the encoder's amount of steps. If the encoder provides 1440 steps, enter 1439 because the counting starts with zero.
Please note that the encoder's range has to be entered in the SensorLink Configuration Tool [2037], too.

E2 Range:
Enter the encoder's amount of steps. If the encoder provides 1440 steps, enter 1439 because the counting starts

with zero.
Please note that the encoder's range has to be entered in the SensorLink Configuration Tool 2037, too.

E1 Relative:
Enter zero to reset the relative step count for the current encoder position or enter an offset value.

E2 Relative:
Enter zero to reset the relative step count for the current encoder position or enter an offset value.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

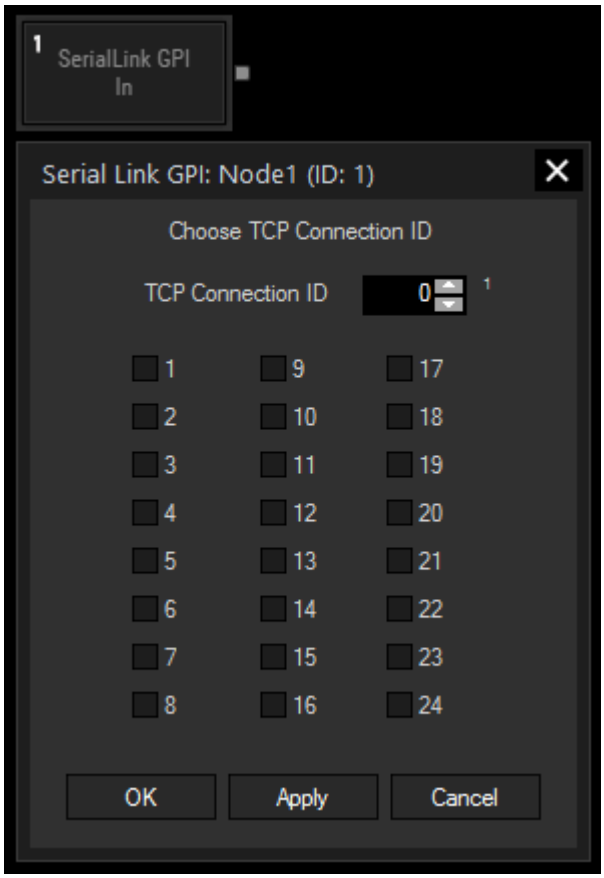
The node generates the following output:

- E1
- E2
- Analog 1
- Analog 2
- Switch 1
- Switch 2
- Switch 3
- Switch 4
- Switch 5
- Switch 6
- Switch 7
- Switch 8
- E1 Delta
- E2 Delta
- E1 Relative
- E2 Relative
- Link Connection State

7.4.4.2.22 SerialLink Input

The SerialLink [2044] input node provides all 24 GPI contact closure states.
Connect a Serial Link via TCP in the Connection Manager [1258] and set its TCP ID in the input node to listen to the incoming packets.

This node can be found under Nodes > Input > Devices > Serial Link GPI



Node Properties

TCP Connection ID:
Enter here the SerialLink's TCP Connection ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

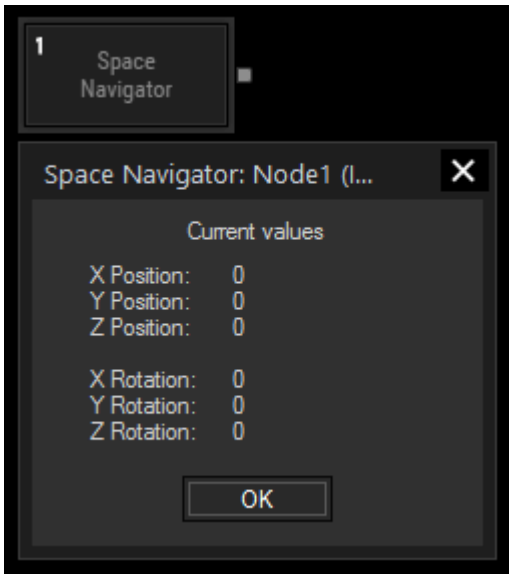
The node generates the following output:

- Status for all 24 GPIs (True (10V)=1 or False (0V)=0)

7.4.4.2.23 Space Navigator Input

The Space Navigator input node provides the 3D Mouse's X, Y and Z position as well as its X, Y and Z Rotation. It is necessary to install the 3D Connexion Driver before using this tool.

This node can be found under Nodes > Input > Devices > Space Navigator



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.

Node output values

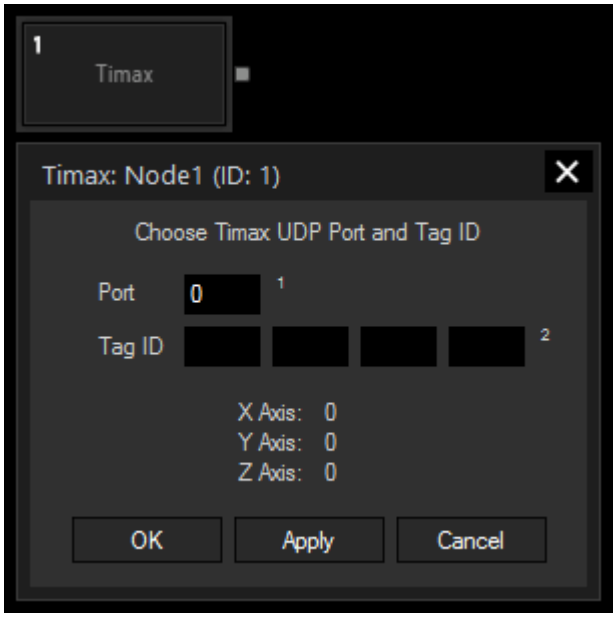
The node generates the following output:

- X Pos
- Y Pos
- Z Pos
- X Rot
- Y Rot
- Z Rot

7.4.4.2.24 Timax Input

The Timax input node provides XYZ data of the Ubisense Timax tracker.

This node can be found under Nodes > Input > Devices > Timax



Node Properties

Port:
Enter the Timax UDP Port.

Tag ID:
Enter the Timex Tag ID.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

- X Position
- Y Position
- Z Position

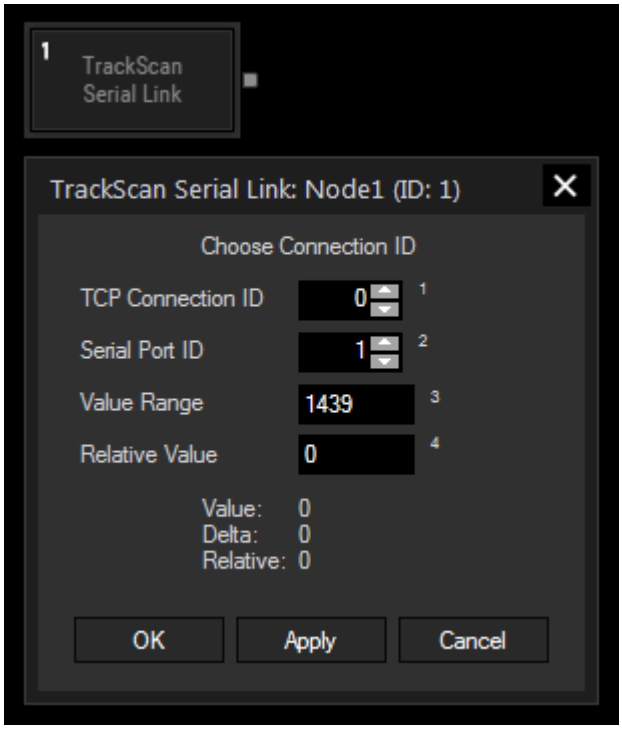
7.4.4.2.25 TrackScan Serial Link Input

The TrackScan Serial Link input node provides the integration of a Barcode scanner to read values in mm for up to 2 km of distance via our SerialLink device. It requires a Serial Link to be connected as TCP client in the Connection Manager.

Please note:
There are two different revisions of the TrackScan device in circulation:
Rev.1 TrackScans (57600/8-N-1) only connect to the WD with the TrackScan Input Node [1032] (via the local COM Port) and not with this TrackScan Serial Link input Node!
Rev.2 TrackScans (9600/8-N-1) can be used with the TrackScan Input Node (via the local COM Port) and can also be connected via this TrackScan Serial Link Input Node (through a Serial Link Device).

Important:
The Serial Link Ports must be set to 9600 Parity off. Please make sure that TX and RX are crossed (e.g. by using gender changers and a null modem cable between Serial Link and each TrackScan)!

This node can be found under Nodes > Input > Devices > TrackScan Serial Link



Node Properties

TCP Connection ID:
Enter here the Serial Links Connection ID. The TCP Connections can be found and setup in the Connection Manager [1258].

Serial Port ID:
Choose here the Serial Links Port ID the TrackScan is connected to.

Value Range:
The range defines the jump point of a maximum value back to zero.
The range is only important for rotating platforms, to determine the index point. Enter the range in mm. This is only important for correct calculation of delta values when using as incremental length count.

Relative Value:
Enter an offset value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

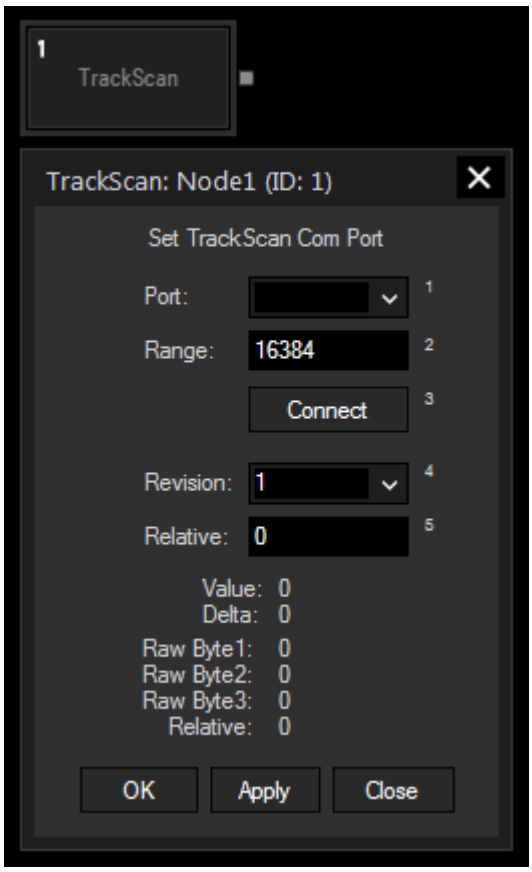
- TrackScan Serial Link Value,
- TrackScan Serial Link Delta.
- TrackScan Serial Link Relative

7.4.4.2.26 TrackScan Input

The TrackScan input node provides the integration of a Barcode scanner to read values in mm for up to 2 km of distance.

Please note:
There are two different revisions of the TrackScan device in circulation:
Rev.1 TrackScans (57600/8-N-1) only connect to the WD with this TrackScan Input Node (via the local COM Port).
Rev.2 TrackScans (9600/8-N-1) can be used with this TrackScan Input Node (via the local COM Port) and can also be connected via the TrackScan Serial Link [1031] Input Node (through a Serial Link Device).

This node can be found under Nodes > Input > Devices > TrackScan



Node Properties

Port:
Choose the TrackScan COM Port to listen to. The COM Port Connections can be found and setup in the Connection Manager [1258].

Range:
The range defines the jump point of a maximum value back to zero.

The range is only important for rotating platforms, to determine the index point. Enter the range in mm. This is only important for correct calculation of delta values when using as incremental length count.

Revision:
Please choose here the Revision of your TrackScan (1 or 2).

Relative Value:
Enter an offset value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

In addition to the general commands, this nodes provides specific function commands that can replace pressing the buttons in the Configuration manually. E.g. `Node1.Disconnect` disrupts the connection with the current device without opening the dialog.

The following additional function commands are available for the TrackScan node:

- Connect
- Disconnect
- FlushPort

Node output values

The Node generates the following output:

- TrackScan Value
- TrackScan Delta
- TrackScan Byte 1
- TrackScan Byte 2
- TrackScan Byte 3
- TrackScan Relative

7.4.4.2.27 WII Input

The Wii input node provides all 4 IR tracking coordinates as well as all button states and accelerometer data for up to 4 Wii controllers. The additional Nunchuk and Balance Board are also supported. The Wii controller needs to be connected via Bluetooth as HID.

This node can be found under Nodes > Input > Devices > Wii Input

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "Node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)` colors the node in purple.

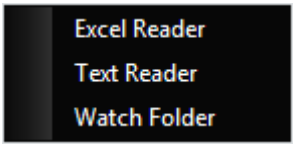
Node output values

The node generates the following output:

- Please see all items listed in the picture above

7.4.4.3 File System Input

The File System input nodes allow you to retrieve data and live changes from Excel files, text files and watch folders.



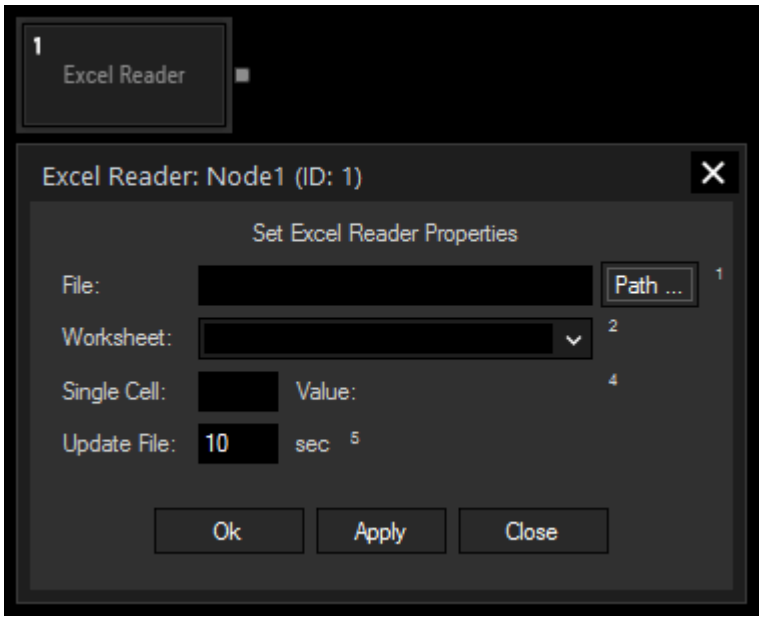
Excel Reader [1035]
Text Reader [1036]
Watch Folder [1037]

7.4.4.3.1 Excel Reader Input

The Excel Reader Input Node allows you to readout cells from an excel sheet as text.

Please note that Microsoft Excel has to be installed on the computer if you want to use this node!

This node can be found under Nodes > Input > File System > Excel Reader



Node Properties

File:
Use the Path-Button to browse to the document you want to read.

Worksheet:
Press apply after you have selected your file.
In the list select now one of the worksheets your file contains.

Single Cell:
Enter the cell you want to read out, e.g. B11.

Update File:
Enter the update interval.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.

In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

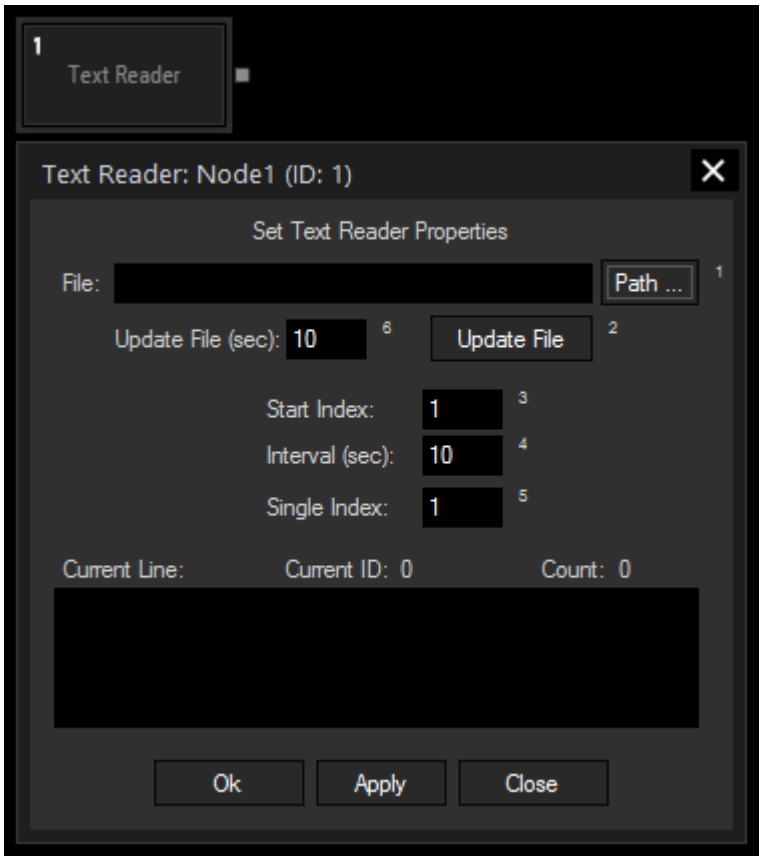
The Node generates the following output:

- Cell (as String)

7.4.4.3.2 Text Reader Input

The Text Reader input node allows you to read out text files (.txt) for further processing.

This node can be found under Nodes > Input > File System > Text Reader



Node Properties

File:
Click on **Path** and choose your text file via the windows browser.

Update File:
Please enter the interval time the files should be updated in seconds.
To do it manually click on **Update File**.

Start Index:
Enter the Index of the line the cycling should start at. This will affect the outputs Current Line and Current ID.

Interval (seconds):
All Text Lines will be cycling in the entered amount of time. This will affect the outputs Current Line and Current Line ID.

Single Index:
Enter the Index of the line that you want to route to the output Single Line.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

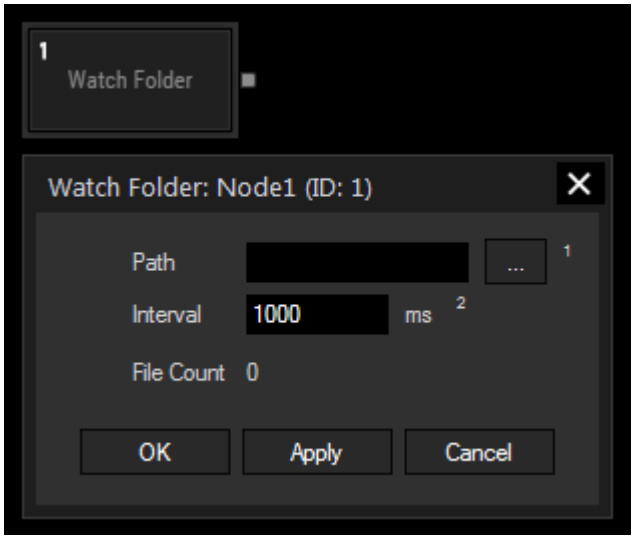
- Line Count
- Last Line
- Current Line
- Current Line ID
- Single Line
- RND Line (Random Line)
- Entire Text

7.4.4.3.3 Watch Folder

The Watch Folder input node monitors changes of a selected folder in your windows explorer. It provides a list of the files located in this folder which is updated as soon as files are added to the watch folder.

For receiving this list, connect a Variable output node [1209] to the Watch Folder node. The Variable node should direct to a global list type variable [1900].

This node can be found under Nodes > Input > File System > Watch Folder



Node properties

Path:
Enter the path of your current watch folder in windows explorer manually or select a path from the explorer with the "..." button.
E.g.: C:\Christie\content\Watch

Interval:
Enter the interval in milliseconds how often the Watch Folder node should check for updated files.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

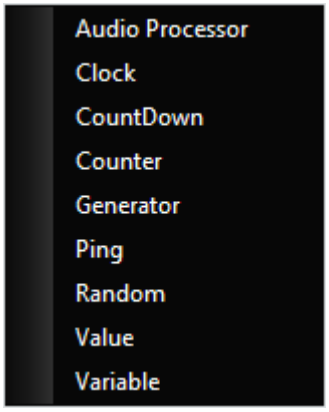
Node output values

The Node generates the following output:

- File Count: The number of files in your watch folder
- File Paths: A list containing the complete file paths of all files located in the folder

7.4.4.4 Generic Input

The Generic input nodes provide all kind of generic system or project related values, like variable [1900] values, the computer time and date, and different automatically generated values like random numbers or a count down.



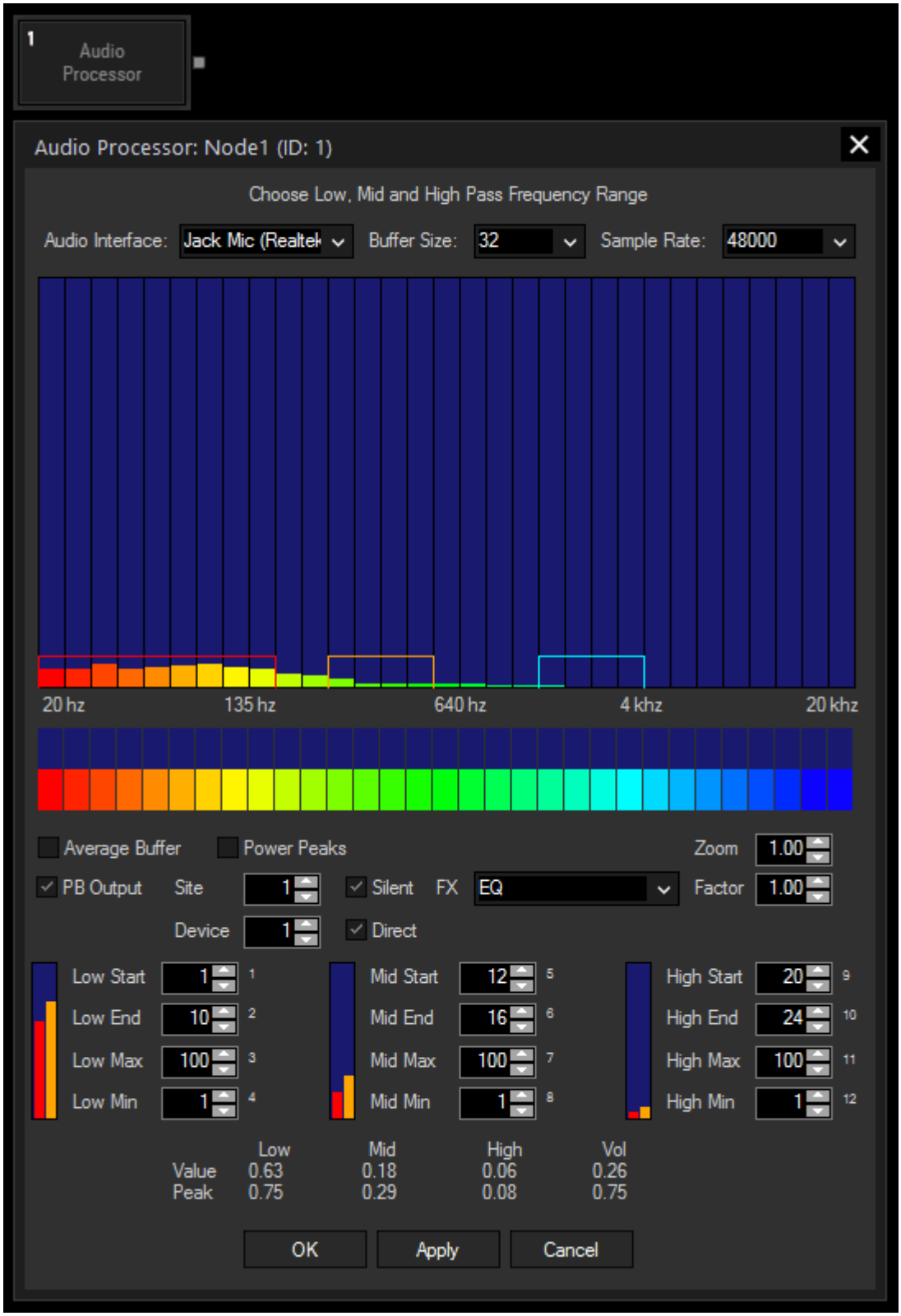
Audio Processor [1038]
Clock [1041]
Count Down [1042]
Counter [1043]
Generator [1044]
Ping [1047]
Random [1048]
Value [1049]
Variable [1209]

7.4.4.4.1 Audio Processor Input

The Audio Processor input node allows you to monitor the level of 32 frequency bands, ranging from 20Hz to 20kHz, of a selectable audio input source. You can also define three frequency band passes (low - red, mid - orange, high - cyan) and monitor their levels as well as peak values.

Additionally, the node can deliver input values for the Pandoras Box FX EQ [591], EQ Media [588] and EQ Waveform [590].

This node can be found under Nodes > Input > Generic > Audio Processor



Node Properties

Audio Interface:
Choose your Audio Interface from the list.

Buffer Size:
Select a buffer size from the drop-down. The larger the buffer, the more values per second will be analyzed.

Sample Rate:
Select the sample rate from the list your sound device is set up with.

Zoom:
This value lets you zoom in or out of the EQ display in this node. It does not influence the output result.

PB Output:
Check this box to make the audio processor node available for the PB EQ effects.

For using one of these FX, please enter the **Site** and **Devic**e ID of the Device whereto the FX is assigned and select in the **FX drop-down** which out of the three available is being used.

Check the box for **Silent** value changes if you do not want to generate active values in Pandoras Box. To automatically avoid the FX Parameter Value Smoothing, check the **Direct** box.

If you want to increase or decrease to the values transmitted to the PB FX, you can do so by entering a respective **Factor**.

Low / Mid / High:
The three available bandpass regions can be further defined in this section

Start / End:
Enter here the start and the end frequency band (out of the 32 available) that should be included in your bandpass region.

Max / Min:
The value and peak output values refer in their size to the maximum and minimum area of their corresponding bandpass. A value/peak of 1 means that the maximum is reached or exceeded.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

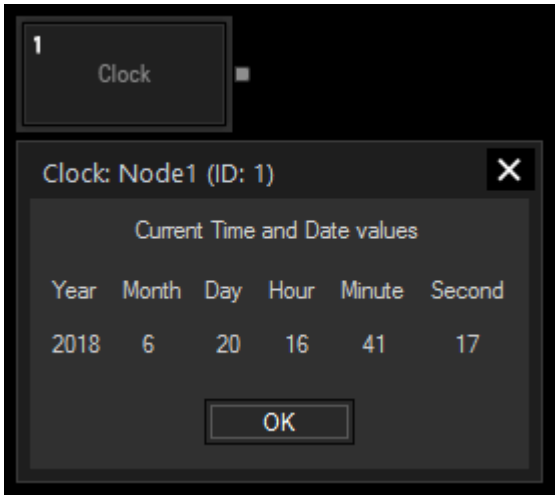
The Node generates the following output:

- Low
- Low Peak
- Mid
- Mid Peak
- High
- High Peak
- Volume
- Volume Peak
- EQ 1 to 32

7.4.4.4.2 Clock Input

The Clock node provides the current Date values: Year, Month, Day, Hour, Minute and Second.

This node can be found under Nodes > Input > Generic > Clock



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

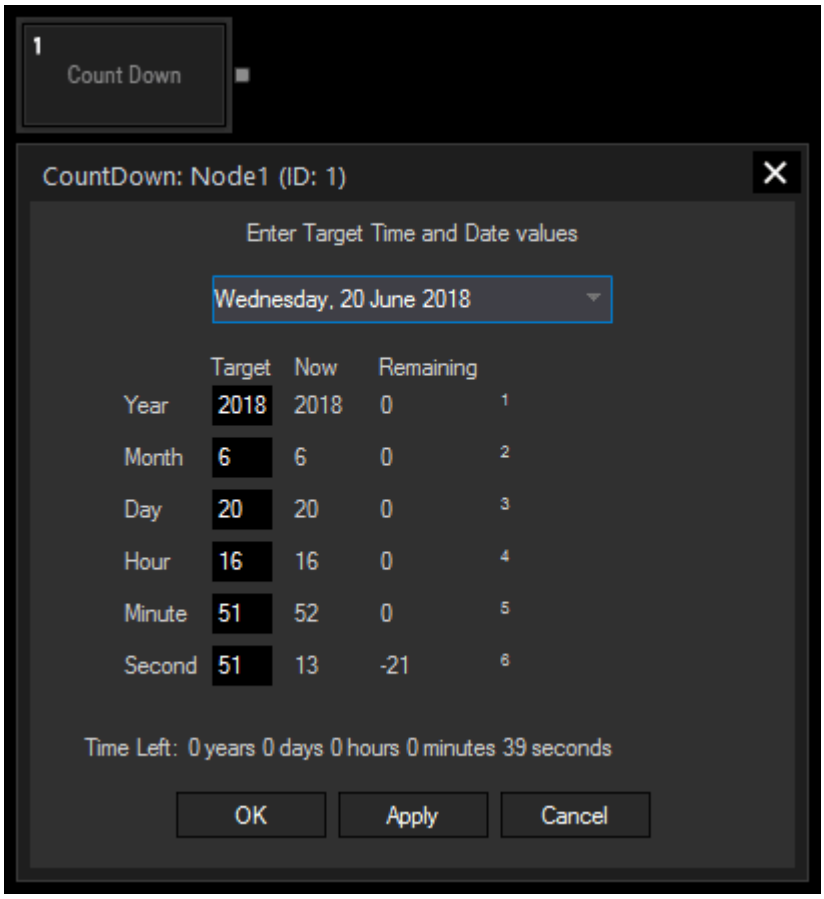
The Node generates the following output:

- Year
- Month
- Day
- Hour
- Minute
- Second

7.4.4.4.3 Count Down Input

The Count Down input node allows you to set a target time to which it counts down.
It will give out both the total amount of remaining years, months, days, hours, minutes and seconds and the Count Down of hours, minutes and seconds.

This node can be found under Nodes > Input > Generic > Count Down



Node Properties

Target Time and Date values:
Please choose the target date from the list or enter it in the text fields below. Enter here also the time target (hours, minutes and seconds).

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

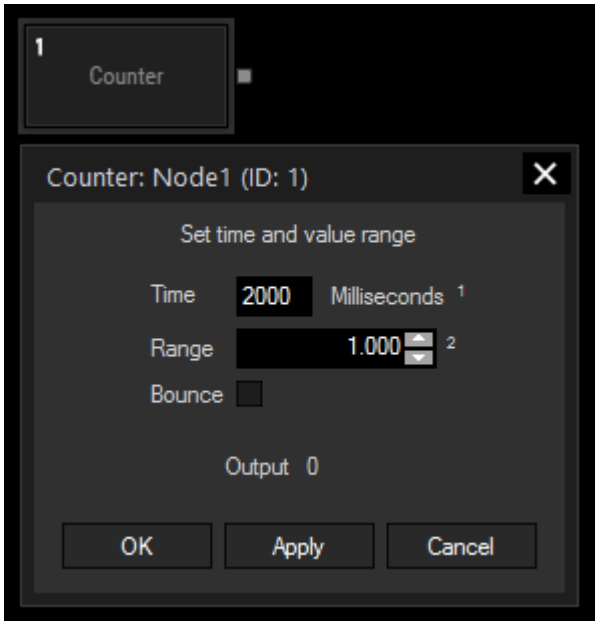
- Years Left
- Months Left
- Days Left
- Hours Left
- Minutes Left
- Seconds Left
- CD Hours

- CD Minutes
- CD Seconds

7.4.4.4.4 Counter Input

The Counter input node can be set up to continuously count a certain value range in a given amount of time. The counter can also be set to bounce up and down.

This node can be found under Nodes > Input > Generic > Counter



Node Properties

Time:
Set the amount of time.

Range:
Set the value range.

For example:
To get a counter that counts continuously in 5 seconds from 0 to 5 enter for Time=5000 ms and for Range=5.
To get a counter that counts continuously in 5 seconds from 0 to 10 enter for Time=5000 ms and for Range=10.
To get a bouncing counter (counting continuously from 0 to X and back to 0 etc.) please tick the **Bounce** check box.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

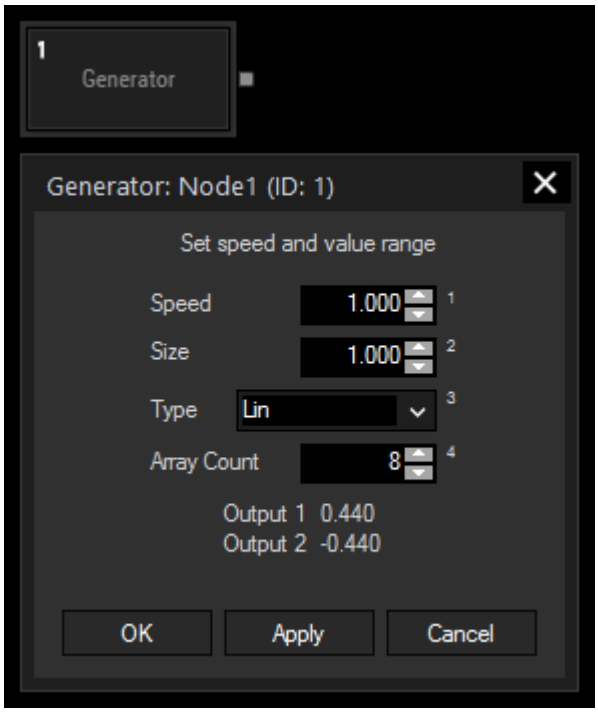
The Node generates the following output:

- Value

7.4.4.4.5 Generator

The Generator input node provides constantly alternating values, different patterns like sine or a ramp function can be selected

This node can be found under Nodes > Input > Generic > Generator



Node properties

Speed:
Enter here the speed that the generated curve should have. A smaller value will result in a longer time one period of the resulting signal would take.

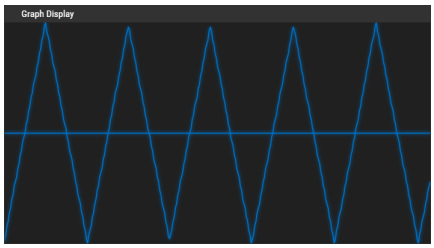
Size:
The Size declares the maximum positive value the generated signal can have. Please note that the types "Sine", "Cosine" and "Circle" always have their maximum at 1, due to being derived from calculating with a unit circle.

Type:
Select in this drop-down what type of curve your output signal is supposed to have. For more information, please refer to the topic Curve Types below this entry

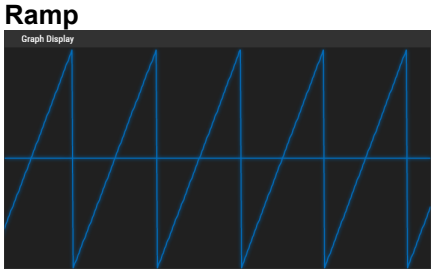
Curve Types

The Generator Node allows setting different curve types. Connected to a GraphDisplay [858], the curve can be made visible:

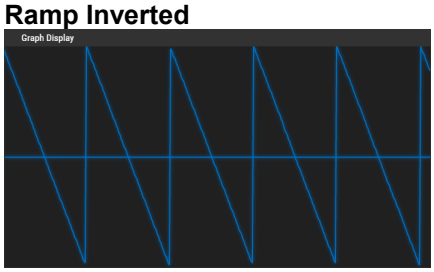
Linear



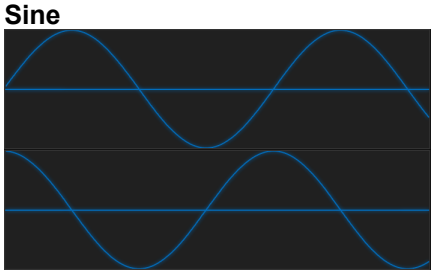
Ramp



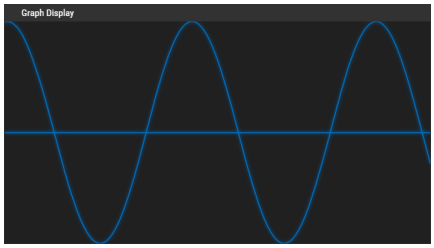
Ramp Inverted



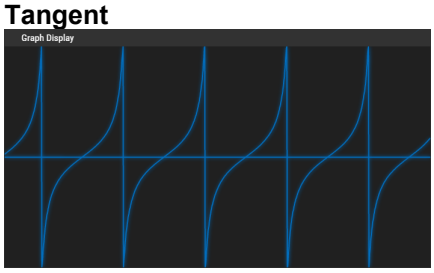
Sine



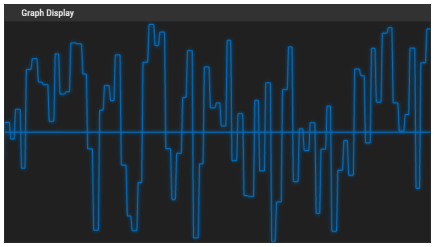
Cosine



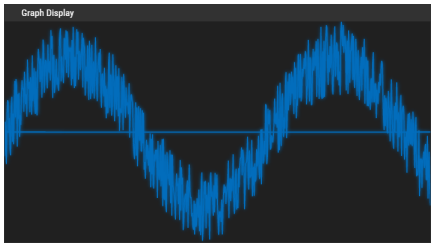
Tangent



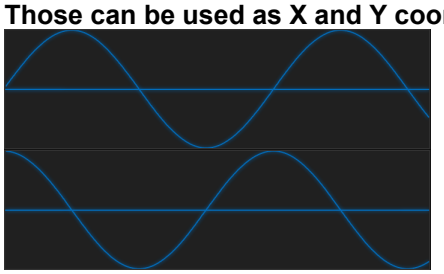
Random



Random Sine / Cosine: generates values with a maximum deviation of +/-0.5 from the original sine / cosine curve



Circle: generates a sine (outputted as Value 1) and the corresponding cosine curve (outputted as Value 2). Those can be used as X and Y coordinates to create a circular motion.



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

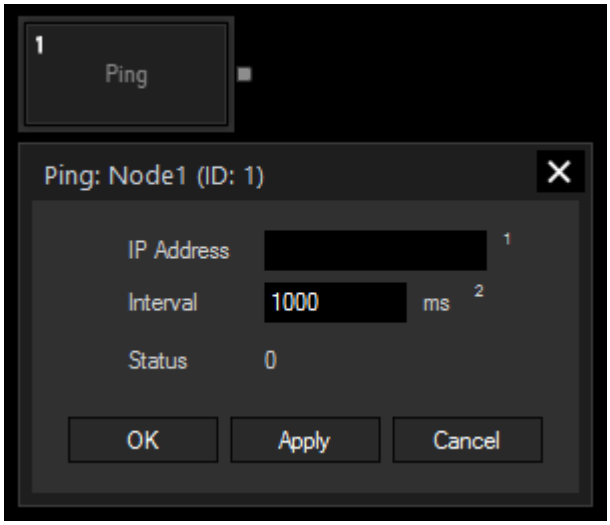
The Node generates the following output:

- Value 1
- Value 2 (which is the inverted Value 1, except for "Circle")

7.4.4.4.6 Ping Input

The Ping input node gives you feedback in a certain time interval if the specified IP is online.

This node can be found under Nodes > Input > Generic > Ping



Node Properties

IP Address:
Enter the IP Address you want to ping.

Interval:
Enter the update interval in milliseconds.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

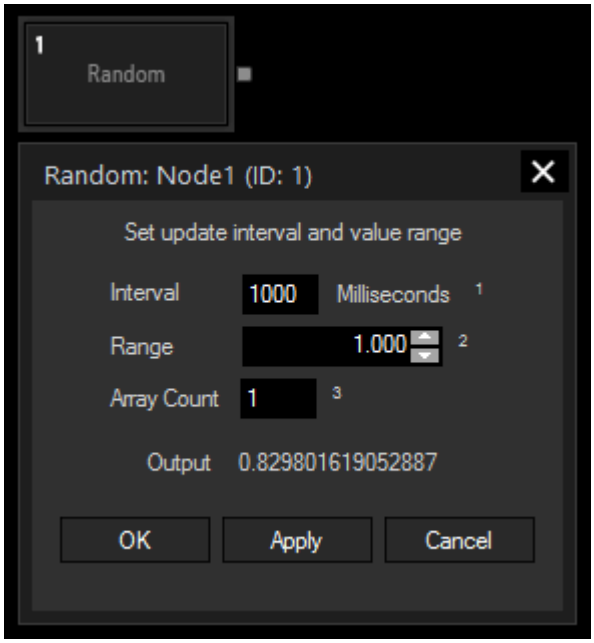
The Node generates the following output:

- Ping status (true or false)

7.4.4.4.7 Random Input

The Random input node provides a random number on a given update interval and value range.

This node can be found under Nodes > Input > Generic > Random



Node Properties

Interval:
Enter the update interval in milliseconds.

Range:
Enter the value range.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

In addition to the general commands, this nodes provides a specific function command, `Node1.Refresh`, that refreshes the output and generates a new random number before the actual interval is finished.

Node output values

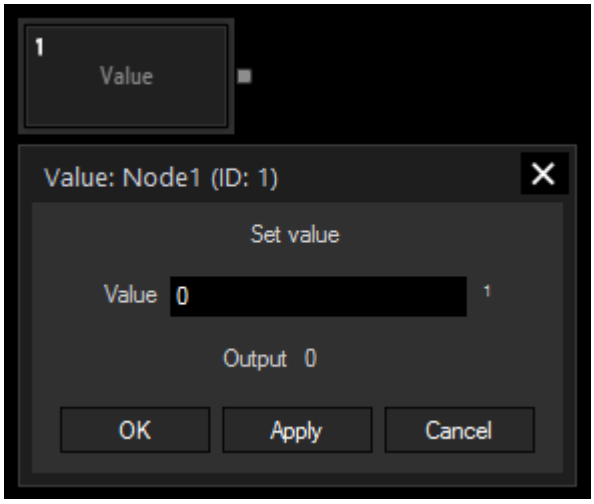
The Node generates the following output:

- Random Value
- Random Values()

7.4.4.4.8 Value Input

The Value input node allows you to set a numeric value for further processing.

This node can be found under Nodes > Input > Generic > Value



Node Properties

Value:
Enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

The value in this node can also be changed using a Value output node [1208], as well as with the command `Node1.SetParam(1,new value)`.

Node output values

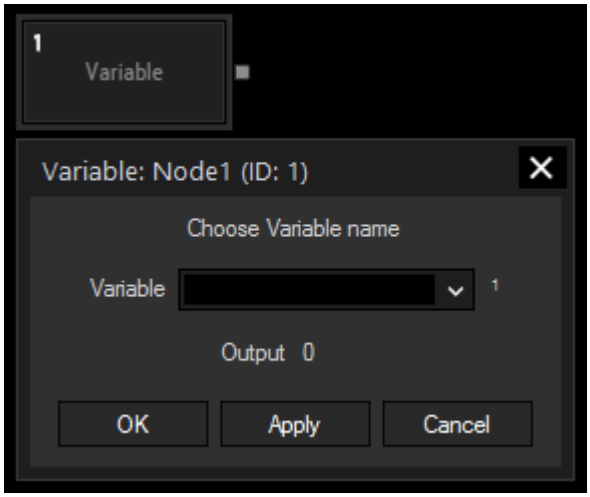
The Node generates the following output:

- Value

7.4.4.4.9 Variabel Input

The Variable input node allows you to use a variable [1900] for further processing.

This node can be found under Nodes > Input > Generic > Variable



Node Properties

Variable:
Select a global variable from the drop-down.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

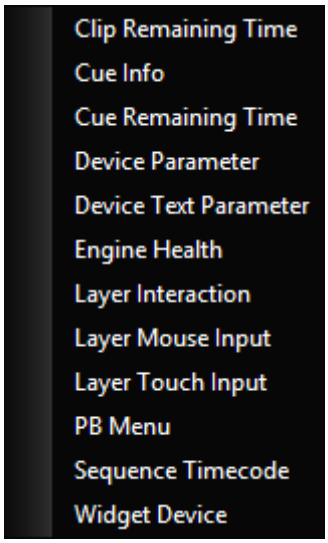
- Variable Value

To set up and process variables, please use these scripts [1620].
To see all existing variables and their values, see the Variable List [1900].

7.4.4.5 Pandoras Box Input

The Pandoras Box input nodes allow you to retrieve live information on nearly all elements of the connected Pandoras Box project. This includes Sequence and specific Device information, as well as system information or even mouse data of all machines connected to the network.

For some nodes, you need a valid Pandoras Box connection [1256] to the Master application, for others WD communicates with the installed PB Menu [2097] on any machine in the network.

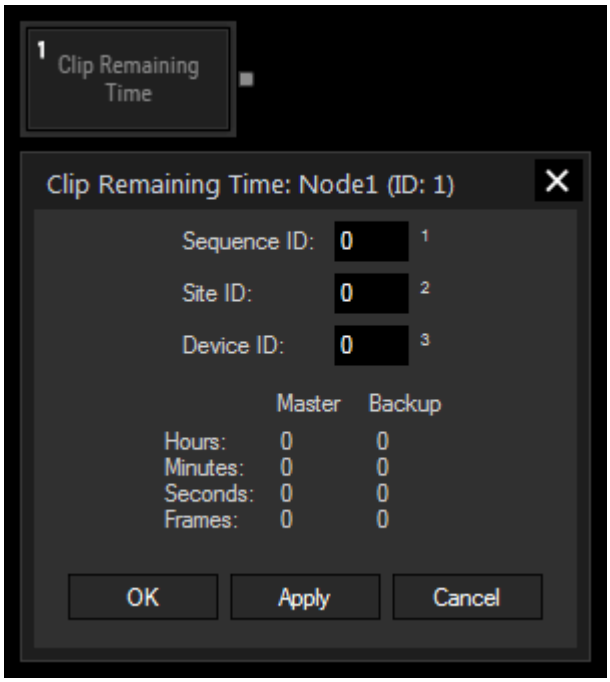


Clip Remaining Time [1052]
Cue Info [1053]
Cue Remaining Time [1054]
Device Parameter [1055]
Device Text Parameter [1056]
Engine Health [1057]
Layer Interaction [1058]
Layer Mouse Input [1059]
Layer Touch Input [1060]
PB Menu [1062]
Sequence Timecode [1065]
Widget Device [1066]

7.4.4.5.1 Clip Remaining Time

The Clip Remaining Time node provides the remaining time of a clip in Pandoras Box. The nowpointer must be inside the clip container, the remaining time is calculated by the nowpointer's position relating to the end of the container.

The node can be found under Nodes > Input Nodes > Pandoras Box > Clip Remaining Time



Node properties

Sequence ID:
Enter the Sequence ID.

Site ID:
Enter the Site ID.

Device ID:
Enter the Device ID.

The PB Master and Backup Connection can be set up in the PB Network Configuration [1256].

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

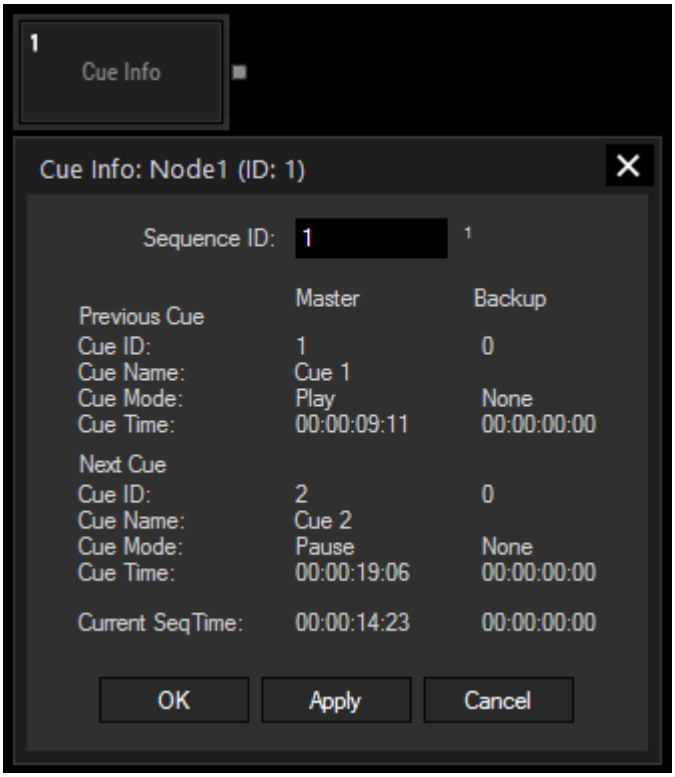
- Hour
- Minute
- Second
- Frame
- Backup Hour
- Backup Minute
- Backup Second

- Backup Frame
- TC Master
- TC Backup

7.4.4.5.2 Cue Info

The Cue Info input node provides information regarding the previous and next Cue in a Pandoras Box Sequence. Further it shows the current time.

The node can be found under Nodes > Input Nodes > Pandoras Box > Cue Info



Node properties

Sequence ID:
Enter the Sequence ID.

The PB Master and Backup Connection can be set up in the PB Network Configuration 1256.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

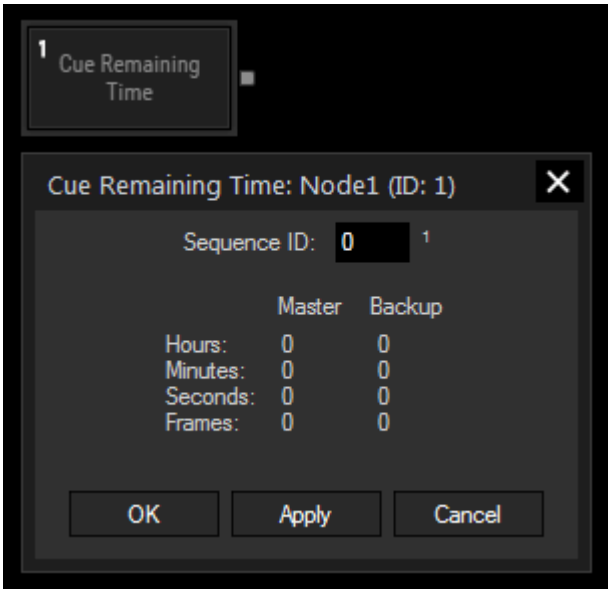
- PrevCueName
- PrevCueID
- PrevCueMode
- PrevCueTime
- NextCueName
- NextCueID
- NextCueMode
- NextCueTime

- PrevCueName_B (for Backup connection)
- PrevCueID_B
- PrevCueMode_B
- PrevCueTime_B
- NextCueName_B
- NextCueID_B
- NextCueMode_B
- NextCueTime_B
- CurrentSeqTime
- CurSeqTime_B

7.4.4.5.3 Cue Remaining Time

The Cue Remaining Time input node provides the remaining time until the next Cue in a Pandoras Box Sequence is reached. The time is calculated by the nowpointer's position relating to the next Cue.

The node can be found under Nodes > Input Nodes > Pandoras Box > Cue Remaining Time



Node properties

Sequence ID:
Enter the Sequence ID.

The PB Master and Backup Connection can be set up in the PB Network Configuration 1256.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

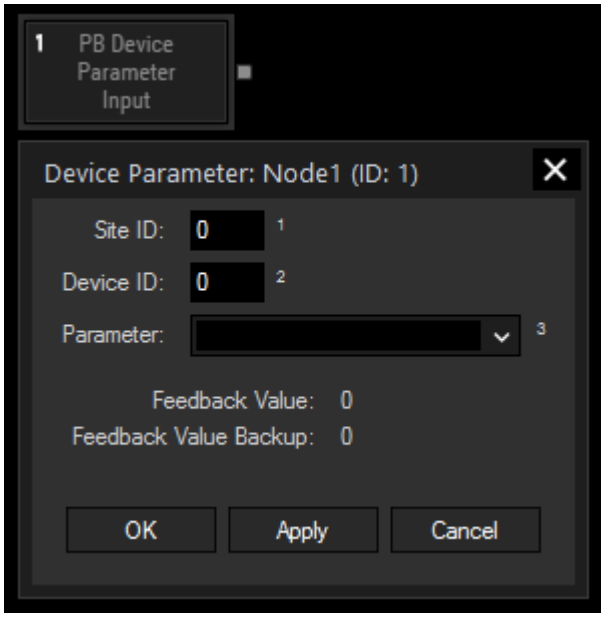
- Hour
- Minute
- Second
- Frame
- Backup Hour
- Backup Minute
- Backup Second

- Backup Frame
- TC Master
- TC Backup

7.4.4.5.4 PB Device Parameter Input

The PB Device Parameter input node provides feedback values for all layer parameters from a Pandoras Box Master and Backup System for further processing.

The node can be found under Nodes > Input Nodes > Pandoras Box > Device Parameter



Node Properties

Site ID:
Enter the Site ID from which you want to get feedback values.

Device ID:
Enter the Device ID from which you want to get feedback values.

Parameter:
Choose the parameter you want to monitor from the drop-down.

Example:
To route the current value of the parameter "X Position" of layer 3 of Site 1 to another node, enter "1" as SiteID, "3" as DeviceID and choose "X Position" from the drop-down list.

The PB Master and Backup Connection can be set up in the PB Network Configuration 1256.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

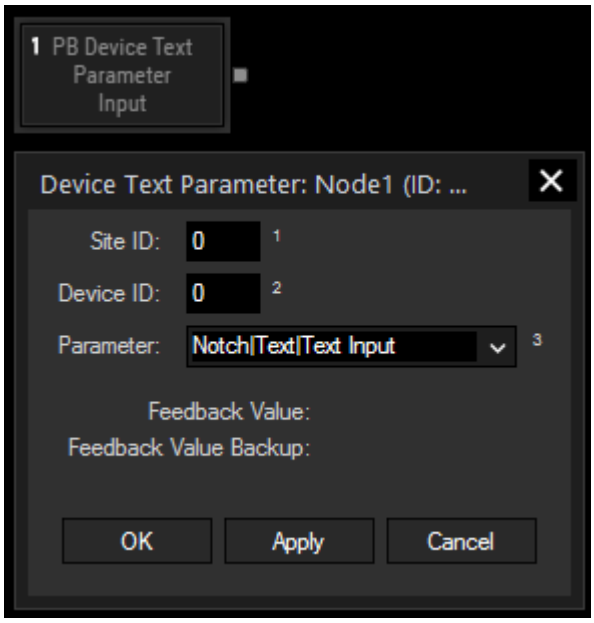
The Node generates the following output:

- Value
- Backup Value

7.4.4.5.5 PB Device Text Parameter Input (2)

The PB Device Text Parameter input node provides feedback values for text layer parameters from a Pandoras Box Master and Backup System for further processing.

The node can be found under Nodes > Input Nodes > Pandoras Box > Device Text Parameter



Node Properties

Site ID:
Enter the Site ID from which you want to get feedback values.

Device ID:
Enter the Device ID from which you want to get feedback values.

Parameter:
Choose the text parameter you want to monitor from the drop-down. Currently, only Notch Layers [671] provide text values.

Example:
To route the current value of the parameter "Name" (in the "Notch" group under "Displays") of layer 3 of Site 1 to another node, enter "1" as SiteID, "3" as DeviceID and choose "Notch|Displays|Name" from the drop-down list or enter it as a value.

The PB Master and Backup Connection can be set up in the PB Network Configuration [1256].

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

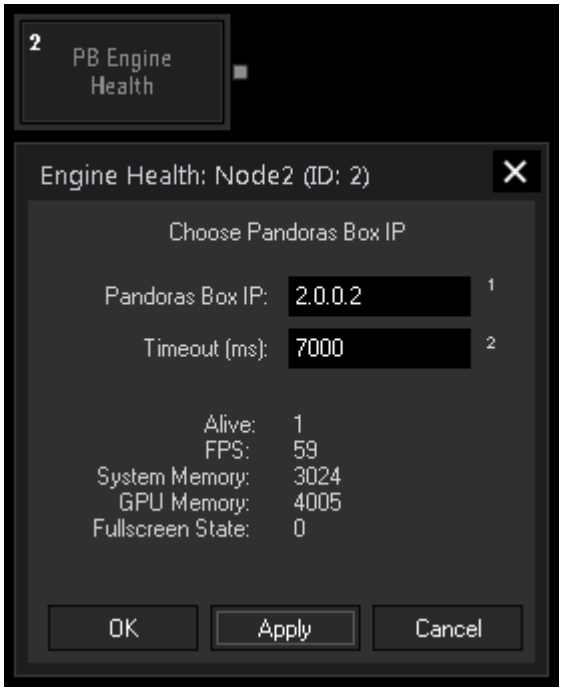
- Value
- Backup Value

7.4.4.5.6 PB Engine Health Input

The Engine Health information is sent by each Master or Client Pandoras Box application and includes information to ensure its running well.

For your information: the commands `WDStartPbHealthListener`[1823] and `WDStopPbHealthListener`[1827] to start and stop the internally used UDP port 8992 that is used to receive the Multicast data send to 239.0.0.11.

The node can be found under Nodes > Input Nodes > Pandoras Box > Engine Health



Node properties

IP:
Enter the IP address from your Pandoras Box Master or Client machine and click Connect.
Note that the PB Health connection is always linked to the PB Master Connection adapter as set in the PB Network Configuration dialog[1256].

Timeout (ms):
The states changes if the node receives no response by Pandoras Box in the given timeout. The timeout starts at 5000ms and cannot be shortened further.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

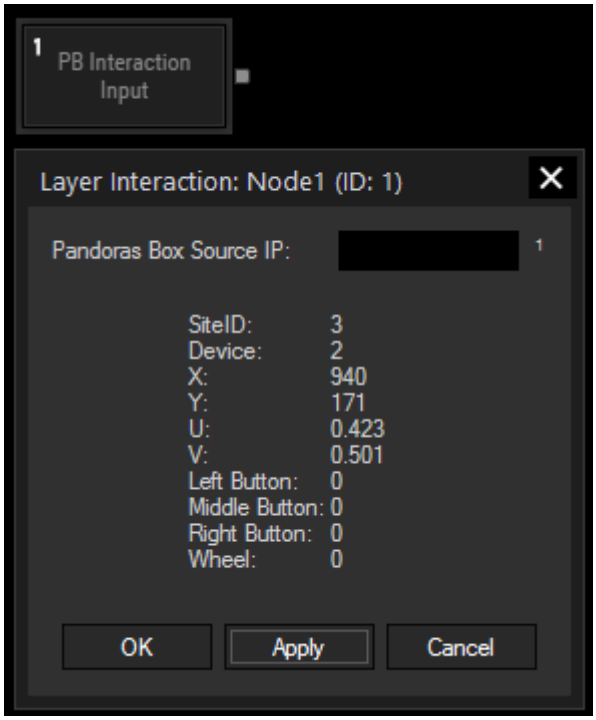
The Node generates the following output:

- Alive: 0 = false, 1 = true
- FPS (frames per second)
- System Memory (depending on hardware the C: drive, C:\Christie\content or C:\coolux\content)
- GPU Memory
- Fullscreen State: 0 = false, 1 = true

7.4.4.5.7 PB Layer Interaction

The PB Interaction Input node provides mouse and touch point feedback from a Pandoras Box system with activated "Input Events". If you are interested in filtering the feedback data based on the Site or Device ID or if you are working with multi-touch inputs, please use the Layer Mouse input [1059] or Layer Touch input [1060] instead.

The node can be found under Nodes > Input Nodes > Pandoras Box > Layer Interaction



The chapter "Picking Mode" [252] explains what to set up on Pandoras Box side but also in Widget Designer.

In short:
- PB Master includes "Widget Designer Device [696]" in Device Tree [173]
- PB Master or Client > Input Event Settings [210] > "Output" and "Widget Designer Device" activated
- PB Master in Picking Mode or Client connected to mouse or touch device
- WD > PB Network Configuration [1256] > connected to Master and "Enable Connections" to Widget Designer Device => "Input Tester"shows data

If you are interested in sending mouse or touch data to a Pandoras Box (drawing) Canvas, please see the chapter "Layer UV Draw to Canvas" [1245].

Node Properties

Source IP:
Optionally, enter the IP from the Pandoras Box Master, otherwise the Master connection from the PB Network Configuration [1256] works.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.connected

Node output values

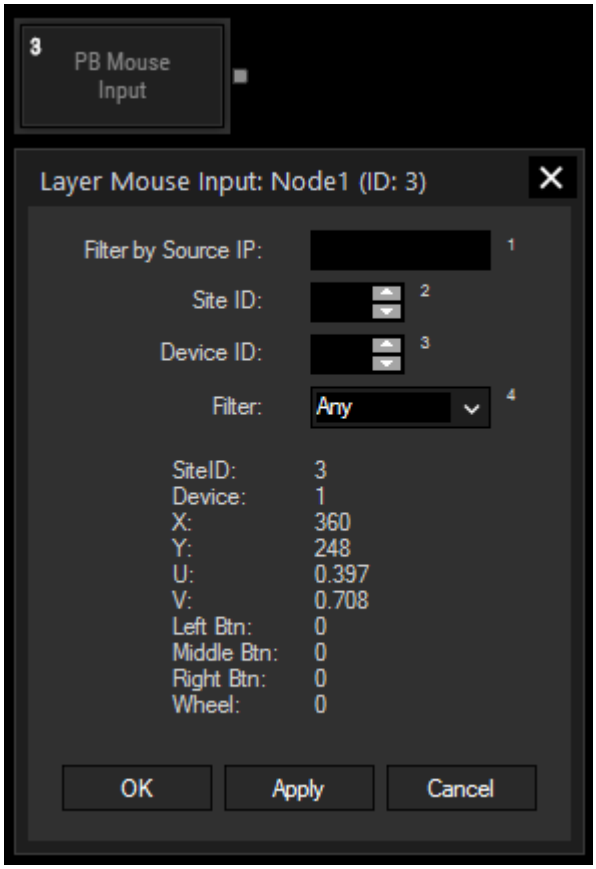
The Node generates the following output: (For an explanation regarding this data, please scroll down in the chapter "Picking Mode" [252] and see the options in the dialog PB Network Configuration [1256].

- Site
- Device
- ScreenX
- ScreenY
- TextureU
- TextureV
- Mouse Left
- Mouse Middle
- Mouse Right
- Mouse Wheel

7.4.4.5.8 PB Layer Mouse Input

The Layer Mouse Input node provides mouse and single touch point feedback from a Pandoras Box system with activated "Input Events". In difference to the Layer Interaction [1058] node, this node can filter data based on the Site or Device ID whilst the Layer Touch [1060] input works with multi-touch input.

The node can be found under Nodes > Input Nodes > Pandoras Box > Layer Mouse Input



The chapter "Picking Mode" [252] explains what to set up on Pandoras Box side but also in Widget Designer.

In short:

- PB Master includes "Widget Designer Device [696]" in Device Tree [173]
- PB Master or Client > Input Event Settings [210] > "Output" and "Widget Designer Device" activated
- PB Master in Picking Mode or Client connected to mouse or touch device
- WD > PB Network Configuration [1256] > connected to Master and "Enable Connections" to Widget Designer Device => "Input Tester"shows data

If you are interested in sending mouse or touch data to a Pandoras Box (drawing) Canvas, please see the chapter "Layer UV Draw to Canvas" [1245].

Node Properties

Source IP:
Optionally, enter the IP from the Pandoras Box Master, otherwise the Master connection from the PB Network Configuration [1256] works.

Site ID:
Enter a Site ID from Pandoras Box, see "Filter". and optionally also a Device ID from Pandoras Box and change below option "Filter" if you like to filter the incoming data to receive only data depending on the Site and optionally also the Device.

Device ID:
Enter a Device ID from Pandoras Box, see "Filter".

Filter:
Set the Filter option to "Site" in case you like to process only the input data sent by the Site ID entered above.
Set the Filter to "Site Device" if you like to filter incoming data even more, that is to receive only feedback from the Site and Device ID entered above.

If the Filter is set to "Any" you will see unfiltered data from all Sites and Devices.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.connected

Node output values

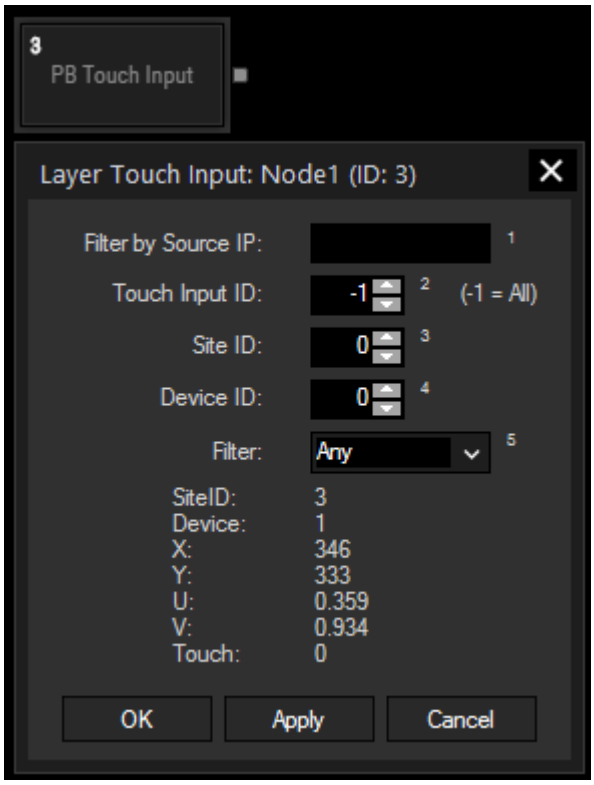
The Node generates the following output: (For an explanation regarding this data, please scroll down in the chapter "Picking Mode" [252] and see the options in the dialog PB Network Configuration [1256].

- Site
- Device
- ScreenX
- ScreenY
- TextureU
- TextureV
- LeftBtn
- MiddleBtn
- RightBtn
- Wheel

7.4.4.5.9 PB Layer Touch Input

The Layer Touch Input node provides single- and multi-touch point feedback from a Pandoras Box system with activated "Input Events". In difference to the Layer Interaction [1058] node, this node can filter data based on the Touch Input, Site or Device ID whilst the Layer Mouse input [1059] works with single-touch input and mouse inputs.

The node can be found under Nodes > Input Nodes > Pandoras Box > Layer Touch Input



The chapter "Picking Mode" [252] explains what to set up on Pandoras Box side but also in Widget Designer.

In short:
- PB Master includes "Widget Designer Device [696]" in Device Tree [173]
- PB Master or Client > Input Event Settings [210] > "Output" and "Widget Designer Device" activated
- PB Master in Picking Mode or Client connected to touch device
- WD > PB Network Configuration [1256] > connected to Master and "Enable Connections" to Widget Designer Device => "Input Tester"shows data

If you are interested in sending touch data to a Pandoras Box (drawing) Canvas, please see the chapter "Layer UV Draw to Canvas" [1245].

Node Properties

Source IP:
Optionally, enter the IP from the Pandoras Box Master, otherwise the Master connection from the PB Network Configuration [1256] works.

Touch Input ID:
Enter an Input ID if you like to see data from this input only. Enter "-1" if all inputs should be processed. See the data protocol in the Input Tester dialog if your touch device sends a touch input ID.

Site ID:
Enter a Site ID from Pandoras Box, see "Filter".

Device ID:
Enter a Device ID from Pandoras Box, see "Filter".

Filter:
Set the Filter option to "Site" in case you like to process only the input data sent by the Site ID entered above.
Set the Filter to "Site Device" if you like to filter incoming data even more, that is to receive only feedback from the Site and Device ID entered above.

If the Filter is set to "Any" you will see unfiltered data from all Sites and Devices.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.connected

Node output values

The Node generates the following output: (For an explanation regarding this data, please scroll down in the chapter "Picking Mode" [252] and see the options in the dialog PB Network Configuration [1256].

- Site
- Device
- ScreenX
- ScreenY
- TextureU
- TextureV
- Touched

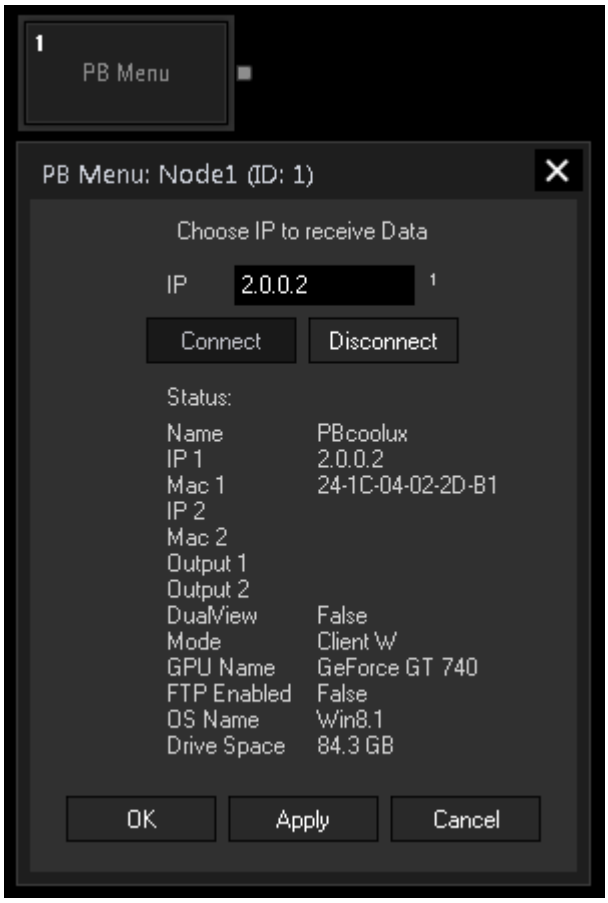
7.4.4.5.10 PB Menu Input

The PB Menu Input node provides technical information about your Master or Client machine where the PB Menu [2097] is already installed. Please make sure to use WD version 6.5 (and above) with PB Menu version 7.

After adding the node to your WD project, please say "Yes" when you get the following message: "Do you wish to enable PB Menu status checks?" This activates the PB Menu option "Check Status" in the PB Network Configuration dialog [1256] which opens the UDP port 4999 that is used internally to receive the Broadcast data. However, as PB Menu [2097] and the Server Management Application [2109] (formerly PB Remote) also listen to that port, it can happen that your input node receives data even though "Check Status" is inactive. Note that the information in the node updates depending on the "Interval" set in the PB Configuration dialog.

Please note: For using the Projector Calibration Manager [1496] in your project, you also need the PB Menu Node.

The node can be found under Nodes > Input Nodes > Pandoras Box > PB Menu



Node properties

IP:
Enter the IP address from your Pandoras Box Master or Client machine and click Connect.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

In addition to the general commands, this nodes provides specific function commands that can replace pressing the buttons in the Configuration manually. E.g. `Node1.EnterFullScreen` sets the PB application running on the connected PC to full screen.

The following additional function commands are available for the PB Menu node:

- Calibrate
- Connect
- Disconnect
- EnterFullScreen
- LeaveFullScreen

Node output values

The Node generates the following output:

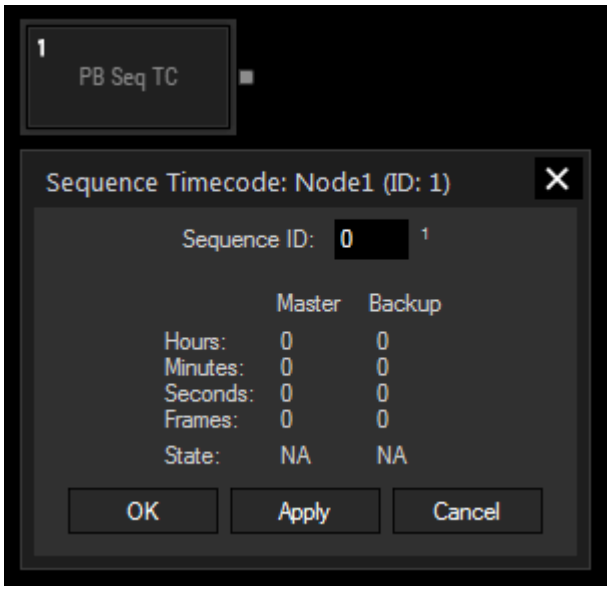
- Name of the PB machine
- IP-address of Lan1
- MAC address 1
- IP-address of Lan2

- MAC address 2
- Output 1 = resolution and FPS of output 1 (the refresh rate is only available for the graphics cards NVIDIA Quadro M4000, P1000 or P4000)
- Output 2 = resolution and FPS of output 2 (the refresh rate is only available for the graphics cards NVIDIA Quadro M4000, P1000 or P4000)
- Dual View = True/ False
- Mode: Master / Client W/F = Master or Client in windowed or fullscreen mode
- GPU Name = Graphics Card
- OS Name = Operating system
- Drive Space
- PB Menu Revision

7.4.4.5.11 PB Sequence Timecode

The Sequence Timecode node provides the Sequence timecode information of the PB Master and Backup machine, as well as the current state of the Sequence.

The node can be found under Nodes > Input Nodes > Pandoras Box > Sequence Timecode



Node properties

Sequence ID:
Enter the Sequence ID.

The PB Master and Backup Connection can be set up in the PB Network Configuration 1256.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

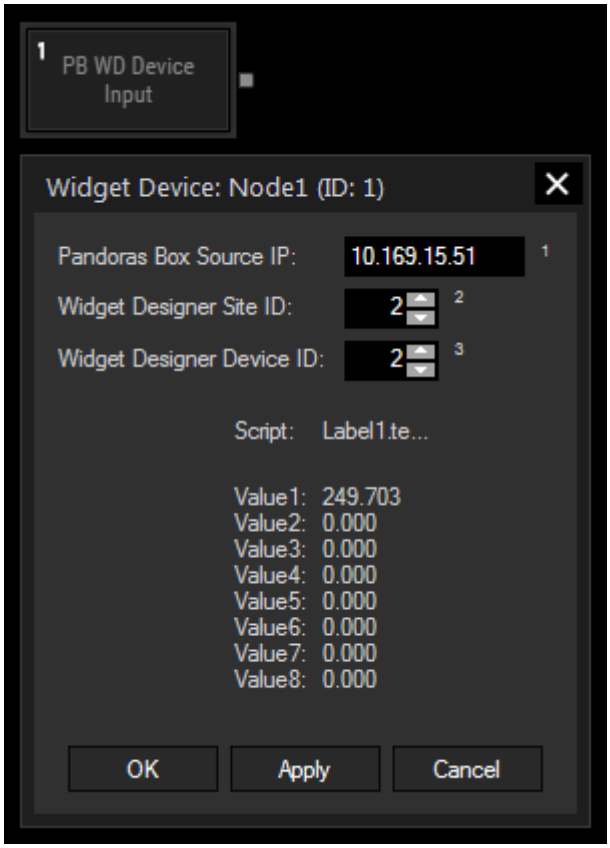
The Node generates the following output, each for PB Master and Backup separately:

- Hour
- Minute
- Second
- Frame
- Timecode
- State: Play / Pause / Stop / NA (Not available)

7.4.4.5.12 PB Widget Device

The Widget Designer Device [696] enables you to control WD from your Pandoras Box Sequence, more specifically from the nowpointer's position. Commands can be sent for direct execution and up to eight numeric values per node can be sent to WD for further processing. While the simple Widget Device executes the entered commands without additional programming, the Widget Device input node can receive those commands as simple string message and thus can be used for more sophisticated scripting options.

The node can be found under Nodes > Input Nodes > Pandoras Box > Widget Device



Node properties

Pandoras Box Source IP:
Enter the IP address of your Pandoras Box machine.

Widget Designer Site ID:
Enter the Site ID of the Widget Designer Device in your PB Project.

Widget Designer Device ID:
Enter the Device ID of the Values Device you want to read out.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

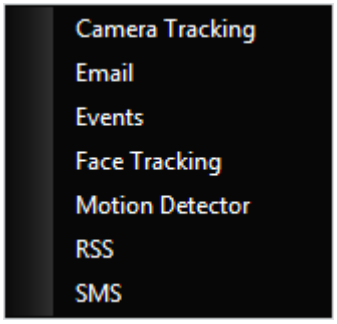
The Node generates the following output:

- Script: Returns the complete script from the Widget Device that is currently executed as a String message.

- Value 1-8: Returns the values 1-8 which are defined in the Values Device of the Widget Designer Device in your PB project.

7.4.4.6 Tools Input

The Tools input nodes provide information from various Tools [1483] like RSS, Email or Events. The video input related Devices [1277] like Face Tracker and Motion Detector.

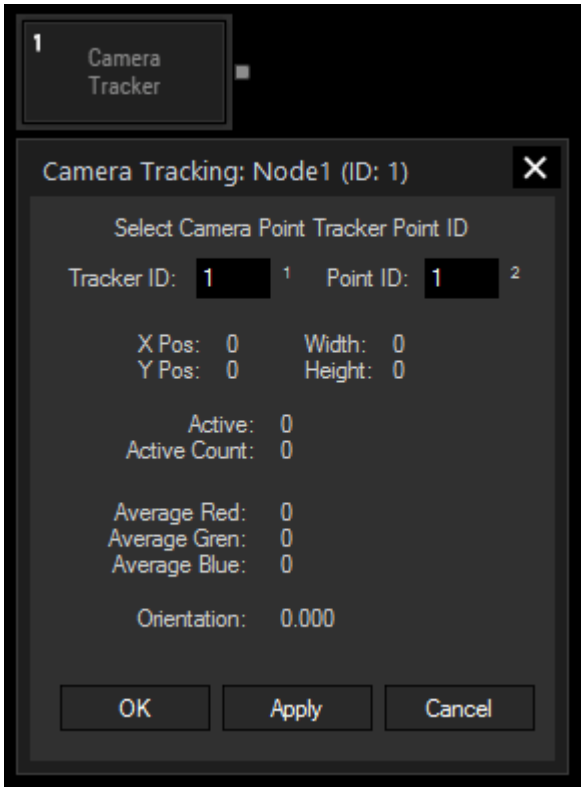


Camera Tracking [1067]
Email [1068]
Events [1069]
Face Tracking [1070]
Motion Detecor [1071]
RSS [1072]
SMS [1073]

7.4.4.6.1 Camera Tracking Input

The Camera Tracker Interface [1291] and the Camera Tracking input node allows you to track up to 99 tracking points.

The node can be found under Nodes > Input Nodes > Tools > Camera Tracking



Node Properties

Tracker ID:
As you can set up up to eight Camera Trackers, enter here the ID of the tracker to be processed.

Point ID:
Select the ID of the point that is tracked via the Camera Tracker.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

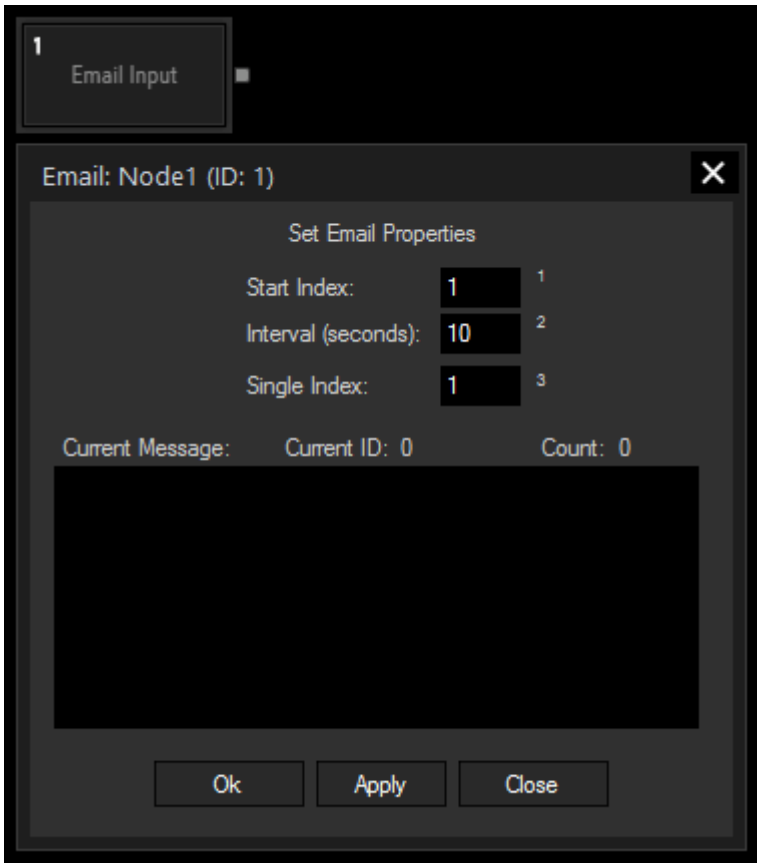
- X Pos
- Y Pos
- Width (of the bounding box)
- Height (of the bounding box)
- Active (0 or 1)
- Active Point Count (amount of currently active points)
- Average Red
- Average Green
- Average Blue
- Orientation

Please note:
Each tracked point you want to get the data from for further processing needs its own Camera Tracker input node.

7.4.4.6.2 Email Input

The Email input node provides all incoming Emails for further processing.
Please configure your mail server in the Email Settings 1490.

The node can be found under Nodes > Input Nodes > Tools > Email



Node Properties

Start Index:
Enter the Index of the email where the cycling should start. This will effect the outputs Current Email and Current ID.

Interval (seconds):
All incoming emails will be cycling in the entered amount of time. This will effect the outputs Current Email and Current Email ID.

Single Index:
Enter the Index of the email that you want to route to the output Single Email.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

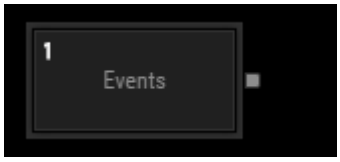
The Node generates the following output:

- Email Count
- Last Email
- Current Email
- Current Email ID
- Single Email
- Random Email

7.4.4.6.3 Events Input

The Events Input Node provides information about the last events, current events and next events.
Setup your events in the Event Editor[1483].

The node can be found under Nodes > Input Nodes > Tools > Events



Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output: (Information about the Current Event):

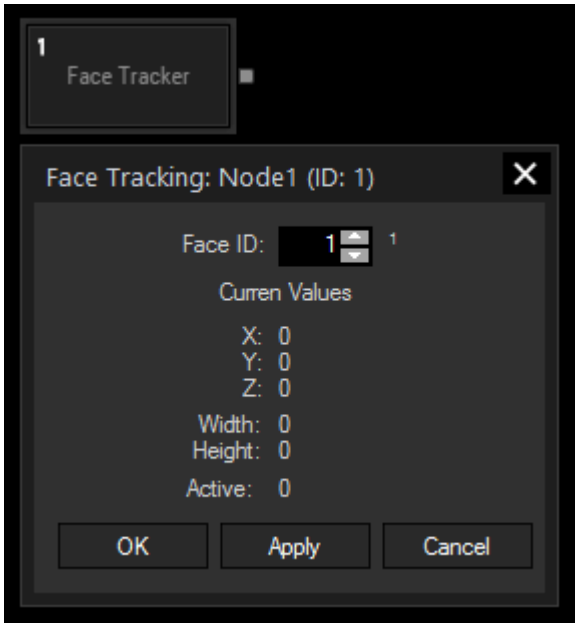
- Event (Event name)
- Year
- Month
- Day
- Hour
- Second

The same information is provided for the last and the next event.

7.4.4.6.4 Face Tracker Input

The Face Tracker input node allows proceeding the position values delivered by the Face Tracker Tool [1287].

The node can be found under Nodes > Input Nodes > Tools > Face Tracker



Node Properties

Face ID:
Choose here the ID of the face to be tracked.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

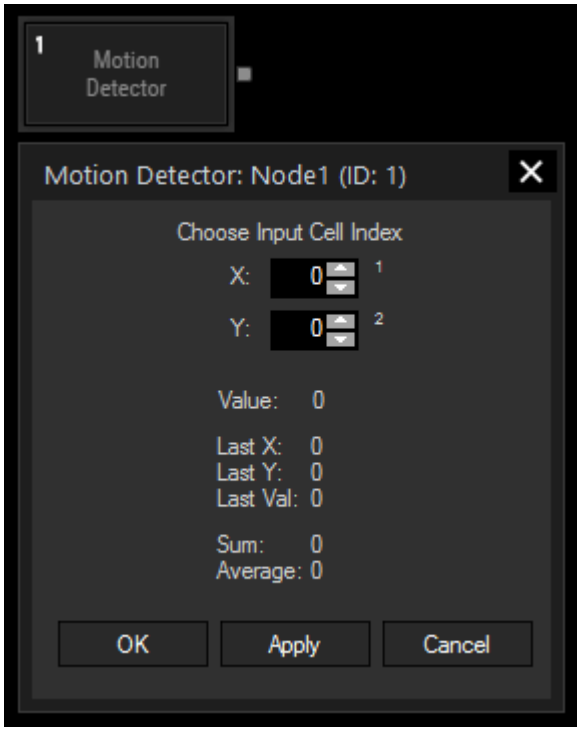
The Node generates the following output:

- X
- Y
- Z
- Width (of the bounding box)
- Height (of the bounding box)
- Active

7.4.4.6.5 Motion Detector Input

The Motion Detection input node provides the data given by the Motion Detector Tool [1289] for further processing.

The node can be found under Nodes > Input Nodes > Tools > Motion Detector



Node Properties

X/Y:
Choose here the Input Cell Index from the Motion Detector Tool whose data you want to use by entering the column number (X) and the row number (Y).

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

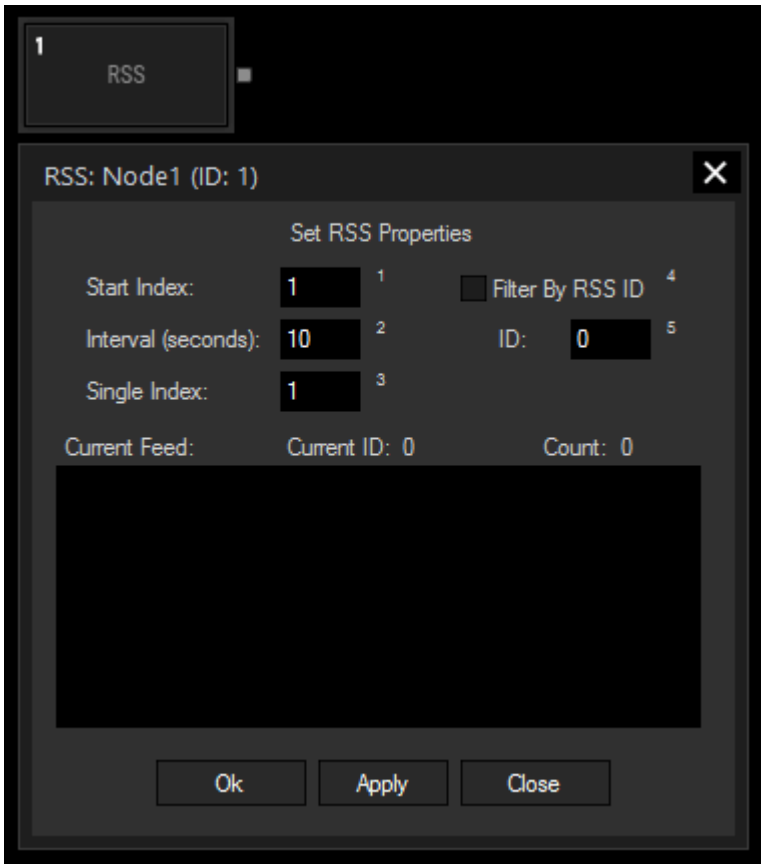
The Node generates the following output:

- Value (the value of the chosen cell)
- Last X (X index of the cell in which the last motion was detected)
- Last Y (Y index of the cell in which the last motion was detected)
- Last Value
- Sum (Sum of all cell values)
- Average (Average of all cell values).

7.4.4.6.6 RSS Input

The RSS input node provides all incoming RSS feeds for further processing.
Please setup the RSS connection in the RSS Settings [1493] before.

The node can be found under Nodes > Input Nodes > Tools > RSS



Node Properties

Start Index:
Enter the Index of the RSS Feed where the cycling should start. This will affect the outputs Current RSS and Current ID.

Interval (seconds):
All incoming RSS Feeds will be cycling in the entered amount of time. This will affect the outputs Current RSS and Current ID.

Single Index:
Enter the Index of the RSS Feed that you want to route to the output Single RSS.

Filter By RSS ID:
As you can receive several RSS feeds at once, you can limit the messages displayed by this node to one single Feed. Check this box and enter the respective RSS **ID**.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

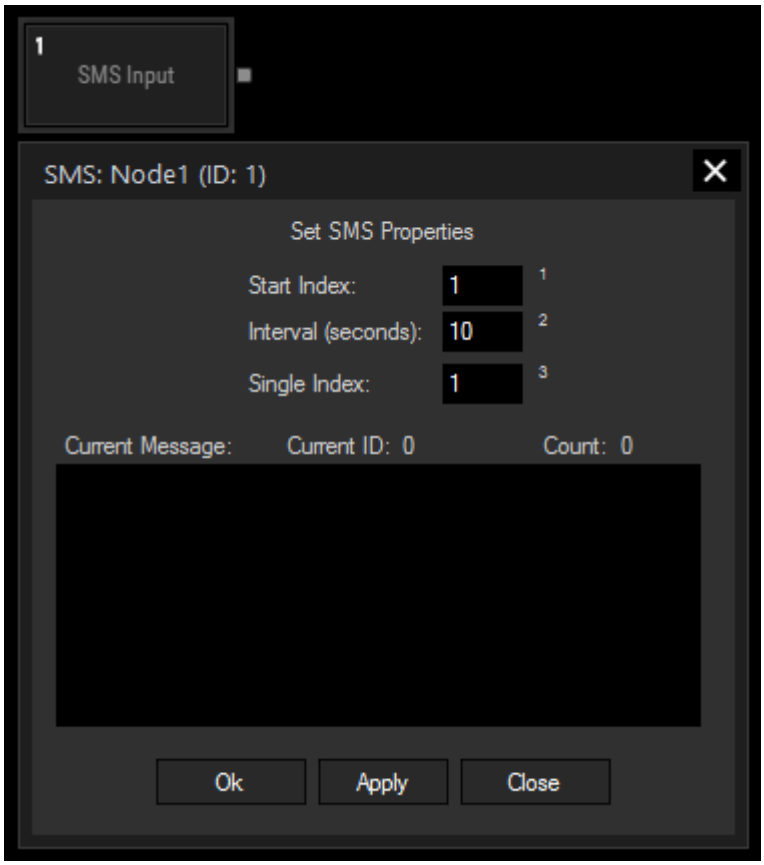
The Node generates the following output:

- RSS Count
- Last RSS
- Current RSS
- Current ID
- Single RSS
- RND RSS
- Last Title
- Last Text
- Current Title
- Current Text
- Single Title
- Single Text
- RND Title
- RND Text

7.4.4.6.7 SMS Input

The SMS input node provides all incoming SMS for further processing.
Please setup the required GSM modem in the SMS Settings [1494] before.

The node can be found under Nodes > Input Nodes > Tools > SMS



Node Properties

Start Index:
Enter the Index of the start SMS, where the cycling should begin. This will effect the outputs Current SMS and Current ID.

Interval (seconds):
All incoming SMS will be cycling in the entered amount of time. This will effect the outputs Current SMS and Current ID.

Single Index:
Enter the Index of the SMS that you want to route to the output Single SMS.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

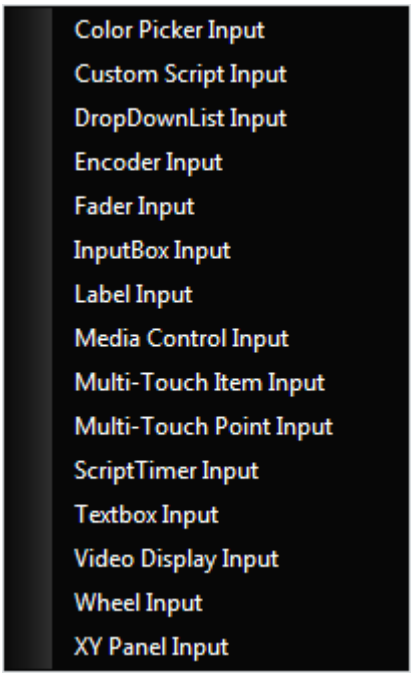
Node output values

The Node generates the following output:

- SMS Count
- Last SMS
- Current SMS
- Current ID
- Single SMS
- RND SMS

7.4.4.7 Widgets Input

Some widgets provide values that can be read out constantly with the Widgets input nodes. They range from a simple Fader [874] value over the ScriptTimer [826] remaining time but can also provide detailed information on items from the Multitouch Panel [899].

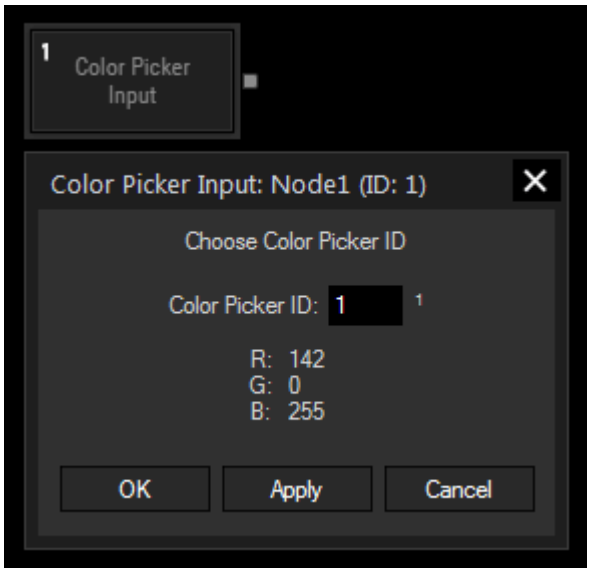


ColorPicker Input [1076]
CustomScript Input [1077]
DropDown List Input [1078]
Encoder Input [1079]
Fader Input [1080]
InputBox Input [1081]
Label Input [1082]
MediaControl Input [1083]
Multi-Touch Item Input [1084]
Muti-Touch Point Input [1086]
ScriptTimer Input [1087]
TextBox Input [1088]
Video Display Input [1089]
Wheel Input [1091]
XY Panel Input [1092]

7.4.4.7.1 Color Picker Input

The Color Picker Input Node provides the RGB and CMY values of the currently selected color of the Color Picker widget.

The node can be found under Nodes > Input Nodes > Widgets > Color Picker Input



Node properties

Color Picker ID:
Enter the ID of the ColorPicker you want to access.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

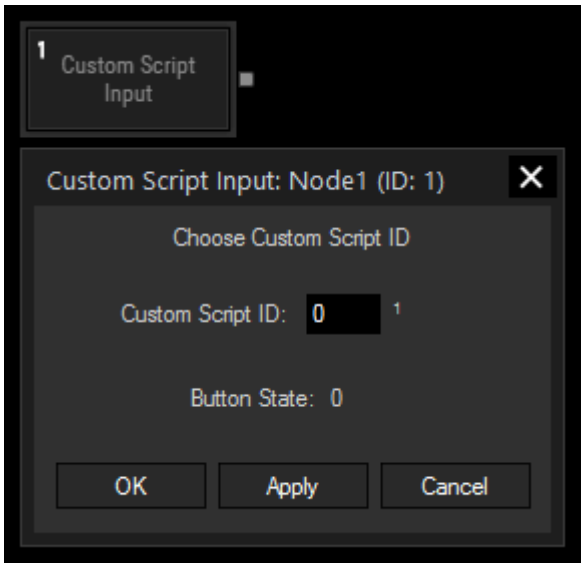
The Node generates the following output:
- Red
- Green
- Blue
- Cyan
- Magenta
- Yellow

7.4.4.7.2 Custom Script Input

The Custom Script Input node allows you to intercept Custom Script Button [822] presses within Widget Designer. Select the CS Button ID to output the current click state. The output value is either 0 or 1.
Please note that the output stays at 1 as long as the button is held down. If you like to count for example how many times a button was used, you need the information when it was clicked, not how long it was pressed. The clicked time can be achieved with the help of other filtering nodes, e.g. Delta [1104] and If [1106] node.

The node can be found under Nodes > Input Nodes > Widgets > Custom Script Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Custom Script Input



Node Properties

Custom Script ID:
Enter here the Custom Script Button ID this node should refer to.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

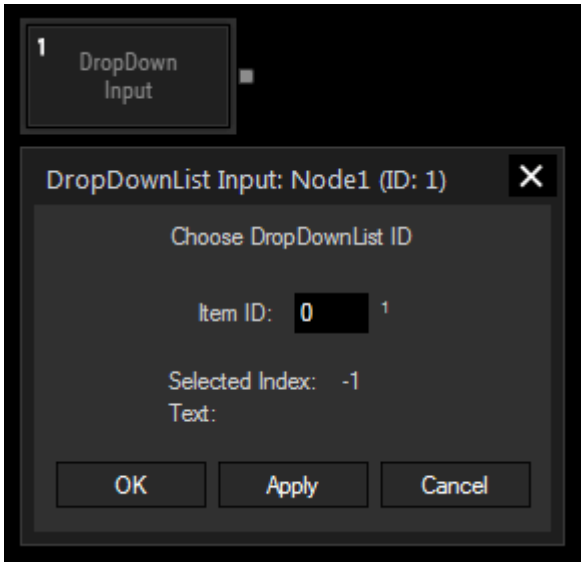
The node generates the following output:

- Value (True=1, False=0)

7.4.4.7.3 DropDownList Input

The DropDownList input node allows intercepting values from a Widget Designer Drop Down List [868] control. Select the Drop Down List ID to output the current text value and the index number of the chosen entry.

The node can be found under Nodes > Input Nodes > Widgets > Drop Down List



Node properties

Item ID:
Choose the Drop Down List ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

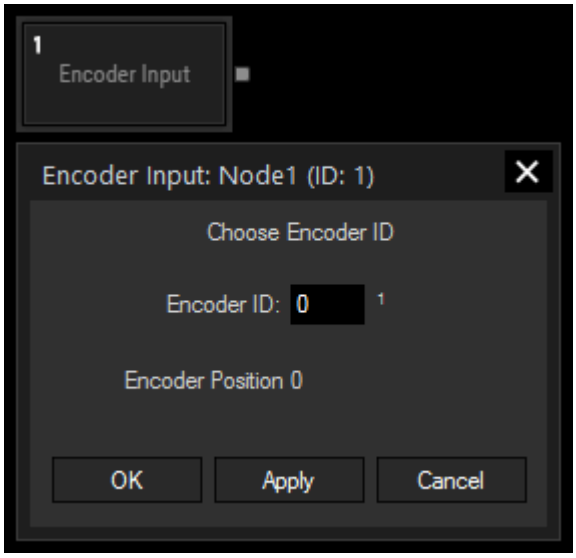
- Selected Index (Index of the selected item: Title = -1, first index = 0 and so on)
- Text (Text of the selected item)

7.4.4.7.4 Encoder Input

The Encoder input node allows intercepting Encoder [877] values within Widget Designer. Enter the Encoder ID to output the current Encoder value.

The node can be found under Nodes > Input Nodes > Widgets > Encoder Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Encoder Input



Node Properties

Encoder ID:
Enter the Encoder ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

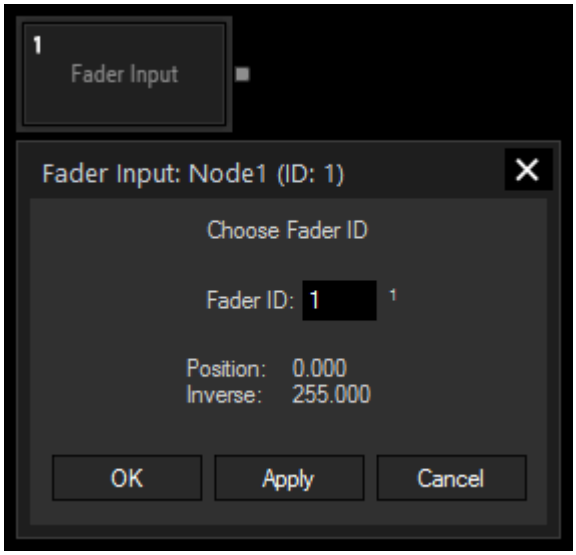
- Encoder Input Value

7.4.4.7.5 Fader Input

The Fader input node allows intercepting Fader [874] Values within Widget Designer. Select the Fader ID to output the current fader value.

The node can be found under Nodes > Input Nodes > Widgets > Fader Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Fader Input



Node Properties

Fader ID:
Enter the Fader ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

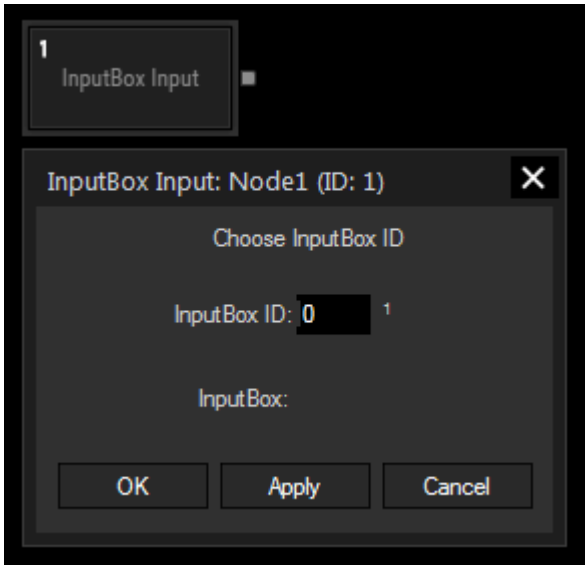
The node generates the following output:

- Value
- Inverse Value

7.4.4.7.6 InputBox Input

The InputBox input node enables you to constantly retrieve the text entered in the selected InputBox [886] widget.

The node can be found under Nodes > Input Nodes > Widgets > InputBox Input



Node properties

InputBox ID:
Enter the InputBox ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

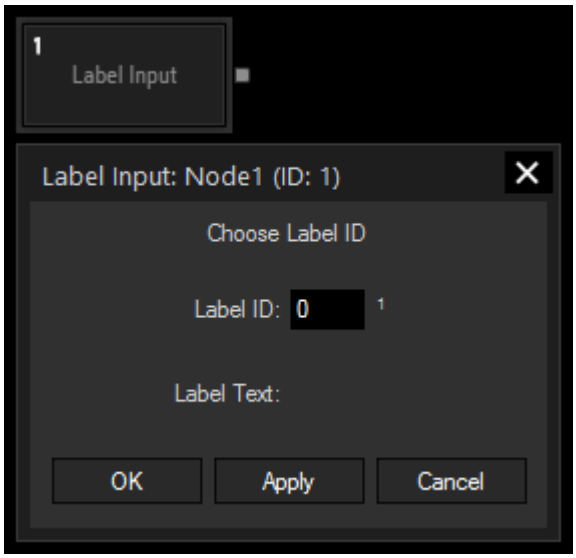
- Value (as String)

7.4.4.7.7 Label Input

The Label input node allows using Label [888] texts for further processing.

The node can be found under Nodes > Input Nodes > Widgets > Label Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Label Input



Node Properties

ID:
Enter the Label ID in the text field.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

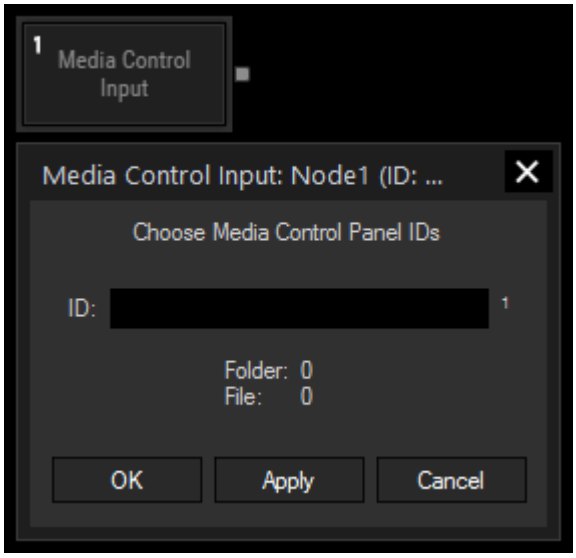
- Value (as String)

7.4.4.7.8 Media Control Input

The Media Control input node allows to intercept the last called Folder and File ID of one or multiple Media Control Panels [828].

The node can be found under Nodes > Input Nodes > Widgets > Media Control Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Media Control Input



Node Properties

ID:
Enter the Media Control Panel ID in the text field. If you want to retrieve information from several Media Control Panels, you can enter multiple IDs, separated with a white space.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

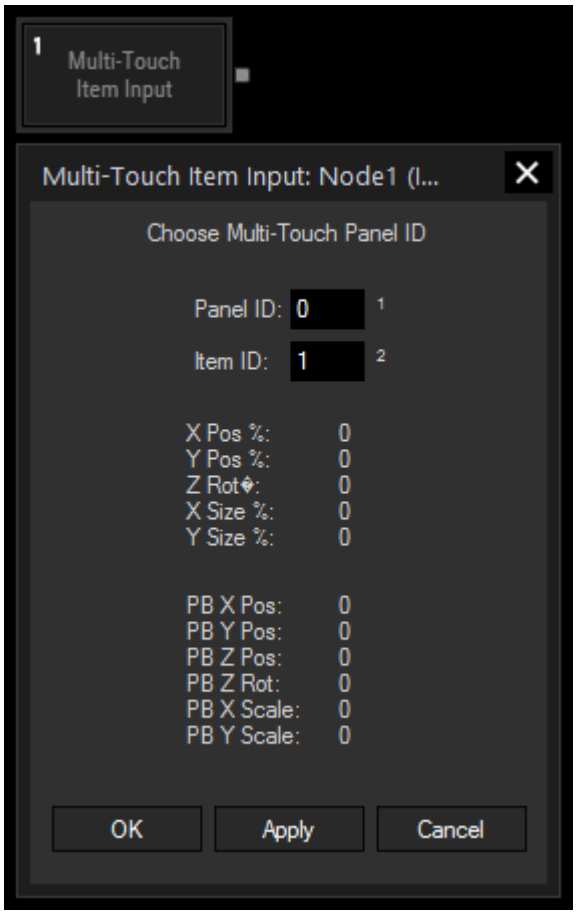
- Folder ID and
- File ID of the last called File

7.4.4.7.9 Multi-Touch Item Input

The Multi-Touch Item input node provides the current state of a Multi-Touch Item (Position, Rotation, Size) as well as the current state of the Layer in PB linked to this Item. See the Multitouch Tool [899] for further information and setup.

The node can be found under Nodes > Input Nodes > Widgets > Multi-Touch Item Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Multi-Touch Item Input



Node Properties

Panel ID:
Enter the ID of the Multi-Touch Panel.

Item ID:
Enter the ID of the Multi-Touch Item you want to use.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

- X Pos %
- Y Pos %
- Z Rot °

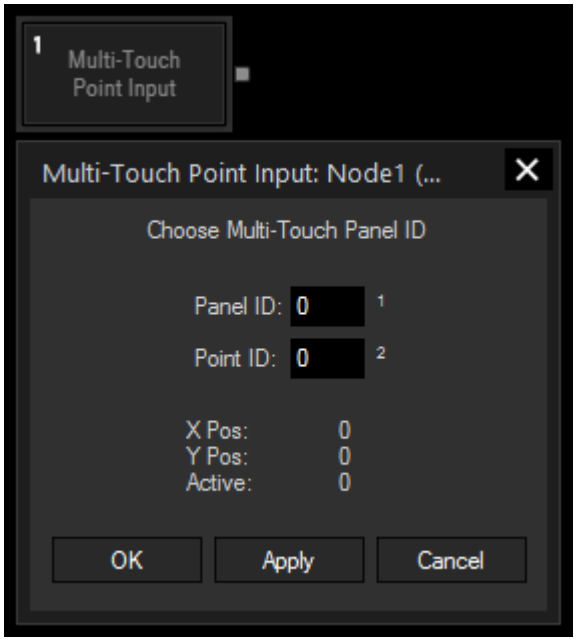
- X Size %
- Y Size %
- X Size %
- X Pos PB
- Y Pos PB
- Z Rot PB
- X Size PB
- Y Size PB
- X Size PB

7.4.4.7.10 Multi-Touch Point Input

The Multi-Touch Point input node provides the current Position of a touch point inside a Multi-Touch Panel. See the Multitouch Tool [899] for further information and setup.

The node can be found under Nodes > Input Nodes > Widgets > Multi-Touch Point Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Multi-Touch Point Input



Node Properties

Panel ID:
Enter the ID of the Multi-Touch Panel.

Point ID:
Enter the ID of the Multi-Touch Point you want to use.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

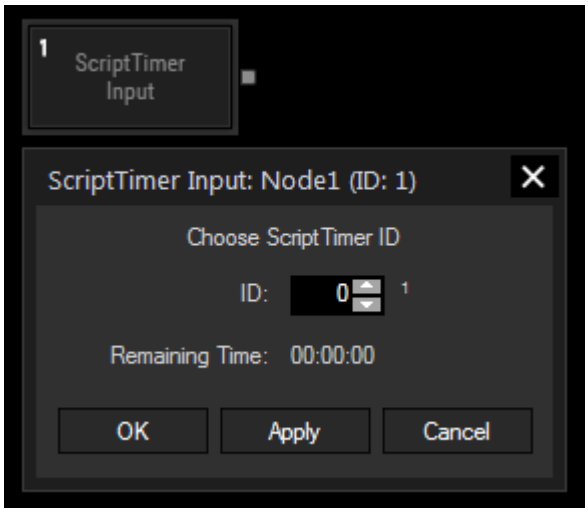
The node generates the following output:

- X Pos
- Y Pos
- Active

7.4.4.7.11 ScriptTimer Input

The ScriptTimer input node allows you to retrieve the remaining time information until the next interval of the ScriptTimer [826] starts.

The node can be found under Nodes > Input Nodes > Widgets > ScriptTimer Input



Node properties

ID:
Enter the ScriptTimer ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output while the time is running out:

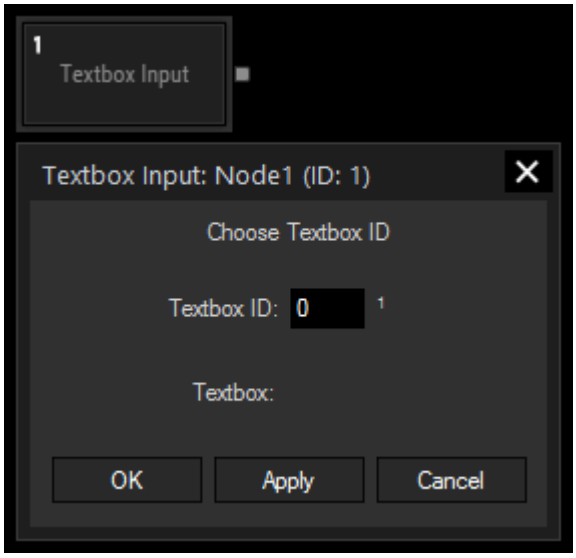
- Hours
- Minutes
- Seconds
- Time Text (complete remaining time as String)

7.4.4.7.12 Textbox Input

The Textbox input node allows intercepting the text inside a TextBox [923] within Widget Designer. Select the Textbox ID to output the current Textbox text.

The node can be found under Nodes > Input Nodes > Widgets > Textbox Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Textbox Input



Node Properties

Textbox ID:
Enter the TextBox ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

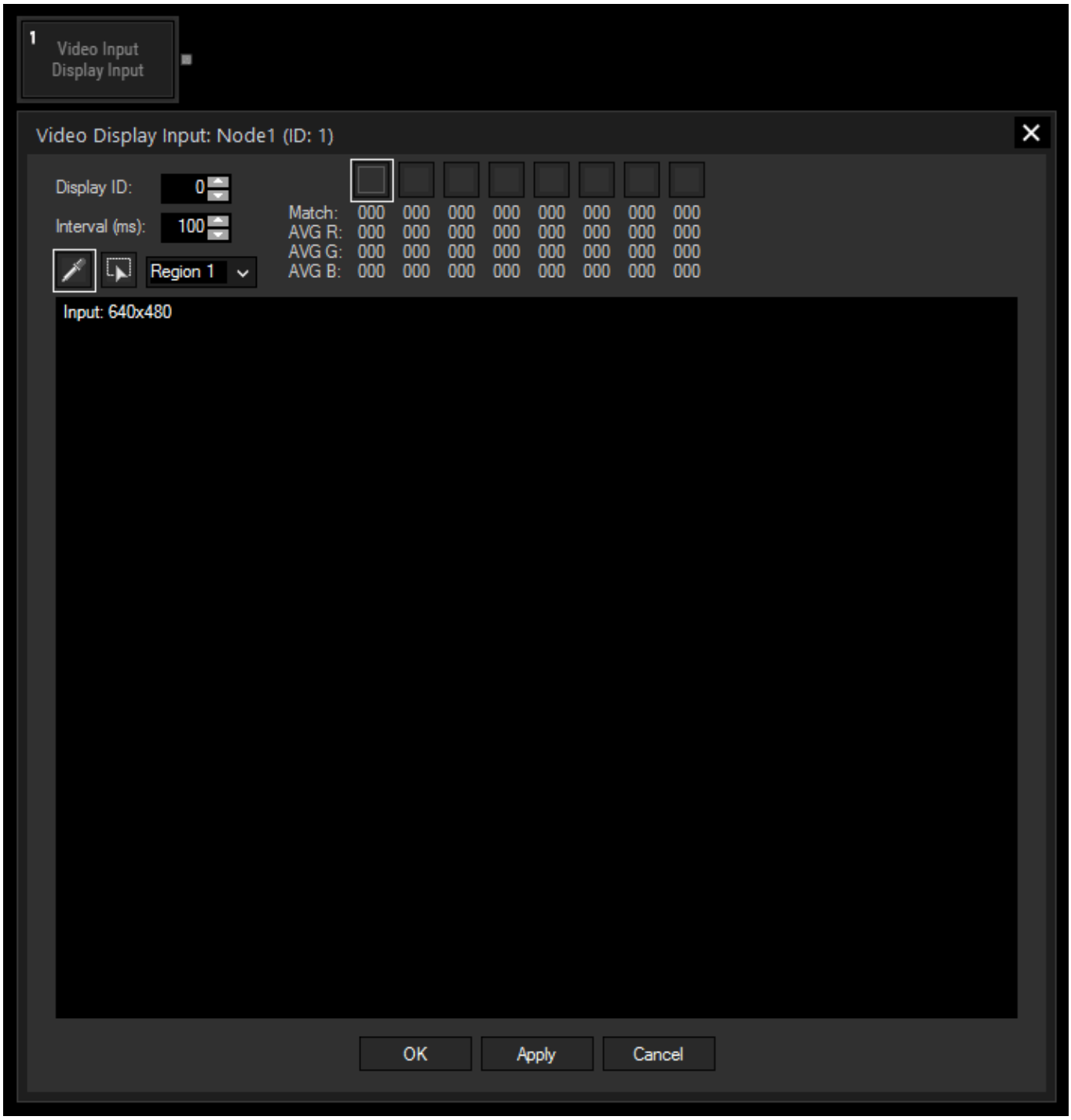
- Value (as String)

7.4.4.7.13 Video Display Input

The Video Input Display input node enables you to analyze the incoming video stream of a Video Input Display[865] widget. Different regions can be set up to retrieve the average R, G and B values of a specific area. Additionally, a certain color can be set up to be detected within the region.

The node can be found under Nodes > Input Nodes > Widgets > Video Display Input

Or by right-clicking on the respective widget itself and selecting Create related Node teams> Video Display Input



Node Properties

Display ID:
Enter the VideoInputDisplay ID.

Interval (ms):
Enter here the refresh interval for the incoming video stream.

Defining a region:
Select one out of the eight available regions from the drop-down.
When you click the **Draw Region** button left of the drop-down, you can draw a rectangle into the video feed corresponding to the selected region. The new region is automatically applied.

Selecting a color:
The eight **color select buttons** on top of the dialog correspond to the eight regions. Press on the respective color button to open a color picker and select a color.
Alternatively, select a region from the drop-down and press the **Pipette** button to extract a color directly from the video feed.

The currently selected region is indicated by a white frame around the color select buttons at the top of the dialog.

A number indicating the **Match** of the selected color within its region, as well as the regions' **average R, G and B** values are displayed below the color select buttons.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.

Node output values

The node generates the following output separately for each region:

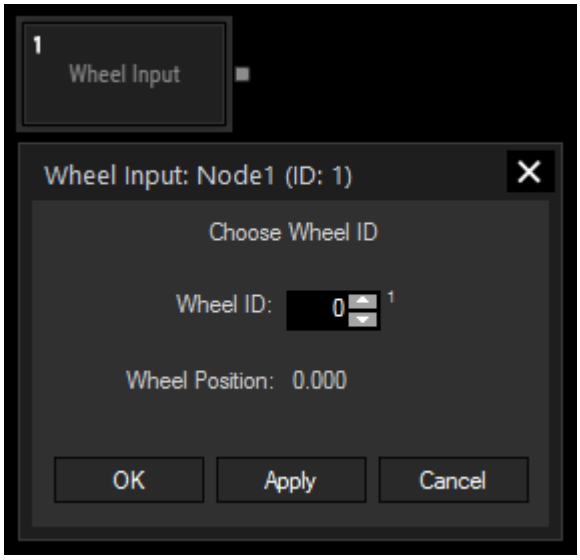
- AVG Red
- AVG Green
- AVG Blue
- Match

7.4.4.7.14 Wheel Input

The Wheel input node allows intercepting Wheel [883] values within Widget Designer. Select the Wheel ID to output the current Wheel value.

The node can be found under Nodes > Input Nodes > Widgets > Wheel Input

Or by right-clicking on the respective widget itself and selecting Create related Node > Wheel Input



Node Properties

Wheel ID:
Enter the Wheel ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The node generates the following output:

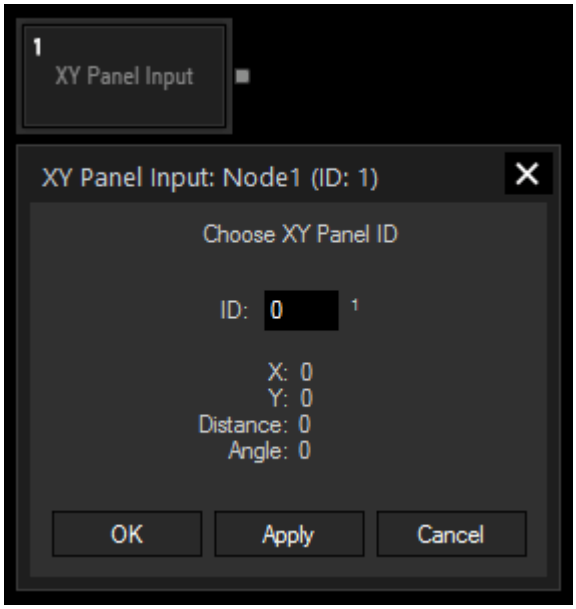
- Value

7.4.4.7.15 XY Panel Input

The XY Panel input node allows intercepting XY Panel [896] values within Widget Designer. Select the XY Panel ID to output the current XY Panel values.

The node can be found under Nodes > Input Nodes > Widgets > XY Panel Input

Or by right-clicking on the respective widget itself and selecting Create related Node > XY Panel Input



Node Properties

ID:
Choose the XY Panel ID.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

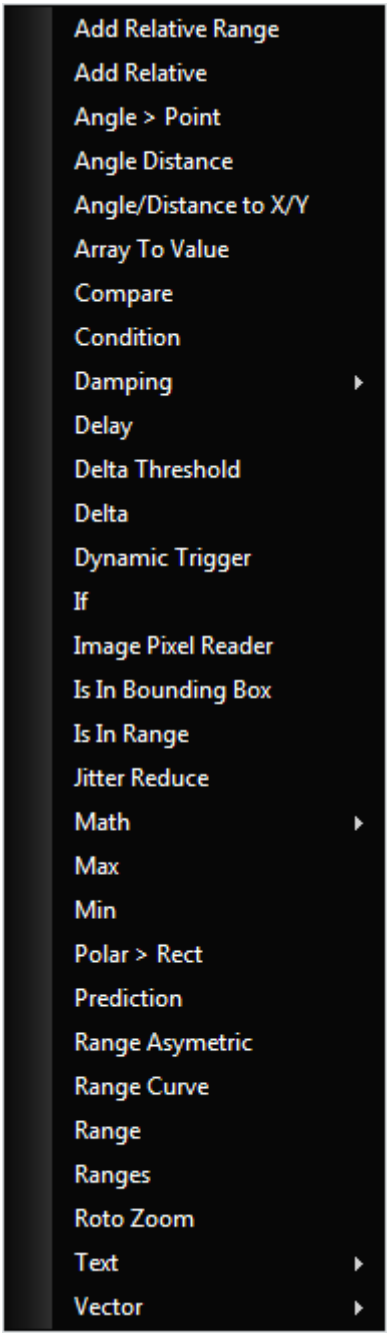
The node generates the following output:

- X Position
- Y Position
- Distance
- Angle

7.4.5 Filter Nodes

Filter nodes are used between Input and Output nodes, they recalculate the input value. Please see the introductory chapter if you like to learn more about other node types or how to create and work with nodes [937] in general.

The following sub chapters describe the various Filter nodes in alphabetic order.

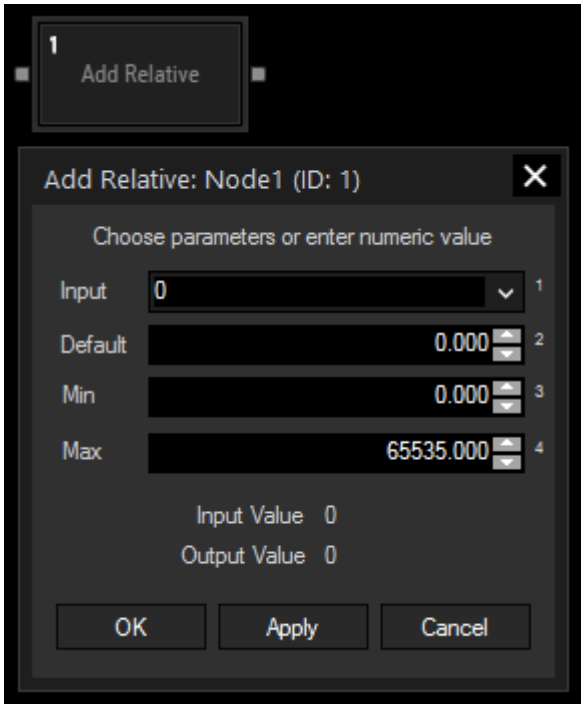


Add Relative Range
Add Relative 1094
Angle > Point 1095
Angle Distance 1096
Angle/Distance to X/Y 1097
Array To Value
Compare 1098
Condition 1099
Damping 1100
Delay 1103
Delta Threshold
Delta 1104
Dynamic Trigger 1105
If 1106
Image Pixel Reader 1107
Is In Bounding Box 1108
Is In Range 1110
Jitter Reduce 1111
Math 1113
Max 1138
Min 1139
Polar > Rect 1140
Prediction 1141
Range Asymetric 1143
Range Curve
Range 1147
Ranges 1148
Roto Zoom
Text 1150
Vector 1166

7.4.5.1 Add Relative Filter

The Add Relative filter node continuously adds up incoming data. The Add Relative filter node is one of the few nodes that is related to the sample rate of the node system (the cycle interval can be inspected and changed in the Options dialog [1510]). When no node property window is opened, the filter works at the given time values. If the updates are increased by open node property dialogs of the node chain, which is 25 times per second, then the timings will be affected. It is recommended to keep all node property dialogs closed when an Add Relative filter node is being used.

The node can be found under Nodes > Filter Nodes > Add Relative



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Default:
Enter the Default Value to which the Input Values will be added.

Min:
Enter the limit for the minimum output value.

Max:
Enter the limit for the maximum output value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

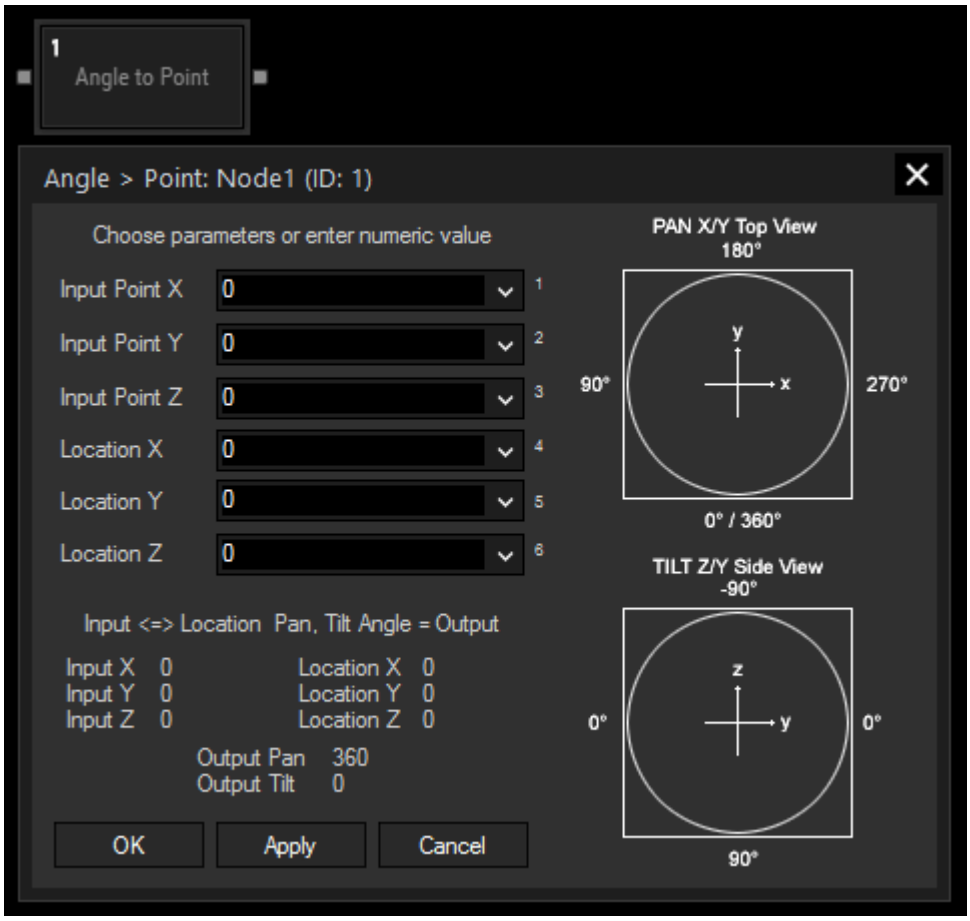
The Node generates the following output:

- Output

7.4.5.2 Angle to Point Filter

The Angle To Point filter node allows to calculate Pan and Tilt angles based on the relationship of moving targets XYZ position input values according to the fixed position of the source. This node is useful to be used for tracking setups that require calculating the exact Pan and Tilt angle for remote cameras or Moving Light projectors.

The node can be found under Nodes > Filter Nodes > Add Angle to Point



Node Properties

Input Point X:
Choose an input node from the drop-down or enter a numeric value.

Input Point Y:
Choose an input node from the drop-down or enter a numeric value.

Input Point Z:
Choose an input node from the drop-down or enter a numeric value.

Location X:
Enter the X Position of the reference point or choose an input node from the drop-down.

Location Y:
Enter the Y Position of the reference point or choose an input node from the drop-down.

Location Z:
Enter the Z Position of the reference point or choose an input node from the drop-down.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

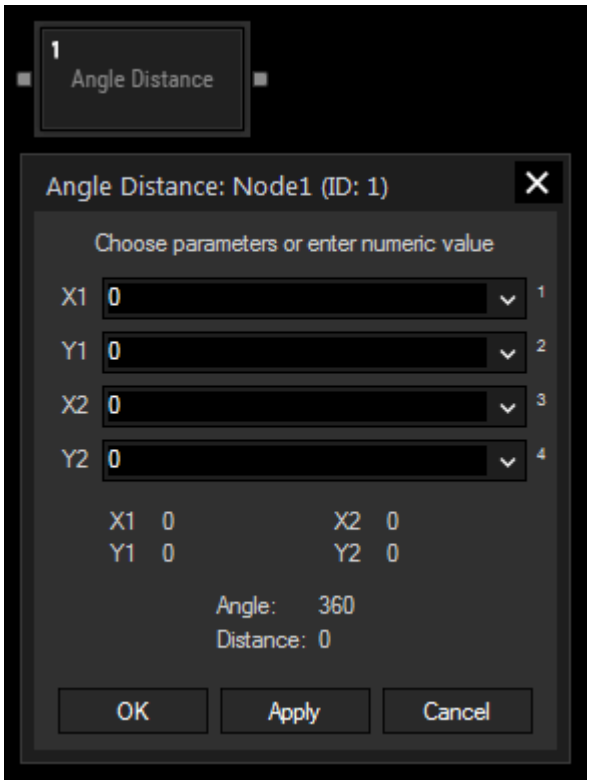
The Node generates the following output:

- Pan Angle
- Tilt Angle

7.4.5.3 Angle Distance Filter

The Angle Distance filter node analyzes two points 1 and 2. It returns the distance between them as well as the angle resulting from the line between the connected points and the X-axis if the origin of the coordinate system would go through point 1.

The node can be found under Nodes > Filter Nodes > Angle Distance



Node properties

X1 and Y1:
Choose an input node from the drop-down or enter a numeric value for the coordinates of the first point.

X2 and Y2:
Choose an input node from the drop-down or enter a numeric value for the coordinates of the second point.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

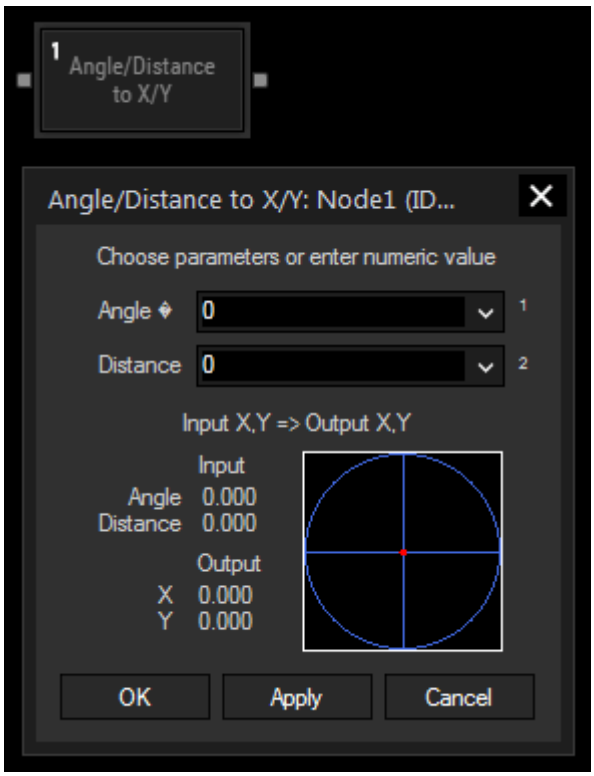
The Node generates the following output:

- Angle (in degrees)
- Distance

7.4.5.4 Angle/Distance to X/Y Filter

The Angle/Distance to X/Y filter node receives an angle and a distance forming a straight line starting at the origin, and calculates the coordinates of the point resulting from it.

The node can be found under Nodes > Filter Nodes > Angle/Distance to X/Y



Node properties

Angle:
Choose an input node from the drop-down or enter a numeric value for the angle (in degrees).

Distance:
Choose an input node from the drop-down or enter a numeric value for the distance between origin and resulting point.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

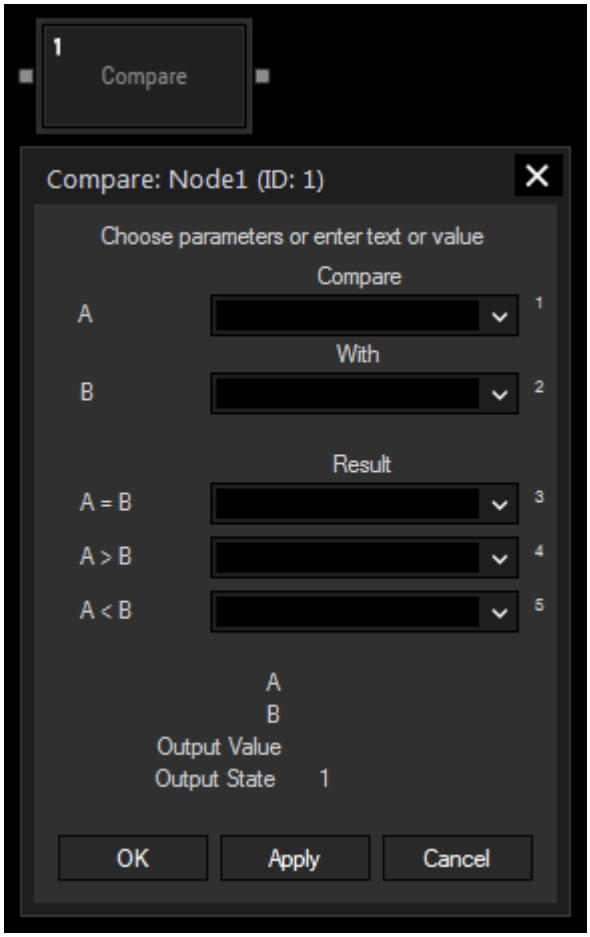
- X (coordinate of the resulting point)
- Y (coordinate of the resulting point)

Please bear in mind that the initial side of the angle is based on the Y-axis instead of the X-axis like in most Cartesian coordinate systems!

7.4.5.5 Compare Filter

The Compare Filter node checks two input values A and B if value A is smaller than, equal to, or greater than value B. Each case can generate a definable output value as well as a representative numeric output state.

The node can be found under Nodes > Filter Nodes > Compare



Node properties

A:
Choose an input node from the drop-down or enter a numeric value.

B:
Choose an input node from the drop-down or enter a numeric value.

Operator:
Assign here specific output values for each case.

A = B: A is equal to B
A > B: A is greater than B
A < B: A is smaller than B

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.

In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

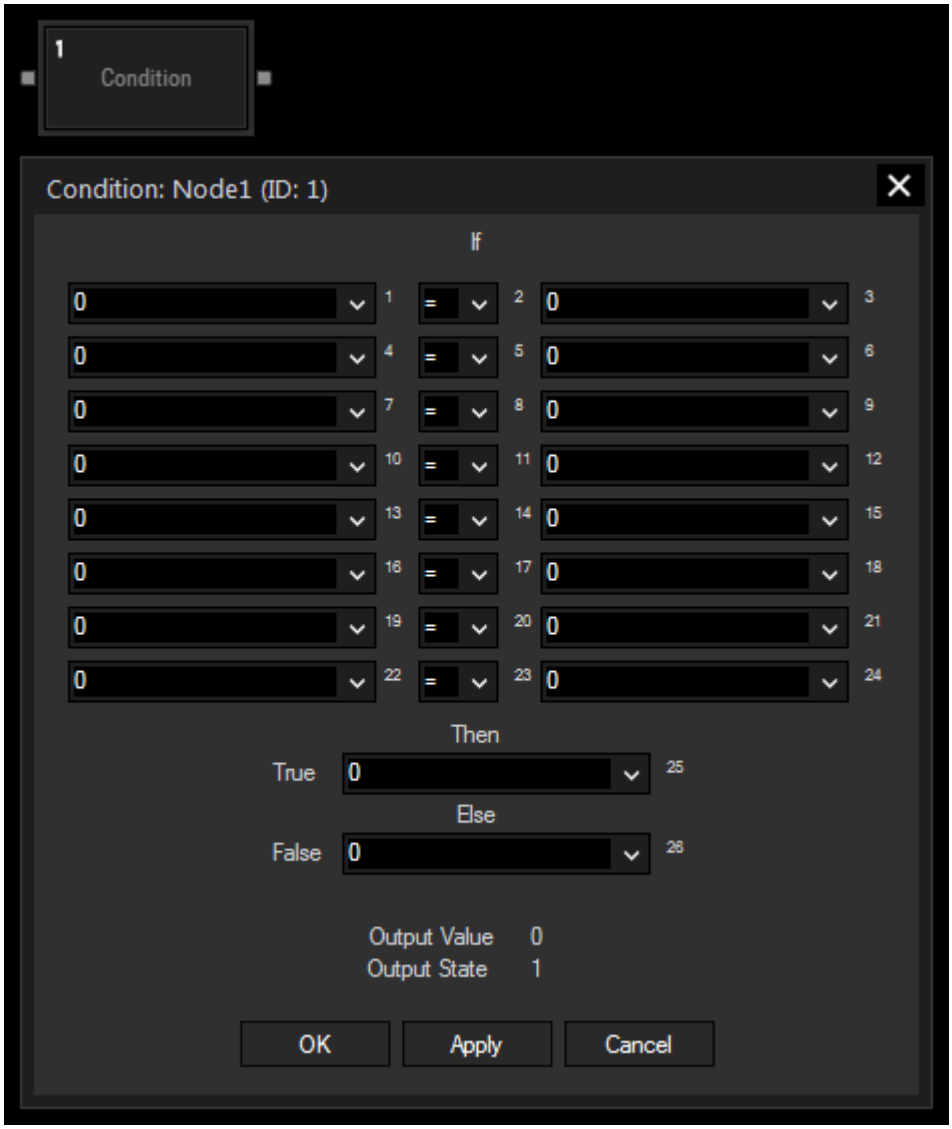
The Node generates the following output:

- Output Value
- Output State: 1 = "A = B", 2 = "A > B", 3 = "A < B"

7.4.5.6 Condition Filter

The Condition Filter Node checks up to eight pairs of values if one is smaller than, equal to, or greater than the other value. The output condition is "true" when all subordinate conditions are also "true". If one or more are "false", the output condition is also "false".

The node can be found under Nodes > Filter Nodes > Condition



Node properties

The value pairs can be defined with the drop-down left and right of the operator.

Value left and right:
Choose an input node from the drop-down or enter a numeric value.

Operator:
= Equal to
=> Greater than or equal to

> Greater than
<= Less than or equal to
< Less than
<> Less than and greater than

True:
If all of the above statements are "true", this input node value or entered numeric value will be given out as output value.

False:
If one or more of the above statements are "false", this input node value or entered numeric value will be given out as output value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

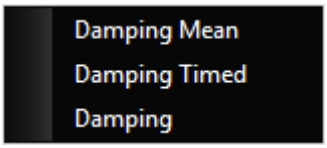
Node output values

The Node generates the following output:

- Output Value
- Output State
  1 = True: All conditions must be true
  0 = False: When at least one of the conditions is not true, the output state is false.

7.4.5.7 Damping Filter Nodes

The Damping filter nodes provide different ways of damping numeric input values.



Damping Mean

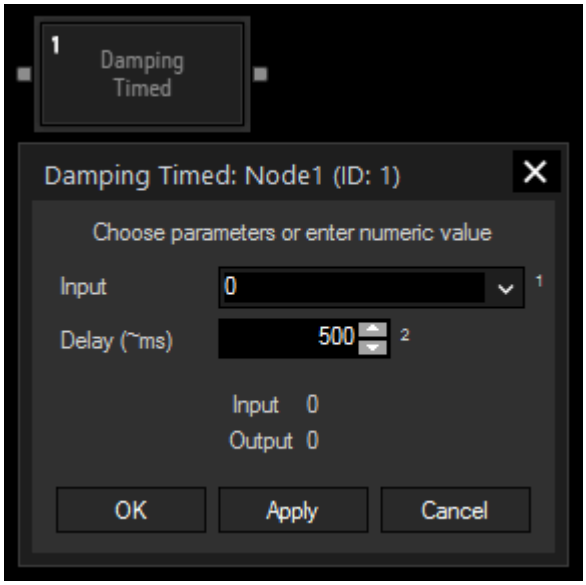
Damping Timed [1101]

Damping [1102]

7.4.5.7.1 Damping Timed Filter

The time based Damping filter node allows setting a delay time for the input data to be damped.

The node can be found under Nodes > Filter Nodes > Damping > Damping Timed



Node properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Delay (ms):
Enter the delay time in milliseconds

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

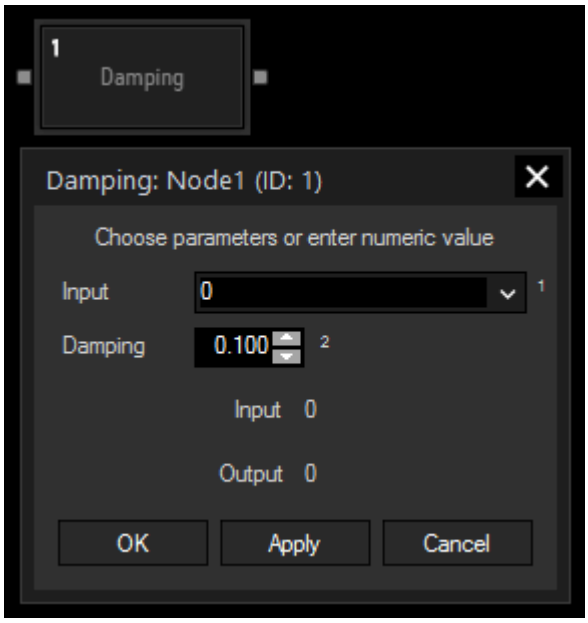
The Node generates the following output:

- Result

7.4.5.7.2 Damping Filter

The prediction based Damping filter node allows reducing noisy input values and extrapolating a trend based on the history of the received values.

The node can be found under Nodes > Filter Nodes > Damping > Damping



Node properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Damping:
Enter the damping factor.
0,1 = maximum damping
1 = no damping

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

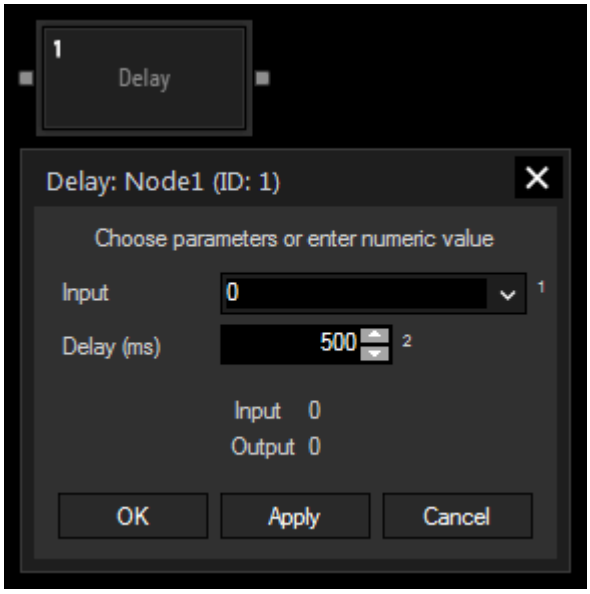
The Node generates the following output:

- Output

7.4.5.8 Delay Filter

The Delay Filter node delays the incoming value and outputs changes after the specified time.

The node can be found under Nodes > Filter Nodes > Delay



Node properties

Input:
Choose an input node from the drop-down.

Delay (ms):
Enter here the amount of delay in milliseconds

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

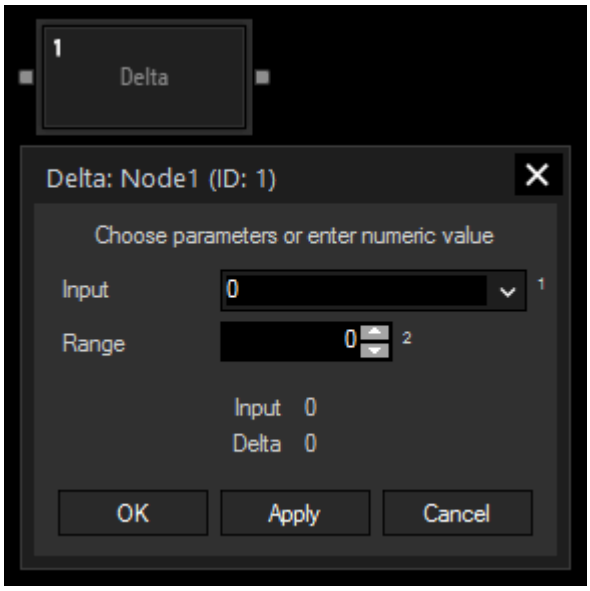
The Node generates the following output:

- Output Value

7.4.5.9 Delta Filter

The Delta filter node allows calculating the incremental delta value of the current value vs. the last input value. The Range defines the maximum value count in case of rotary encoders that would provide the same value range per 360°.

The node can be found under Nodes > Filter Nodes > Delta



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Range:
Enter maximum value if incremental values are used to calculate correct delta values between the start and end value.
You will need this for example when using encoders from 0 to 1439.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

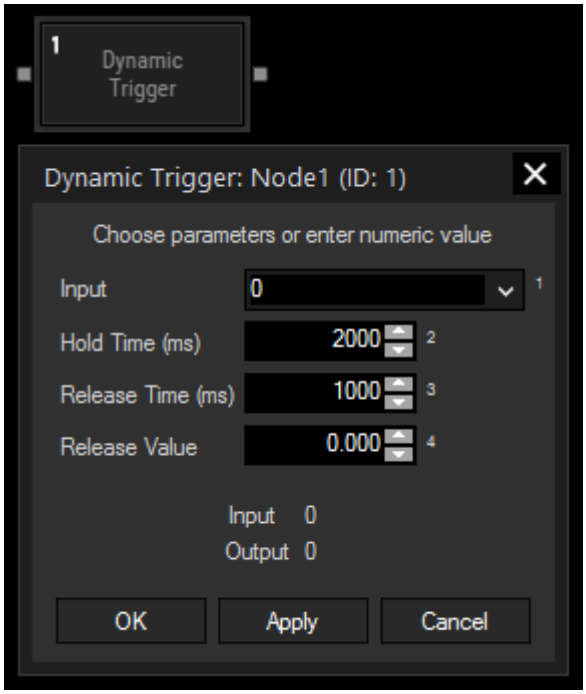
The Node generates the following output:

- Output

7.4.5.10 Dynamic Trigger Filter

The Dynamic Trigger filter node allows acting as a gate with a given hold and release time and an additional target value. The Dynamic Trigger filter node is one of the few nodes that is related to the sample rate of the node system (the cycle interval can be inspected and changed in the Options dialog [1510]). When no node property window is opened, the filter works at the given time values. If the updates are increased by open node property dialogs of the node chain, which is 25 times per second, then the timings will be affected. It is recommended to keep all node property dialogs closed when a Dynamic Trigger node is being used.

The node can be found under Nodes > Filter Nodes > Dynamic Trigger



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Hold Time (ms):
Enter the Hold Time in milliseconds

Release Time (ms):
Enter the Release Time in milliseconds

Release Value:
Enter the Release Value, this will be the minimum value the node gives out.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

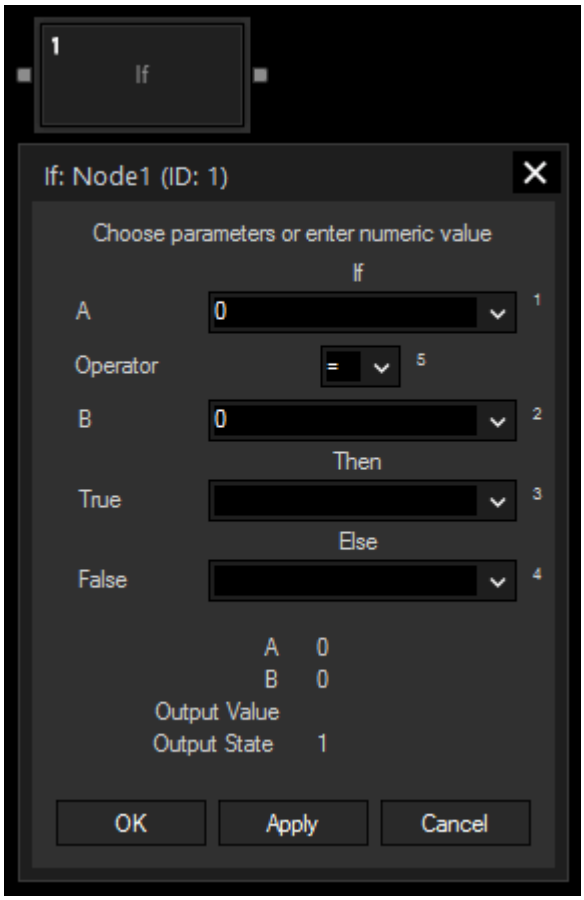
The Node generates the following output:

- Output

7.4.5.11 If Filter

The IF filter node allows comparing incoming data. It can output specific values if the statement is true or false, or simply the state as 1 and 0.

The node can be found under Nodes > Filter Nodes > If



Node Properties

A:
Choose an input node from the drop-down or enter a numeric value or text.

Operator:
Choose the Operator.

B:
Choose an input node from the drop-down or enter a numeric value or text.

True:
If the operation is True, the selected input node value or entered value will be given out as Output Value.

False:
If the operation is False, the selected input node value or entered value will be given out as Output Value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

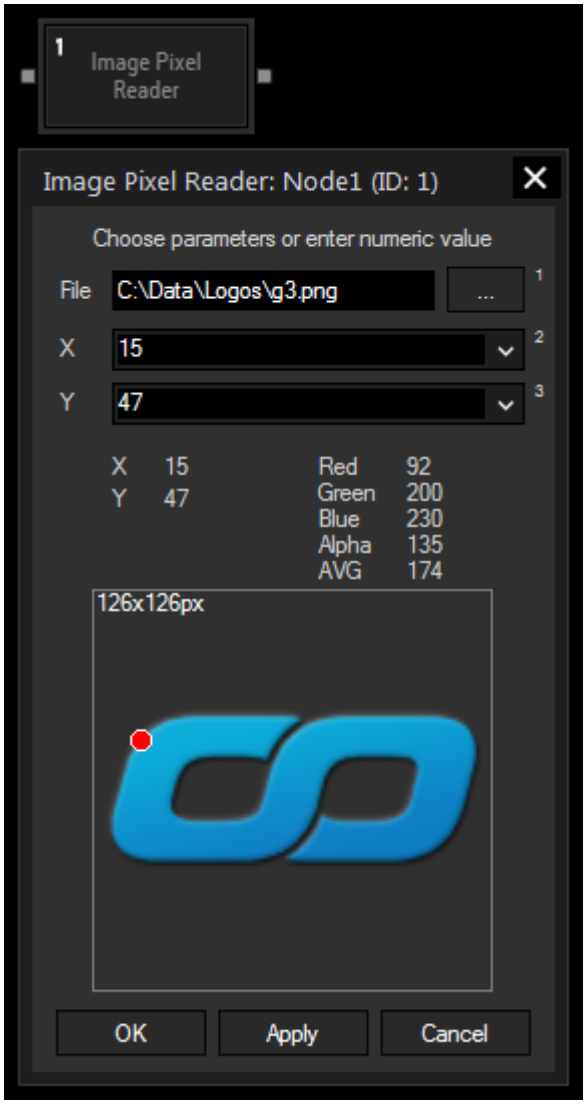
The Node generates the following output:

- Output Value
- Output State (1 = True, 0 = False)

7.4.5.12 Image Pixel Reader Filter

The Image Pixel Reader filter node enables you to read out pixel information of a specified image file. It can return red, green, blue and alpha value of a pixel within the image, as well as an average value for the three colors.

This node can be found under Nodes > Filter > Image Pixel Reader



Node properties

File:
Click the button on the right to browse for an image file. Accepted file formats are BMP, PNG, JPG and TIFF.

X:
Choose an input node from the drop-down or enter a numeric value for the X position of the pixel to be analyzed.

Y:
Choose an input node from the drop-down or enter a numeric value for the Y position of the pixel to be analyzed. Please bear in mind that the pixel count starts with 1/1 in the top left corner.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

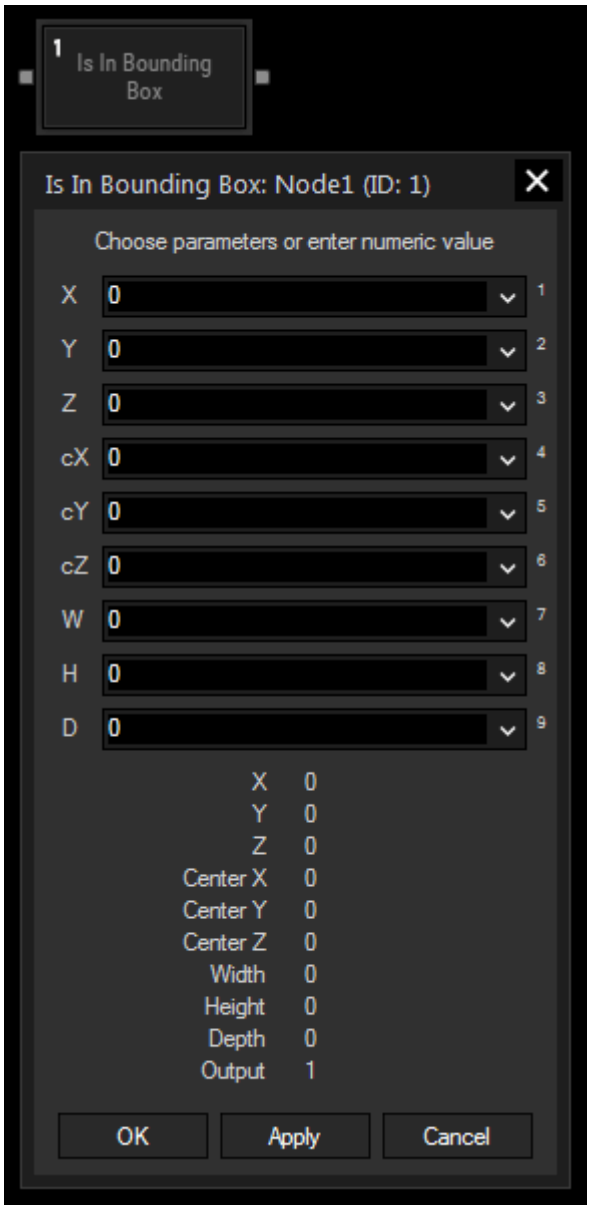
The Node generates the following output:

- Result: "1" if the input point is within the bounding box and "0" if it is outside the bounding box

7.4.5.13 Is In Bounding Box Filter

The Is In Bounding Box filter node creates a virtual box in a 3D coordinate space and checks whether the coordinates of the input point lie within or outside this box.

This node can be found under Nodes > Filter > Is In Bounding Box



Node properties

X, Y and Z:
Choose an input node from the drop-down or enter a numeric value for the X, Y and Z coordinates of the incoming point.

cX, cY and cZ:
Choose an input node from the drop-down or enter a numeric value for the X, Y and Z coordinates of the bounding box' center point.

W:
Choose an input node from the drop-down or enter a numeric value for the bounding box width. The border of the box in X direction is then calculated by the center point X coordinate +/- half the width.

H:
Choose an input node from the drop-down or enter a numeric value for the bounding box height. The border of the box in Y direction is then calculated by the center point Y coordinate +/- half the height.

D:
Choose an input node from the drop-down or enter a numeric value for the bounding box depth. The border of the box in Z direction is then calculated by the center point Z coordinate +/- half the depth.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

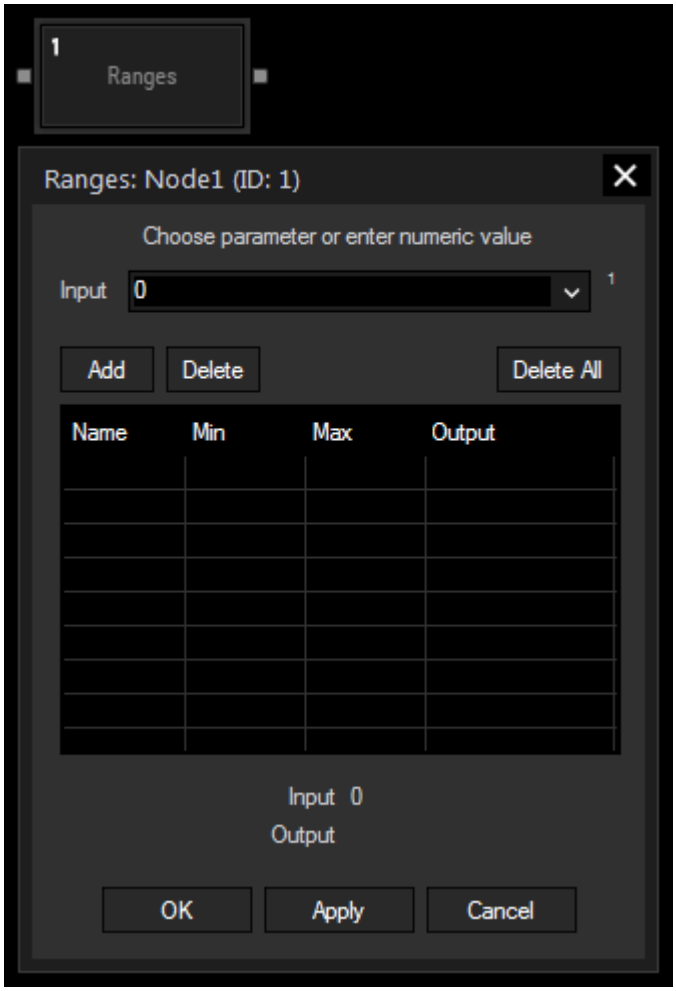
The Node generates the following output:

- Result: "1" if the input point is within the bounding box and "0" if it is outside the bounding box

7.4.5.14 Is In Range Filter

The Is In Range filter node checks whether the input point lies within a specific range or outside of it.

This node can be found under Nodes > Filter > Is In Range



Node properties

Input Value:
Choose an input node from list or enter numeric value.

Min:
Choose an input node from the drop-down or enter a numeric value for the minimum value the input may have to be within the range.

Max:
Choose an input node from the drop-down or enter a numeric value for the maximum value the input may have to be within the range.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

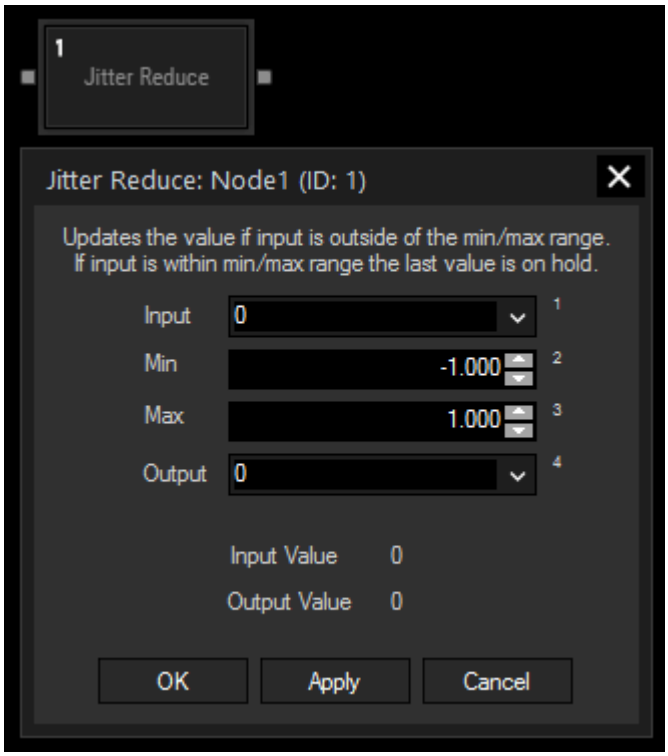
- Output Value: returns the input value if it is within the range. If it is outside, it returns the maximum or minimum value, respective to the input value being higher or lower than the range.
- Result: "-1" if the input value is lower than the minimum, "0" if it is within the range and "1" if it is higher than the maximum

7.4.5.15 Jitter Reduce Filter

The Jitter Reduce filter node allows to eliminate jittering input data, for example when using an AirScan [1277].

The filter updates the output value, if the input is outside the min/max range. If the input is within the min/max range, the last value is on hold.

The node can be found under Nodes > Filter Nodes > Jitter Reduce



Node Properties

Input:
Choose an input node from the list (a delta value).

Min:
Enter minimum value for filtering the jitter from the delta input.

Max:
Enter maximum value for filtering the jitter from the delta input.

Output:
Choose an input node from the list.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

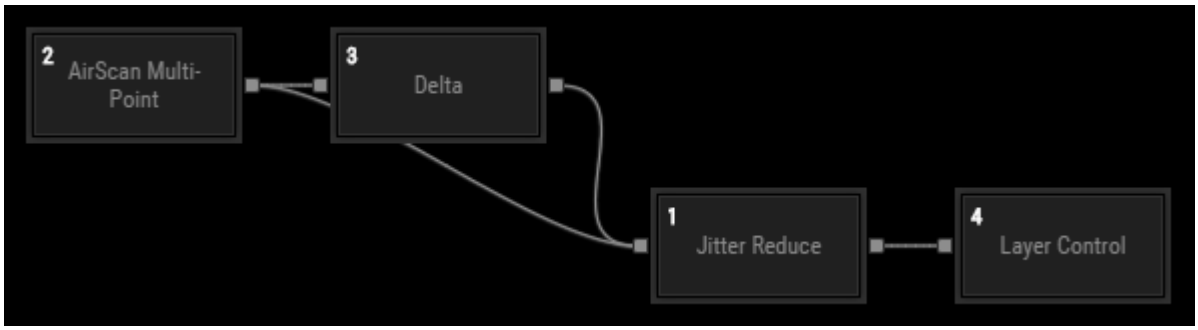
Node output values

The Node generates the following output:

- Jitter Reduce Output.

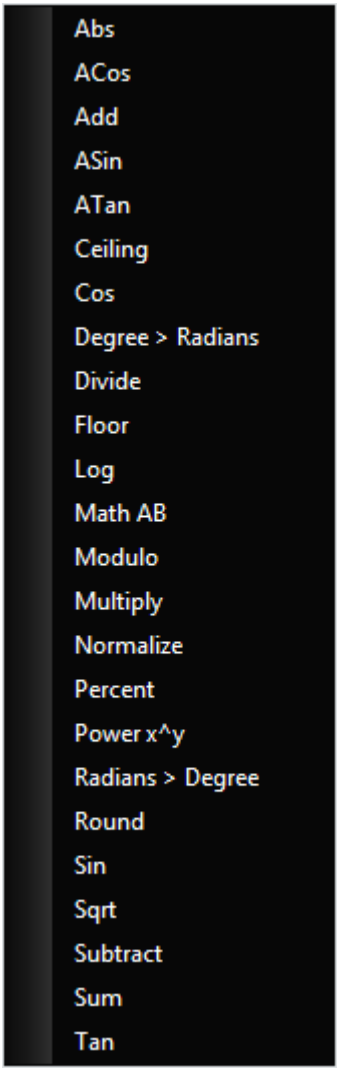
Example

Place a Delta filter node behind your sensor input node to calculate the jittering delta. In the Jitter Reduce node, choose the Delta node output as "Input". If you do not want to route the sensor input data to an output node as long as the delta is between e.g.. -1 and 1, enter these values for min and max. Connect the sensor output node as well to the Jitter Reduce filter node and choose the sensor's output data for "Output".



7.4.5.16 Math Filter Nodes

The Math Filter nodes provide access to all standard math calculations. The following sub chapters describe the various Math Filter nodes in alphabetic order.

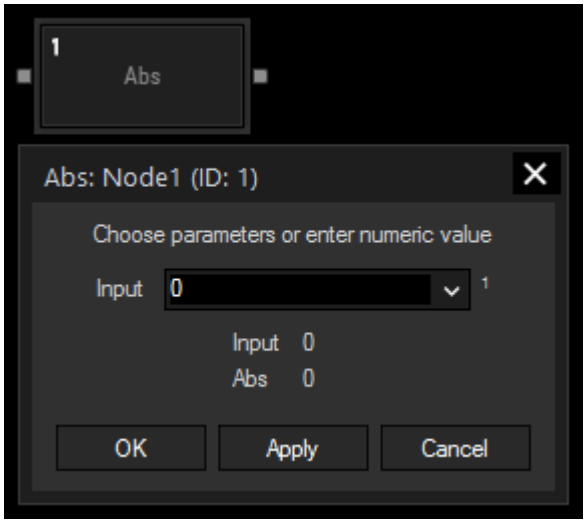


Abs 1114
ACos 1115
Add 1116
ASin 1117
ATan 1118
Ceiling 1119
Cos 1120
Degree > Radians 1121
Divide 1122
Floor 1123
Log 1124
Math AB 1125
Modulo 1126
Multiply 1127
Normalize 1128
Percent 1129
Power x^y 1130
Radians > Degree 1131
Round 1132
Sin 1133
Sqrt 1134
Subtract 1135
Sum 1136
Tan 1137

7.4.5.16.1 Abs Filter

Returns an absolute positive value of the input source value.

The node can be found under Nodes > Filter Nodes > Math > Abs



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

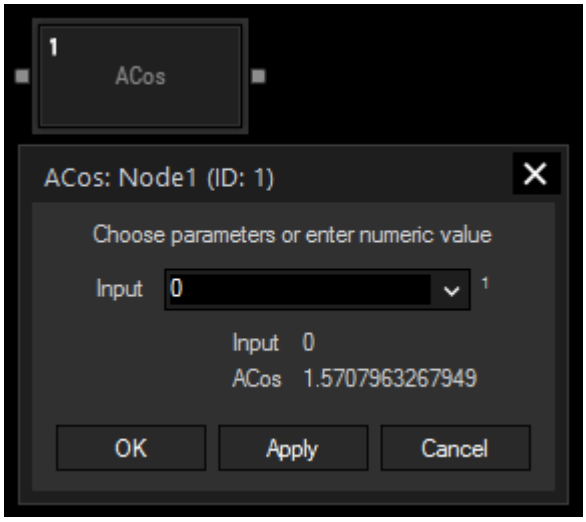
The Node generates the following output:

- Result

7.4.5.16.2 ACos Filter

Arc Cosine calculation.

The node can be found under Nodes > Filter Nodes > Math > ACos



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

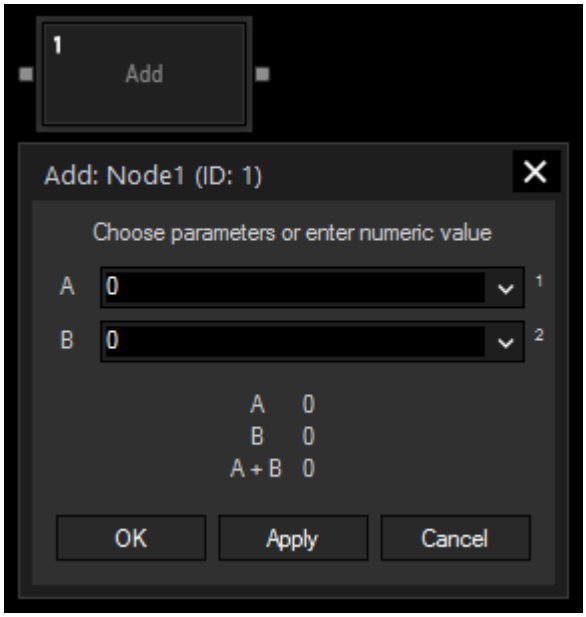
- Result (in rad)

If you need a result in degree, use the Radians to Degree [1131] filter node for converting the result.

7.4.5.16.3 Add Filter

Additive calculation of two source values.

The node can be found under Nodes > Filter Nodes > Math > Add



Node Properties

Input A:
Choose the first input node from the drop-down or enter a numeric value.

Input B:
Choose the second input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

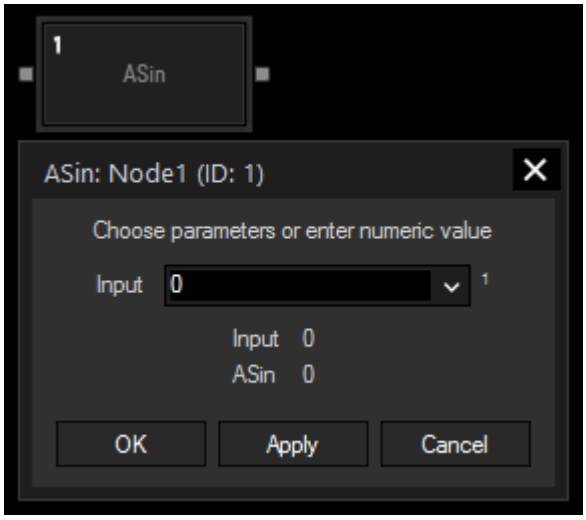
The Node generates the following output:

- Result

7.4.5.16.4 ASin Filter

Arc Sine calculation.

The node can be found under Nodes > Filter Nodes > Math > ASin



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

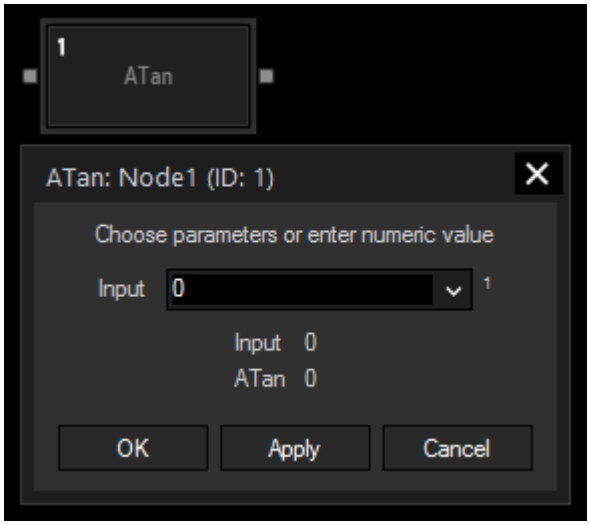
- Result (in rad)

If you need a result in degree, use the Radians to Degree [1131] filter node for converting the result.

7.4.5.16.5 ATan Filter

Arc Tangent calculation.

The node can be found under Nodes > Filter Nodes > Math > ATan



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

- Result (in rad)

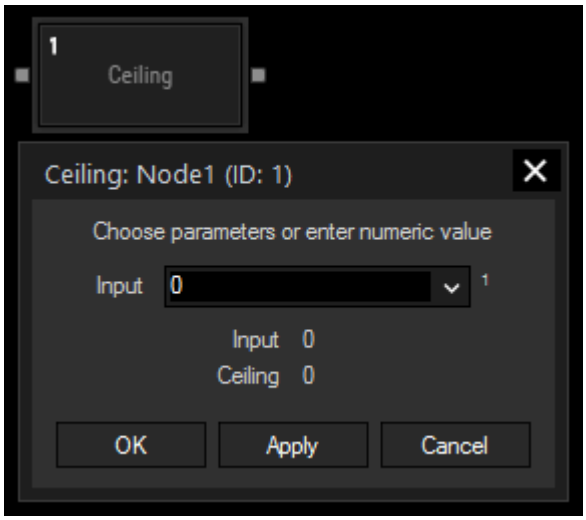
If you need a result in degree, use the Radians to Degree [1131] filter node for converting the result.

7.4.5.16.6 Ceiling Filter

Returns the rounded up value of a decimal input value.

For example: Input value 5,1 results in the output value 6.

The node can be found under Nodes > Filter Nodes > Math > Ceiling



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

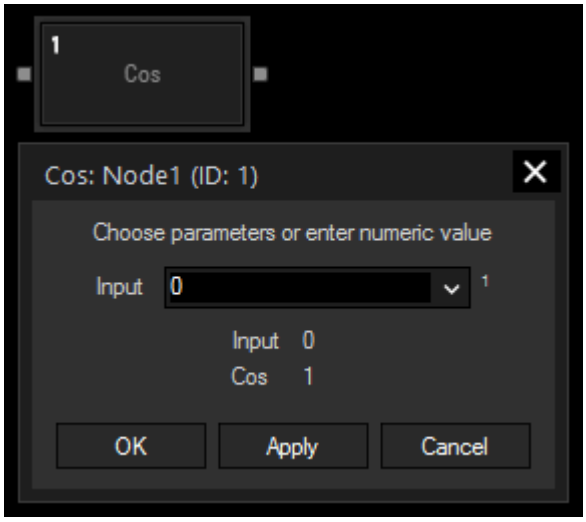
The Node generates the following output:

- Result

7.4.5.16.7 Cos Filter

Cosine calculation of an input angle.

The node can be found under Nodes > Filter Nodes > Math > Cos



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Please bear in mind that the input angle must be in radians. Use the Degree to Radians [1121] node if you need to convert the angle first.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

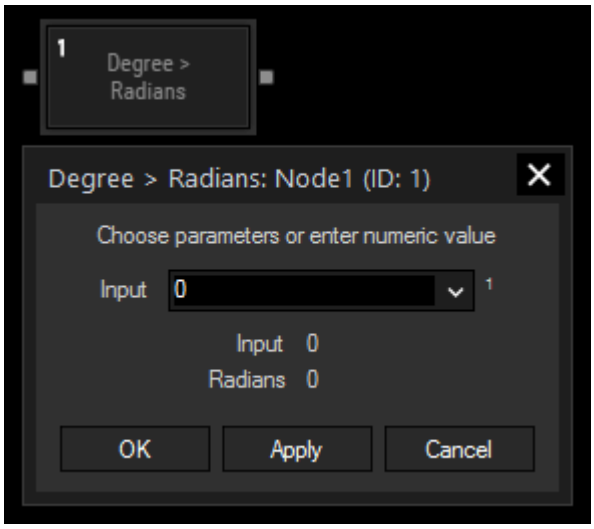
The Node generates the following output:

- Cosine Result.

7.4.5.16.8 Degree > Radians Filter

Converts a value from degrees to radians. This filter node is useful for cosine, sine and tangent calculation and value conversion.

The node can be found under Nodes > Filter Nodes > Math > Degree > Radians



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

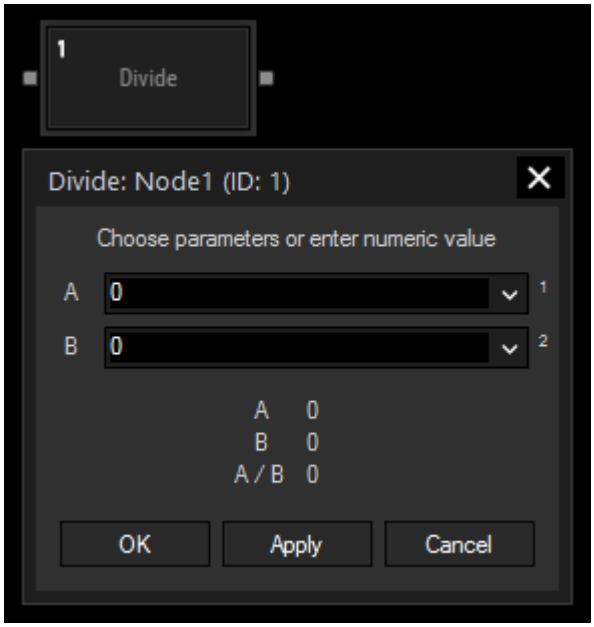
The Node generates the following output:

- Result (in rad)

7.4.5.16.9 Divide Filter

Division calculation of two source values.

The node can be found under Nodes > Filter Nodes > Math > Divide



Node Properties

Input A:
Choose an input node from the drop-down or enter a numeric value.

Input B:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

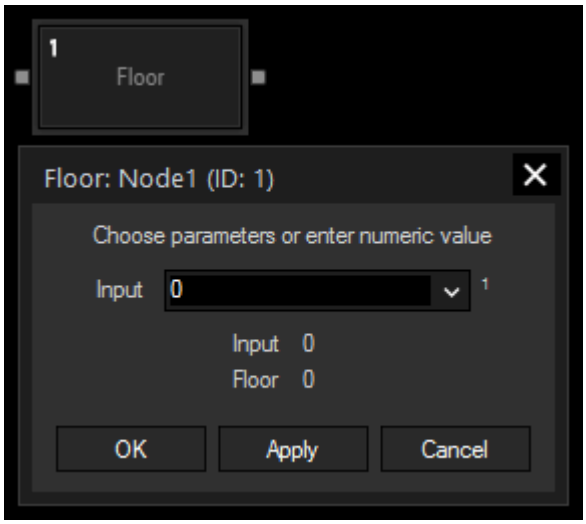
- Result

7.4.5.16.10 Floor Filter

Returns the rounded down value of a decimal input value.

For example: Input value of 4,3 will result in the Output value 4.

The node can be found under Nodes > Filter Nodes > Math > Floor



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

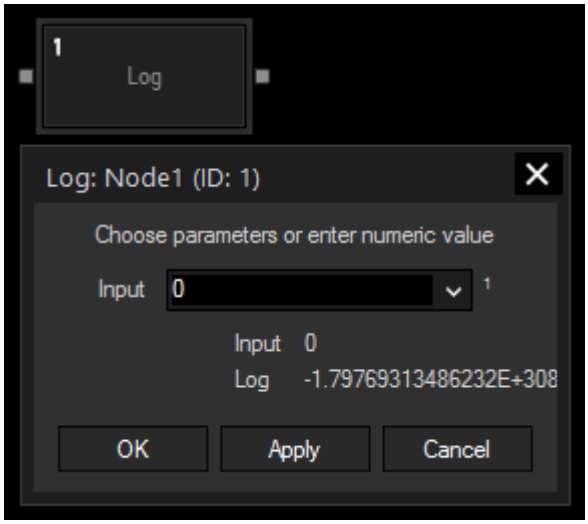
The Node generates the following output:

- Result

7.4.5.16.11 Log Filter

Logarithm calculation of the input source value to the base **e**.

The node can be found under Nodes > Filter Nodes > Math > Log



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

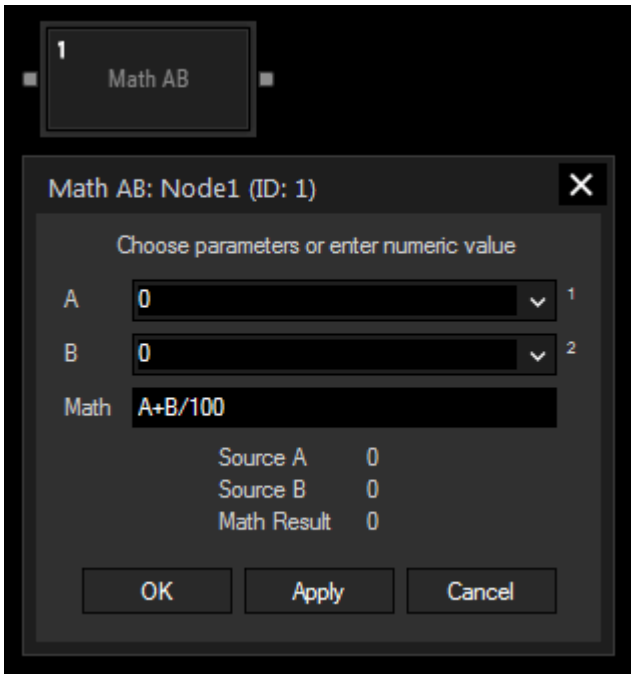
The Node generates the following output:

- Result

7.4.5.16.12 Math AB Filter

The Math AB Filter Node enables you to add your own mathematical formula with two input values A and B. It is useful for reducing the amount of nodes and thus saves performance for larger node systems calculating mathematical operations.

The node can be found under Nodes > Filter Nodes > Math > Math AB



Node properties

A:
Choose an input node from the drop-down or enter a numeric value.

B:
Choose an input node from the drop-down or enter a numeric value.

Math:
Enter here the mathematical formula with the variables A and B you want to calculate.
Usable expressions are "+", "-", "*" and "/", as well as "( )".

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

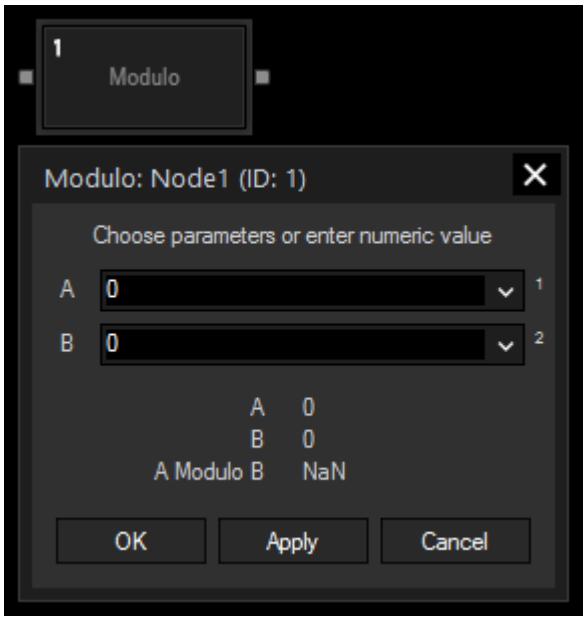
The Node generates the following output:

- Result

7.4.5.16.13 Modulo Filter

Modulo calculation of the input source value.
For Example: 7 modulo 3 = 1.

The node can be found under Nodes > Filter Nodes > Math > Modulo



Node Properties

Input A:
Choose an input node from the drop-down or enter a numeric value for the dividend.

Input B:
Choose an input node from the drop-down or enter a numeric value for the divisor.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

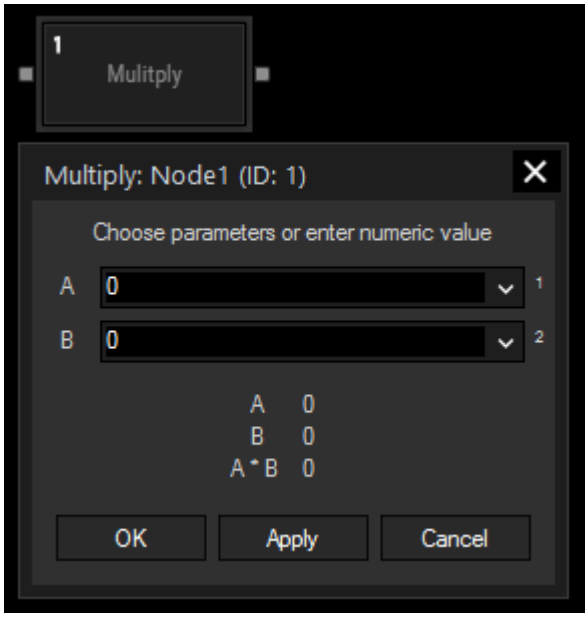
The Node generates the following output:

- Result

7.4.5.16.14 Multiply Filter

Multiplies two input source values.

The node can be found under Nodes > Filter Nodes > Math > Multiply



Node Properties

Input A:
Choose an input node from the drop-down or enter a numeric value.

Input B:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

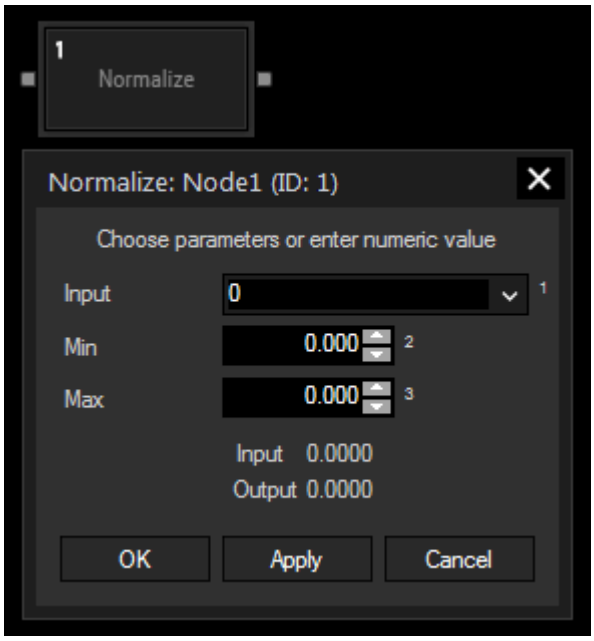
- Result

7.4.5.16.15 Normalize Filter

The Normalize Filter node normalizes the incoming value. The input is mapped on a value between 0 and 1, where Min represents the value mapped on 0 and Max represents the value mapped on 1.

Values outside the range of Min and Max will be normalized with modulo(input/(Max-Min)), normalizations of values smaller than Min will have a negative sign.

The node can be found under Nodes > Filter Nodes > Math > Normalize



Node properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Min:
Enter the numeric value that should be mapped to 0.

Max:
Enter the numeric value that should be mapped to 1.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

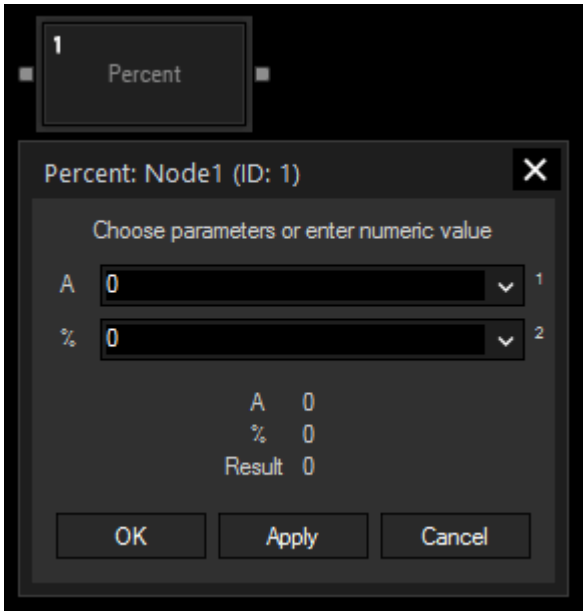
The Node generates the following output:

- Result

7.4.5.16.16 Percent Filter

Calculates the percentage of an input source value.

The node can be found under Nodes > Filter Nodes > Math > Percent



Node Properties

Input A:
Choose an input node from the drop-down or enter a numeric value.

Input %:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

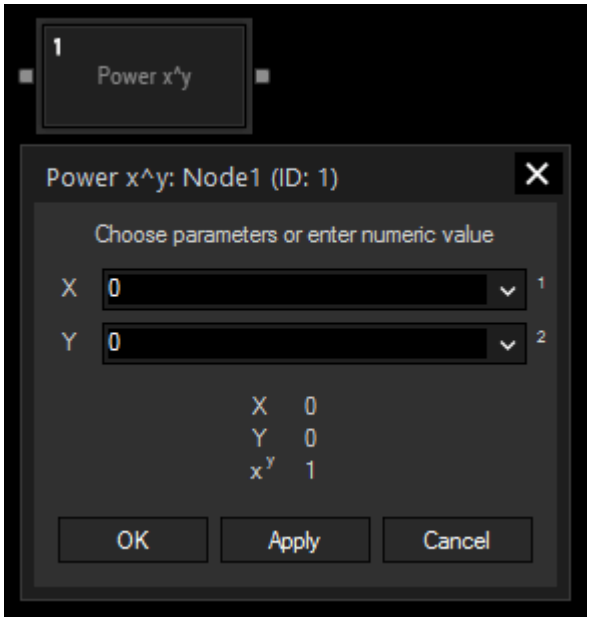
The Node generates the following output:

- Result

7.4.5.16.17 Power x^y Filter

Power of two input source values.

The node can be found under Nodes > Filter Nodes > Math > Power x^y



Node Properties

Input X:
Choose an input node from the drop-down or enter a numeric value for the base.

Input Y:
Choose an input node from the drop-down or enter a numeric value for the exponent.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

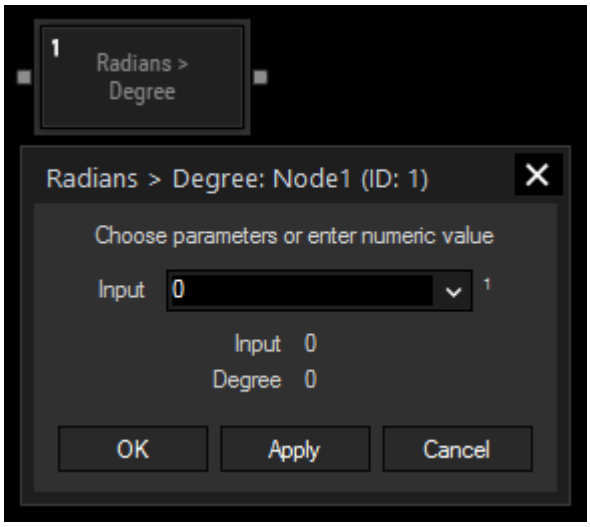
The Node generates the following output:

- Power Result.

7.4.5.16.18 Radians > Degree Filter

Converts a value from radians to degrees. This filter node is useful for cosine sine and tangent calculation and value conversion.

The node can be found under Nodes > Filter Nodes > Math > Radians > Degree



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

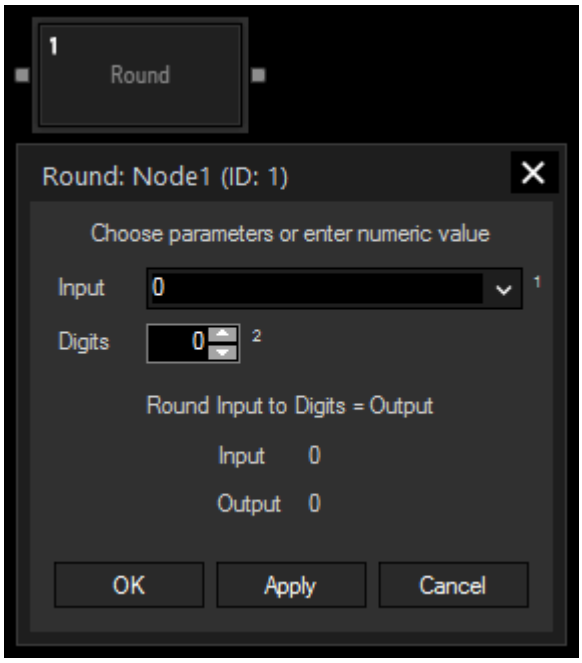
- Result

7.4.5.16.19 Round Filter

This filter allows to round decimal values to a given amount of decimal places.

For Example: Input value 7.633 rounded to 1 digit results in the output value 7.6.

The node can be found under Nodes > Filter Nodes > Math > Round



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Digits:
Enter the amount of decimal places.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

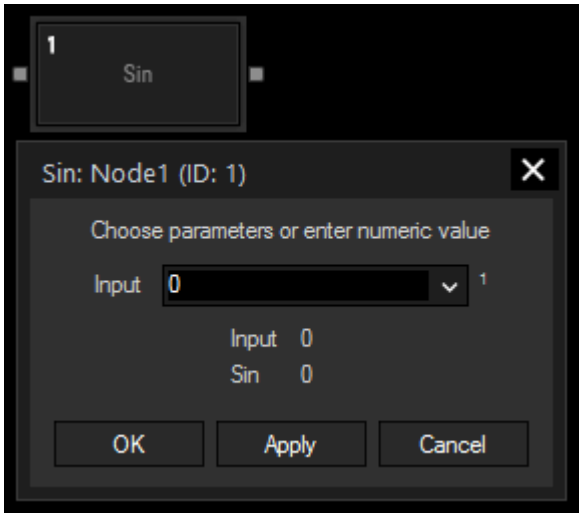
The Node generates the following output:

- Result

7.4.5.16.20 Sin Filter

Sine calculation of an input angle.

The node can be found under Nodes > Filter Nodes > Math > Sin



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Please bear in mind that the input angle must be in radians. Use the Degree to Radians [1121] node if you need to convert the angle first.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

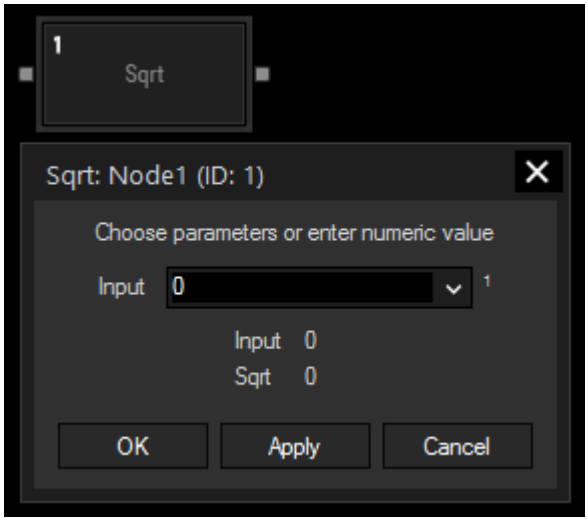
The Node generates the following output:

- Result

7.4.5.16.21 Sqrt Filter

Square Root calculation of the input source value.

The node can be found under Nodes > Filter Nodes > Math > Sqrt



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

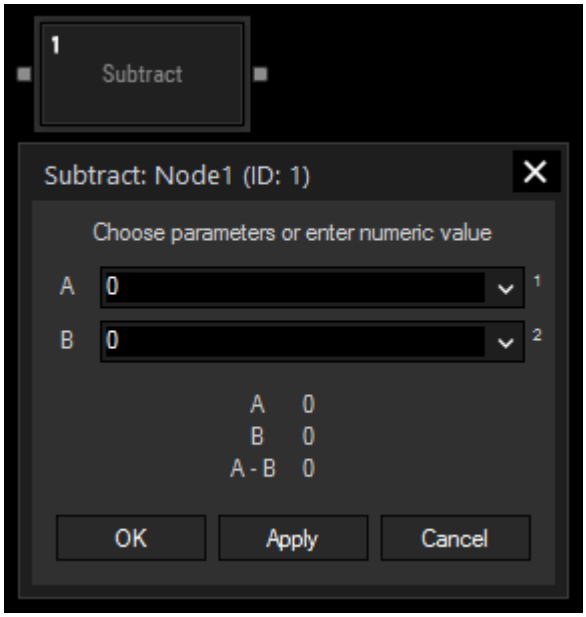
The Node generates the following output:

- Result

7.4.5.16.22 Subtract Filter

Subtraction of two input source values.

The node can be found under Nodes > Filter Nodes > Math > Subtract



Node Properties

Input A:
Choose an input node from the drop-down or enter a numeric value.

Input B:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

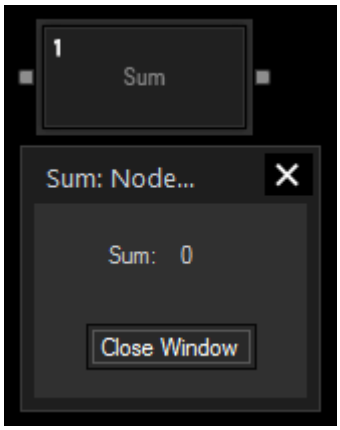
The Node generates the following output:

- Result

7.4.5.16.23 Sum Filter

The Filter Node SUM allows summing up all input values attached.
This is useful when connecting many delta nodes to determine easily when one of them has changed its value.

The node can be found under Nodes > Filter Nodes > Math > Sum



Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

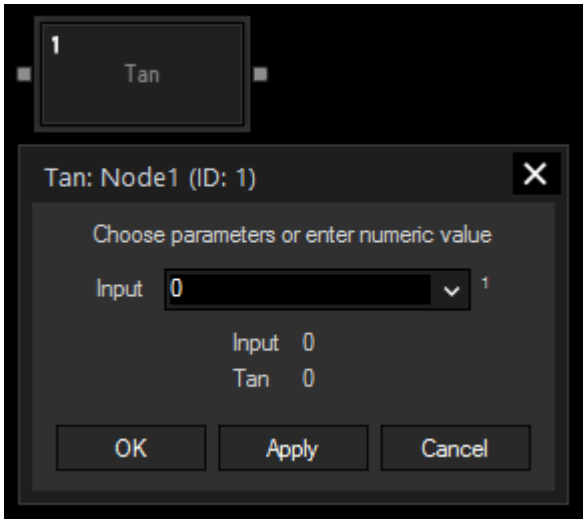
The Node generates the following output:

- Sum

7.4.5.16.24 Tan Filter

Tangent calculation of an input angle.

The node can be found under Nodes > Filter Nodes > Math > Tan



Node Properties

Input A:
Choose an input node from the drop-down or enter a numeric value.

Please bear in mind that the input angle must be in radians. Use the Degree to Radians [1121] node if you need to convert the angle first.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

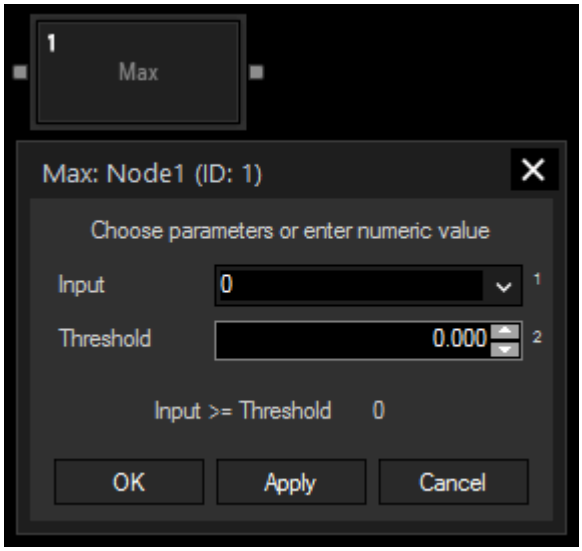
The Node generates the following output:

- Result

7.4.5.17 Max Filter

The Maximum filter node allows setting a maximum limit to input values.

The node can be found under Nodes > Filter Nodes > Max



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Threshold:
Choose the Threshold (maximum output value).

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

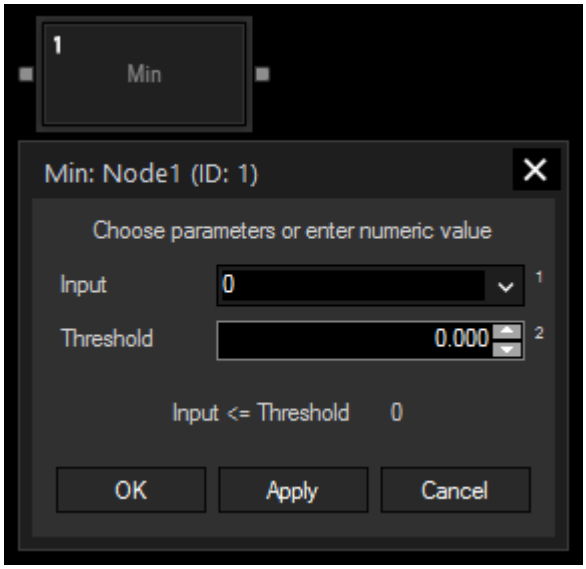
The Node generates the following output:

- Output Value

7.4.5.18 Min Filter

The Minimum filter node allows setting a minimum limit to input values.

The node can be found under Nodes > Filter Nodes > Min



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Threshold:
Choose the Threshold (minimum output value).

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

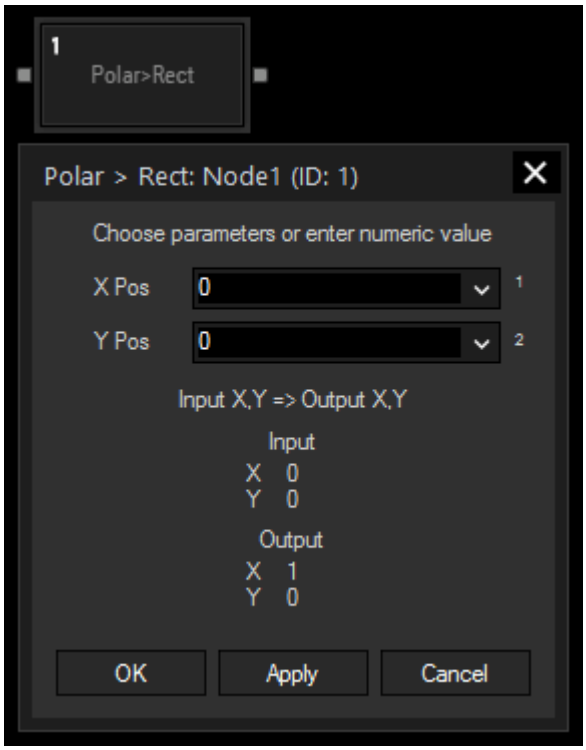
The Node generates the following output:

- Output Value

7.4.5.19 Polar>Rectangular Filter

The Polar to Rectangular filter node allows converting radial XY values to rectangular XY values. This node is useful to convert tracking values received from a spherical surface to apply them to a two dimensional texture coordinate space.

The node can be found under Nodes > Filter Nodes > Polar > Rect



Node Properties

X Pos:
Choose an input node from the drop-down or enter a numeric value.

Y Pos:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

- X
- Y

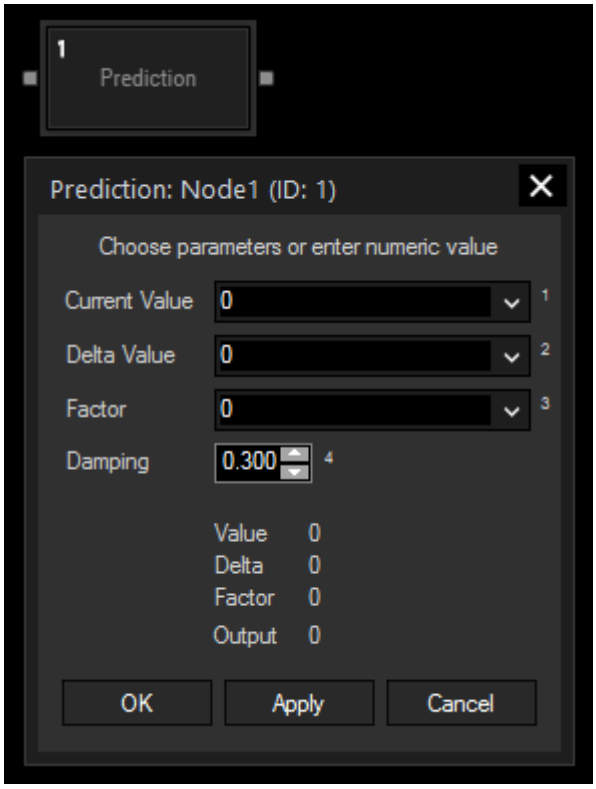
7.4.5.20 Prediction Filter

The Prediction filter node helps improving timing and smoothness of tracked values, especially with any kind of motion tracking.
As processing tracked values always contains a certain delay, it is often necessary to predict the expected outcome to make up for this delay.

Prediction of future values is based on the current value and the delta to the preceding value. A dedicated factor for the amount of prediction needs to be set, depending on the specific setup. As predicted values tend to have a decent amount of jitter, an additional damping value can be set for smoothing the output.

What calculation is being performed by this node can be described with the equation: PredictedValue = CurrentValue + (Factor * DeltaValue)

The node can be found under Nodes > Filter Nodes > Prediction



Node Properties

Current Value:
Choose an input node from the drop-down or enter a numeric value for the current value.

Delta Value:
Choose an input node from the drop-down or enter a numeric value for the delta value between current and last value.
Some nodes, like e.g. the Sensor Link input node [1026], provide a delta output by themselves. For other input sources, it is recommended to set up a Delta filter node [1104] to calculate the necessary delta value and link its output to this input source.

Factor:
Choose an input node from the drop-down or enter a numeric value for the factor the delta value needs to be multiplied with for compensating the delay of the previous processing.
A higher factor means a larger predicted value. If it is too high, the predicted point might even appear ahead of the tracked point.
The factor is determined by the system, trial and error is a common way to find out its value.

Damping:
Enter here the value for damping the output values.

The used method of prediction leads to jitter, as every incoming delta value is being multiplied with the same factor. To counter this jitter, a damping usually needs to be set up.
Please bear in mind that this damping again adds a delay to the output value, depending on the amount of damping.
If you are outputting these values to Pandoras Box, please keep in mind that there already is a Parameter Value Smoothing [210] set up for each Site that might require adjustment in combination with this node.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

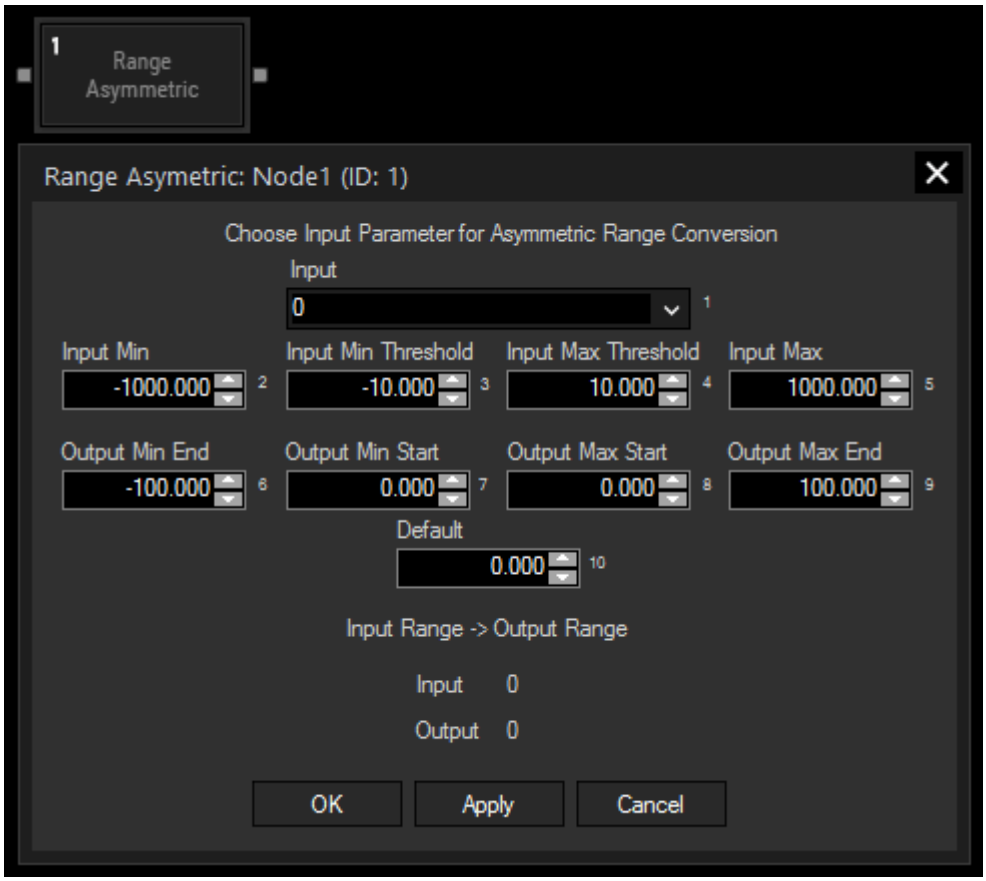
The Node generates the following output:

- Result

7.4.5.21 Range Asymetric Filter

The Range Asymetric node allows to convert a low and high range of an input source value and to leave a gap between the ranges. This filter is especially useful for filtering inconsistent joystick data, for example when the XY motion is mechanically snapping back to center, the center value might not always return the same values. Instead of creating a long node chain with multiple Range and min max nodes this nodes does it all in one step.

The node can be found under Nodes > Filter Nodes > Range Asymetric



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Input Min / Output Min End:
For all values below the Input Minimum the Output Min End Value will be the output.

Input Min Threshold / Output Min End :
For all values from Input Min up to Input Min Threshold the range between Output Min End and Output Min Start will be given out.

Input Max Threshold / Input Max:
For all values between Input Min Threshold and Input Max Threshold the output will be the Default Value.

Input Max / Output Max End:
For all values above this Input Maximum the Output Max End Value will be the output.

Default:
This value will be given out for all input values between Input Min and Input Max Threshold.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

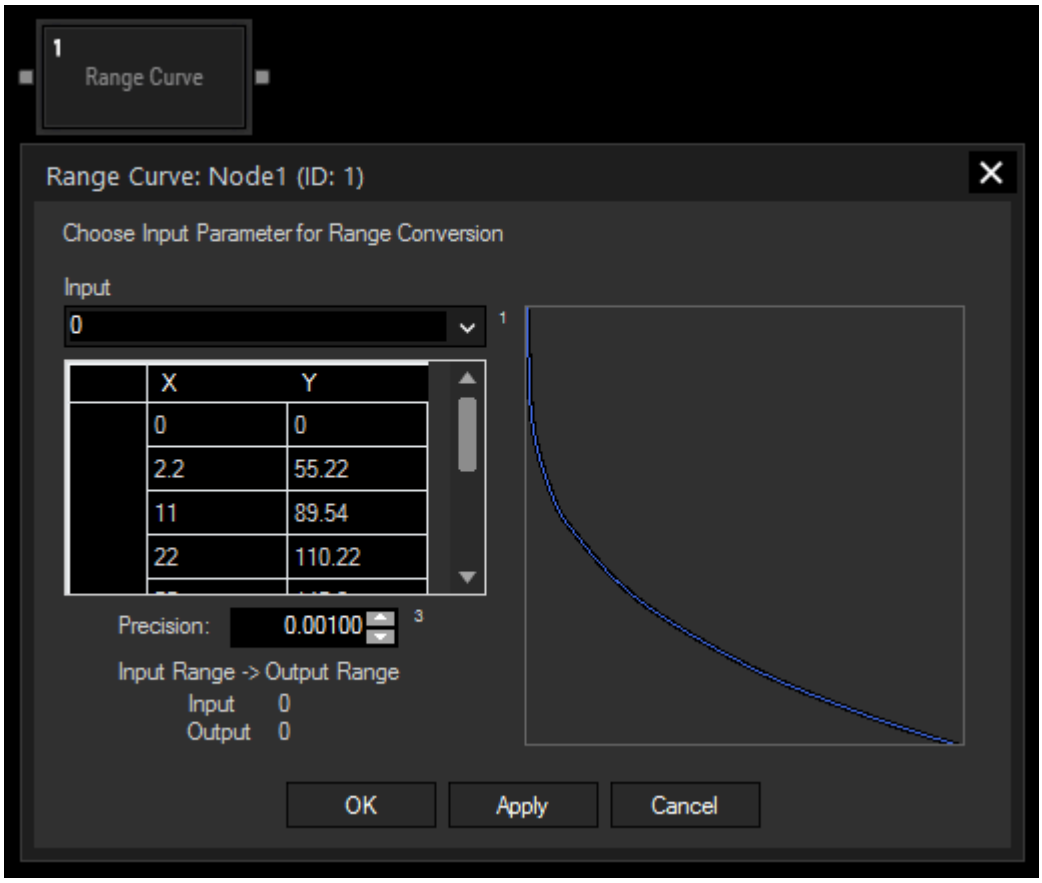
- Output

7.4.5.22 Range Curve Filter

The Range Curve filter node allows assigning any input value to a new output value, where the mapping of input to output is defined by a non-linear function. For defining this function, X and Y coordinates of points can be set and will be automatically connected by a smooth curve. The input value represents the X coordinate of a point on the curve, the output value respectively the Y coordinate of the same point.

It is for example helpful to transform the linear value distribution of a Fader [874] widget to the logarithmic scale of a PB Device Volume [663] fader (see below Example 2 [1145]).

The node can be found under Nodes > Filter Nodes > Range Curve



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value that will be mapped on the X axis.

Set Points:
Enter here the X and Y coordinates of the points for defining the mapping function. Please bear in mind that the X coordinates need to consecutively follow each other from smallest to largest value if you want to avoid loops (and thus inconclusive mapping of one input to several output values).

The curve at the right side of the dialog displays the graph derived from the entered points and is being updated with each new coordinate. The positive X axis goes to the right, the positive Y axis to the bottom, a maximum of ~220 is visible for each.
Larger values are not displayed, but are taken into calculation of the curve.

Precision:
Enter here the precision with which the curve should be written. It is recommended to set the precision to at least as much as your most precise result is required to be, e.g. two decimals.

Examples

Example 1:
The image above pictures a curve derived from a simple gamma curve with γ = 0.3, normed on input values from 0 to 220 for a more significant display:
$Y = (X/220)^{0.3} * 220$

The more points are given for the Range Curve node, the more exact the result will be. In this case, a selection of X and Y value pairs was calculated with the above equation and added as points to the node:

X	Y
0	0
2.2	55.22
11	98.54
22	110.22
55	145.2
110	178.64
165	201.74
220	220

Input values that now arrive as X value of the equation will be mapped and outputted as the respective Y equivalent, e.g.:
Input: 78.64
Output: 161.17 (exact value using the equation for calculation: 161.58)

The outputted result deviates from the actual result, but with more points and more precise coordinates (i.e. more decimals), it will also converge more.

Example 2:
Linking a linear Fader [874] widget with a logarithmic (dB) Volume fader in PB

Volume Faders in PB can be found in Audio Track or Video Layer Device Parameters.
The needed input values of a Volume Fader range from 0 to 2, those values will be converted within PB into the logarithmic dB values you can see at the label (please refer to the chapter Volume [663] for more information about the value conversion).
If you simply use the Fader widget's own Control option, select the respective Device and Volume parameter and set the Minimum to 0 and the Maximum to 2, you will see that the movement of the WD Fader and the movement of the PB fader do not conform, due to the logarithmic calculation.

If you want both movements to be identical, you need to send the values already with a logarithmic distribution from WD. This is how you can easily achieve it:
1. Create a Fader widget, set the Minimum to 0 and the Maximum to 100.
2. Create a Fader Input node and link it to the Fader, connect this to a Range Curve filter node and connect this to a PB Device Parameter output node.
3. Select the Fader input node as Input in the Range Curve Node and apply the following values to the table:

X	Y
0	0
12.5	0.03125

25	0.125
50	0.5
75	1.125
100	2

4. Open the Device Control node, set up the Device and Parameter you would like to address and choose the Range Curve node as input value.
5. Enjoy!

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e. methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

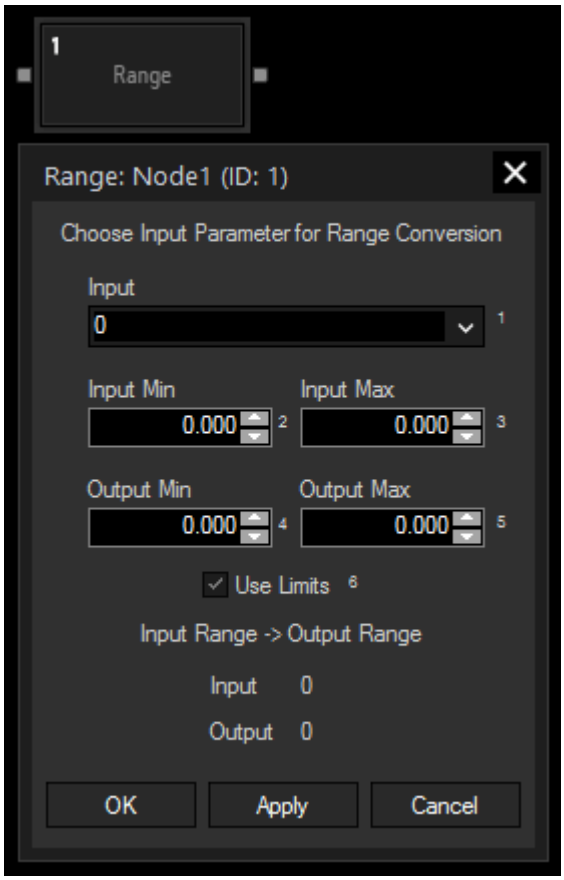
The Node generates the following output:

- Output

7.4.5.23 Range Filter

The Range filter node allows mapping any input range to a new output range. You may set limits, otherwise if the input range is exceeded in both, positive or negative values, the conversion will still be applied.

The node can be found under Nodes > Filter Nodes > Range



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Input Min:
Enter the Input Minimum value.

Input Max:
Enter the Input Maximum value.

Output Min:
Enter the Output Minimum value. The Input Minimum value will be mapped to the Output Minimum value.

Output Max:
Enter the Output Minimum value. The Input Maximum value will be mapped to the Output Maximum value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

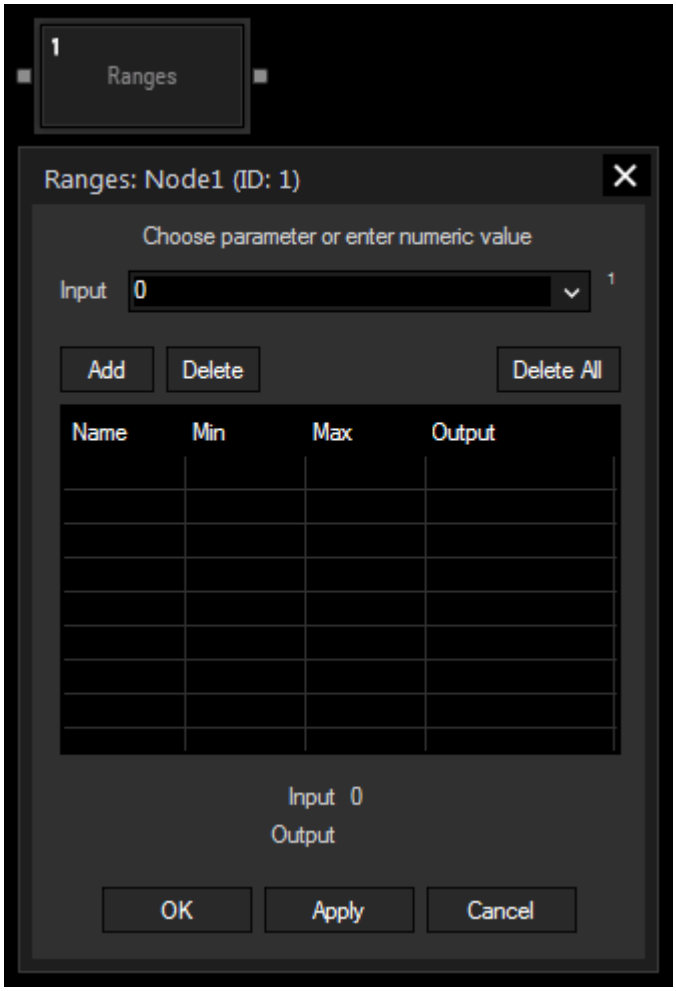
The Node generates the following output:

- Output

7.4.5.24 Ranges Filter

The Ranges filter node checks if the input point lies within the specifiable ranges and outputs the set value corresponding to this active range.

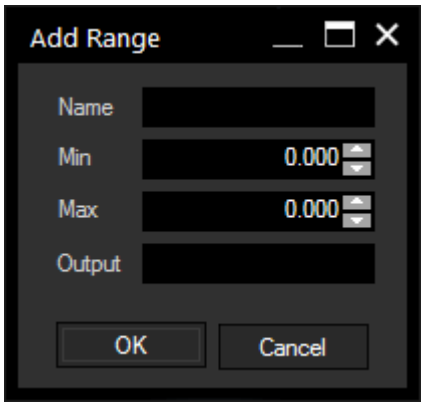
This node can be found under Nodes > Filter > Ranges



Node properties

Input:
Choose an input node from list or enter numeric value.

Add:
Open the Add Range dialog to configure a new range.



Name:
Define here a name for your range

Min / Max:
Enter here the minimum and maximum values representing the range. Please note that the Max value needs to be larger than the Min value.

Output:
Define a string or numeric value that should be outputted when the input value lies within this range.
If different ranges overlap in this area, the output value of the range positioned higher in the list will be chosen as output.

If the input value is out of any range, the output will hold the last active range output value.

Delete:
Select an entry from the list and press this button to delete it. An entry can also be deleted by right-clicking on it an selecting "Delete".

Delete All:
Press this button to reset the whole list.

Edit:
If you want to edit an existing range, right-click on one of the entries from the list and select "Edit" to open the Edit dialog (corresponding to the Add dialog).

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

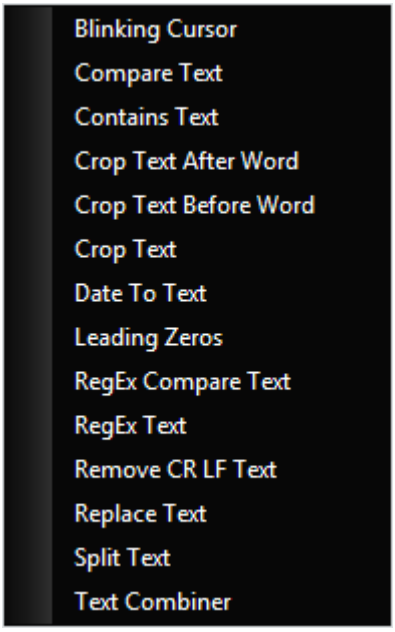
Node output values

The Node generates the following output:

- Output Value (set output value of the currently active range)

7.4.5.25 Text Filter Nodes

Use these nodes to handle text assets, e.g. to compare texts. The following sub chapters describe the various Text Filter nodes in alphabetic order.

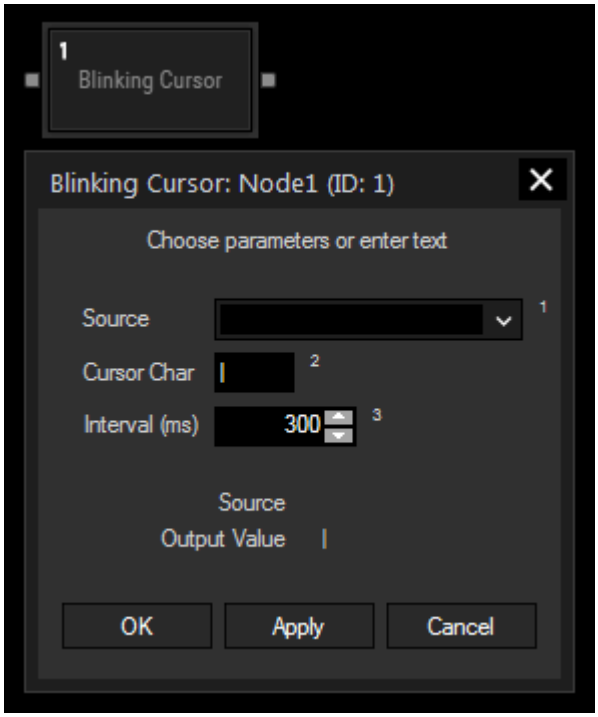


Blinking Cursor 1151
Compare Text 1152
Contains Text 1153
Crop text After Word 1154
Crop text Before Word 1155
Crop Text 1155
Date To Text 1157
Leading Zeros 1158
RegEx Compare Text 1158
RegEx Text 1160
Remove CR LF Text 1162
Replace Text 1163
Split Text 1164
Text Combiner 1165

7.4.5.25.1 Blinking Cursor Filter

The Blinking Cursor node adds a freely selectable, blinking character value after the input text string.

The node can be found under Nodes > Filter Nodes > Text > Blinking Cursor



Node properties

Source:
Choose an input node from the drop-down or enter text.

Cursor Char:
Enter here the character that should be added to the source string.

Interval (ms):
Enter the blinking interval in milliseconds.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

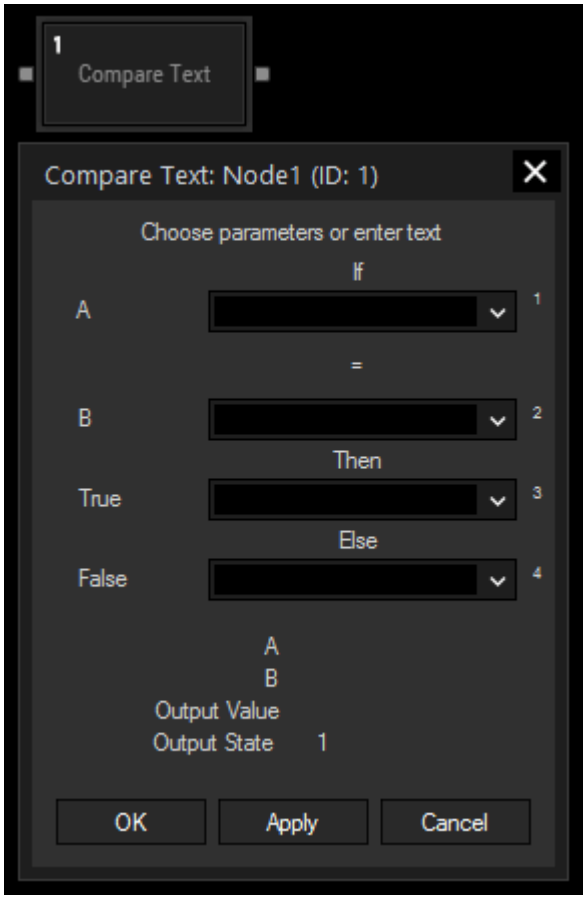
The Node generates the following output:

- Output Value

7.4.5.25.2 Compare Text Filter

Compares two text sources.

The node can be found under Nodes > Filter Nodes > Text > Compare Text



Node Properties

Input A:
Choose an input node from the drop-down or enter a text.

Input B:
Choose an input node from the drop-down or enter a text.

True:
Choose an input node from the drop-down or enter a text that should be the output if the result is true.

False:
Choose an input node from the drop-down or enter a text that should be the output if the result is false.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

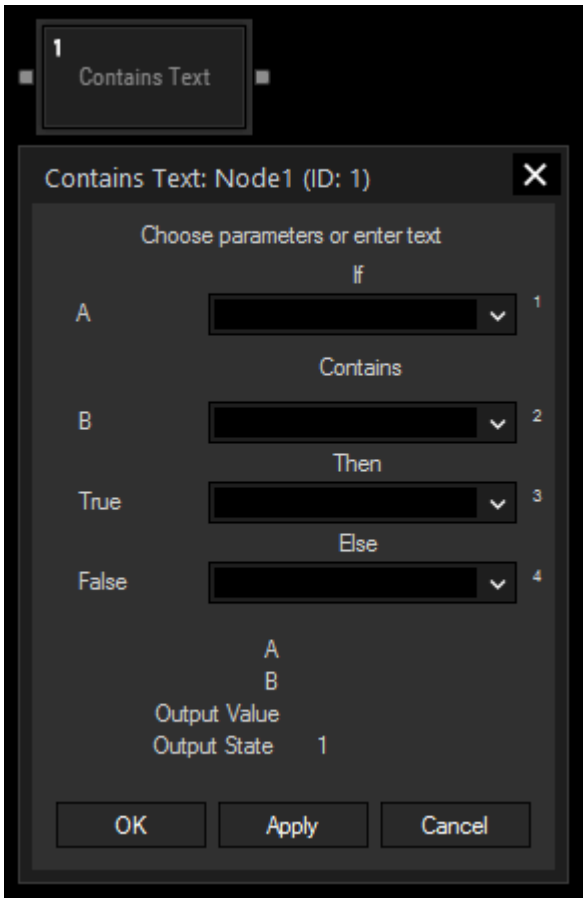
The Node generates the following output:

- Output value
- State (true = 1 or false = 0)

7.4.5.25.3 Contains Text Filter

This filter allows scanning a text input source for specific character strings.

The node can be found under Nodes > Filter Nodes > Text > Contains Text



Node Properties

Input A:
Choose an input node from the drop-down or enter a text.

Input B:
Choose an input node from the drop-down or enter a text.

True:
Choose an input node from the drop-down or enter a text that should be the output if the result is true.

False:
Choose an input node from the drop-down or enter a text that should be the output if the result is false.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

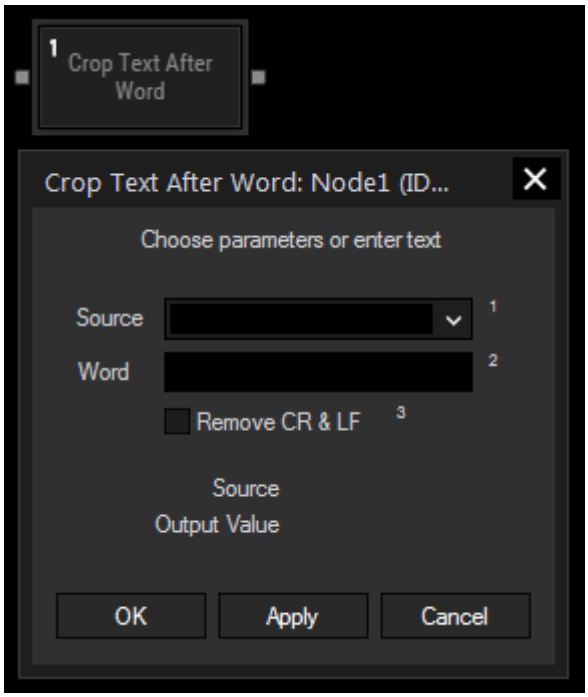
The Node generates the following output:

- Output Value
- State (true = 1 or false = 0)

7.4.5.25.4 Crop Text After Word

This filter node crops the rest of the text after a current word. A word is considered to be closed when a space character comes after it, if only a part of the word is entered, the cropping automatically starts before the next white space.

The node can be found under Nodes > Filter Nodes > Text > Crop Text After Word



Node properties

Source:
Choose an input node from the drop-down or enter text.

Word:
Enter the word after which the source text will be cropped. If the word does not exist in the source text, an empty text string will be outputted.

Remove CR&LF:
Check this box to remove \r and \n inside the text.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

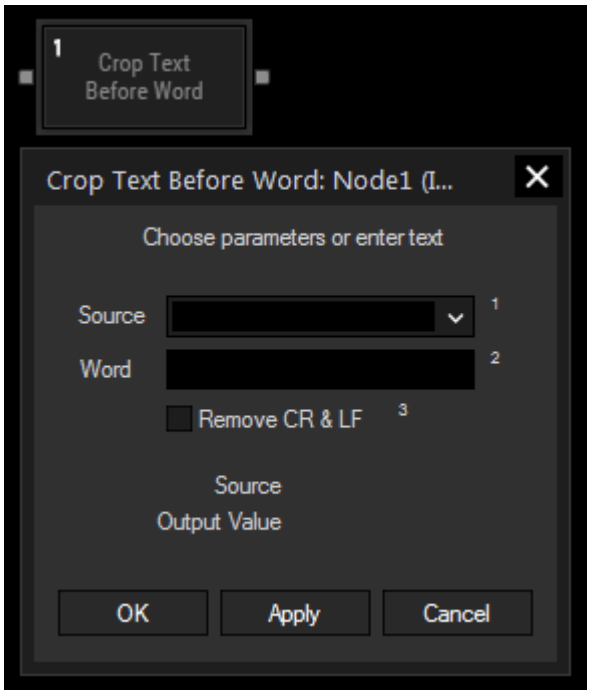
The Node generates the following output:

- Output Value: source text without cropped parts.

7.4.5.25.5 Crop Text Before Word

This filter node crops the rest of the text after a current word. A word is considered to be started when a space character comes before it, if only a part of the word is entered, the cropping automatically starts with the previous space character.

The node can be found under Nodes > Filter Nodes > Text > Crop Text Before Word



Node properties

Source:
Choose an input node from the drop-down or enter text.

Word:
Enter the word before which the source text will be cropped. If the word does not exist in the source text, an empty text string will be outputted.

Remove CR&LF:
Removes \r and \n inside the text.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

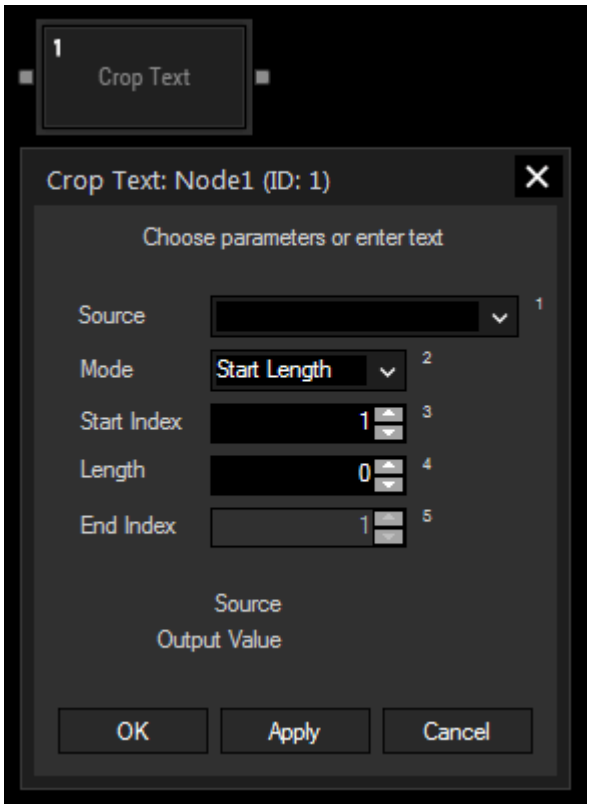
The Node generates the following output:

- Output Value: source text without cropped parts.

7.4.5.25.6 Crop Text Filter

The Crop Text Filter node crops the text before and after a defined part. There are different modes for cropping available.

The node can be found under Nodes > Filter Nodes > Text > Crop Text



Node properties

Source:
Choose an input node from the drop-down or enter a numeric value.

Mode:
Start Length: Crops the source text before the start index and after a defined length.
Start End: Crops the source text before and after the defined start- and end Index.
Start Trim: Crops the source text before the defined start index.
End Length: Crops the text after the end index and leaves the amount of characters defined with the length before the end index.

The following three parameters are only available if needed by the selected mode

Start Index:
Select the start index for your cropping mode. The first character of the source string has index 1.

Length:
Select the length of your output string. A length of 1 will output only one character.

End Index:
Select the end index of your cropping mode. The first character of the source string has index 1.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

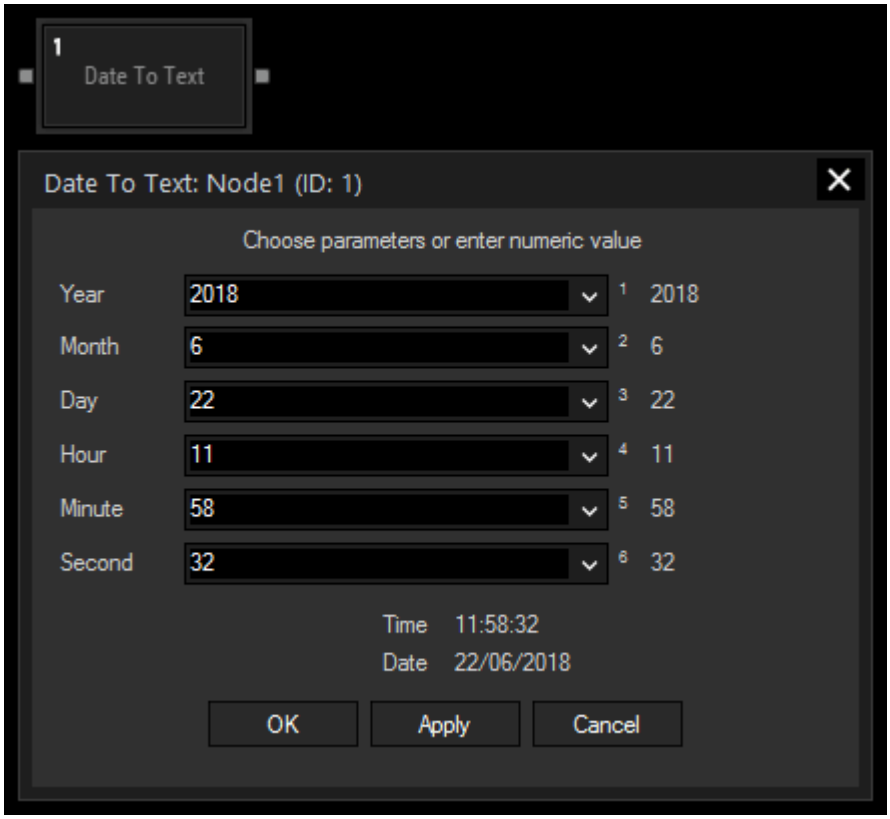
The Node generates the following output:

- Output Value: source text without cropped parts.

7.4.5.25.7 Date To Text Filter

Converts numeric input values to date string.

The node can be found under Nodes > Filter Nodes > Text > Date To Text



Node Properties

Year:
Choose an input node from the drop-down or enter a numeric value.

Month:
Choose an input node from the drop-down or enter a numeric value.

Day:
Choose an input node from the drop-down or enter a numeric value.

Hour:
Choose an input node from the drop-down or enter a numeric value.

Minute:
Choose an input node from the drop-down or enter a numeric value.

Second:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

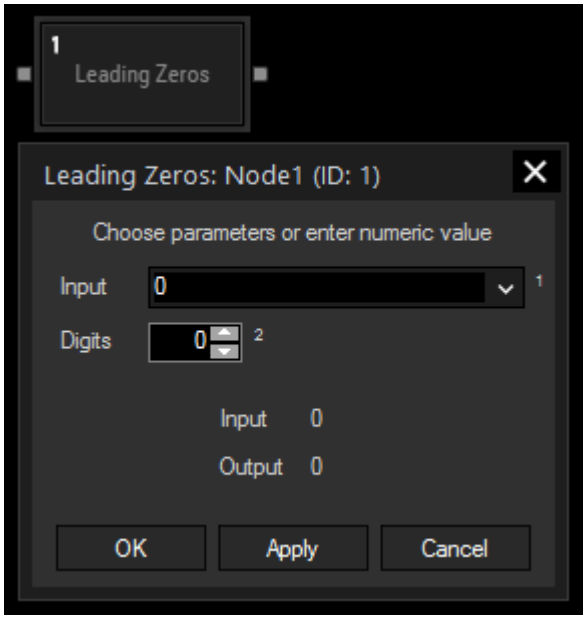
The Node generates the following output:

- Time Text Output
- Date Text Output

7.4.5.25.8 Leading Zeros Filter

Generates Leading Zeros for Text. This is especially useful for scoreboard counts.

The node can be found under Nodes > Filter Nodes > Text > Leading Zeros



Node Properties

Input:
Choose an input node from the drop-down or enter a numeric value.

Digits:
Enter the amount of digits the filter should generate in front of the input.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

- Leading Zeros Number

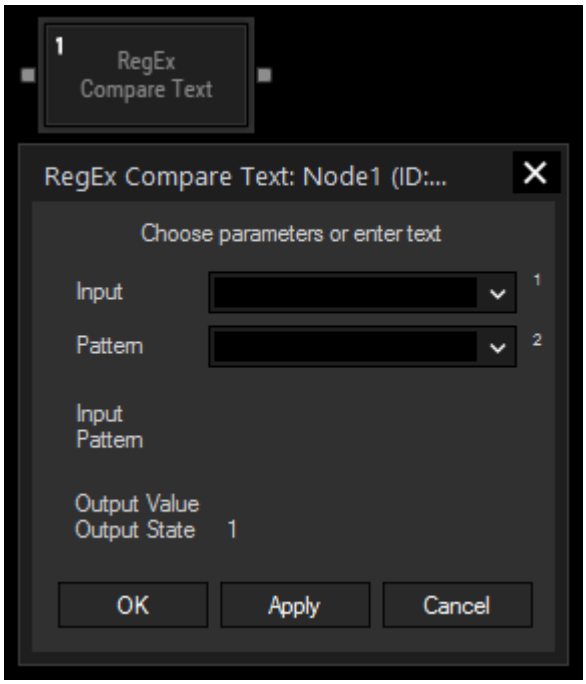
7.4.5.25.9 RegEx Compare Filter

The RegEx Text Filter node extends the "Find" feature known from common text editor software with RegEx expressions. A Regular Expression (short: RegEx) is a pattern that describes a certain combination / amount of text including word, number and (non-) printable characters. By supporting character classes (wild cards), grouping, back-references, etc. the RegEx engine allows very flexible and efficient text processing.

There are different RegEx flavors i.e. implementations in various programming languages. Widget Designer uses the RegEx the .NET framework library.

Note: The Filter node "RegEx Text [1160]" also searches for text that matches the used RegEx, but in addition to the "RegExCompare" node it can also replace it. In the chapter for "RegEx Text [1160]" node you will find a few examples how regular expressions work. For further information and complete description of supported character classes, operators and constructs, please visit:
https://msdn.microsoft.com/en-us/library/az24scfc%28v=vs.110%29.aspx

The node can be found under Nodes > Filter Nodes > Text > RegEx Compare



Node properties

Input:
Choose an input node from the drop-down or enter a text.

Pattern:
Enter a regular expression describing the text you like to search for.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

- Output
- State

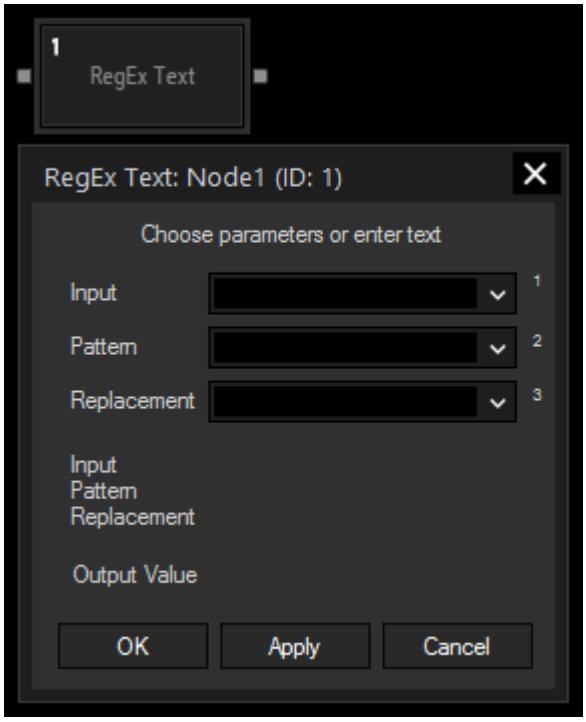
7.4.5.25.10 RegEx Text Filter

The RegEx Text Filter node extends the "Find and Replace" feature known from common text editor software with RegEx expressions. A Regular Expression (short: RegEx) is a pattern that describes a certain combination / amount of text including word, number and (non-) printable characters. By supporting character classes (wild cards), grouping, back-references, etc. the RegEx engine allows very flexible and efficient text processing.
There are different RegEx flavors i.e. implementations in various programming languages. Widget Designer uses the RegEx the .NET framework library.

Below you will find a few examples how regular expressions work. For further information and complete description of supported character classes, operators and constructs, please visit:
https://msdn.microsoft.com/en-us/library/az24scfc%28v=vs.110%29.aspx

Note: The Filter node "RegEx Compare 1158" also searches for text that matches the used RegEx but does not replace it, it simply returns a TRUE or FALSE.

The node can be found under Nodes > Filter Nodes > Text > RegEx Text



Node properties

Input:
Choose an input node from the drop-down or enter a text.

Pattern:
Enter a regular expression describing the text you like to search for.

Replacement:
Enter a regular expression that should replace a found pattern.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

- Output

Examples for Regular Expressions

These examples are based on the input string:
A cat wears 99 hats - What? - Yes, yes!

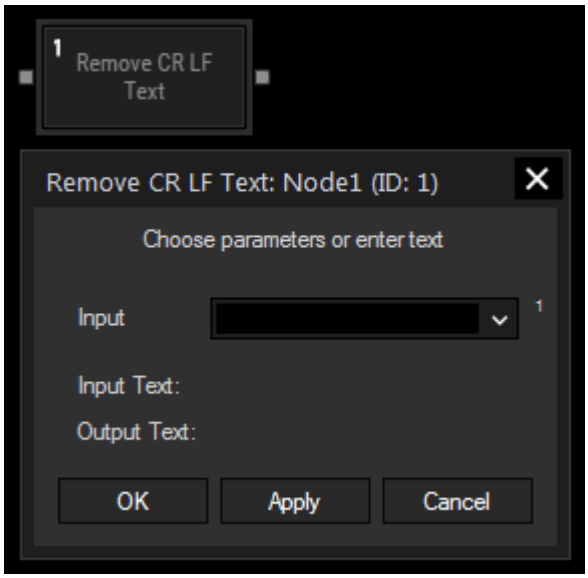
Pattern	Replace	Output	Meaning
[a-z]+	1	A 1 1 99 1 - W1? - Y1, 1!	Searches case-sensitive any letter from character group "a-z" that occurs at least once. Replaces with digit "1"
(cat)[^\?]*	$1	A cat? - Yes, yes!	Searches for a string that starts with "cat" followed by any character that is not a "?" occurring zero or more times. By putting "cat" in round brackets, it becomes group no.1. Replaces with first group.
[^a-zA-Z0-9]		A cat wears 22 hats What Yes yes	Searches for any character that is not a-z nor A-Z nor 0-9 nor a space. Replaces with nothing (i.e. it's erased). The mentioned characters can be described shorter with the "character class" word character "\w". Hence, we can search alternatively for the negated group.
[^\w]			
\W			
(?i)yes	yes	A cat wears 22 hats - What? - yes, yes!	Searches case-insensitive for the word "yes". Replaces with "yes".

For further information and complete description please visit:
https://msdn.microsoft.com/en-us/library/az24scfc%28v=vs.110%29.aspx

7.4.5.25.11 Remove CR LF Text Filter

This Filter node removes carriage returns and line feeds (CR LF) inside the text input. Please enter \r for CR and \n for LF

The node can be found under Nodes > Filter Nodes > Text > Remove CR LF Text



Node properties

Input:
Choose an input node from list or enter text.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

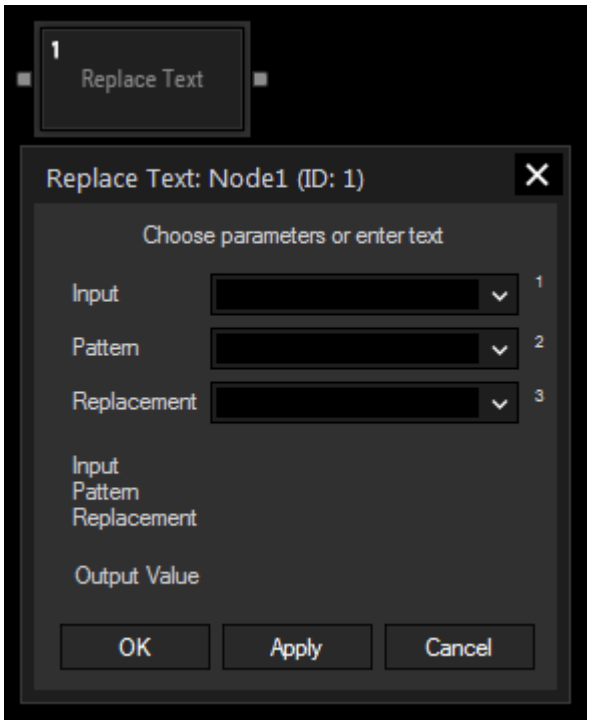
The Node generates the following output:

- Output Text: source text without carriage returns and line feeds.

7.4.5.25.12 Replace Text Filter

The Replace Text filter node finds a selectable pattern of characters in an input string and replaces all found parts with another specified character string.

The node can be found under Nodes > Filter Nodes > Text > Replace Text



Node properties

Input:
Choose an input node from the drop-down or enter a text.

Pattern:
Choose an input node from the drop-down or enter a text for the pattern that is to be detected.

Replacement:
Choose an input node from the drop-down or enter a text for the character string to replace the pattern.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

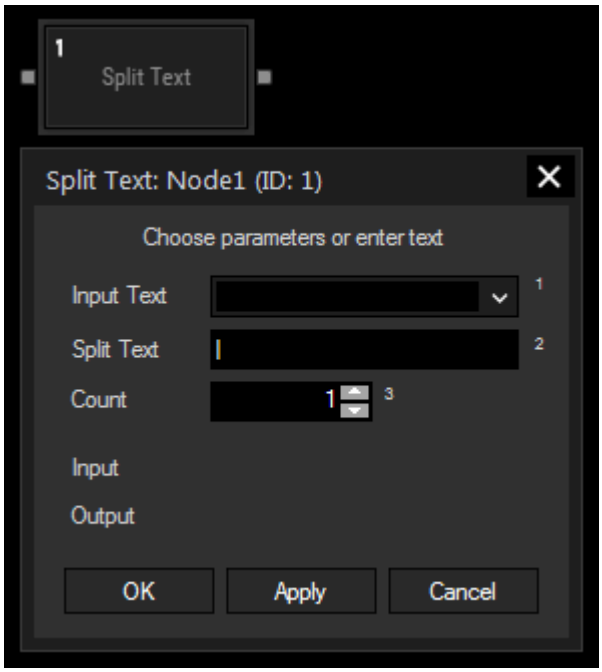
The Node generates the following output:

- Output Value: source text with the replaced parts

7.4.5.25.13 Split Text Filter

The Split Text filter node is able to split an input character string and output each section independently from each other. A definable split character marks the pieces to be split from each other, this character will be removed after splitting.

The node can be found under Nodes > Filter Nodes > Text > Split Text



Node properties

Input Text:
Choose an input node from the drop-down or enter a text

Split Text:
Enter here the split character that marks the different sections

Count:
Enter here the amount of outputs that should be generated with splitting. Please not that it will not be automatically detected how many split parts are generated in this node.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

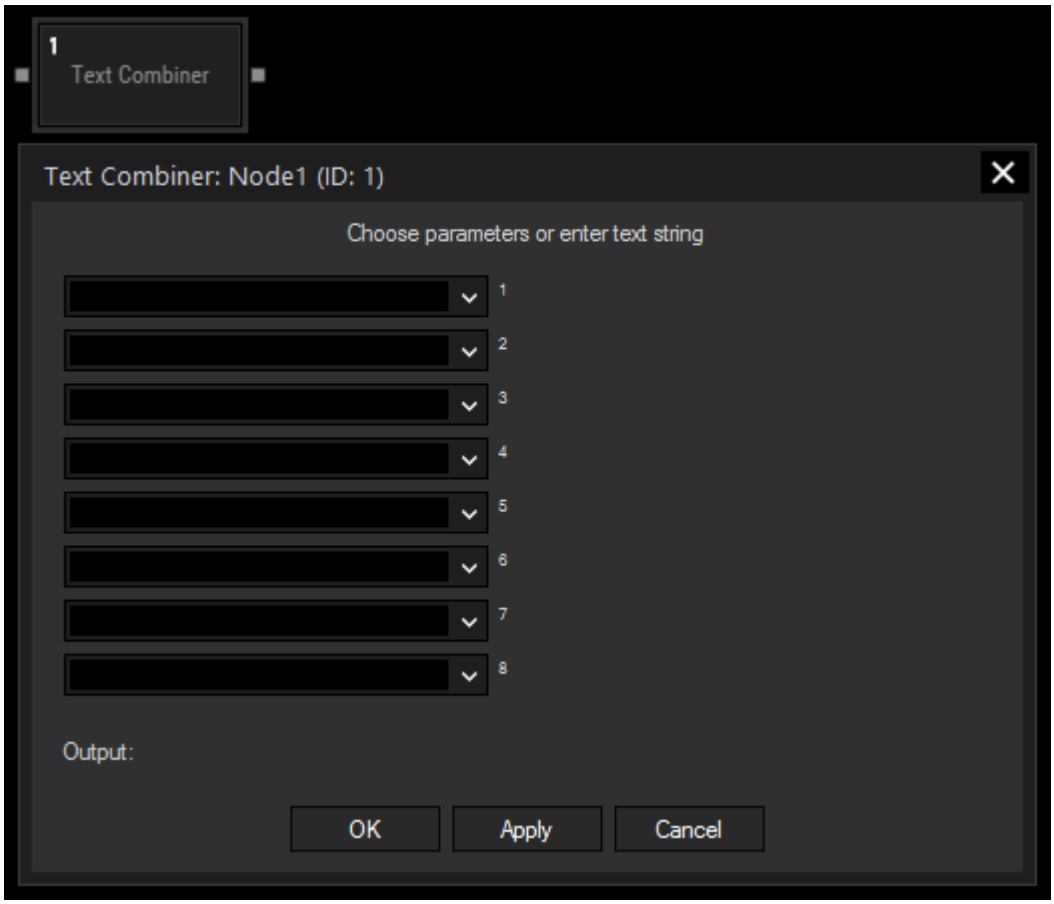
The Node generates the following output:

- Output 1: split section 1
- Output 2: split section 2
- …
(The number of outputs depends on the Count parameter)

7.4.5.25.14 Text Combiner Filter

Combines up to eight text input sources or entered text to one string.
The Text Combine Filter now supports single entry for Line Feed [LF].

The node can be found under Nodes > Filter Nodes > Text > Text Combiner



Node Properties

Inputs (8x):
Choose an input node from the drop-down or enter a text.

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

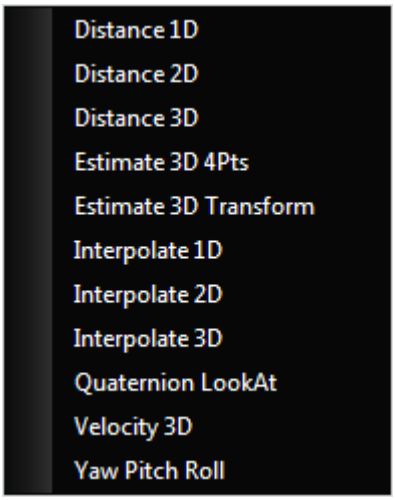
Node output values

The Node generates the following output:

- Combined String Output

7.4.5.26 Vector Filter Nodes

Vector filter nodes provide useful tools for tracking and for calculations around movements. Predicting values for smoothly tracked movements as well as angle, distance and position calculations simplify the workflow for all kinds of tracking environments.

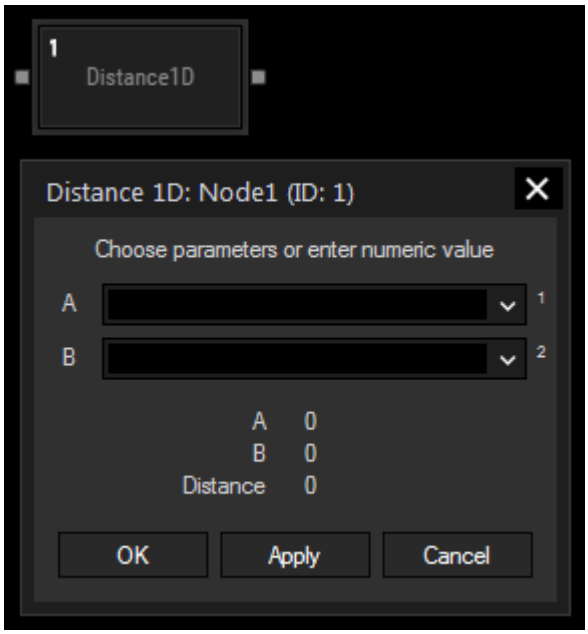


Distance 1D 1167
Distance 2D 1168
Distance 3D 1169
Estimate 3D 4Pts
Estimate 3D Transform
Interpolate 1D 1170
Interpolate 2D 1171
Interpolate 3D 1172
Quaternion LookAt 1174
Velocity 3D 1176
Yaw Pitch Roll

7.4.5.26.1 Distance 1D Filter

The Distance 1D filter node enables you to easily measure the distance between two points A and B in a one dimensional coordinate system.

This node can be found under Nodes > Filter > Vector > Distance 1D



Node properties

A and B:
Choose an input node from the drop-down or enter a numeric value for the two points to be tracked.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

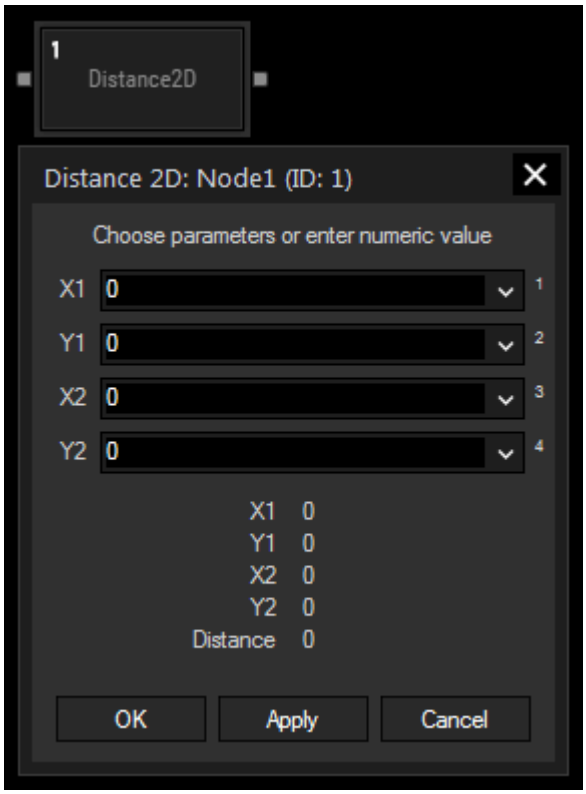
The Node generates the following output:

- Result (distance)
- UnitsPerSec (velocity with which the points are moving towards each other / away from each other)

7.4.5.26.2 Distance 2D Filter

The Distance 2D filter node enables you to easily measure the distance between two points 1 and 2 in a two dimensional coordinate system.

This node can be found under Nodes > Filter > Vector > Distance 2D



Node properties

X1 and Y1:
Choose an input node from the drop-down or enter a numeric value for the X and Y coordinates of the first point.

X2 and Y2:
Choose an input node from the drop-down or enter a numeric value for the X and Y coordinates of the second point.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

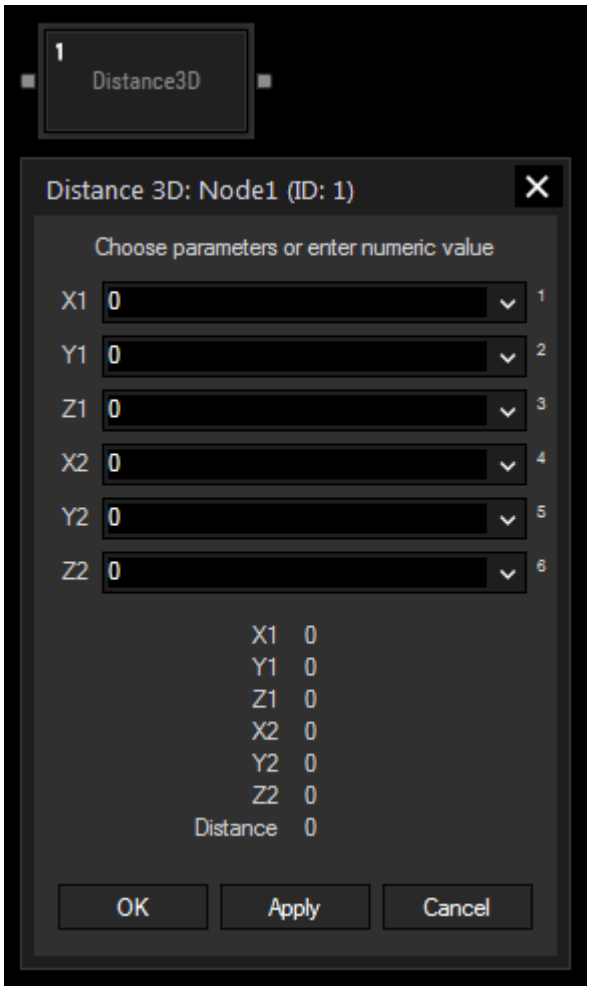
The Node generates the following output:

- Result (distance)
- UnitsPerSec (velocity with which the points are moving towards each other / away from each other)

7.4.5.26.3 Distance 3D Filter

The Distance 3D filter node enables you to easily measure the distance between two points 1 and 2 in a three dimensional coordinate system.

This node can be found under Nodes > Filter > Vector > Distance 3D



Node properties

X1, Y1 and Z1:
Choose an input node from the drop-down or enter a numeric value for the X, Y and Z coordinates of the first point.

X2, Y2 and Z2:
Choose an input node from the drop-down or enter a numeric value for the X, Y and Z coordinates of the second point.

Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

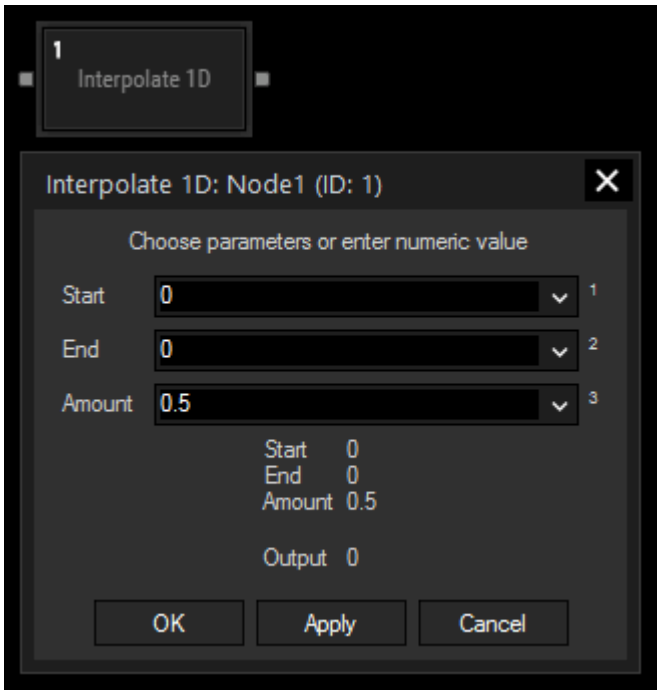
The Node generates the following output:

- Result (distance)
- UnitsPerSec (velocity with which the points are moving towards each other / away from each other)

7.4.5.26.4 Interpolate 1D Filter

The Interpolate 1D filter node interpolates a point between a defined start and end point in a one dimensional coordinate system and outputs its position. The percentage of how close the interpolated point is supposed to be to the start or end point can be adjusted with the "Amount" parameter.

This node can be found under Nodes > Filter > Vector > Distance 1D



Node properties

Start:
Choose an input node from the drop-down or enter a numeric value for the position value of the start point.

End:
Choose an input node from the drop-down or enter a numeric value for the position value of the end point.

Amount:
Enter here the percentage how close the interpolated point should be to the start or end point. 0.5 means that the point is exactly in the middle between start and end. A value approaching 0 puts the interpolated point closer to the start, a value approaching 1 closer to the end point.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

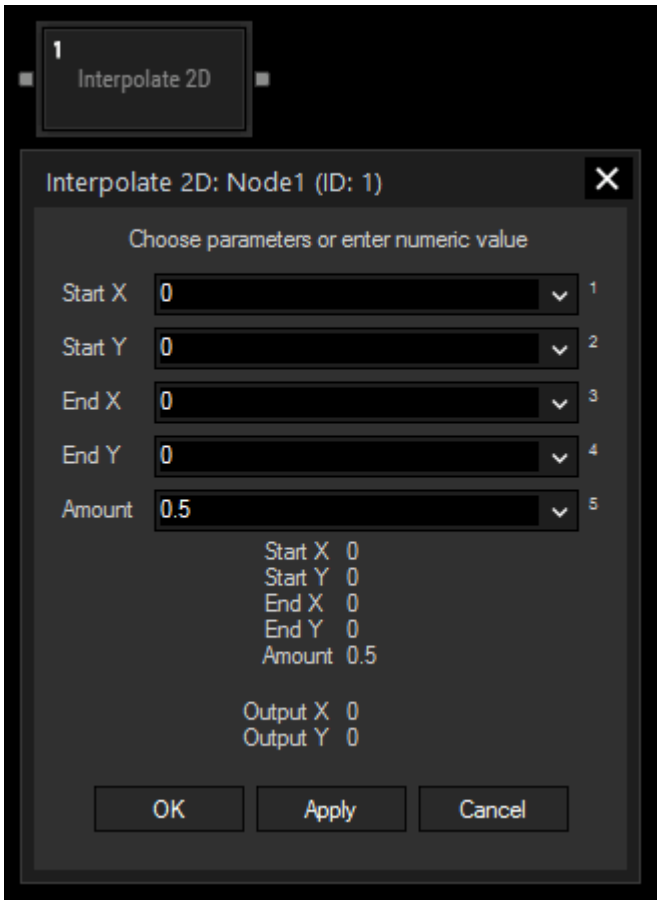
The Node generates the following output:

- Output (position of the interpolated point)

7.4.5.26.5 Interpolate 2D Filter

The Interpolate 2D filter node interpolates a point between a defined start and end point in a two dimensional coordinate system and outputs its X and Y coordinates. The percentage of how close the interpolated point is supposed to be to the start or end point can be adjusted with the "Amount" parameter.

This node can be found under Nodes > Filter > Vector > Distance 2D



Node properties

Start X and Y:
Choose an input node from the drop-down or enter a numeric value for the X and Y coordinates of the start point.

End X and Y:
Choose an input node from the drop-down or enter a numeric value for the X and Y coordinates of the end point.

Amount:
Enter here the percentage how close the interpolated point should be to the start or end point. 0.5 means that the point is exactly in the middle between start and end. A value approaching 0 puts the interpolated point closer to the start, a value approaching 1 closer to the end point.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

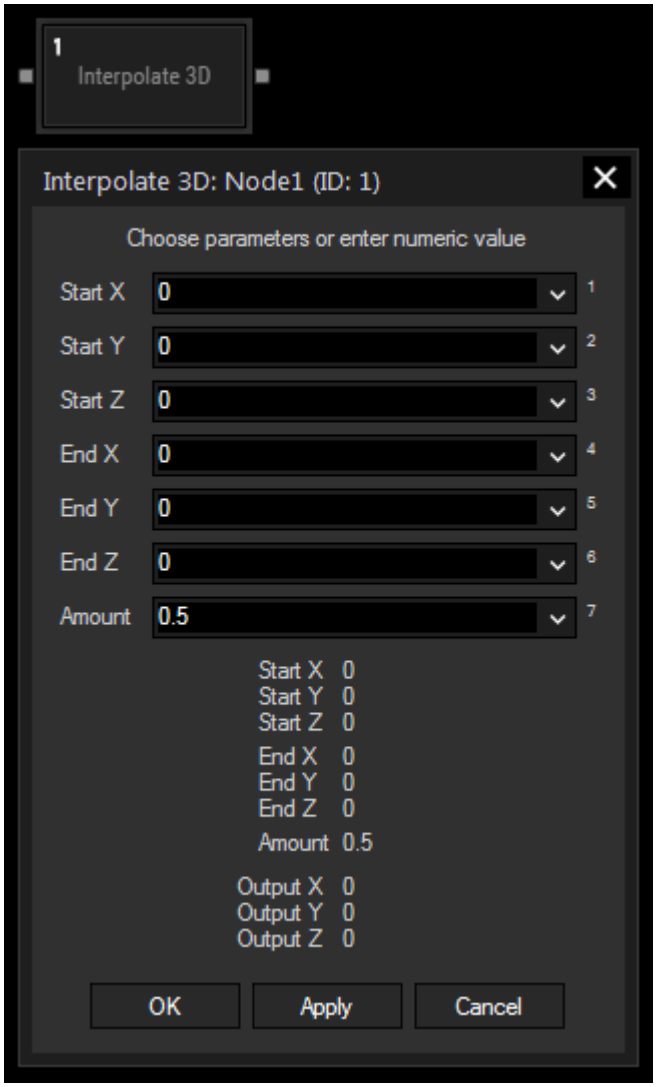
The Node generates the following output:

- Output X (X coordinate of the interpolated point)
- Output Y (Y coordinate of the interpolated point)

7.4.5.26.6 Interpolate 3D Filter

The Interpolate 3D filter node interpolates a point between a defined start and end point in a three dimensional coordinate system and outputs its X, Y and Z coordinates. The percentage of how close the interpolated point is supposed to be to the start or end point can be adjusted with the "Amount" parameter.

This node can be found under Nodes > Filter > Vector > Distance 3D



Node properties

Start X, Y and Z:
Choose an input node from the drop-down or enter a numeric value for the X, Y and Z coordinates of the start point.

End X, Y and Z:
Choose an input node from the drop-down or enter a numeric value for the X, Y and Z coordinates of the end point.

Amount:
Enter here the percentage how close the interpolated point should be to the start or end point. 0.5 means that the point is exactly in the middle between start and end. A value approaching 0 puts the interpolated point closer to the start, a value approaching 1 closer to the end point.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

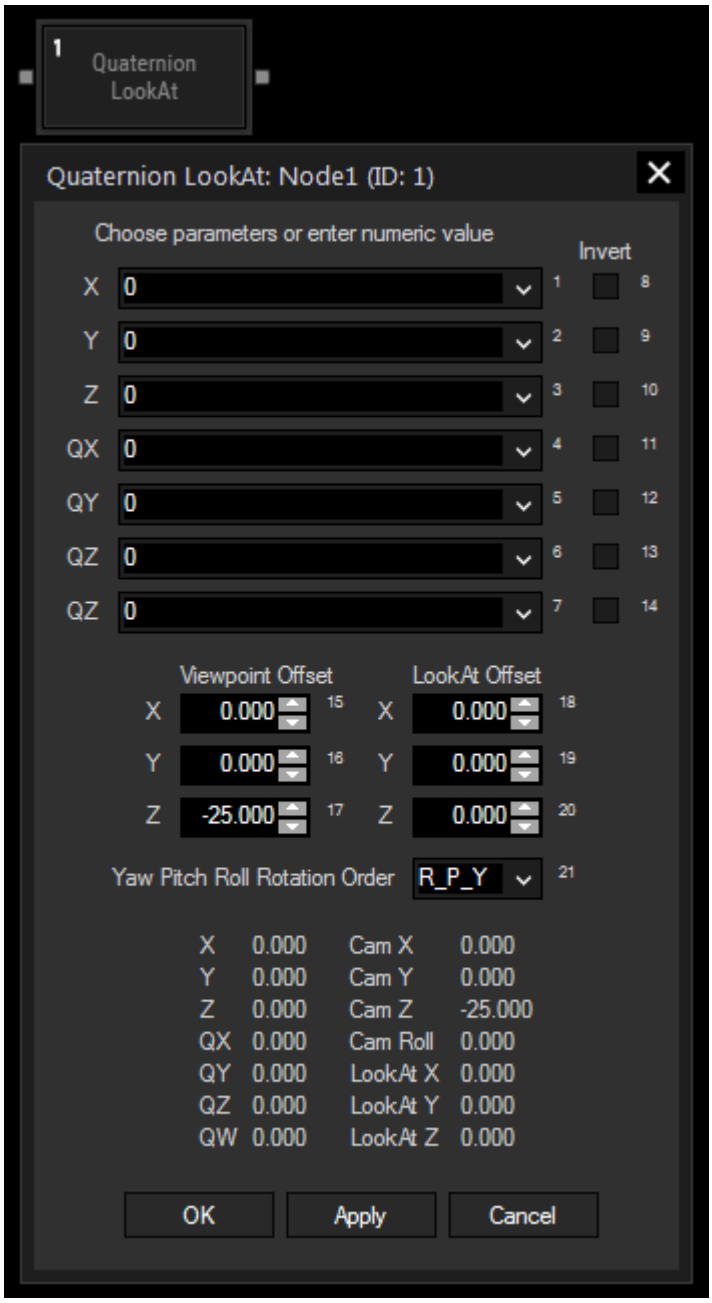
- Output X (X coordinate of the interpolated point)
- Output Y (Y coordinate of the interpolated point)
- Output Z (Z coordinate of the interpolated point)

7.4.5.26.7 Quaternion LookAt Filter

This node is designed for calculating Pandoras Box Camera position and rotation, based on quaternion values. It is especially useful for situations where transmitting Euler angles can lead to a gimbal lock of the Camera.

Quaternions are also easier to handle and more efficient in processing multidimensional rotating systems. If you have a device delivering quaternion values, like e.g. many industrial robots, you can easily track the Camera position with this node.

This node can be found under Nodes > Filter > Vector > Quaternion LookAt



Node properties

X, Y and Z:
Choose an input node from the drop-down or enter a numeric value for the X-, Y- and Z-coordinates of the point (the real world camera position) to be tracked.

QX, QY, QZ and QW:
Choose an input node from the drop-down or enter a numeric value for the X-, Y-, Z- and W-factors of the unit quaternion specifying your system. The factors follow the equation describing the unit quaternion q:

q = w + x*i + y*j + z*k

Invert:
Check this box for one or more of the previous parameters for inverting it

Viewpoint Offset:
Here you can enter an offset for the X-, Y- and Z-coordinates of your PB Camera position that will be included in the calculation

LookAt Offset:
Here you can enter an offset for the X-, Y- and Z-coordinates of your PB Camera target position that will be included in the calculation

Yaw Pitch Roll Rotation Order:
Select here a different order for your axis calculation if the coordinate system of the input device does not match the PB coordinate system

Node control

With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

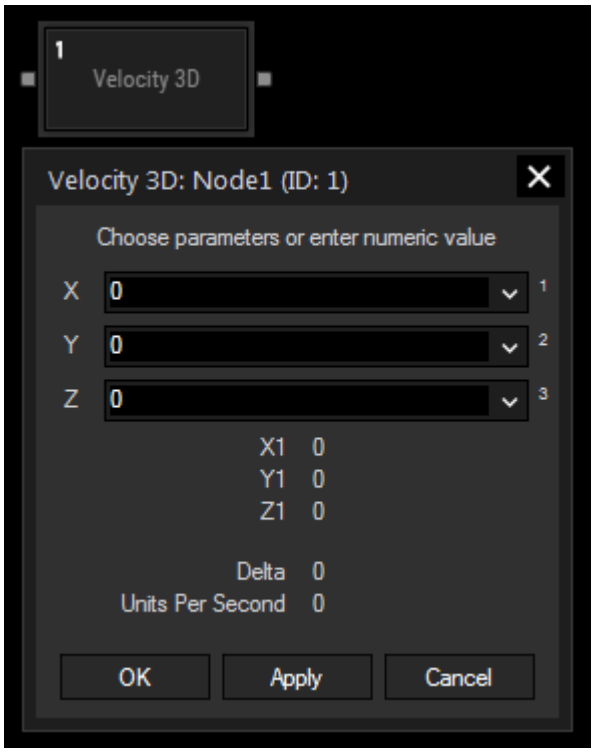
The Node generates the following output:

- Cam X
- Cam Y
- Cam Z
- Cam Roll
- LookAt X
- LookAt Y
- LookAt Z

7.4.5.26.8 Velocity 3D Filter

The Velocity 3D node enables you to calculate the current velocity of a point moving in a 3D space as well as the delta of its crossed distance.

This node can be found under Nodes > Filter > Vector > Velocity 3D



Node properties

X, Y and Z:
Choose an input node from the drop-down or enter a numeric value for the X-, Y- and Z-coordinates of the tracked point.

If you want to track a two or one dimensional movement, select only the respective axis and leave the others on "0"

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

Node output values

The Node generates the following output:

- Units per second (velocity)
- Delta

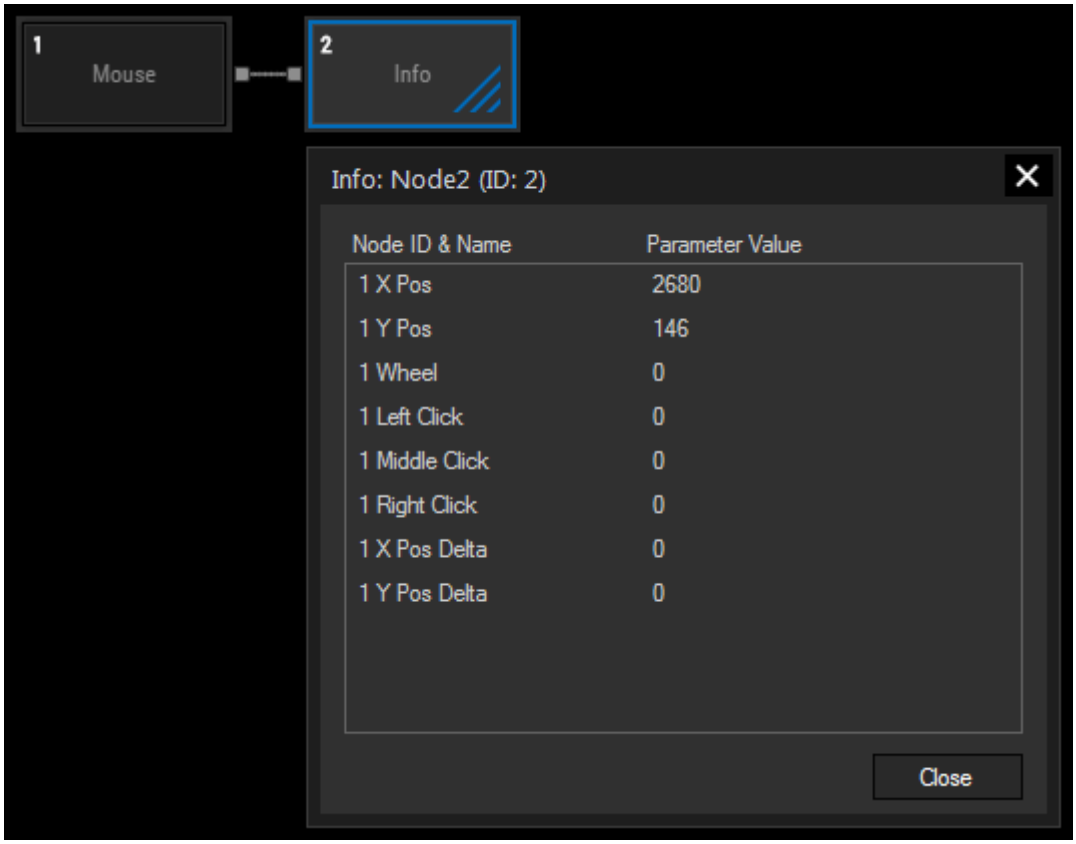
7.4.6 Output Nodes

Output nodes are used to send values out that are provided by other nodes. Please see the introductory chapter if you like to learn more about other node types or how to create and work with nodes [937] in general.

The following sub chapters describe the various Output nodes in alphabetic order.

7.4.6.1 Info Output

The Info output node allows you to display and monitor all values delivered by the connected input or filter nodes. The example picture below shows an Info node displaying all outputted values of a preceding Mouse input node [1016].



Node Properties

This node simply displays all incoming data, even when several nodes are connected to it.

Please note that several filter nodes which buffer data, such as e.g. Delta, Prediction or Estimate 3D filter nodes, will not display data if connected solely to an Info output node. The Info node itself does not actively pull data from the preceding nodes, therefore you will need an additional "real" output node (e.g. a Label output node) to trigger data flow in the node chain.

The Info output node does not appear in the Output Node Count [1510], and, as it only monitors the data flows rather than activating them, does not influence the node system's performance.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

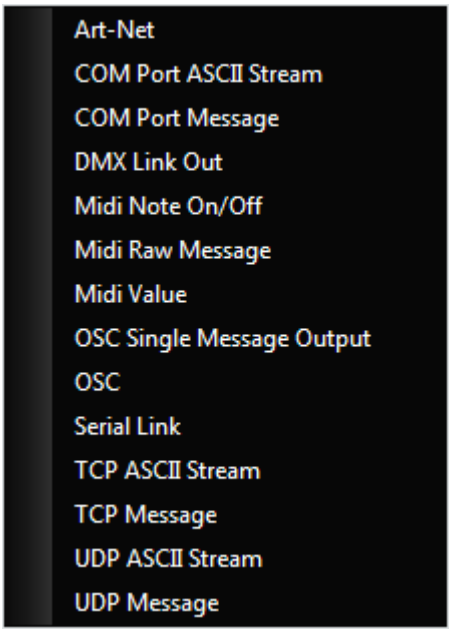
Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node2.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node2.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2 Connections Output

The Connection output nodes can send data via different kinds of output protocols, mostly over network. All of the here listed connections can be set in the Connection Manager [1258].

Please note that OSC is a UDP based protocols and thus needs a UDP connection [1267] set up.

The available nodes are:

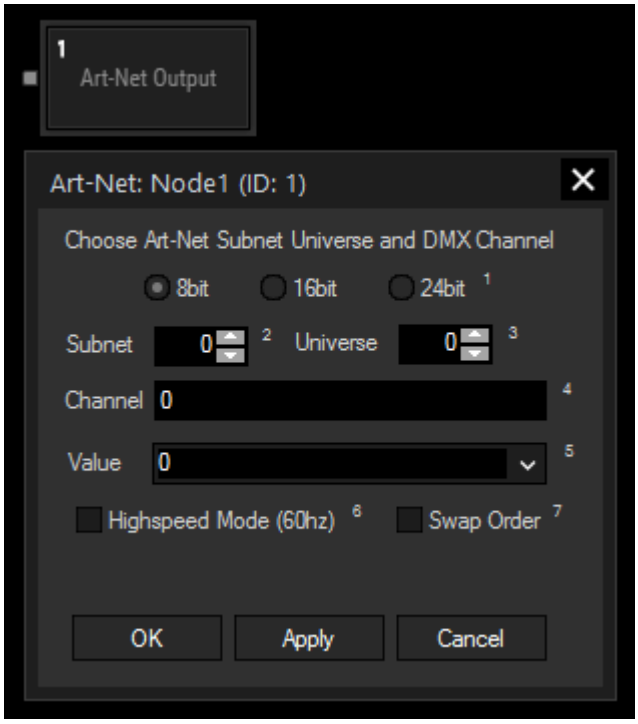


Art-Net [1179]
COM Port ASCII Stream [1180]
COM Port Message [1181]
DMX Link Out [1182]
Midi Note On/Off [1183]
Midi Raw Message [1184]
Midi Value [1185]
OSC Single Message Output [1186]
OSC [1186]
Serial Link [1189]
TCP ASCII Stream [1191]
TCP Message [1192]
UDP ASCII Stream [1193]
UDP Message [1194]

7.4.6.2.1 Art-Net Output

The Art-Net output node allows to assign incoming source values to multiple 8, 16 or 24bit DMX output Channels on a given Art-Net Subnet and Universe.
Simply type in all DMX start addresses and use a white space in between them i.e.:1 2 123.
The Art-Net output node updates Art-Net only on change of the input value. Please enable the Art-Net output first in the Connection Manager [1259]. Keep in mind, that a Universe is sent even if all Art-Net output nodes were deleted. If you wish to stop sending specific Universes to the network you can open the "Art-Net Universe List" from the Connection Manager and remove a Universe from the output list by right-clicking on the desired Subnet and Universe. Right-click and choose "Refresh List" e.g. if you started sending a Universe again.

This node can be found under Nodes > Output > Connections > Art-Net



Node Properties

Check one of the radio buttons to give out either **8bit, 16bit or 24bit** DMX values.

Art-Net Subnet, Universe:
Set here the Art-Net Subnet and Universe.

Channel:
Set here the DMX channel. To send Art-Net to multiple channels, use a white space in between the channel IDs, e.g. 1 2 509.

Value:
Choose an input node from the drop-down or enter a numeric value.

Highspeed Mode (60 Hz):
Enable this option to use the Highspeed Mode that does a refresh rate of 60 Hz (Default is off)

Please note:
The Highspeed Mode is not supported by GrandMA consoles.

Swap Order:
Check this box to swap the order of bytes when you are using 16 or 24bit mode.

Node control

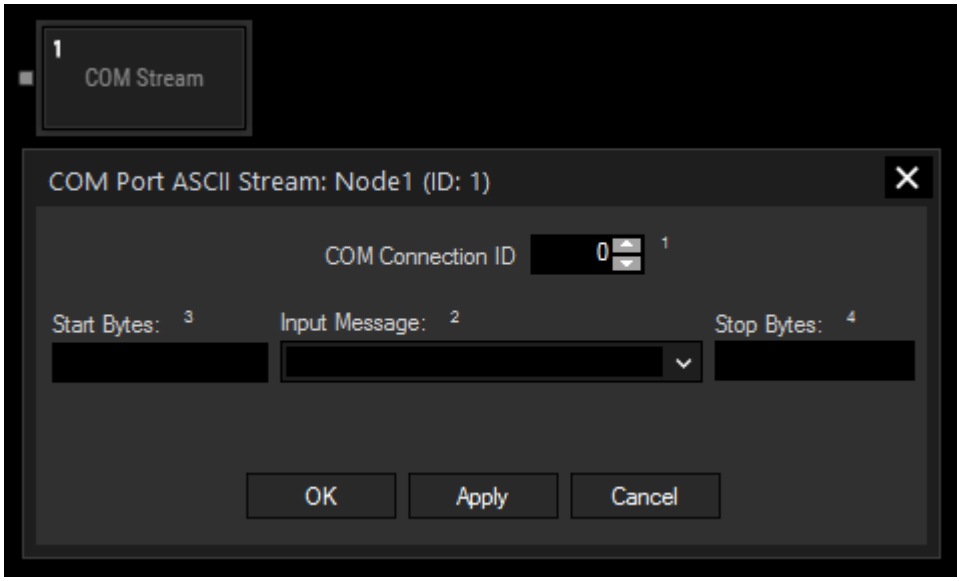
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.2 COM Port ASCII Stream Output

The Com Port ASCII Stream output node allows sending ASCII values over the local COM Port connection.
Set up the COM Port Connection in the Connection Manager [1258].

This node can be found under Nodes > Output > Connections > COM Stream



Node Properties

COM Connection ID:
Enter the COM Port Connection ID.

Start and Stop Bytes:
Enter Start and Stop Bytes that should frame the message.
Enter these values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages [944].

Input Message:
Please select the input node source that should be given out via this COM Port output node from the drop-down.

Node control

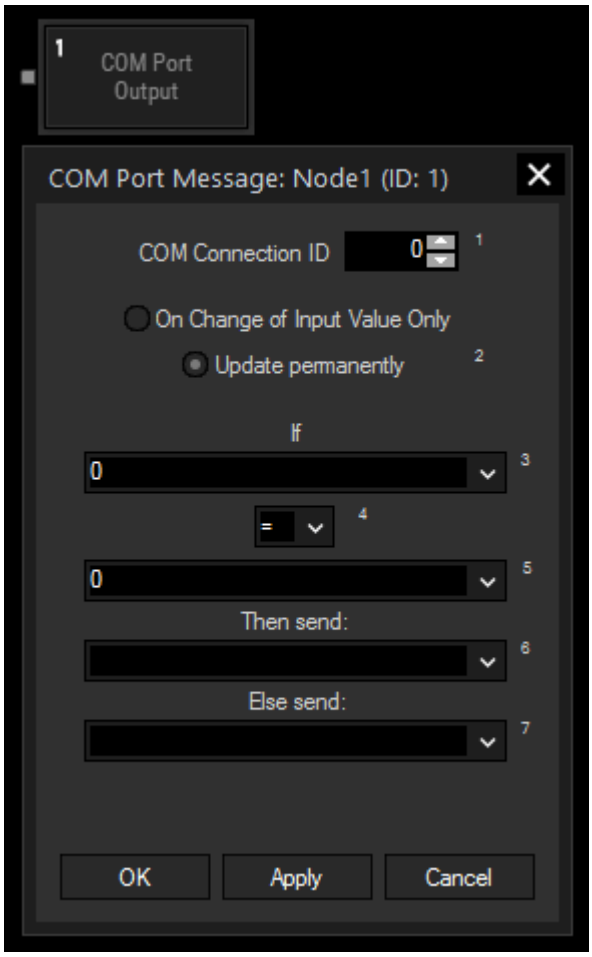
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.3 COM Port Output

The Com Port Message value acts like an If node [1106]. The incoming data can be compared and depending on the true or false result a dedicated ASCII or Byte packet can be sent to the local COM Port. Set up the COM connection in the Connection Manager [1258].

This node can be found under Nodes > Output > Connections > Com Port



Node Properties

COM Connection ID:
Enter the COM Port Connection ID.

Update:
Choose if the output should be updated either "On Change of Input Value Only" or permanently.

If:
Choose the first input node from the drop-down or enter a numeric value. This input will be compared to the second input.
Choose the Operator.
Choose the second input node from the drop-down or enter a numeric value.

Then send:
If the operation is True, this input node value or entered numeric or text value will be given out as Output Value.

Else send:
If the operation is False, this input node value or entered numeric or text value will be given out as Output Value.

Node control

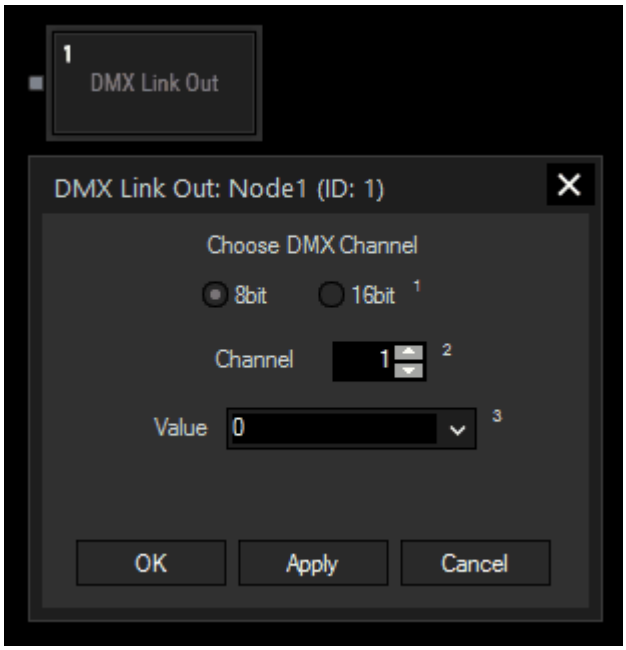
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.4 DMX Link Output

The DMX Link output node allows sending out DMX data via the DMX Link Out interface [1999].

This node can be found under Nodes > Output > Connections > DMX Link



Node Properties

DMX Channel:
Choose the channel and if it is an **8bit or 16bit** value.

Value:
Choose an input node from the drop-down or enter a numeric value that should be given out via this DMX Link Output Node.

Node control

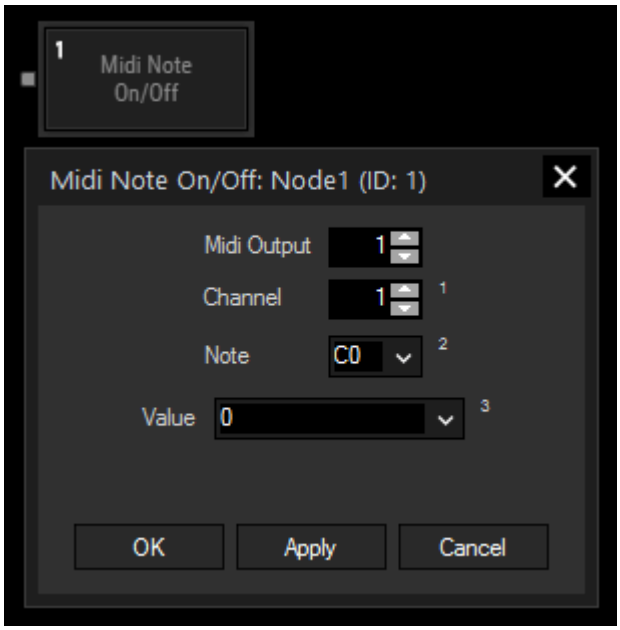
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.5 Midi Note On/Off Output

The Midi Note On/Off node allows to send Midi Note on/off values to a given note on a given Midi Channel.
The Midi connection needs to be enabled in the Connection Manager[1258].

This node can be found under Nodes > Output > Connections > Midi Note On/Off



Node Properties

Midi Output:
As several Midi connections can be established at once, enter here the ID of the respective Midi output device.

Channel:
Please choose the Midi Channel.

Note:
Please choose the Midi Note.

Value:
Choose an input node from the drop-down or enter a numeric value. It will be sent as Midi Note on/off values.

Node control

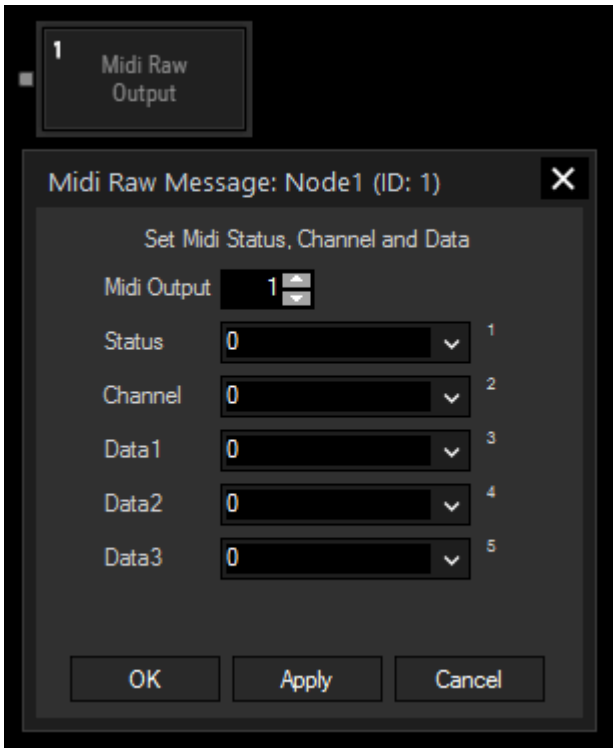
With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.6 Midi Raw Message Output

The Midi Raw Message node allows mapping individual input source values to all 5 midi control bytes.
The Midi connection needs to be enabled in the Connection Manager[1258].

This node can be found under Nodes > Output > Connections > Midi Raw Message



Node Properties

Status:
Choose an input source from the list or enter a numeric value.

Channel:
Choose an input source from the list or enter a numeric value.

Data1:
Choose an input source from the list or enter a numeric value.

Data2:
Choose an input source from the list or enter a numeric value.

Data3:
Choose an input source from the list or enter a numeric value.

Node control

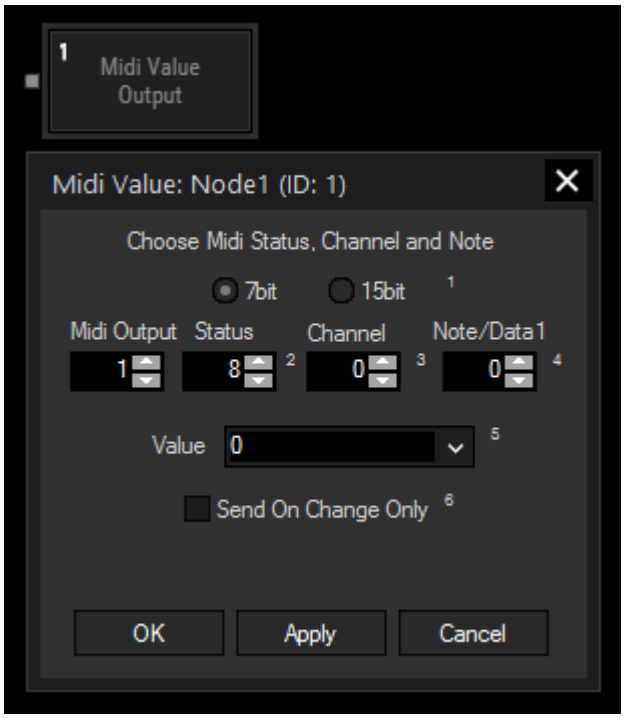
With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.7 Midi Value Output

The Midi Value output node allows sending a 7 or 15bit value on a given Midi Channel.
The Midi connection needs to be enabled in the Connection Manager [1258].

This node can be found under Nodes > Output > Connections > Midi Value



Node Properties

Please choose if you want to send a **7 or 15bit value**.

Status:
Enter the Midi Status.

Channel:
Enter the Midi Channel.

Note/Data1:
Enter the Midi Note / Data1.

Value:
Choose the Input Source from the list to send on this Midi Channel.

Check the box **Send on change Only** if you do not want to constantly send out the current input value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

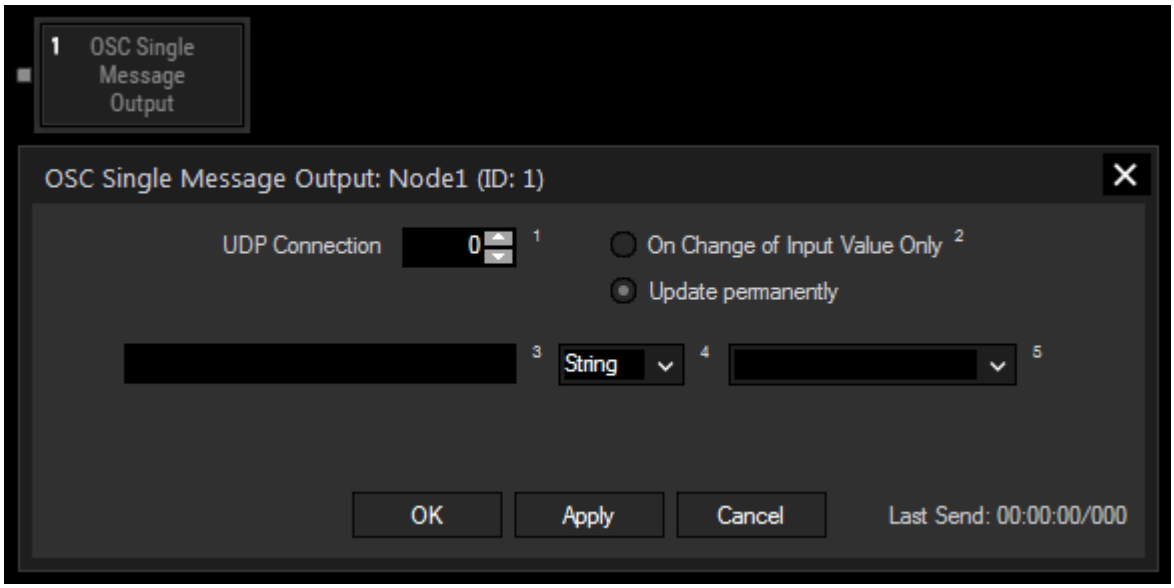
Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.8 OSC Single Message Output

The OSC Single Message Output node enables you to send data via the Open Sound Control Protocol. Unlike the OSC Output node [1186] which allows sending bundle messages, this node only processes single OSC messages.

A UDP Broadcast Input Connection is required, see Connection Manager [1258].

This node can be found under Nodes > Output > Connections > OSC Single Message Output



Node Properties

UDP Connection:
Enter here the ID of your UDP connection.

On Change of Input Value Only / Update permanently:
Choose here whether the current value is supposed to be sent permanently or only once when it changes.

Address Field:
Enter here the address of your OSC message, e.g. /synthesizer/filter/abc

Data Type:
Select the data type of your message (integer, float or string) from the drop-down.

Message:
Choose an input node from the drop-down or enter a numeric value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.9 OSC Output

The OSC Single Message Output node enables you to send data via the Open Sound Control Protocol. Unlike the OSC Single Message Output node [961] which only allows sending messages with one value, this node can processes bundled OSC messages with up to 17 values.

A UDP Broadcast Input Connection is required, see Connection Manager [1258].

This node can be found under Nodes > Output > Connections > OSC Output

Node Properties

UDP Connection:
Enter here the ID of your UDP connection.

On Change of Input Value Only / Update permanently:
Choose here whether the current value is supposed to be sent permanently or only once when it changes.

Message:
Choose an input node from the drop-down or enter a numeric value for each value that should be bundled in the message. You can transmit up to eight floats (i.e. floating point values) plus eight integers as well as one string value per message.

Address:
Enter here the address of your OSC message, e.g. /synthesizer/filter/abc

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

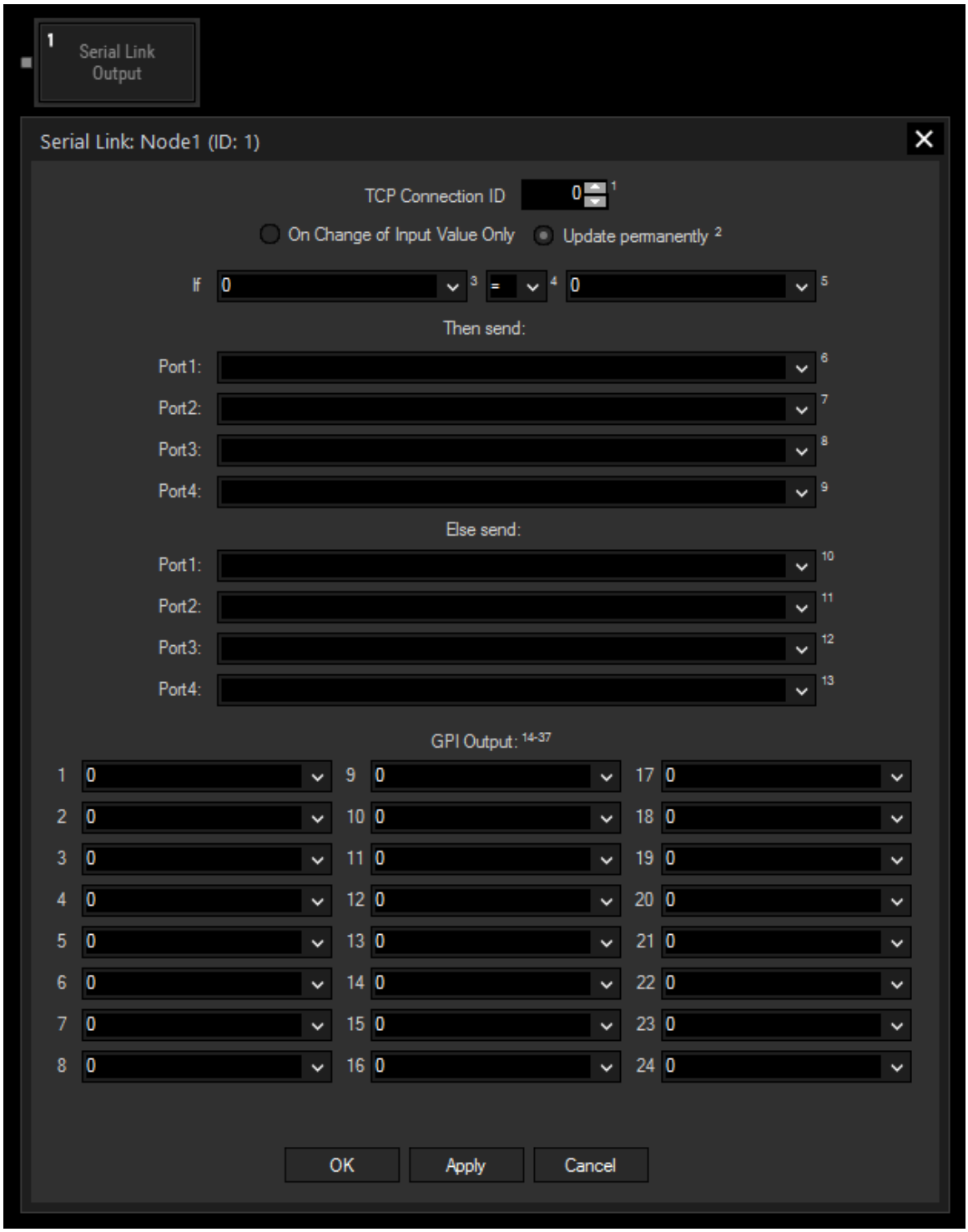
Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.10 Serial Link Output

The Serial Link output node connects to a Serial Link device via TCP. Incoming values can be compared to send ASCII or Byte packets to any of the four serial ports of the connected Serial Link [2044]. All 24 GPI contact closures can be mapped and assigned to input values as well 0 is off, 1 is contact closed.

A TCP Connection [1265] to the Serial Link has to be established before using this node.

This node can be found under Nodes > Output > Connections > Serial Link



Node Properties

TCP Connection ID:
Enter the TCP Connection ID. The TCP Connections can be found and setup in the Connection Manager [1258].

Update:
Choose if the output should be updated either **On Change of Input Value Only** or **permanently**. Please note that permanently updated values draw a lot of performance.

If:
Choose the first input node from the drop-down or enter a numeric value. This input will be compared to the second input.
Choose the Operator.
Choose the second input node from the drop-down or enter a numeric value.

Then send:
If the operation is True, the True script will be executed. You have the possibility to assign a different true send command for each Serial Link port. The topic Script Language [1511] explains commands and how to use them in more detail.
See here a list of all commands [1520].

Else send:
If the operation is False, the Else script will be executed. You have the possibility to assign a different else send command for each Serial Link port.

GPI Outputs:
All 24 GPI contact closures can be mapped and assigned to input values as well 0 is off, 1 is contact closed.

Node control

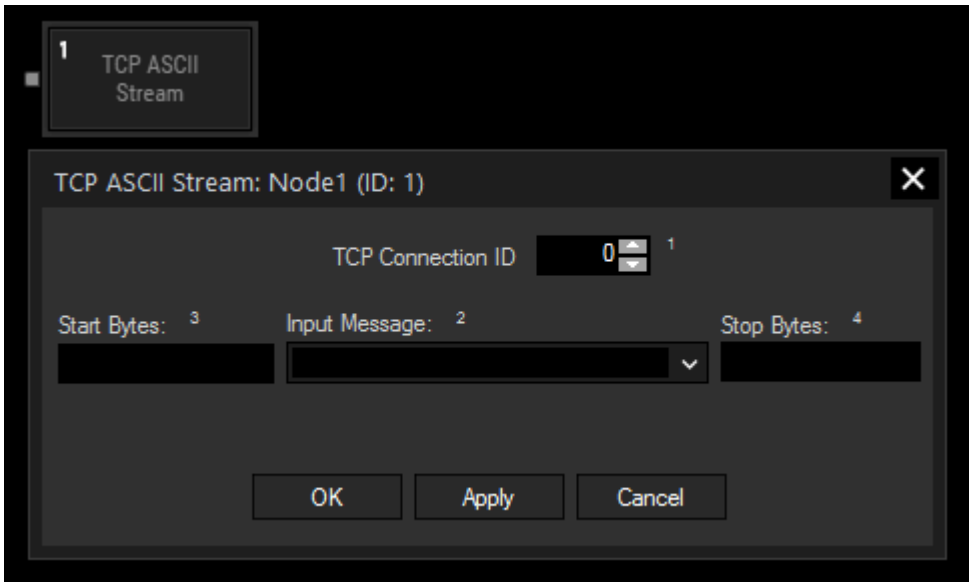
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.11 TCP ASCII Stream Output

The TCP ASCII Stream output node allows sending ASCII values over a TCP connection.
Set up the TCP Connection in the Connection Manager[1258].

This node can be found under Nodes > Output > Connections > TCP ASCII Stream



Node Properties

TCP Connection ID:
Enter the TCP Connection ID.

Start and Stop Bytes:
Enter Start and Stop Bytes that should frame the message.
Enter these values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages[944].

Input Message:
Please select the input node source that should be given out via this TCP output node from the drop-down.

Node control

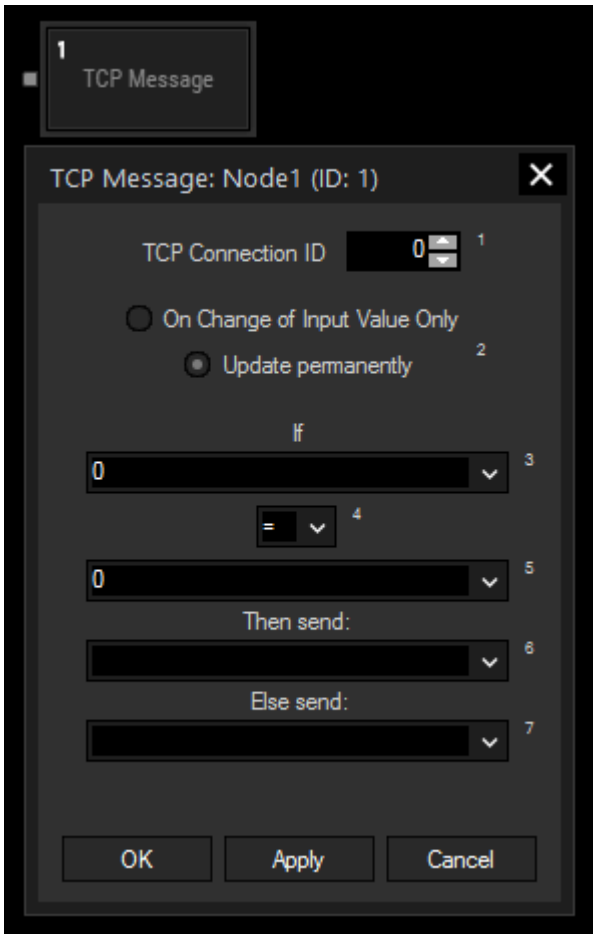
With Widget Designer version 6, all nodes support so called node commands[1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.12 TCP Message Output

The TCP Message value acts like an If node [1106]. The incoming data can be compared and depending on the true or false result a dedicated ASCII or Byte packet can be sent to the TCP connection setup by the its ID in the Connection Manager [1258].

This node can be found under Nodes > Output > Connections > TCP Message



Node Properties

TCP Connection ID:
Enter the TCP Connection ID.

Update:
Choose if the output should be updated either "On Change of Input Value Only" or permanently.

If:
Choose the first input node from the drop-down or enter a numeric value. This input will be compared to the second input.
Choose the Operator.
Choose the second input node from the drop-down or enter a numeric value.

Then send:
If the operation is True, this input node value or entered numeric or text value will be given out as Output Value.

Else send:
If the operation is False, this input node value or entered numeric or text value will be given out as Output Value.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

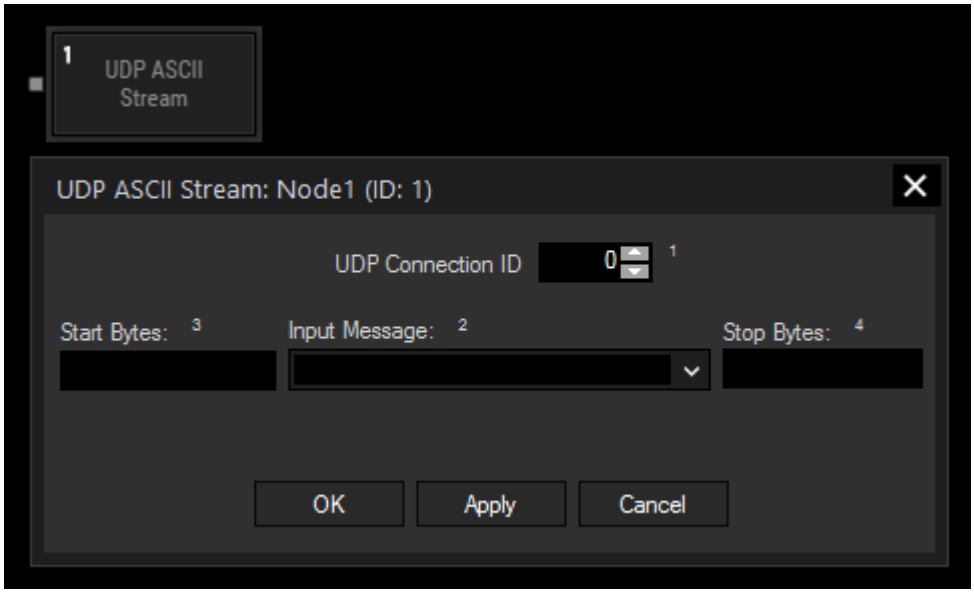
Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.13 UDP ASCII Stream Output

The UDP ASCII Stream output node allows sending ASCII values over an UDP connection.

Set up the UDP Connection in the Connection Manager [1258].

This node can be found under Nodes > Output > Connections > UDP ASCII Stream



Node Properties

TCP Connection ID:
Enter the UDP Connection ID.

Start and Stop Bytes:
Enter Start and Stop Bytes that should frame the message.
Enter these values as string, dec or hex value. Please refer to Syntax TCP- / UDP- / Serial messages [944].

Input Message:
Please select the input node source that should be given out via this UDP output node from the drop-down.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.2.14 UDP Message Output

The UDP Message value acts like an If node [1106]. The incoming data can be compared and depending on the true or false result a dedicated ASCII or Byte packet can be sent to the UDP connection setup by the its ID in the Connection Manager [1258].

This node can be found under Nodes > Output > Connections > UDP Message

Node Properties

UDP Connection ID:
Enter the UDP Connection ID.

Update:
Choose if the output should be updated either "On Change of Input Value Only" or permanently.

If:
Choose the first input node from the drop-down or enter a numeric value. This input will be compared to the second input.
Choose the Operator.
Choose the second input node from the drop-down or enter a numeric value.

Then send:
If the operation is True, this input node value or entered numeric or text value will be given out as Output Value.

Else send:
If the operation is False, this input node value or entered numeric or text value will be given out as Output Value.

Node control

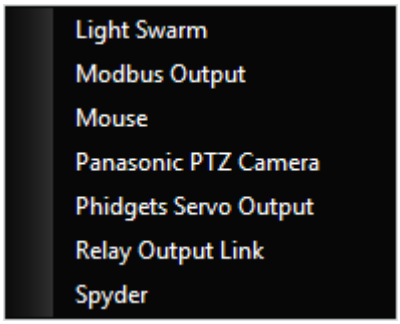
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.3 Devices Output

The Devices output nodes can send data to specific, implemented hardware devices. This can be the computer's mouse pointer, as well as Phidget [1383] output devices, the NET Link Relay output [2007] or even sending commands to a Christie Spyder image processor.

The communication with the here listed devices is implemented in the Widget Designer software, which means that the application transmits data already in the correct format the device requires.
For most devices, it is necessary to establish a connection first, e.g. if it is connected via network, and configuration. The devices listed in the Devices Menu can be set up with the respective entry there. All others only need the configuration the node itself provides.



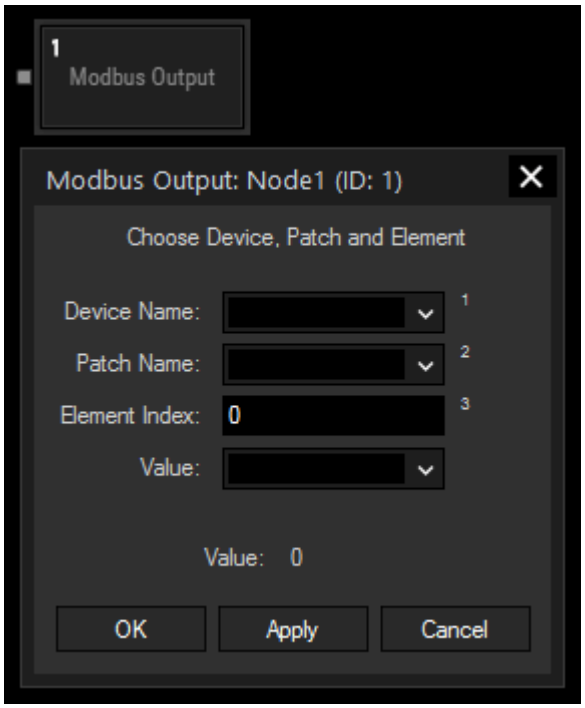
Light Swarm
Modbus Output [1196]
Mouse [1197]
Panasonic PTZ Camera [1198]
Phidgets Servo Output [1201]
Relay Output Link [1202]
Spyder

7.4.6.3.1 Modbus Output

The Modbus output node allows you to set the output value of a specific Modbus element. The patch this element belongs to has to be configured as output type. The respective Master and patch need to be set up before using this node, for more information please refer to the chapters Modbus Master Configuration [1299] and Modbus Patch Configuration [1300].

Modbus output values can also be set via scripting, as described in the chapter Using Modbus [1302].

This node can be found under Nodes > Output > Devices > Modbus



Node Properties

Device Name:
Select the Master device from the drop-down.

Patch Name:
Select the patch from the drop-down.

Element Index:
Enter the index of the element of which you would like to retrieve the current value. Please note that the index starts with "0" for the first element. If you have specified names for your elements in the patch configuration [1300], you can as well enter the element's name instead of the index.

Value:
Choose an input node from the drop-down or enter a numeric value.

Node control

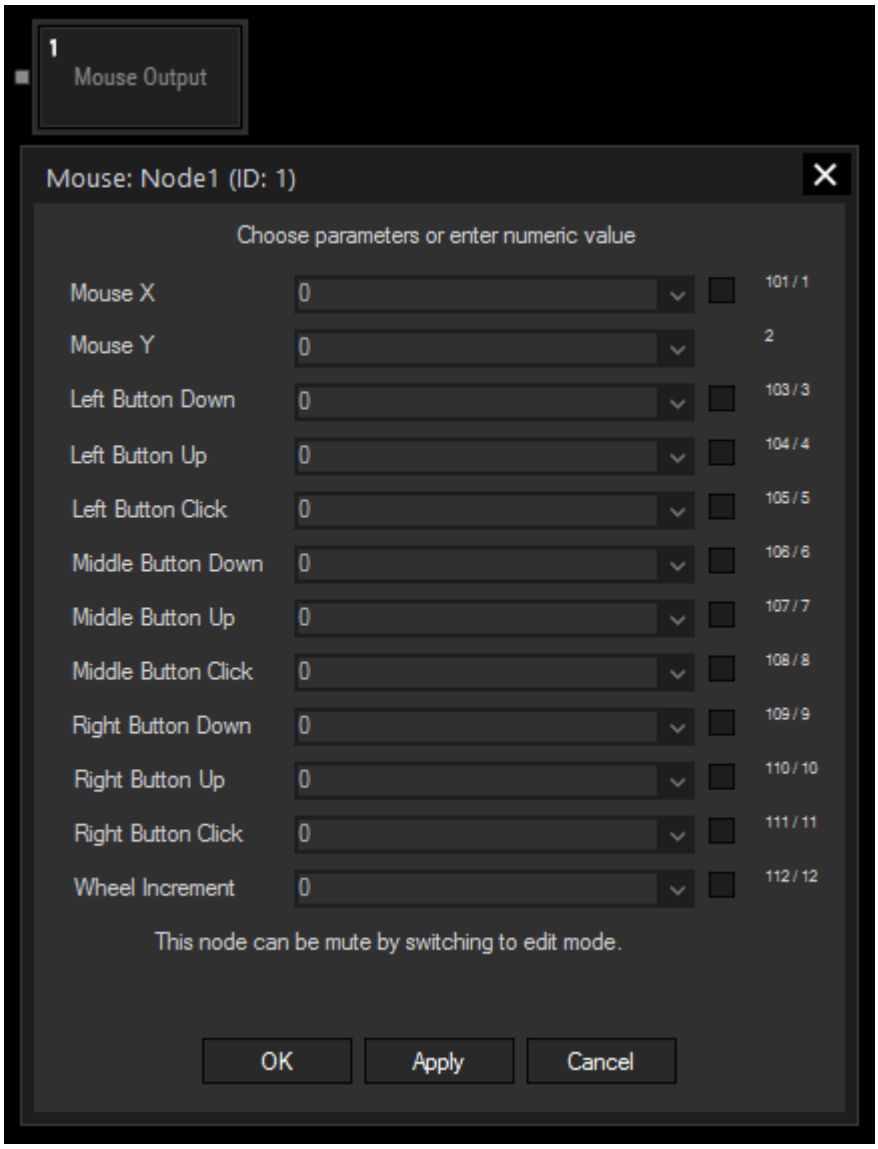
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.3.2 Mouse Output

The Mouse output node allows to remote control the local mouse and map input values to emulate mouse clicks and mouse motion.

This node can be found under Nodes > Output > Devices > Mouse



Node Properties

Enable the parameter of your local mouse that you want to remote control by checking its box and select an input node from the drop-down.

Please note:
The values for Mouse X and Mouse Y range from 0 to the max. value of your screen resolution.
The values for all other parameters have to be either 0 (False) or 1 (True).

If you have an external device, such as an AirScan or incoming touch data, used to control your computer mouse, it is recommended to add a keyboard shortcut [1484] or similar for disabling the mouse control. The command WDNodeDisableOutput [1779] and the corresponding enable command [1780] are helpful here.

Additionally, the node can be muted by switching to Edit mode by pressing [F8].

Node control

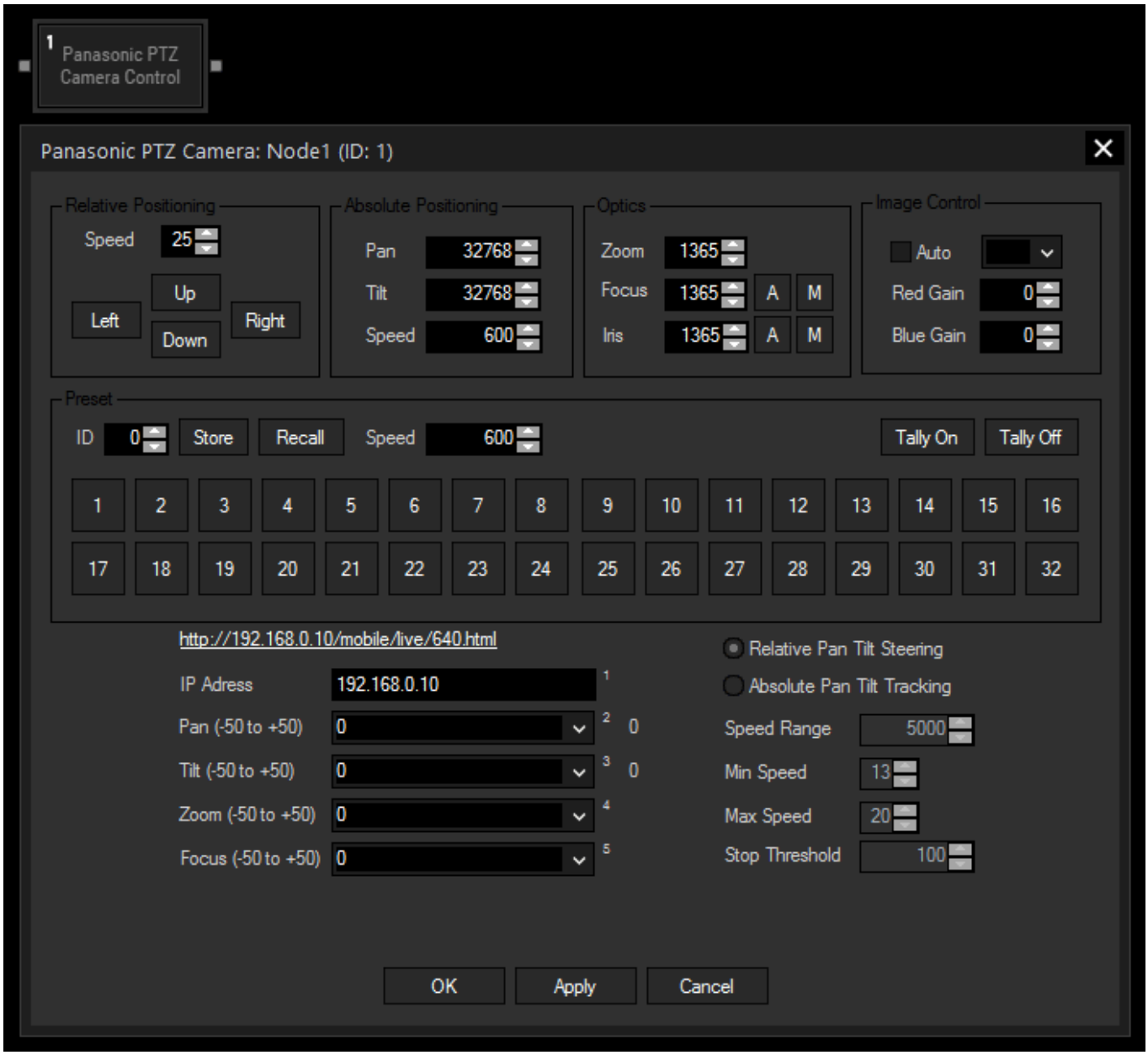
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.3.3 Panasonic PTZ Camera Output

The Panasonic PTZ Camera output node allows easy and quick remote access to the functions of a Panasonic PTZ dome camera. It enables you to control movement and lens as well as image adjustments. Storing and calling presets is possible too, the node additionally provides the absolute current pan and tilt values of the camera as an output.

This node can be found under Nodes > Output > Devices > Panasonic PTZ Camera



Node Properties

Speed:
Enter here the speed with which the camera should perform position adjustments

Up / Down / Left / Right:
Press these buttons to change the camera direction manually.

Pan / Tilt:
If you want to set your camera to a specific position, enter here the absolute values of pan and tilt direction.

Speed:
Enter here the speed with which the camera should move to the absolute position values adjusted above.

Zoom:
Enter here how much the camera should zoom in or out.

Focus:
For manual focus control press the **M** button and adjust the camera's focus with the value on the left until the image is clear.
Press the **A** button for switching to auto-focus mode.

Iris:
For manual iris control press the **M** button and adjust the camera's iris with the value on the left.
Press the **A** button for switching to auto-iris mode.

Image Control:
Check the box to toggle between **auto** and **manual** mode. In auto mode, the functions Red / Blue Gain as well as Iris are not available for manual adjustment.

Select a **Gain value** for your image from the drop-down if you are in manual mode. If the colors if the camera image need adjustment, you can use the **Red** and **Blue Gain** value to correct it.

ID:
If you want to **Store** the current position and lens values in a preset, or **Recall** those values from an existing preset, enter the preset ID here and press the respective button afterward. For recalling a preset, you can alternatively press one of the **32 preset buttons**.

Speed:
Enter here the speed with which the camera should move to the values recalled from a preset.

Tally On / Off:
Press one of these buttons to switch tally on or off.

IP:
Enter here the IP address of your camera. The link above this field will redirect you to the camera's web control interface.

Pan / Tilt:
Choose an input node from the drop-down or enter a numeric value. For relative steering, the input value needs to range from -50 to +50, for absolute tracking between 0 and 65535.

Zoom:
Choose an input node from the drop-down or enter a numeric value. The input value needs to range from -50 to +50.

Focus:
Choose an input node from the drop-down or enter a numeric value. The input value needs to range from -50 to +50.

Relative Pan Tilt Steering:
Select this mode for applying the input values for Pan and Tilt relatively to the current camera value.

Absolute Pan Tilt Tracking:
Select this mode for directly applying the input values for Pan and Tilt to the camera. This box also activates adjustment for the points below.

Speed Range:
As Pan and Tilt position is, by design of the camera control protocol, not being sent to the camera with the absolute values that arrive at the Widget Designer, but in alternating speed values.
Widget Designer calculates these speed values, whereby the Speed Range is a factor influencing the calculation.

The higher the speed range gets, the slower the movement becomes
The default value of 5000 should be adequate for most applications.

Min / Max Speed:
These values also influence the calculation of the mentioned alternating speed values.

Stop Threshold:
The stop threshold indicates the accuracy of the camera reaction to the transmitted absolute values. This means that sending a value of 32000 as absolute position can result in an actual camera position of 32050 or 31090. The smaller the threshold gets, the more exact the position is being taken.
A threshold of 1 would force the camera to take the exact position it was set to, but it might need to adjust. This results in jerking, so it is necessary to find a compromise between the precision of the position and a smooth picture.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

This node additionally provides specific node commands for directly interacting with the node settings. For instance, `Node1.RecallPreset(3)` will execute the function automatically without the need of opening the Configuration dialog and clicking the according button manually.

The following additional function commands are available for the Panasonic PTZ Camera node:

- AbsolutePosition
- RecallPreset
- RelativeDown
- RelativeLeft
- RelativeRight
- RelativeStop
- RelativeUp
- SetBlueGain
- SetFocus
- SetFocusAuto
- SetFocusManual
- SetGainAuto
- SetGainLevel
- SetGainManual
- SetIris
- SetIrisAuto
- SetIrisManual
- SetPresetSpeed
- SetRedGain
- SetZoom
- StorePreset
- TallyOff
- TallyOn

Node output values

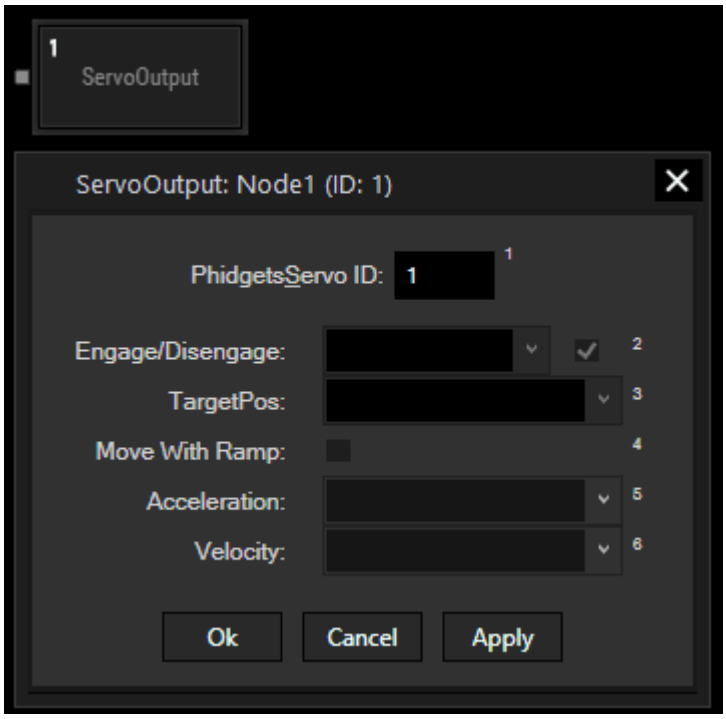
The Node generates the following output:

- Pan (exact current pan position of the camera)
- Tilt (exact current tilt position of the camera)

7.4.6.3.4 Phidgets Servo Output

The Phidgets Servo output node allows to remote control a connected Phidgets Servo device. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Phidgets Servo [1428]" which also shows how to work with the device using the scripting language.

This node can be found under Nodes > Output > Devices > Phidgets Servo Output



Node Properties

PhidgetsServo ID:
Enter the ID you used in the Configuration dialog [1428] to connect to the according Phidget.

Engage/Disengage:
If you want to also be able to engage or disengage the motor with a node rather than scripting, check the box and select an input from the drop-down or enter a fixed value.
The expected vales are "true" for engaging and "false" for disengaging the device.

TargetPos:
Here you can either enter a fixed value for the final position. Alternatively, choose an input from the drop-down list in case you connected an input or filter node beforehand which sends data.

Move with Ramp:
Activate the check box if you like to send values for "Acceleration" and "Velocity" that should be applied when moving to another position.
If you do not use the Ramp mode, the position change will be executed at maximum acceleration and velocity.

Acceleration:
Here you can either enter a fixed value for the acceleration that should apply when the Servo motor changes position. Alternatively, choose an input from the drop-down list in case you connected an input or filter node beforehand which sends data.

Velocity:
Here you can either enter a fixed value for the velocity that should apply when the Servo motor goes to another position. Alternatively, choose an input from the drop-down list in case you connected an input or filter node beforehand which sends data.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

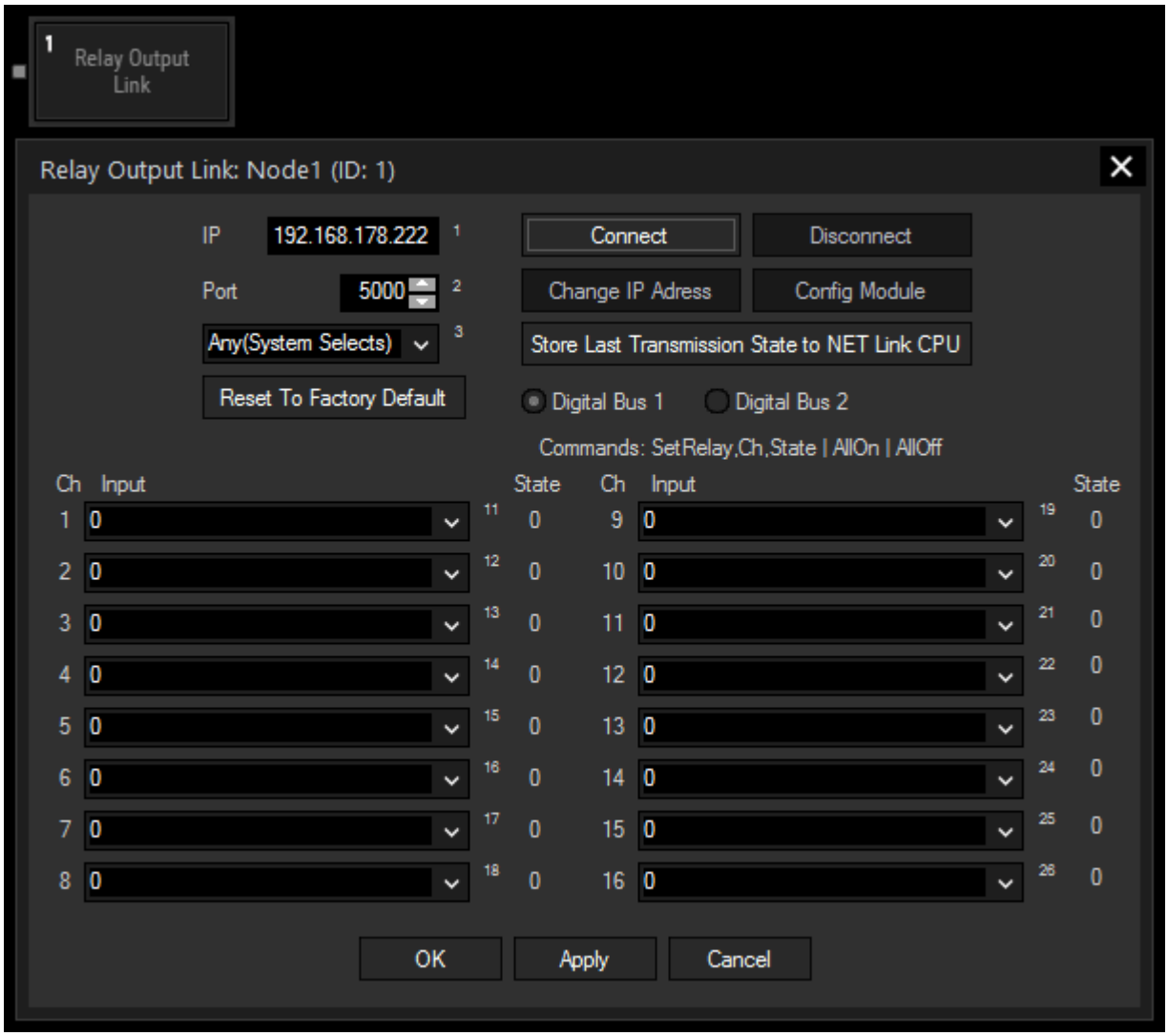
Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.3.5 Relay Output Link

The Relay Output Link node allows to communicate with the NET Link equipped with output boards. Please see more information in the chapter covering the NET Link [2000].
The node configures the device and sends commands to it, for instance to close a contact.

The NET Link Input node [1016] might also be of interest for you.

This node can be found under Nodes > Output > Devices > Relay Output Link



Node properties

IP and Port:
Enter the correct IP address and port from the NET Link's processor or from the Calibration Link.

Reset To Factory Default:
This buttons resets the above settings in the Widget Designer interface. To reset the device itself (to the IP

address 192.168.178.222 and the port to 5000), hold the "Reset" button down whilst plugging the power into the device. Release the button again.

Connect / Disconnect:
Before starting to communicate with the device, for instance receive data, the node must be connected to the device. The connection itself consumes no performance.

Change IP Address:
This button opens a new dialog whereto you may enter another IP address and port for the device. Power-cycle the device to apply the changes.

Config Module:
A NET Link / Calibration Link is configured by us as you have ordered it. However, if you have changed some input / output boards, the processor must be configured in terms of giving him the information which boards are connected to it. Click the "Config Module" button to open a new dialog where you may choose the according boards per processor connection. Find more details in the NET Link hardware chapter [2002].

Start Data / Stop Data:
As soon as data is processed via the network from the device to Widget Designer, performance is drawn.

Store Last Transmission State to NET Link CPU:
Click this button and power-cycle the device if you wish that it remembers whether it should (not) send data as soon as it is powered up.

Digital Bus:
Select here the digital bus the node is supposed to control

Channel, Input list and State:
First, connect an Input node or Filter node to the Relay node to provide the information "0" or "1". Then open the Item Properties and choose the incoming source data for the channel you like to control. The "State" informs you about the current state of the what the incoming signal. "0" leaves the contact open, whilst "1" closes it.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

The Relay Output Link node additionally provides specific node commands for directly interacting with the node settings. For instance, `Node1.StartData` will execute the function automatically without the need of opening the Configuration dialog and clicking the according button manually. `Node1.SetRelay(1,1)` will set the first channel to the state "1".

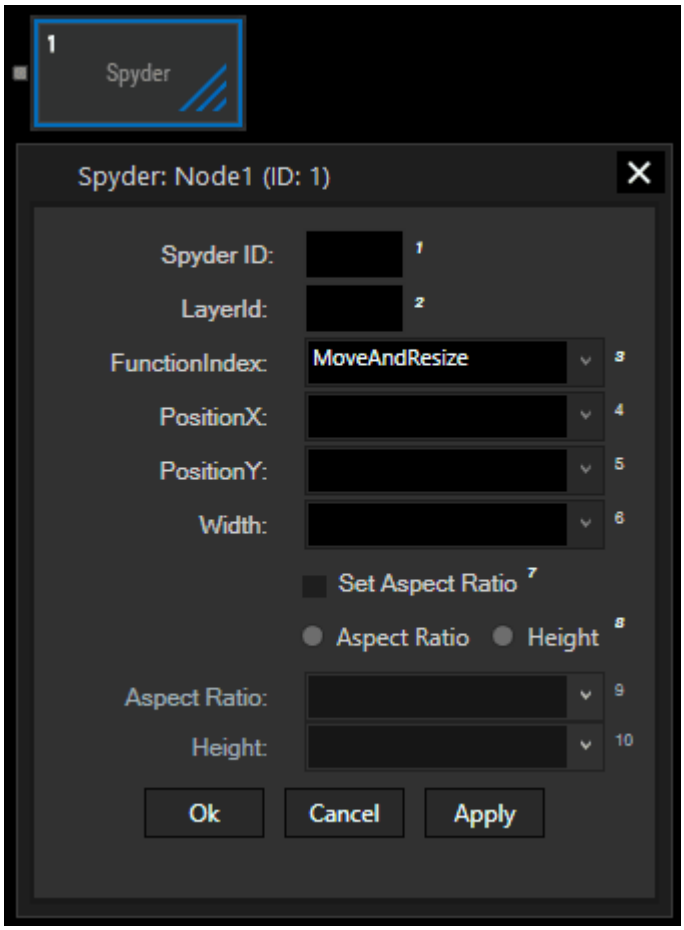
The following additional function commands are available for the Net Link Generic I/O node:

- AllOff
- AllOn
- ChangeIp
- Connect
- Disconnect
- GetConfig
- SetConfig
- SetRelay
- StartData
- StopData
- StoreLastTxMode

7.4.6.3.6 Spyder Output

The Spyder output node allows to remote control a connected Spyder device in terms of scaling and positioning its layers. Please add the device in the Configuration dialog first to connect to it. This is described in the chapter "Spyder" 1322 which also shows how to work with the device using the scripting language.

This node can be found under Nodes > Output > Devices > Spyder



Node Properties

Spyder ID:
Enter the ID you used in the Configuration dialog 1322 to connect to the according IP address.

Layer ID:
Enter the ID of the Spyder layer you like to scale and position.

FunctionIndex:
Choose which parameters you like to control of the Spyder layer:
MoveAndResize: This is based on the Spyder command "LSP–Layer Size and Position Change". It sets the size and horizontal and vertical position of a layer as an absolute position setting.

MoveAndResizeRelative: This is based on the Spyder command "LSP–Layer Size and Position Change". It sets the size and moves the horizontal and vertical position of a layer relative to the layer's current position.

MoveLayer: This is based on the Spyder command "KPS–Layer Position Change". It sets the horizontal and vertical position of a layer as an absolute position setting. Note: Positions are mapped in pixels, relative to the top-left pixelspace corner associated with the layer.

MoveLayerRelative: This is based on the Spyder command "KPS–Layer Position Change". It sets the horizontal and vertical position of a layer relative to the layer's current position. Note: Positions are mapped in pixels, relative to the top-left pixelspace corner associated with the layer.

ResizeLayer: This is based on the Spyder command "KSZ–Layer Size Change". It sets the size of the specified layer.

PositionX and Y:
This option is available if a "Move" function was chosen above. Here you can either enter a fixed pixel value for the horizontal and vertical layer location. Alternatively, choose an input from the drop-down list in case you connected an input or filter node beforehand which sends data.

Width:
This option is available if a "Resize" function was chosen above. Here you can either enter a fixed pixel value for the horizontal layer size. Alternatively, choose an input from the drop-down list in case you connected an input or filter node beforehand which sends data.

Set Aspect Ratio, Aspect Ratio and Height:
This option is available if a "Resize" function was chosen above.

If "Set Aspect Ratio" is not checked, resizing will affect the layer's width and the height adjusts automatically to ensure the layer's aspect ratio.
If it is checked, you can choose between "Aspect Ratio" and "Height" to either choose your own aspect ratio or adjust the height individually.

For "Aspect Ratio" you can either enter a fixed value (e.g. 1,77) or choose an input from the drop-down list in case you connected an input or filter node beforehand which sends data.
For "Height" you can either enter a fixed pixel value for the vertical layer size or choose an input from the drop-down list.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.4 File System Output

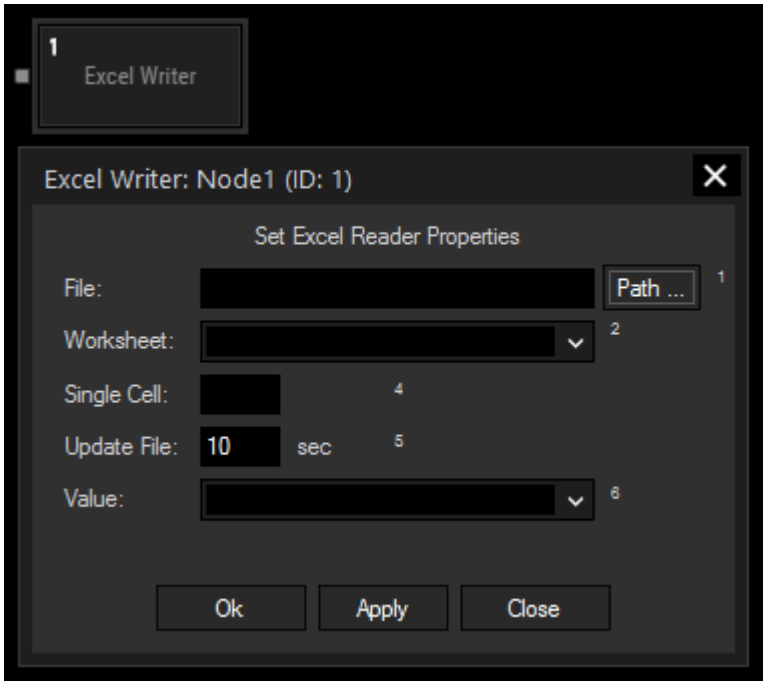
The File System output nodes currently only contain one node, the Excel Writer [1206] node. It enables you to write incoming data into an existing Excel file.

7.4.6.4.1 Excel Writer Output

The Excel Writer output node allows writing incoming data to Excel sheets.
Please note that as long the output node is active you won't be able to open the file in Excel at the same time.

Please note that Microsoft Excel has to be installed on the computer if you want to use this node!

This node can be found under Nodes > Output > File System > Excel Writer



Node Properties

File:
Click on [Path] to browse to the Excel file to which the node should write the incoming data. Press [Apply] in order to load the file before proceeding to the worksheet.

Worksheet:
Please choose the worksheet from the drop-down that should be edited.

Single Cell:
Enter the cell in which the node should write the incoming data, for example A1.

Update File:
By default the Excel File will be updated every 10 seconds. You may increase or decrease this value here.

Value:
Please choose the input source that should be written into the specified document.

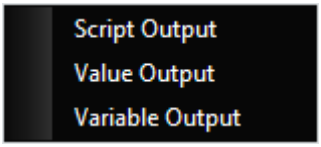
Node control

With Widget Designer version 6, all nodes support so called node commands 1928. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.5 Generic Output

The Generic input nodes provide output for generic actions, like sending incoming data to a variable or trigger a script under certain conditions.

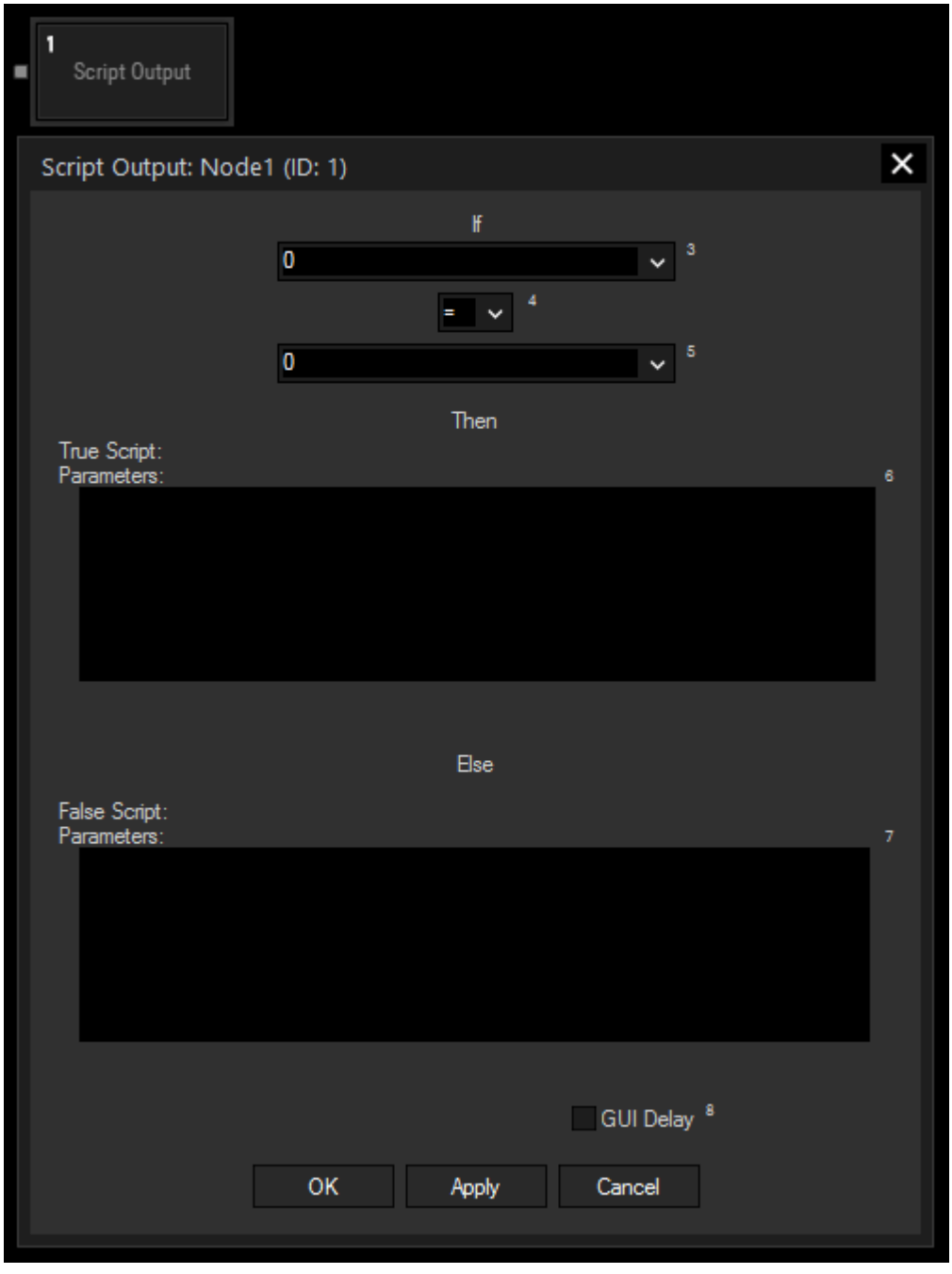


Script Output [1207]
Value Output [1208]
Variable Output [1209]

7.4.6.5.1 Script Output

The Script output node acts like an If Node, the incoming data can be compared and depending on the true or false result a dedicated script can be executed within Widget Designer.

This node can be found under Nodes > Output > Generic > Script Output



Node Properties

If:
Choose the first input node from the list or enter a numeric value. This input will be compared to the second input.
Choose the Operator.
Choose the second input node from the list or enter a numeric value.

True and False Script:
In the Script sections you may enter commands to be executed. If the operation is true, the True script will be executed. If the operation is False, the False script will be executed.
For a better overview, it is recommended to make use of Macros and Functions [1897] for sophisticated scripts.

Node control

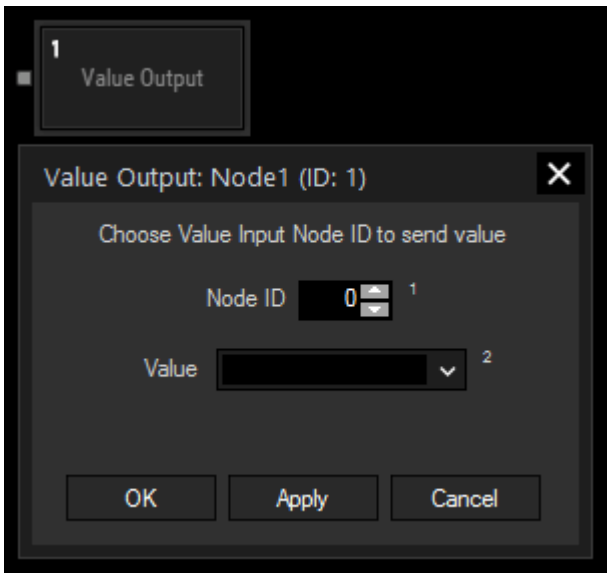
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.5.2 Value Output

The Value output node allows changing the value of a Value input node [1049].

This node can be found under Nodes > Output > Generic > Value Output



Node Properties

Node ID:
Enter the ID of the Value Input Node whose value you want to change.

Value:
Choose the input source that should be routed to the specified Value Input Node or enter a numeric value.

Node control

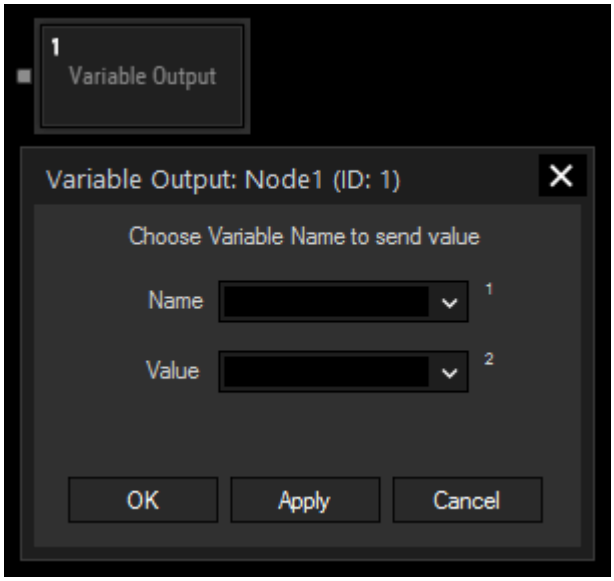
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.5.3 Variabel Output

The Variable output node allows assigning new values to an existing global variable [1900].

This node can be found under Nodes > Output > Generic > Variable Output



Node Properties

Name:
Enter here the variable's name.

Please note that only certain nodes like the Watchfolder [1037] node are able to provide data suitable for list variables. It is not possible to relate a Variable output node to a single element of a list variable.

Value:
Choose the input source that should be routed to the specified Variable.

Node control

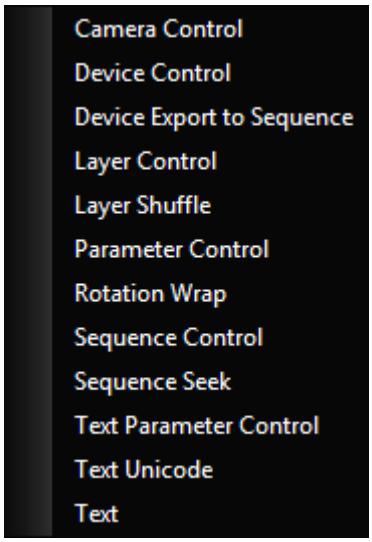
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6 Pandoras Box Output

The Pandoras Box output nodes allow you to send incoming data live to different elements in the Pandoras Box project. This includes Sequence and Device parameters, as well as sending text to Text Inputs [307].

For all nodes, you need a valid Pandoras Box connection [1256] to the Master application.

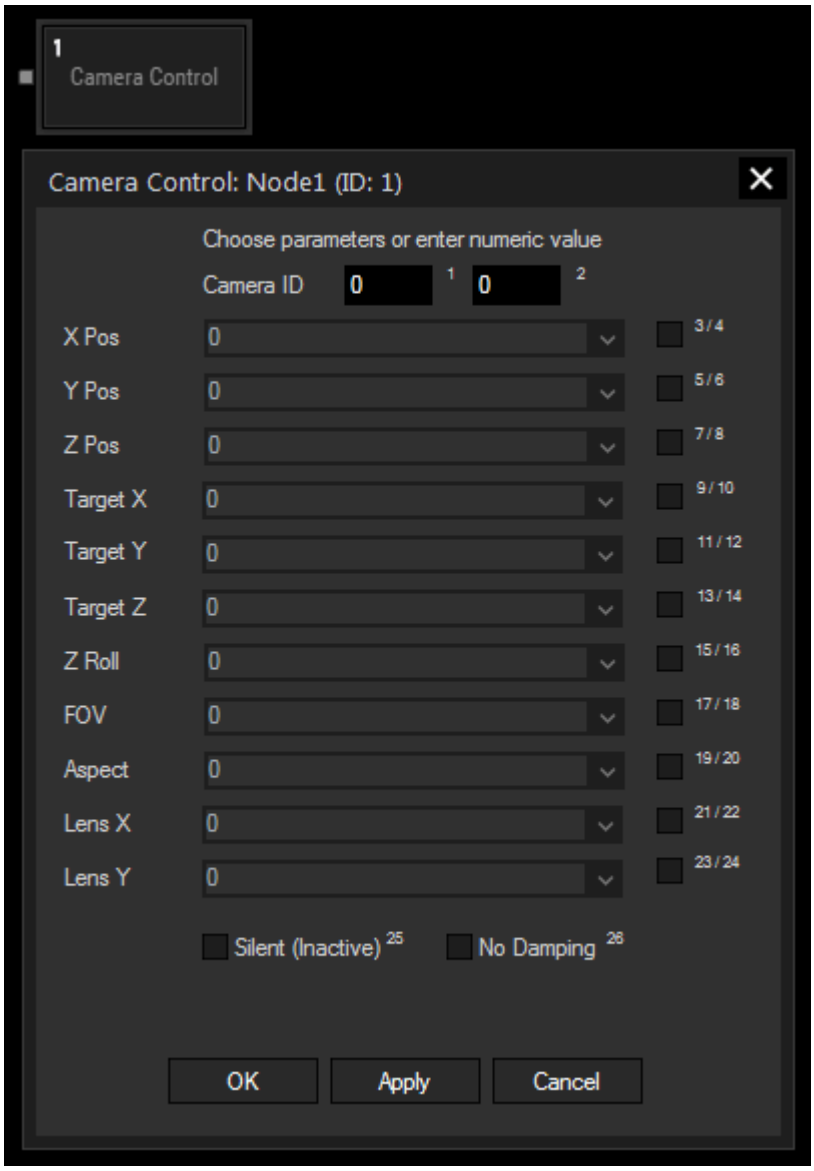


Camera Control [1211]
Device Control [1212]
Device Export to Sequence [1214]
Layer Control [1216]
Layer Shuffle
Parameter Control [1218]
Rotation Wrap [1219]
Sequence Control [1221]
Sequence Seek [1222]
Text Parameter Control [1223]
Text Unicode [1224]
Text [1225]

7.4.6.6.1 PB Camera Control Output

The Camera Control Node allows controlling camera parameters of any Pandoras Box Device such as Cameras and Outputs.

This node can be found under Nodes > Output > Pandoras Box > Camera Control



Node Properties

Camera ID:
Enter here Site and Device ID of the Device to be controlled.

The following **Device parameters** can be controlled by choosing an input node from the drop-down or entering a numeric value. The check-box next to the respective parameter needs to be ticked to enable it.
- X Pos
- Y Pos
- Z Pos
- Target X
- Target Y
- Target Z
- Z Roll
- FOV
- Aspect
- Lens X
- Lens Y

Silent (Inactive):
Check this box for silent value changes which means that no active values are generated in Pandoras Box.

No Damping:
Check this box to turn off the parameter value smoothing [210] in Pandoras Box.

Node control

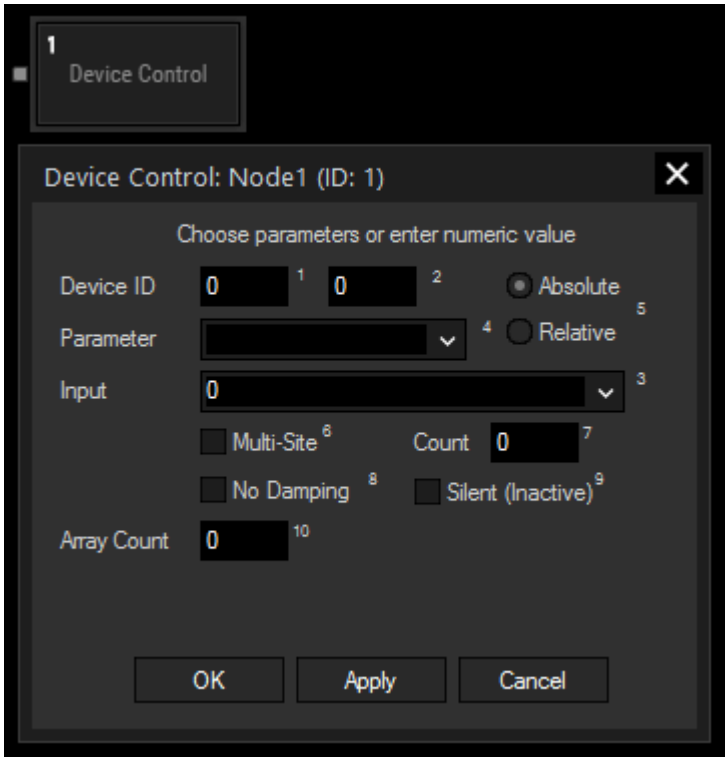
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.2 PB Device Control Output

The Device Control node allows controlling one selectable parameter of a Pandoras Box Device in both absolute or relative mode. Any device such a Video Layer, Audio Track, Camera, Output, ... can be controlled.

This node can be found under Nodes > Output > Pandoras Box > Device Control



Node Properties

Device ID:
Enter the Device ID, e.g. 1 in the first text field and 2 in the second text field. This will control Layer 2 of Site 1.

Absolute / Relative:
Choose absolute or relative mode. In absolute mode the PB device will take the input value, in relative mode the input value will be added or subtracted to the PB Device parameter's value.

Parameter:
Choose the parameter you want to control from the list.

For controlling an FX parameter, write the FX name as it appears in PB and the parameter name, separated by a pipe symbol. E.g.: RGB Multiply|Blue
The same goes for Particle System parameters.

Input:
Choose the input source from the list that should provide the values.

Multi-Site:
In order to control several devices simultaneously the Multi-Site option can be used instead of adding a second PB Device Control Output Node. To do so please tick the check box and enter the amount of sites.

Example:
To control Layer 1 of Site 1 and Layer 1 of Site 2, enter "1 1" in the text fields for the Device ID.
Enable the Multi-Site option and enter the amount of sites: "2".
To control Layer 1 of Site 2 and Layer 1 of Site 3, enter "2 1" in the text fields for the Device ID and "2" for the Multi-Site Count.

Silent (Inactive):
Check this box for silent value changes which means that no active values are generated in Pandoras Box.

No Damping:
Check this box to turn off the parameter value smoothing [210] in Pandoras Box.

Array Count:
If you also want to control not only several Sites but also several Devices at once (with IDs consecutively following the one entered at the top), you can enter the amount of additional Devices here.

Node control

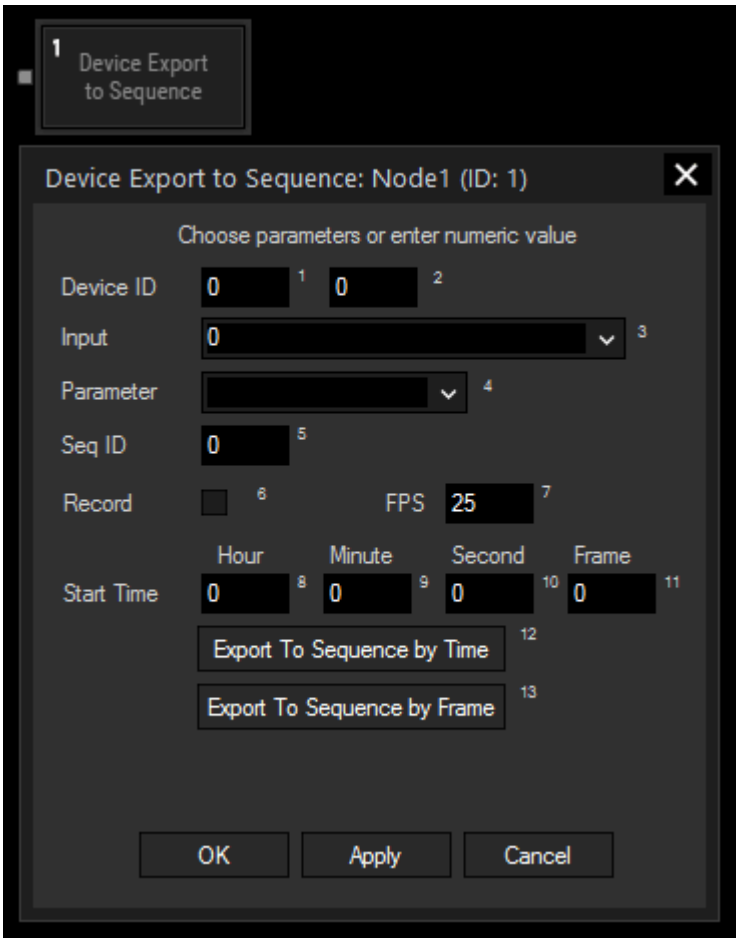
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.3 PB Device Export To Sequence Output

This node allows you to record any incoming data and export the recorded value sequence as Keys directly to a Device parameter in the specified Sequence.

This node can be found under Nodes > Output > Pandoras Box > Device Export to Sequence



Node Properties

Device ID:
Enter the Device ID, e.g. 1 in the first text field and 2 in the second text field. This will control Layer 2 of Site 1.

Input:
Choose the input source from the list that should provide the values for the record.

Parameter:
Choose the parameter you want to export the keys to from the list.

Seq ID:
Enter the Sequence ID.

Record:
Tick this check box and press "Apply" to start recording the input values, untick it and press "Apply" again to stop recording.

FPS:
Enter the frames per seconds with which should be recorded.

Start Time:
Enter the hours, minutes, seconds and frames at which you want to export your value sequence first.

Export To Sequence by Time:
Clicking this button after having recorded a value sequence will export those values as keys to the specified device parameter on the Pandoras Box sequence. Please pay attention to have a container already prepared that enfolds the time of the whole recorded sequence. Otherwise, containers with a default length of 10s will be created and set to the sequence. If your record is longer than this, it will become disrupted in several containers.
Every further click will add the key sequence another time to your sequence, directly subsequent to the previous one. This behavior can be reset by clicking "Apply" again.

Export To Sequence by Frame:
Clicking this button after having recorded a value sequence will export two keys to the specified device parameter on the Pandoras Box sequence. The first key will be placed at the Start Time and contains the first value of your recorded value sequence. The second key will be placed at the current position of the now pointer during the record and contains the last recorded value.

Node control

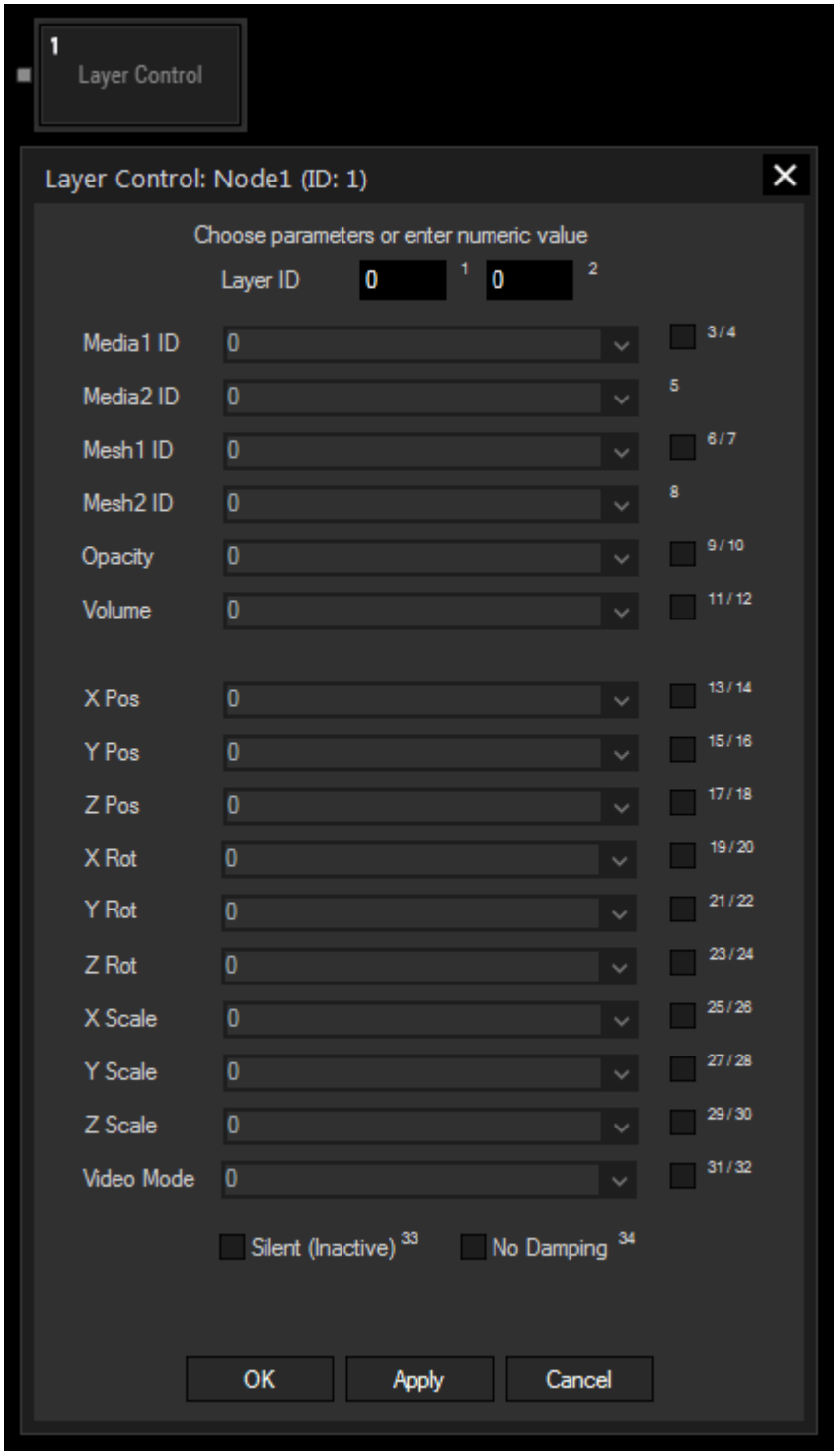
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.4 PB Layer Control Output

The Layer Control Node allows controlling certain parameters of any Pandoras Box Device [318].

This node can be found under Nodes > Output > Pandoras Box > Layer Control



Node Properties

Layer ID:
Enter here Site and Device ID of the Device to be controlled.

The following **Device parameters** can be controlled by choosing an input node from the drop-down or entering a numeric value. The check-box next to the respective parameter needs to be ticked to enable it.
Please also consider the different value ranges [1514] expected by the different parameters.
- Media1 and 2 ID (Folder and File ID of the resource)

- Mesh1 and 2 ID (Folder and File ID of the resource)
- Opacity
- Volume
- X Pos
- Y Pos
- Z Pos
- X Rot
- Y Rot
- Z Rot
- X Scale
- Y Scale
- Z Scale
- Video Mode

Silent (Inactive):
Check this box for silent value changes which means that no active values are generated in Pandoras Box.

No Damping:
Check this box to turn off the parameter value smoothing [210] in Pandoras Box.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.5 PB Parameter Control Output

The Layer Control Node allows controlling up to 16 parameters of any Pandoras Box Device [318] at once. As the parameters are freely selectable, this node is more flexible than the Layer Control [1216] node and it allows to control more than just one parameter like the PB Device Control Output [1212] node.

This node can be found under Nodes > Output > Pandoras Box > Parameter Control

Node Properties

Site and Device ID:
Enter here Site and Device ID of the Device to be controlled.

Input:
Choose an input node from the drop-down or enter a numeric value for controlling the parameter.

Parameter:
Select the parameters you want to control from the drop-down.

Absolute / Relative:
Select here if the values should be directly transmitted as the parameter value (absolute) or if it should be added to the current value (relative).

Multi-Site:
In order to control several Sites simultaneously the Multi-Site option can be used instead of adding a second PB Device Control Output Node. To do so please tick the check box and enter the amount of sites in the field below.

Array Count:
If you also want to control not only several Sites but also several Devices at once (with IDs consecutively following the one entered at the top), you can enter the amount of additional Devices here.

No Damping:
Check this box to turn off the parameter value smoothing [210] in Pandoras Box.

Silent (Inactive):
Check this box for silent value changes which means that no active values are generated in Pandoras Box.

Node control

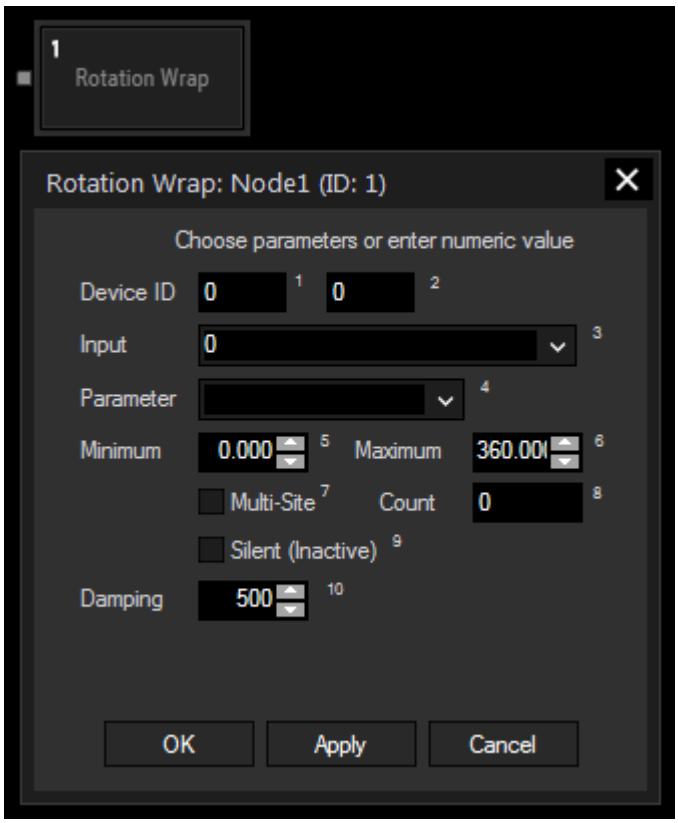
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.6 PB Rotation Wrap Output

The Rotation Wrap output node sends a value to a Pandoras Box Device parameter and maps the input value on the entered range between minimum and maximum. Although all PB Device parameters are available, this node is especially useful for rotations.

This node can be found under Nodes > Output > Pandoras Box > Layer Control



Node Properties

Device ID:
Enter here Site and Device ID of the Device to be controlled.

Input:
Choose an input node from the drop-down or enter a numeric value for controlling the parameter.

Parameter:
Select the parameter you want to control from the drop-down.

Minimum / Maximum:
Enter here the range on which the input value should be mapped.

Multi-Site:
In order to control several Sites simultaneously the Multi-Site option can be used instead of adding a second PB Device Control Output Node. To do so please tick the check box and enter the amount of sites in the **Count** field.

Silent (Inactive):
Check this box for silent value changes which means that no active values are generated in Pandoras Box.

Damping:
Enter here the value for the parameter value smoothing [210] in Pandoras Box.

Node control

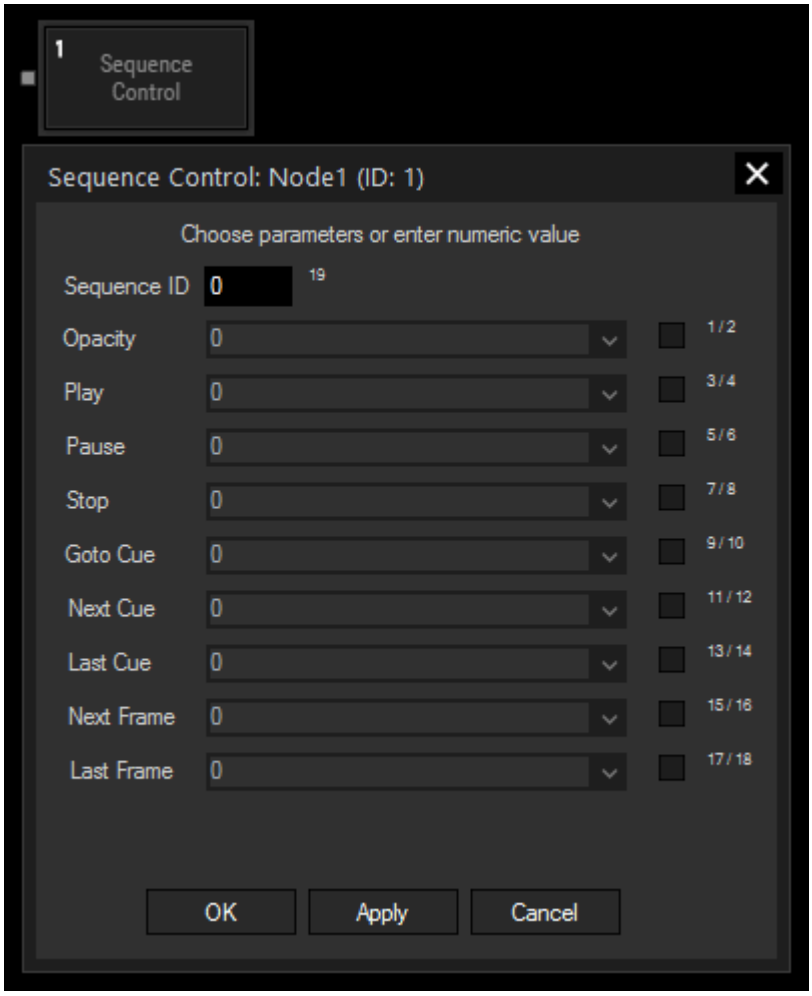
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.7 PB Sequence Control Output

The Sequence Control output node allows mapping any input or filter node data to any sequence parameter value.

This node can be found under Nodes > Output > Pandoras Box > Sequence Control



Node Properties

Sequence ID:
Enter the Sequence ID.

Parameter:
Enable all parameters you want to control by ticking its check box. Choose the input source for the parameter from the list.
The parameters accept different value ranges:

Opacity	0 to 255
Play, Pause, Stop	0=false, 1=true
GotoCue	Cue ID
Next/Last Cue and Next / Last Frame	0=false, 1=true

Node control

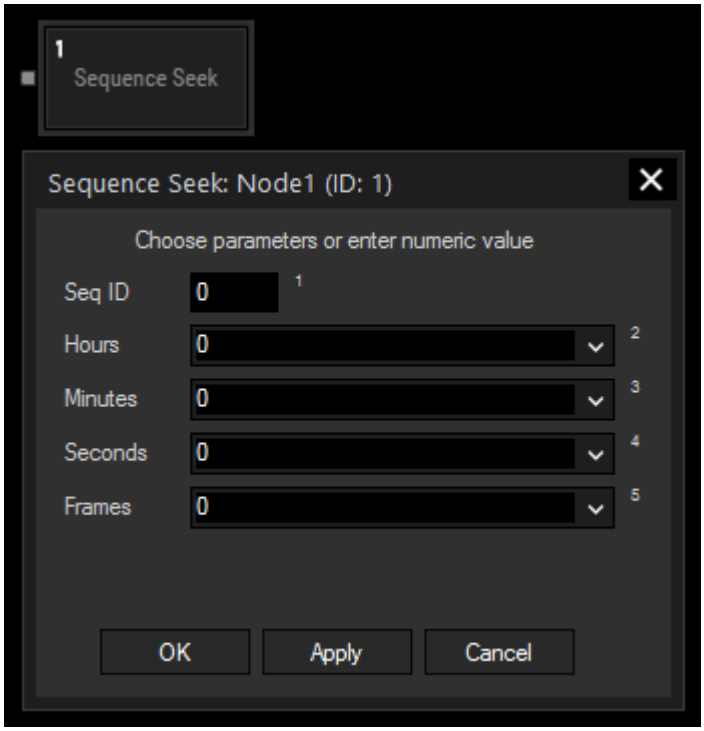
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.8 PB Sequence Seek Output

The Sequence Seek output node allows mapping any input or filter node data in order to set any sequence to any timecode.

This node can be found under Nodes > Output > Pandoras Box > Sequence Seek



Node Properties

Sequence ID:
Enter the Sequence ID.

Hours:
Choose the input source from the list.

Minutes:
Choose the input source from the list.

Seconds:
Choose the input source from the list.

Frames:
Choose the input source from the list.

Node control

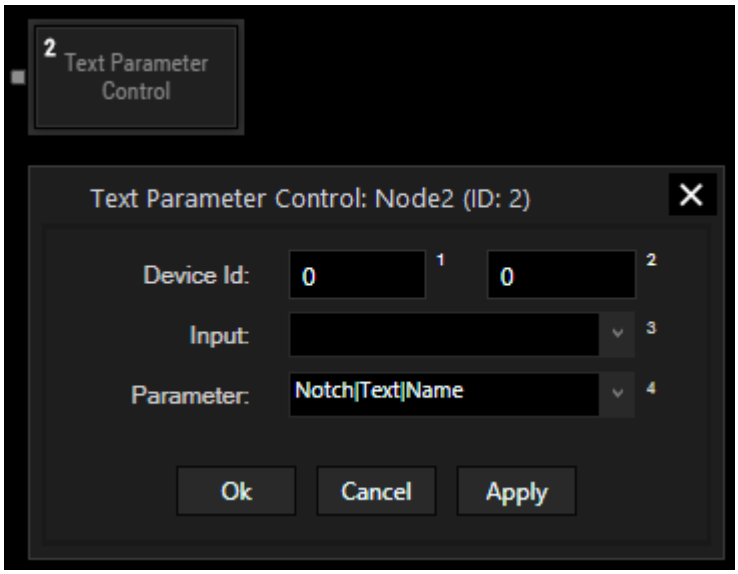
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.9 PB Text Parameter Control

The Text Parameter Control node allows controlling one selectable parameter of a Pandoras Box Device that contains text. For the time being, text parameters can only be found on Notch Layers [671].
In addition to the node you can also use the scripting language to change text parameters in PB, e.g. via the command `DeviceSetTextParam` [1537].

This node can be found under Nodes > Output > Pandoras Box > Text Parameter Control



Node Properties

Device ID:
Enter the Device ID, e.g. 1 in the first text field and 2 in the second text field. This will control Layer 2 of Site 1.

Input:
Choose the input source from the list that should provide the string values.

Parameter:
Choose the parameter you want to control from the list.
For controlling a Notch parameter, write the full parameter name as it appears in PB. The groups are shown in the Device Control tab [171] but also in the Device Tree [173] when unfolding entries. The Device Tree is helpful too, if a name is too long to be displayed in the Device Controls tab.
The name is case-sensitive, all spaces count and the hierarchy needs to be represented using the pipe "|" character, e.g. Notch|Text|Param Name

Node control

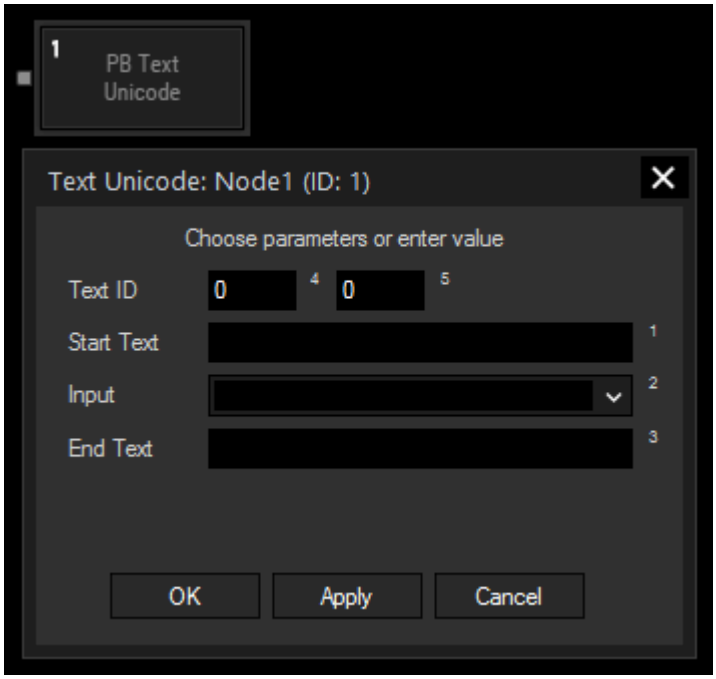
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.10 PB Text Unicode Output

The PB Text Unicode output node allows sending text strings in any language format to Pandoras Box text assets.

This node can be found under Nodes > Output > Pandoras Box > Text Unicode



Node Properties

Text ID:
Enter the Text Asset's DMX File and Folder ID. Set up this Text ID in Pandoras Box.

Start Text:
Enter any text the Text Asset should start with.

Input:
Choose the input source from the list.

End Text:
Enter any text the Text Asset should end with.

Node control

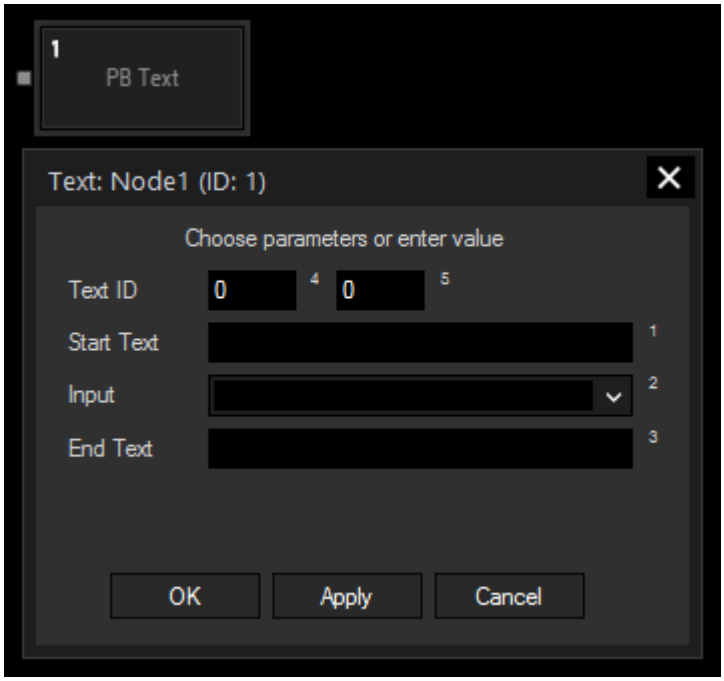
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.6.11 PB Text Output

The PB Text output node allows routing any input or filter node data to a PB Text Asset.

This node can be found under Nodes > Output > Pandoras Box > Text



Node Properties

Text ID:
Enter the Text Asset's DMX File and Folder ID. Set up this Text ID in Pandoras Box.

Start Text:
Enter any text the Text Asset should start with.

Input:
Choose the input source from the list.

End Text:
Enter any text the Text Asset should end with.

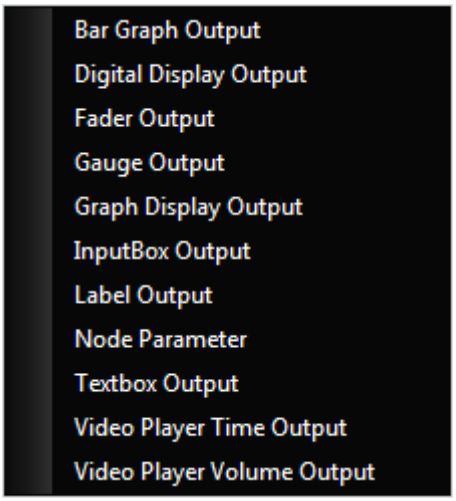
Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7 Widgets Output

It is possible to continuously send data to some widgets. An easy example would to link a Fader [874] with values of an incoming DMX channel, or to display data on a Label [888]. The Display widgets [851] on the other hand are designed as a visual representation for values and can only be used in combination with their respective output nodes. Even the dynamic control of other nodes' parameters can be achieved with the Node Parameter output node.

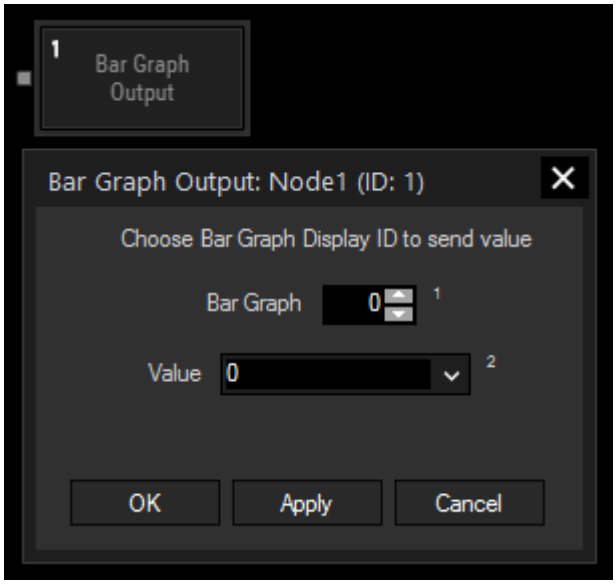


Bar Graph Output [1226]
Digital Display Output [1227]
Fader Output [1228]
Gauge Output [1229]
Graph Display Output [1230]
InputBox Output [1231]
Label Output [1232]
Node Parameter [1233]
Textbox Output [1234]
Video Player Time Output [1235]
Video Player Volume Output [1236]

7.4.6.7.1 Bar Graph Output

The Bar Graph output node allows assigning incoming source values to a Bar Graph [852].

This node can be found under Nodes > Output > Widgets > Bar Graph Output



Node Properties

Bar Graph:
Please enter the ID of the Bar Graph.

Value:
Please choose the input source that should be displayed via this Bar Graph.

Node control

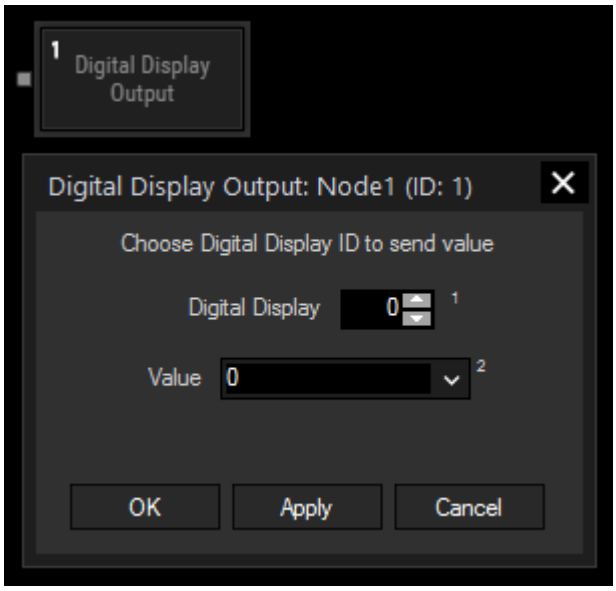
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.2 Digital Display Output

The Digital Display output node allows assigning incoming source values to a Digital Display [854].

This node can be found under Nodes > Output > Widgets > Digital Display Output



Node Properties

Digital Display:
Please enter the ID of the Digital Display.

Value:
Please choose the input source that should be displayed via this Digital Display.

Node control

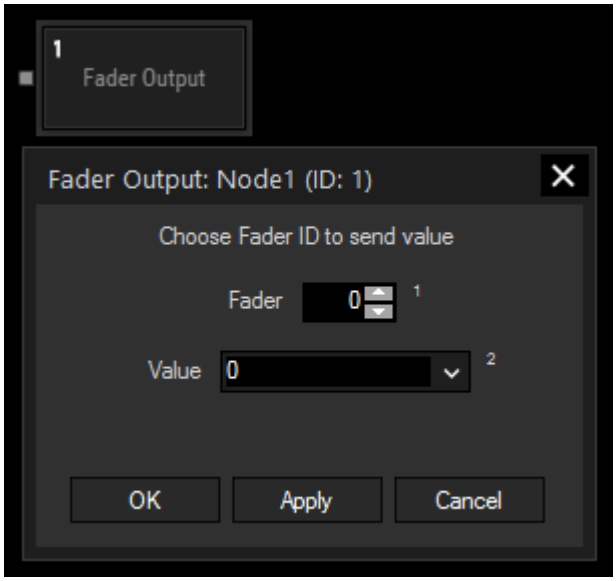
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.3 Fader Output

The Fader output node allows to remote control any Fader [874] based on its ID within a Widget Designer project.

This node can be found under Nodes > Output > Widgets > Fader Output



Node Properties

Fader:
Please enter the ID of the Fader.

Value:
Choose the input source that should remote control this Fader.

Node control

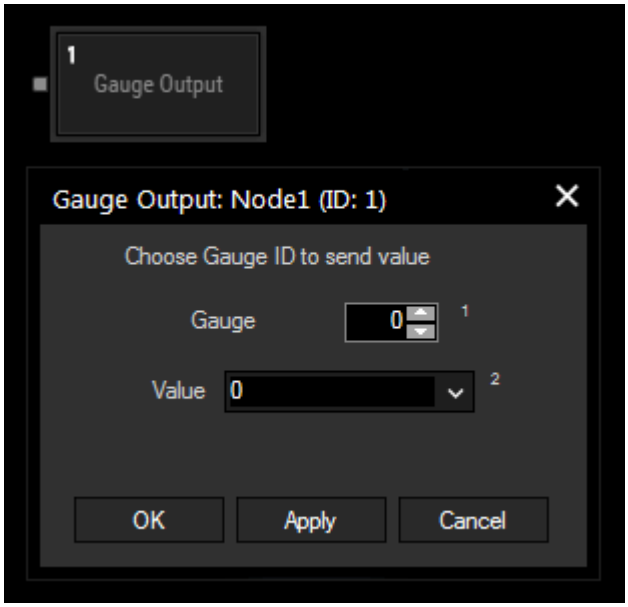
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.4 Gauge Output

The Gauge output node allows routing any incoming numeric value to be displayed on a Gauge [856].

This node can be found under Nodes > Output > Widgets > Gauge



Node Properties

Gauge:
Enter here the ID of the Gauge.

Value:
Choose an input node from the drop-down or enter a numeric value.

Node control

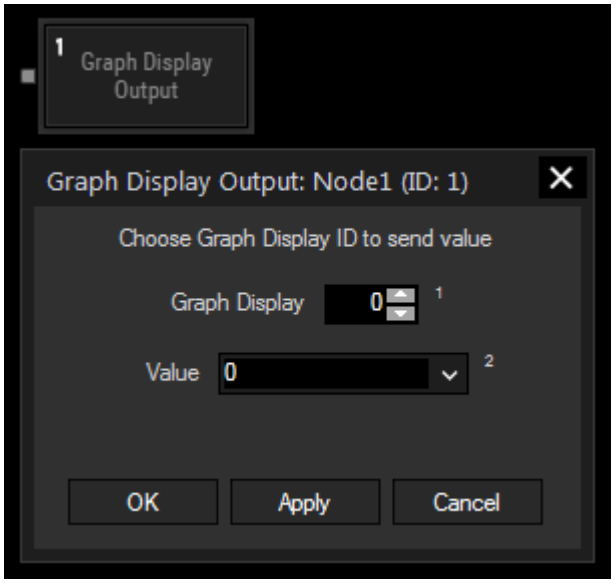
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.5 Graph Display Output

The Graph Display output node allows routing any incoming numeric value to be displayed on a Graph Display [858].

This node can be found under Nodes > Output > Widgets > Graph Display



Node Properties

Graph Display:
Enter here the ID of the Graph Display.

Value:
Choose an input node from the drop-down or enter a numeric value.

Node control

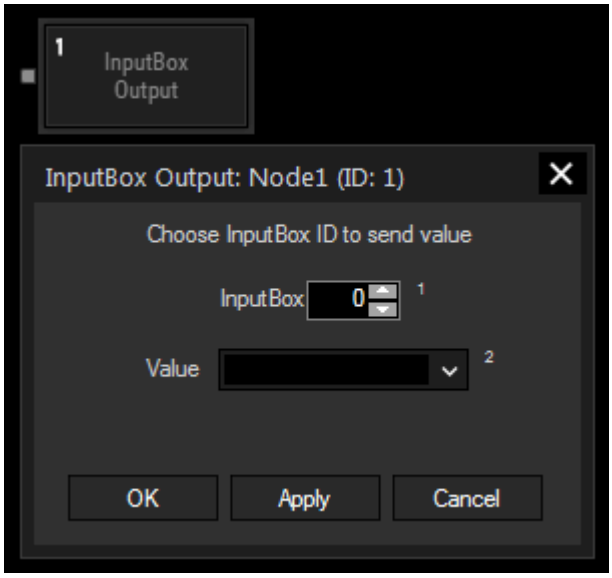
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.6 Input Box Output

The InputBox output node allows routing any incoming value to be displayed as text in an InputBox [886].

This node can be found under Nodes > Output > Widgets > InputBox



Node Properties

Gauge:
Enter here the ID of the InputBox.

Value:
Choose an input node from the drop-down or enter a text.

Node control

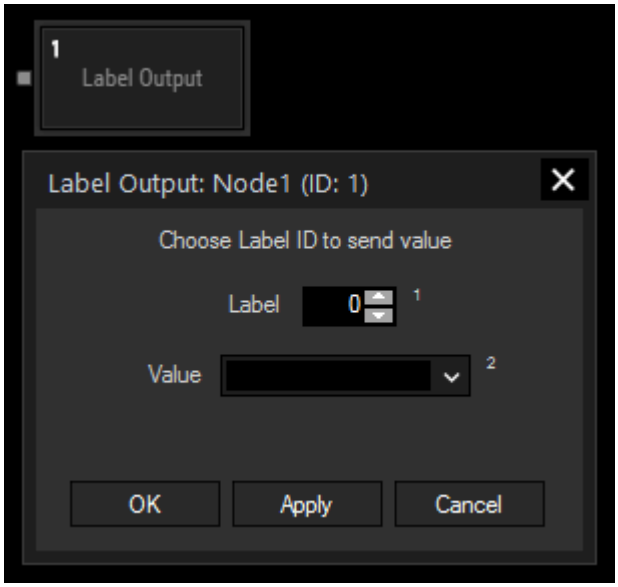
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.7 Label Output

The Label output node allows displaying any input value to a Label [888] inside the Widget Designer.

This node can be found under Nodes > Output > Widgets > Label Output



Node Properties

Fader:
Please choose the Label ID.

Value:
Please choose the input source that should be displayed in this Label.

Node control

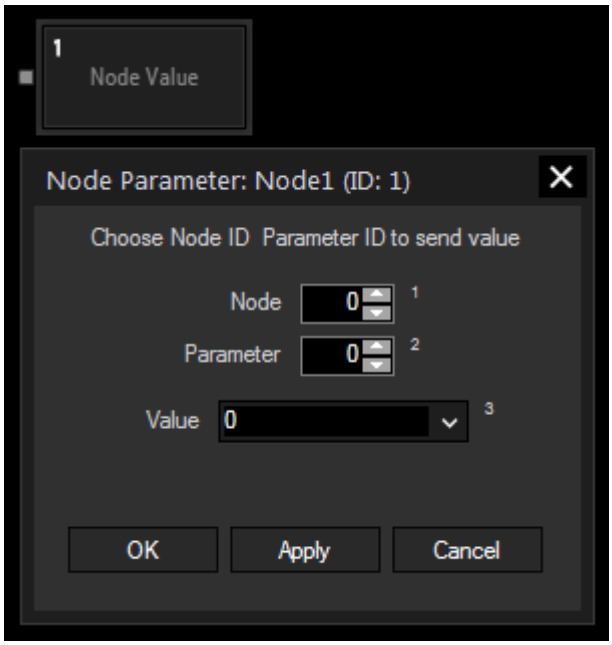
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.8 Node Parameter

Similar to the scripting command WDNodeSetParam [1782], this node allows you to control any node parameter. Please note that changing a node source for a filter or output node with this node will result in setting a numeric or string value for this parameter. Remotely changing the node source is only possible with the command WDNodeSetParamSource [1785].

This node can be found under Nodes > Output > Widgets > Node Parameter



Node Properties

Node:
Enter here the ID of the node.

Parameter:
Enter here the ID of the parameter you want to change. The ID is displayed as small superscript number behind the parameter value.

Value:
Choose an input node from the drop-down or enter a numeric or text value.

Node control

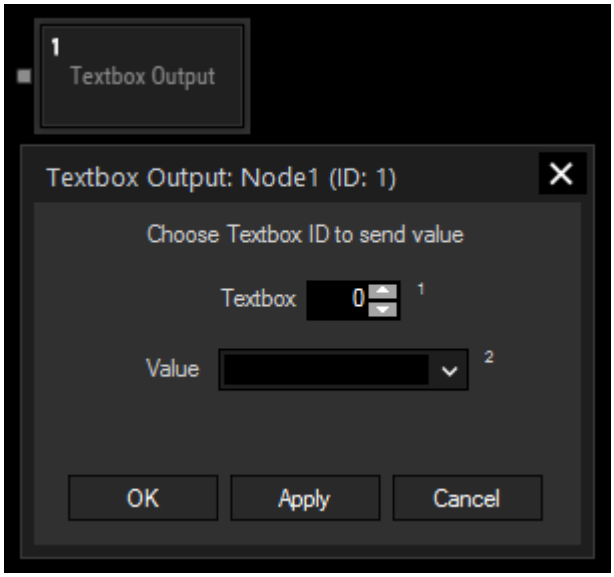
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.9 Textbox Output

The Textbox output node allows sending incoming texts to a TextBox [923] within Widget Designer.

This node can be found under Nodes > Output > Widgets > Textbox Output



Node Properties

Textbox ID:
Enter the TextBox ID the values should be sent to.

Value:
Please choose the input source.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

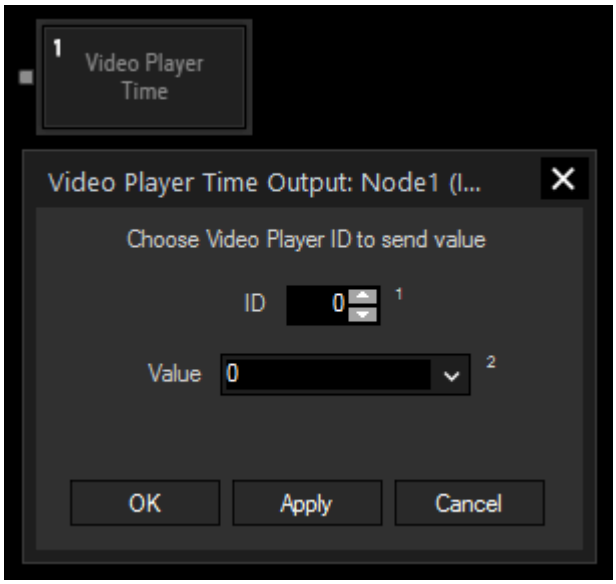
Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.10 Video Player Time Output

The Video Player Time output node allows controlling the current time of a video played in the Video Player [932] that is integrated within Widget Designer.
The time is always handled in seconds.

Example:
To jump to the timecode 1min 25 seconds, the input source has to take the value 85 seconds.

This node can be found under Nodes > Output > Widgets > Video Player Time Output



Node Properties

Video Player ID:
Enter the Video Player ID.

Value:
Choose the input source that should remote control the Video Player time.

Node control

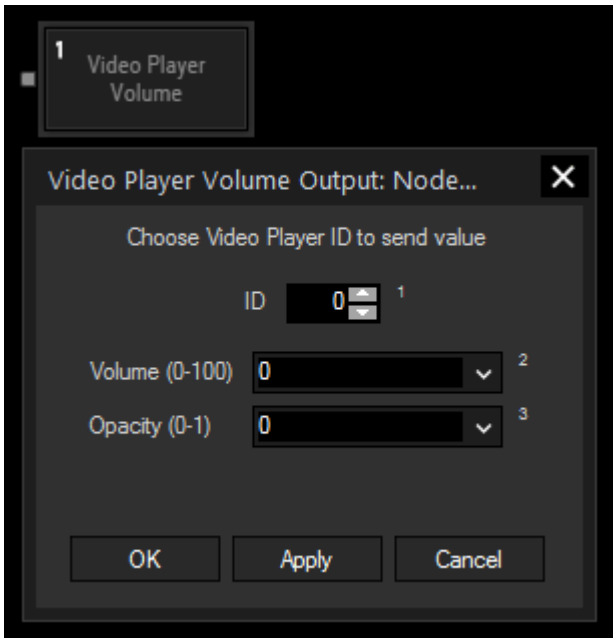
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.6.7.11 Video Player Volume Output

The Video Player Volume output node allows controlling the Volume of the Video Player [932] integrated within Widget Designer.

This node can be found under Nodes > Output > Widgets > Video Player Volume Output



Node Properties

Video Player ID:
Enter the Video Player ID.

Value:
Choose the input source that should remote control the Video Player volume.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.7 Script Nodes

Script nodes execute your customized script; they are triggered through a certain action. Please see the introductory chapter if you like to learn more about other node types or how to create and work with nodes [937] in general.

The following sub chapters describe the various Script nodes. Since version 6.1.3 the IR Phidget is supported via the Configuration dialog > Phidgets [1383], the RFID sensor will follow soon.

Action [1237]
COM [1239]
TCP [1242]
UDP [1240]
Gesture [1243]

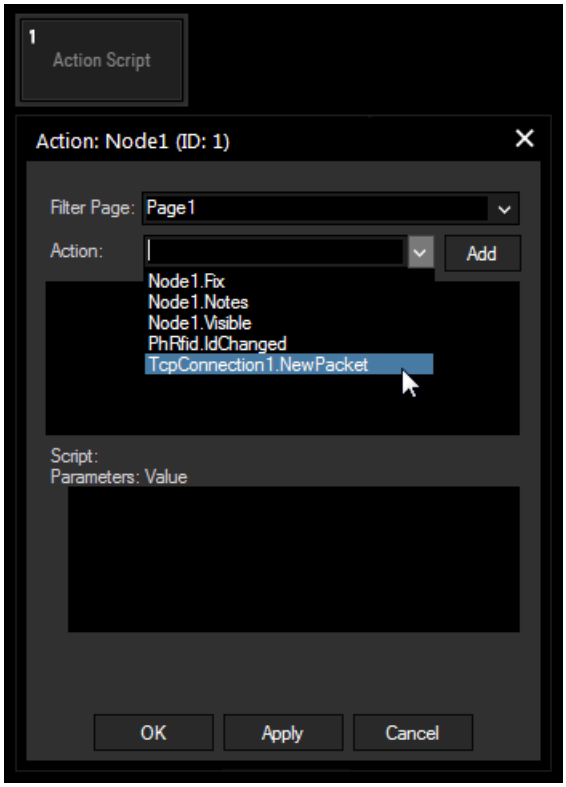
7.4.7.1 Action Script

The Action Script node allows triggering scripts with different events and actions. This action can be the change of a widget property, as well as a new package being received on a TCP, UDP or COM connection [1272].
Note: The trigger for the Phidget RFID was removed from the Action Script node in version 6.1.2 because it is now available in the Event Listener [1353].

Nearly all widget properties that are also available as object member properties [1904] can be inserted as trigger, multiple triggers are possible, too. A blue highlight provides you a visual indication when an action being detected and the script is performed.

The "Value" parameter also allows you to make use of the dedicated value of the action triggering the script.

This node can be found under Nodes > Scripts > Action



Node Properties

Filter Page:
Select here the page on which the widget is located you want to add as a trigger. All windows and pages are available, the node is not limited to the page it is located on.

Action:
This drop-down offers you a list of all possible trigger actions. It includes all widgets and their properties of the page you selected above, as well as the Phidget RFID device and all set TCP, UDP and COM connections. Select one entry of the drop-down and add it to your list by pressing the "**Add**" button. Multiple triggers can be added, the Action Script node will listen to alterations of all elements in the list and perform its script as soon as it detects one.

It is a good workaround for triggers not listed in the drop-down, to use a Label [888] for displaying the wished trigger and set the Action Script action to the Label text property. This way, even variables [1900] or values from input nodes [952] can act as a trigger.

Script:
In the Script section you may enter commands to be executed when the according data is received. For a better overview, it is recommended to make use of Macros and Functions [1897] for sophisticated scripts.
The actual value of the action triggering the script is handed over with the "Value" parameter. "Value" is of the data type "object", so you have to convert it to the estimated data type before using it in the script for most applications.

The value of a TCP, UDP and COM package is committed as a list of integers, representing the decimal bytes of the whole package. Please also refer to the chapter Syntax TCP- / UDP- / Serial Messages [944] for further information concerning byte transcription.
Example: The string "Hello World!" is represented by the byte list
`[72,101,108,108,111,32,87,111,114,108,100,33].`

Examples

Example: Label Text
Task: Every time the text of Label1 changes, the respective text should be written in the DebugLogger [812], together with the current timestamp. For the timestamp, the global variable "Now" can be used.

Action: Label.Text

Script:
```
DebugMessage(Now + ": " + Value)
```
If you have for example three lines of a Japanese Haiku (by Matsuo Basho) displayed in this Label one after the other, the resulting DebugLogger might look like this:

```
2017-10-25 12:37:22.187: iza saraba (an ancient pond)
2017-10-25 12:38:19.707: yukimi ni korobu (a frog jumps in)
2017-10-25 12:41:17.788: tokoromade (the splash of water)
```

Example 2: TCP Input Packet
Task: The Action Script node listens to TCP connection ID 1. The received packets should be analyzed, the header of the packets of interest is "WD!". The payload after the header contains text information and should be displayed on Label1.

Action: TcpConnection1.NewPacket

Script:

```
var byte_list = Value.ToList.Copy
if (byte_list[0] = 87) AND (byte_list[1] = 68) AND (byte_list[2] = 33) {
      byte_list.Remove(0)
      byte_list.Remove(0)
      byte_list.Remove(0)
      Label1.Text = byte_list.DecodeBytes
}
```

Node control

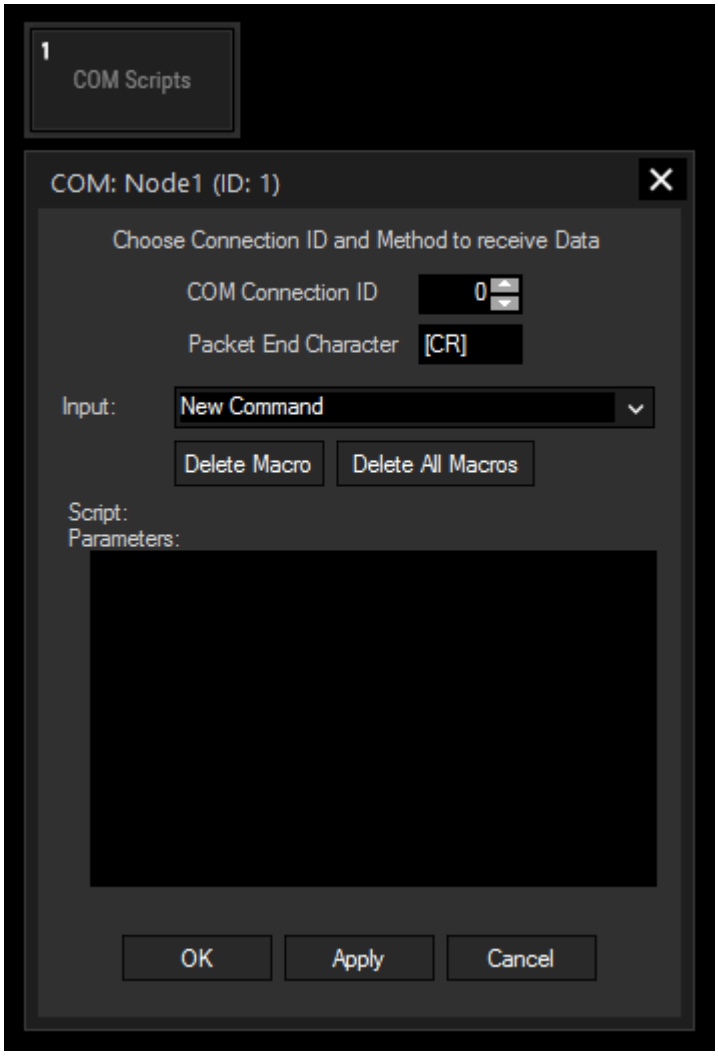
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.7.2 COM Scripts

The COM script node allows receiving either ASCII, decimal or hexadecimal values over a local COM port connection. Any input data can be linked to directly execute a dedicated script within WD. The COM Port connection needs to be enabled in the Connection Manager [1258].

This node can be found under Nodes > Scripts > COM



Node Properties

COM Connection ID:
Enter the ID of the COM Port Connection [1269].

Packet End Character:
Define here the character marking the end of a received packet.

Input:
Please enter the string of the incoming data for which you want a command to be executed. Please note that values can only be entered in ASCII, not hexadecimal, decimal or mixed. Special symbols are not supported. If you need those, please use the COM Port Input node [954] or COM ASCII Stream Input node [956] instead.

Script:
In the Script section you may enter commands to be executed when the according data is received. Each "Input"

string may have a different script assigned.
The topic Script Language [1511] explains this in more detail, for sophisticated scripts the use of Functions or Macros [1897] is recommended.

Type in a new string for the received data packet, type in a script in the Script section and press "Apply". The string will then be added to the **Input** drop-down and the script will be displayed upon selection. If you edit existing scripts, please remember to press "Apply" after changes.

Use **Delete Macro** to delete the selected script.
Use **Delete All Macros** to delete all scripts.

Node control

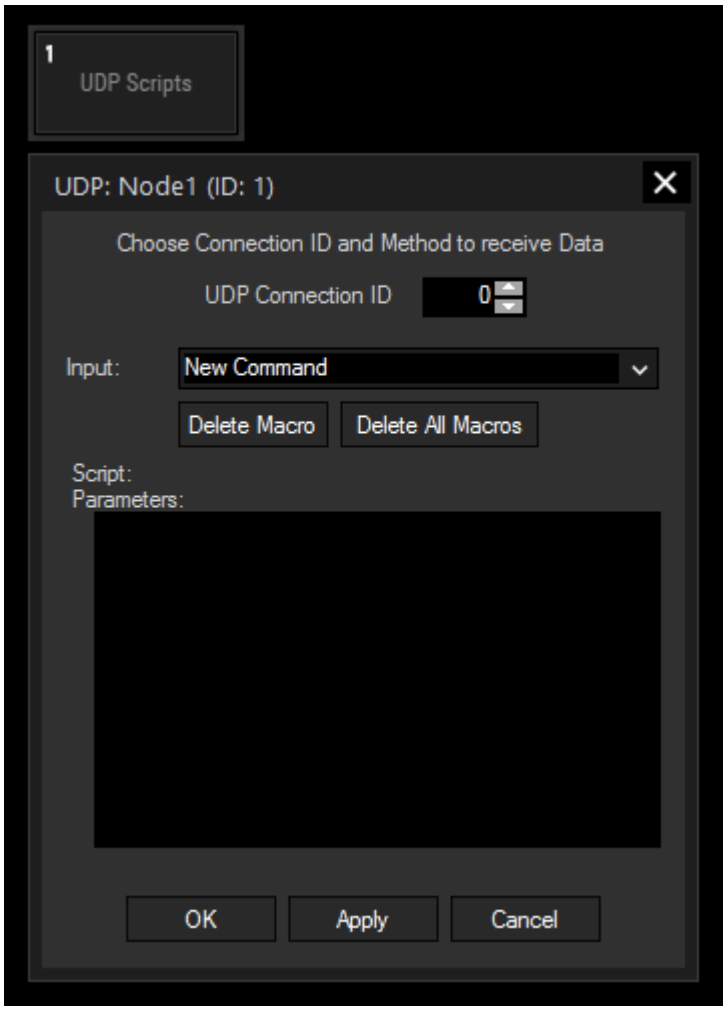
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.7.3 UDP Scripts

The UDP script node allows receiving either ASCII, decimal or hexadecimal values over a local UDP connection. Any input data can be linked to directly execute a dedicated script within WD. The UDP connection needs to be enabled in the Connection Manager [1258].

This node can be found under Nodes > Scripts > UDP



Node Properties

UDP Connection ID:
Enter the ID of the UDP Port Connection [1265].

Packet End Character:
Define here the character marking the end of a received packet.

Input:
Please enter the string of the incoming data for which you want a command to be executed. Please note that values can only be entered in ASCII, not hexadecimal, decimal or mixed. Special symbols are not supported. If you need those, please use the UDP Input node [975] or UDP ASCII Stream Input node [972] instead.

Script:
In the Script section you may enter commands to be executed when the according data is received. Each "Input" string may have a different script assigned.
The topic Script Language [1511] explains this in more detail, for sophisticated scripts the use of Functions or Macros [1897] is recommended.

Type in a new string for the received data packet, type in a script in the Script section and press "Apply". The string will then be added to the **Input** drop-down and the script will be displayed upon selection. If you edit existing scripts, please remember to press "Apply" after changes.

Use **Delete Macro** to delete the selected script.
Use **Delete All Macros** to delete all scripts.

Node control

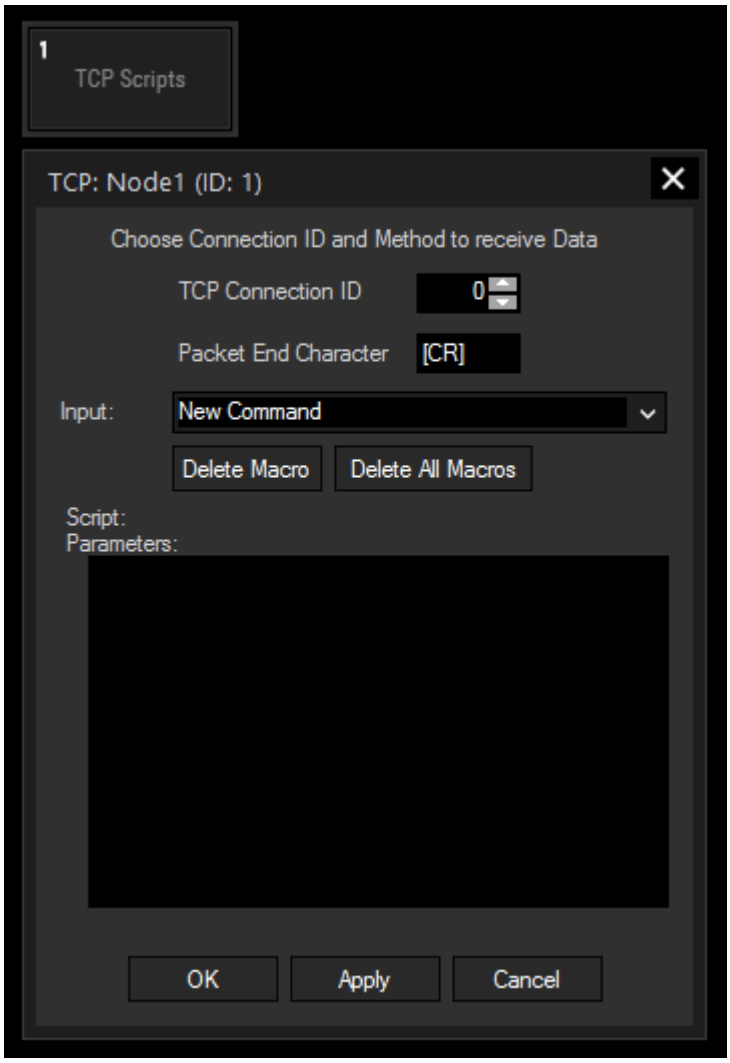
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.7.4 TCP Scripts

The TCP script node allows receiving either ASCII, decimal or hexadecimal values over a local TCP connection. Any input data can be linked to directly execute a dedicated script within WD. The TCP connection needs to be enabled in the Connection Manager [1258].

This node can be found under Nodes > Scripts > TCP



Node Properties

TCP Connection ID:
Enter the ID of the TCP Port Connection [1265].

Packet End Character:
Define here the character marking the end of a received packet.

Input:
Please enter the string of the incoming data for which you want a command to be executed. Please note that values can only be entered in ASCII, not hexadecimal, decimal or mixed. Special symbols are not supported. If you need those, please use the TCP Input node [968] or TCP ASCII Stream Input node [965] instead.

Script:
In the Script section you may enter commands to be executed when the according data is received. Each "Input" string may have a different script assigned.
The topic Script Language [1511] explains this in more detail, for sophisticated scripts the use of Functions or Macros [1897] is recommended.

Type in a new string for the received data packet, type in a script in the Script section and press "Apply". The string will then be added to the **Input** drop-down and the script will be displayed upon selection. If you edit existing scripts, please remember to press "Apply" after changes.

Use **Delete Macro** to delete the selected script.
Use **Delete All Macros** to delete all scripts.

Node control

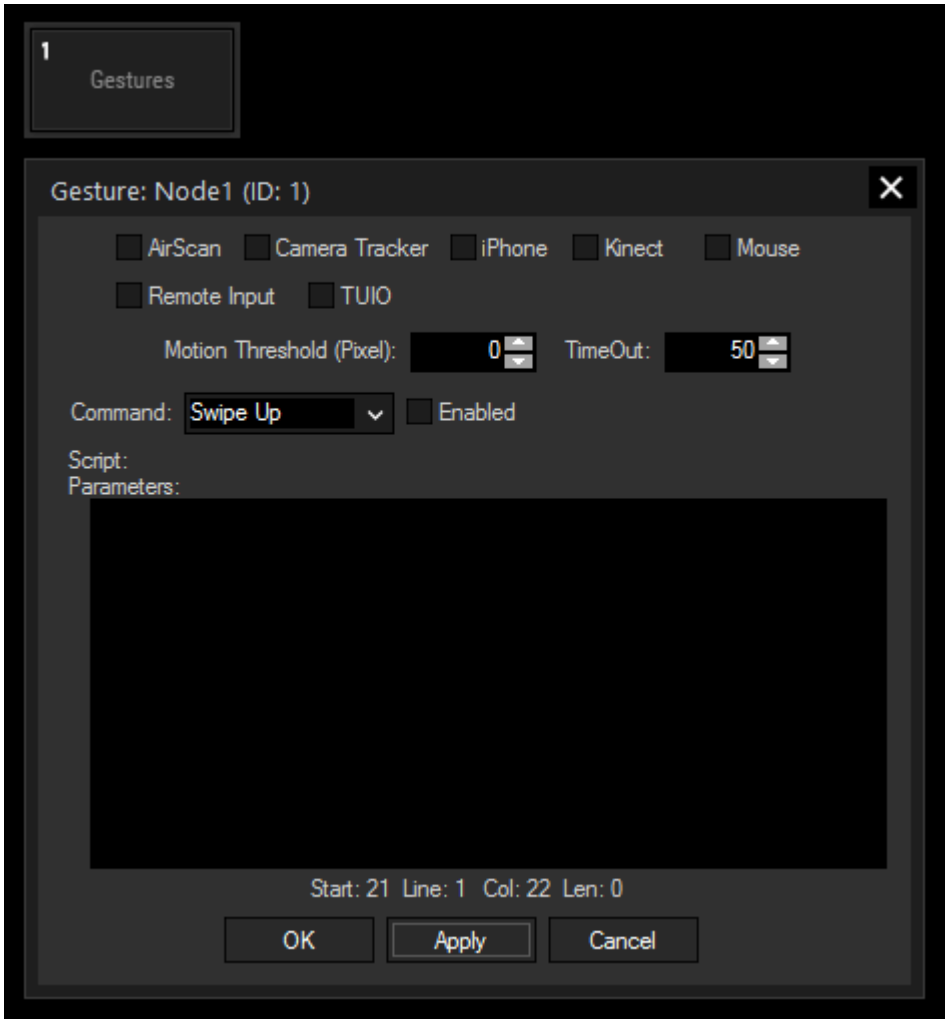
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.7.5 Gesture Scripts

The Gesture node enables you to track incoming touch or position values and execute specific scripts when a certain movement was detected. As soon as an active point is available in one of the tracking sources, the gesture detection starts.

This node can be found under Nodes > Scripts > Gesture



Node Properties

Data Sources:
Select here one or more sources for your gesture detection. Available as source are AirScan [1277], Camera

Tracker [1291], the WD iPhone [1275] app, the Microsoft Kinect [1283] controller, Mouse, Remote Touch input [1260] and TUIO [1261] data. If you select the mouse input, it will also react to touch data interpreted by the OS as mouse data. Mouse gestures will only be tracked when the movement is executed while the mouse is clicked.

Motion Threshold:
If you only want to recognize gestures that have minimum length, you can enter the pixel value here. If e.g. 300px is entered, only movements covering 300 pixels or more will be detected.

TimeOut (ms):
This setting allows you to track only movements with a certain minimum duration. A low value means that quick, short movements are registered, a high value will let the node react to long time moves.

Command:
Select here for which kind of movement you want to edit the script to be executed. Available gestures are Swipe Up, Down, Left and Right. Don't forget to check the **enable** box to activate the script for this gesture.

Script:
In the Script section you may enter commands to be executed when the according data is received. For a better overview, it is recommended to make use of Macros and Functions [1897] for sophisticated scripts.

Node control

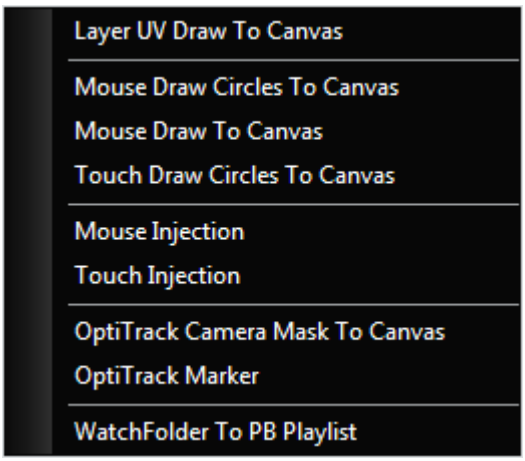
With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
In addition, the node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`.

7.4.8 Interaction Nodes

Interaction nodes execute an underlying application that enables to interact with a remote system. Please see the introductory chapter if you like to learn more about other node types or how to create and work with nodes [937] in general.

The following chapters describe the various Interaction nodes.

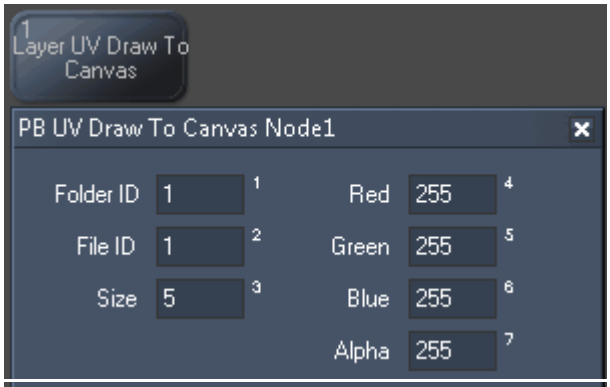


Layer UV Draw To Canvas [1245]
Mouse Draw Circles To Canvas
Mouse Draw To Canvas [1249]
Touch Draw Circles To Canvas
Mouse Injection [1250]
Touch Injection [1251]
OptiTrack Camera Mask To Canvas
OptiTrack Marker
WatchFolder To PB Playlist [1252]

7.4.8.1 Layer UV Draw to Canvas

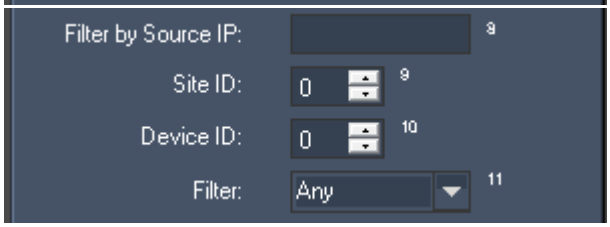
With the Layer UV Draw to Canvas Interaction node you may send mouse click data and mouse move data to a dedicated Canvas asset [280] in Pandoras Box to draw on it. In difference to the "Mouse Draw to Canvas [1249]" node, this node does not work with the mouse connected to the Widget Designer system but with a mouse / touch /... input connected to the PB Client or Master directly. This means that this node routes the remote input events to a Canvas.

Below the explanation of the node's parameters, you will find a step-by-step description of the settings from Widget Designer and Pandoras Box that are necessary when drawing on a Canvas, or receiving mouse or touch input events in general.

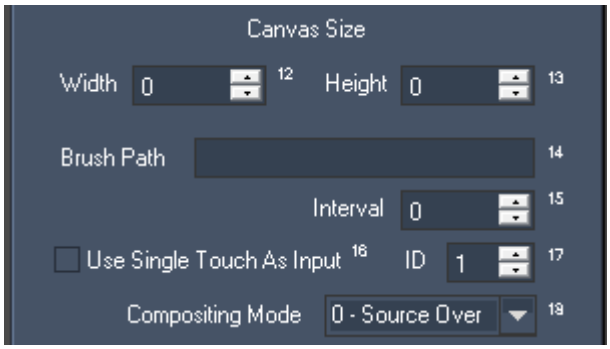


Set up the Folder and File ID from the Canvas Asset.

The Size parameter influences whether you draw a fine line or with a thicker brush.
With Red, Green, Blue and Alpha you set up the color of the brush. Note that small numbers next to parameter fields indicate that they can be changed with the command NodeSetParam [1782].



If you like to filter input data from one Site or even only a Layer, enter the PB Masters IP address, the Site and Device ID and set the filter type.



Enter the same Canvas width and height as set up in the File Inspector in Pandoras Box.

Enter a Brush Path if you like to draw with a media e.g an image. The asset name is the one displayed in the PB Master's Project tab (case-sensitive!), e.g. "Sub folder/Image.png".

The interval parameter influences how continuous or interrupted the drawn line looks like.

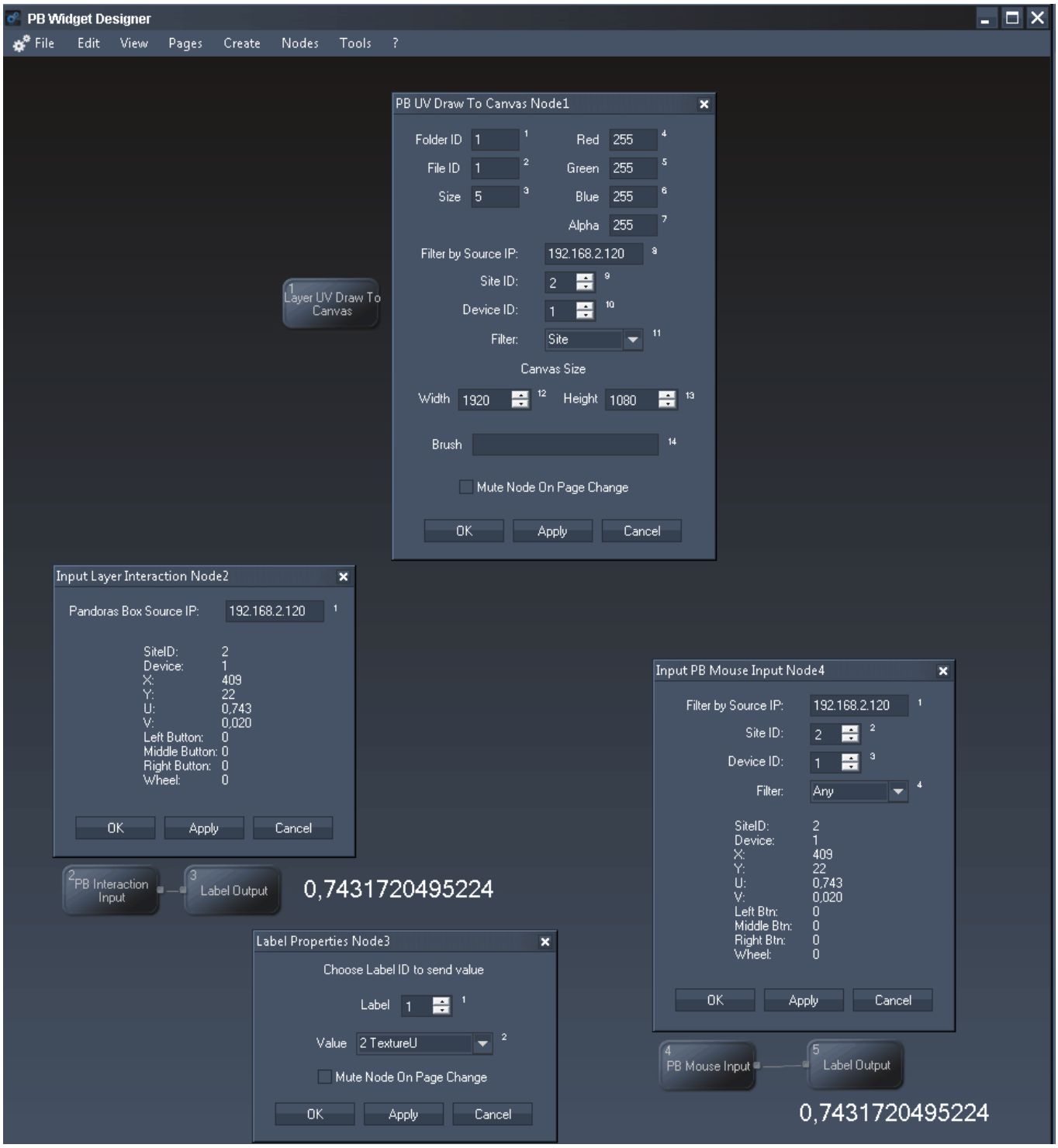
"Use Single Touch As Input" discards multiple input or touch data and takes only the input with the specific "ID".

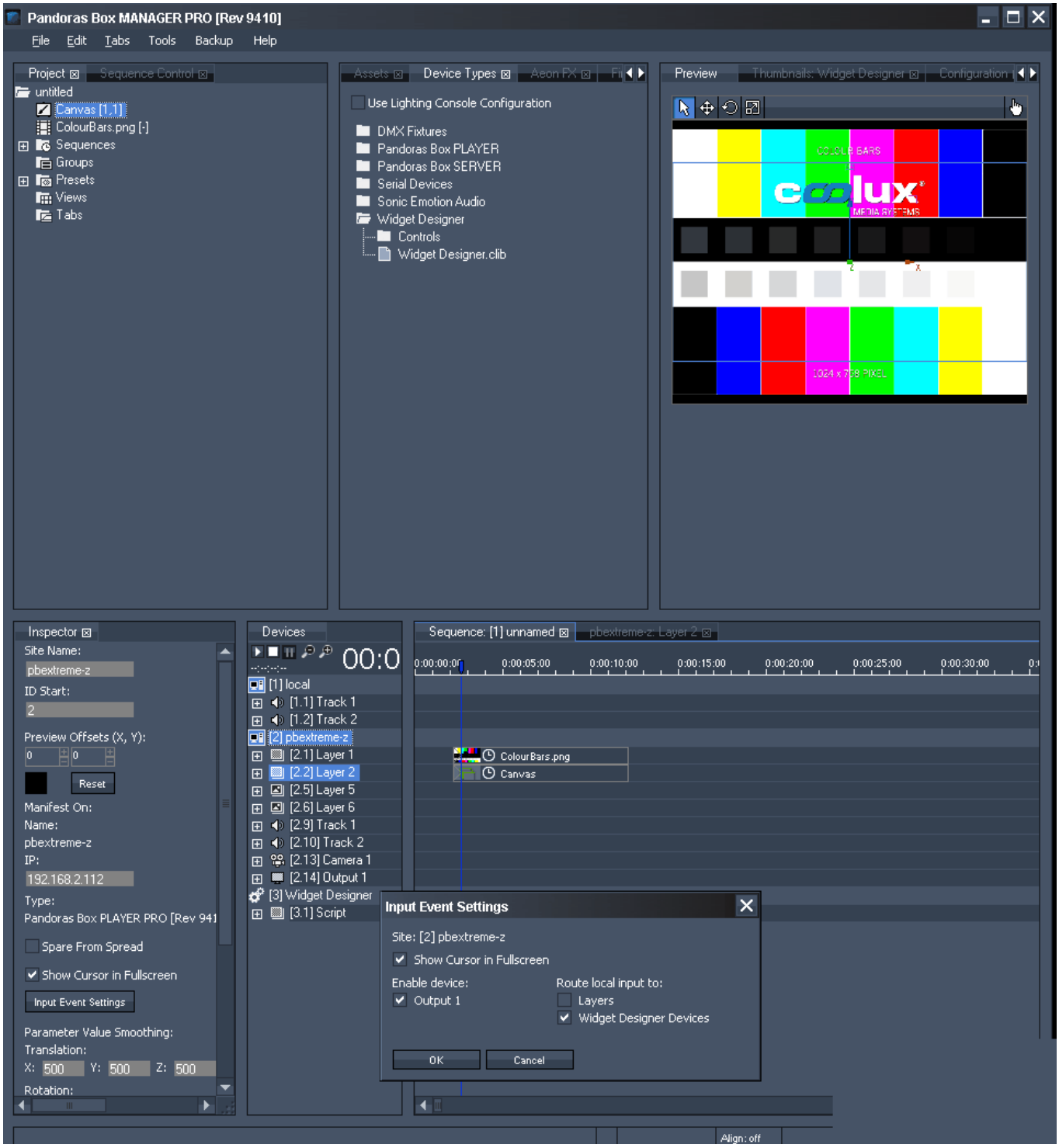
The Compositing Mode influences the Overlay behavior of semi-transparent (Alpha < 255) lines drawn on the Canvas. "Source Over" covers any underlaying layers, the Alpha level simply darkens the color. "Source Copy" does not cover existing layers fully but only according to the Alpha level, i.e. they shine through the Canvas Layer.

Tick the check box "Mute Node on Page Change" if the node should not process and output data when another page is active.

General connection settings in WD and PB required to communicate remote UV data

The below screenshot depicts settings in Pandoras Box and Widget Designer for a test setup that shows how to draw on a Canvas, or receive mouse or touch input events in general:
- one PB Master with WD on IP 192.168.2.120
- one PB Client (in fullscreen mode) on another PC with a mouse device attached


PB Widget Designer
File Edit View Pages Create Nodes Tools ?
PB UV Draw To Canvas Node1
Folder ID 1
File ID 1
Size 5
Red 255
Green 255
Blue 255
Alpha 255
Filter by Source IP: 192.168.2.120
Site ID: 2
Device ID: 1
Filter: Site
Canvas Size
Width 1920
Height 1080
Brush
Mute Node On Page Change
OK Apply Cancel
1 Layer UV Draw To Canvas
Input Layer Interaction Node2
Pandoras Box Source IP: 192.168.2.120
SiteID: 2
Device: 1
X: 409
Y: 22
U: 0,743
V: 0,020
Left Button: 0
Middle Button: 0
Right Button: 0
Wheel: 0
OK Apply Cancel
2 PB Interaction Input
3 Label Output
0,7431720495224
Label Properties Node3
Choose Label ID to send value
Label 1
Value 2 TextureU
Mute Node On Page Change
OK Apply Cancel
Input PB Mouse Input Node4
Filter by Source IP: 192.168.2.120
Site ID: 2
Device ID: 1
Filter: Any
SiteID: 2
Device: 1
X: 409
Y: 22
U: 0,743
V: 0,020
Left Btn: 0
Middle Btn: 0
Right Btn: 0
Wheel: 0
OK Apply Cancel
4 PB Mouse Input
5 Label Output
0,7431720495224



- in the PB Master, drag the Client into the Device tab and in addition a Widget Designer device [696] (from the Device Types tab > Widget Designer > Widget Designer.clib)
- select the WD Device and check the IP address in the Inspector [190] (in our case, as it runs on the same machine, it should be the local IP)
- in WD, go to Connections > PB Configuration [1256] and check the IP address from PB Master then scroll down to the "PB Widget Device Connection" section and tick the check box "Enable Connections"
- now, in the PB Master's Devices tab, the WD Device should not have a red exclamation mark anymore which means that it now is able to transmit (touch) data from the PB application to the WD application and its nodes)

Receiving remote mouse / touch data from a PB Client or PB Master

Before drawing on a Canvas we will check that the UV touch data is send from the PB Client to the PB Master and from the PB Master to WD.

- first drag a testpattern into the PB project and display it on the first layer of the Client (Site/Device ID 2.1)
- in the PB Master's Devices tab, select the Client to display the settings in the Inspector tab [210] and click the button "Input Event Settings [210]". Tick the options for enabling the Output as well as for the Widget Designer Device. Optionally, tick the check box to render the mouse pointer in the fullscreen window

- in WD, go again to Connections > PB Configuration [1256] to the "PB Widget Device Connection" section and click the button "Input Tester" to open a dialog that should display incoming information when moving the input device in the PB Client
- alternatively, you can receive UV data from an input device connected to the PB Master: toggle the Client into the local PB Master's Preview and change to the Picking mode (by clicking the hand icon), more information about the Picking Mode... [252]

- if you like to view the incoming data outside this dialog, create a small node chain and a Label. 1) Nodes > Input > Pandoras Box > Layer Mouse Input [1059] and 2) Nodes > Output > Controls > Label
- in the PB Mouse Input node, enter the Master's (!!) IP address. If you like not to receive data from all sites and devices, enter the according Site ID and Device ID and set the filter drop-down menu to suit your needs. For our purpose, this is not needed, so the filter can say "Any"
- in WD, set the Label Output node to display the "Texture U" data in the according Label. If you now move the Client's attached mouse from left to right in the fullscreen render window, the Label should display texture data from 0 - 1

If this is working, the data transmission works fine. Now we can set up to draw in the Clients screen.

Forwarding remote mouse / touch data to a Drawing Canvas

- first in the PB Master's Project tab, create a Drawing Canvas [280] asset and assign Folder / File ID 1/1 using the Inspector. Set up the resolution you need and then assign it to the Client's second layer (2.2)

- in WD, create the node Nodes > Interaction > Layer UV Draw to Canvas Pandoras Box. The obligatory settings are what Folder / File ID the Canvas asset has (in our case 1/1), what width / height it has and what the Source IP is (the PB Master is in our case the local IP). Optionally you can set up a color and size for the "pen", this is described at the top of this chapter.

- as already mentioned above, you can as well draw on the Canvas with the mouse connected to the PB Master: toggle the Client into the local PB Master's Preview and change to the Picking mode (by clicking the hand icon), more information about the Picking Mode... [252]

7.4.8.2 Mouse Draw to Canvas

With the Mouse Draw to Canvas Interaction node you may send mouse click data and mouse move data to a dedicated Canvas asset [280] in Pandoras Box to draw on it. The mouse data (happening in Widget Designer) is injected into the Pandoras Box system meaning that it is as if it was generated on the remote machine using a connected mouse for instance. Please see the next chapter about the node "Layer UV Draw to Canvas [1245]" if you like to draw on the Canvas with a mouse / touch /... input connected to the PB Client or Master directly.

Set up the Folder and File ID from the Canvas Asset

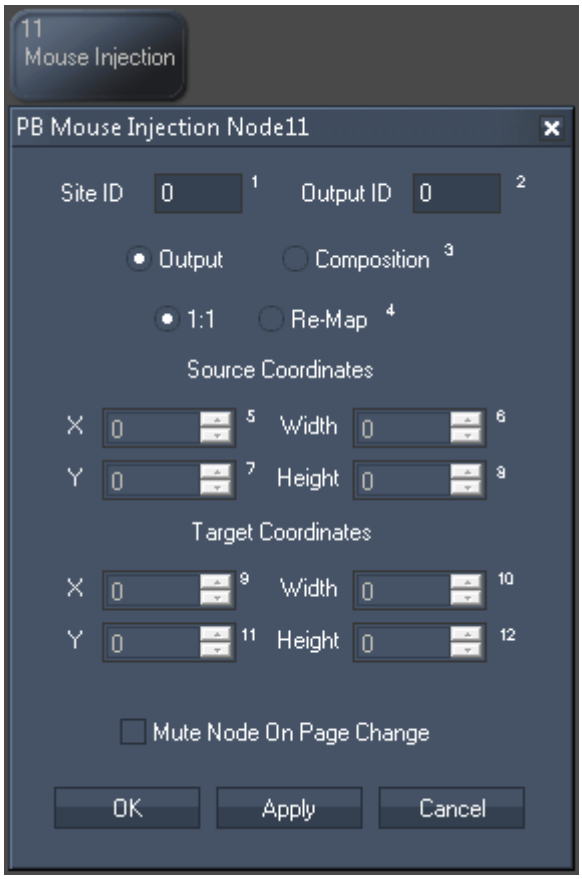
The Size parameter influences whether you draw a fine line or with a thicker brush.
With Red, Green, Blue and Alpha you set up the color of the brush. Note that small numbers next to parameter fields indicate that they can be changed with the command NodeSetParam [1782].

Choose between
- "1:1" to keep the resolution as on your local screen and
- "Re-Map" with more settings to send other coordinates to the remote computer
In case you send. This is not only of interest when the screen resolution of the systems you inject input events in is different to the local resolution. But also when the resolution is much higher due to more outputs. In case the outputs are softedged you need to set up two Injection nodes with different source and target coordinates.

Tick the check box "Mute Node on Page Change" if the node should not process and output data when another page is active.

7.4.8.3 Mouse Injection

With the Mouse Injection Interaction node you may send mouse click data and mouse move data to a dedicated Output of a Site. The mouse data is injected there meaning that it is as if it was generated on the remote machine using a connected mouse for instance.



Set up the Site ID and Output ID from the Master or Client that you like to send the data to.

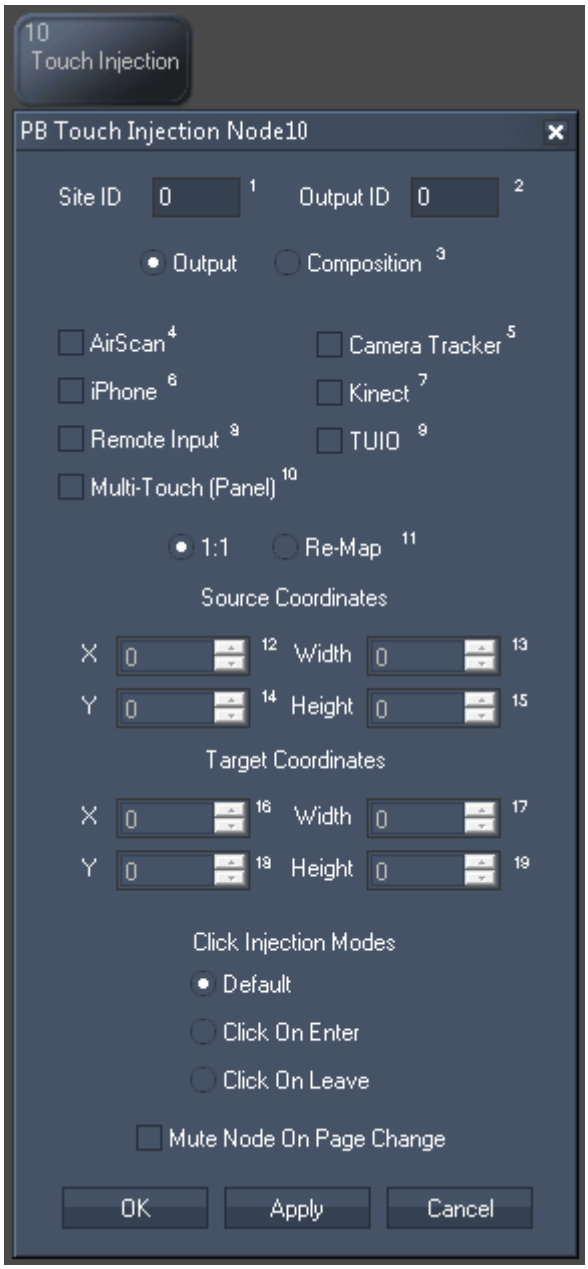
Decide whether the data should be injected to the Output or the Composition pass. More information... 320

Choose between
- "1:1" to keep the resolution as on your local screen and
- "Re-Map" with more settings to send other coordinates to the remote computer. This is not only of interest when the screen resolution of the systems you inject input events in is different to the local resolution. But also when the resolution is much higher due to more outputs. In case the outputs are softedged you need to set up two Injection nodes with different source and target coordinates.

Tick the check box "Mute Node on Page Change" if the node should not process and output data when another page is active.

7.4.8.4 Touch Injection

With the Touch Injection Interaction node you may send touch data to a dedicated Output of a Site. The touch data is injected there meaning that it is as if it was generated on the remote machine using a connected touch screen for instance.



Set up the Site ID and Output ID from the Master or Client that you like to send the data to.

Decide whether the data should be injected to the Output or the Composition pass. More information... 320

Tick the according check-box for each tool that should generate touch data.

Choose between
- "1:1" to keep the resolution as on your local screen and
- "Re-Map" with more settings to send other coordinates to the remote computer. This is not only of interest when the screen resolution of the systems you inject input events in is different to the local resolution. But also when the resolution is much higher due to more outputs. In case the outputs are softedged you need to set up two Injection nodes with different source and target coordinates.

Lastly, you may set up what mode generates a click.

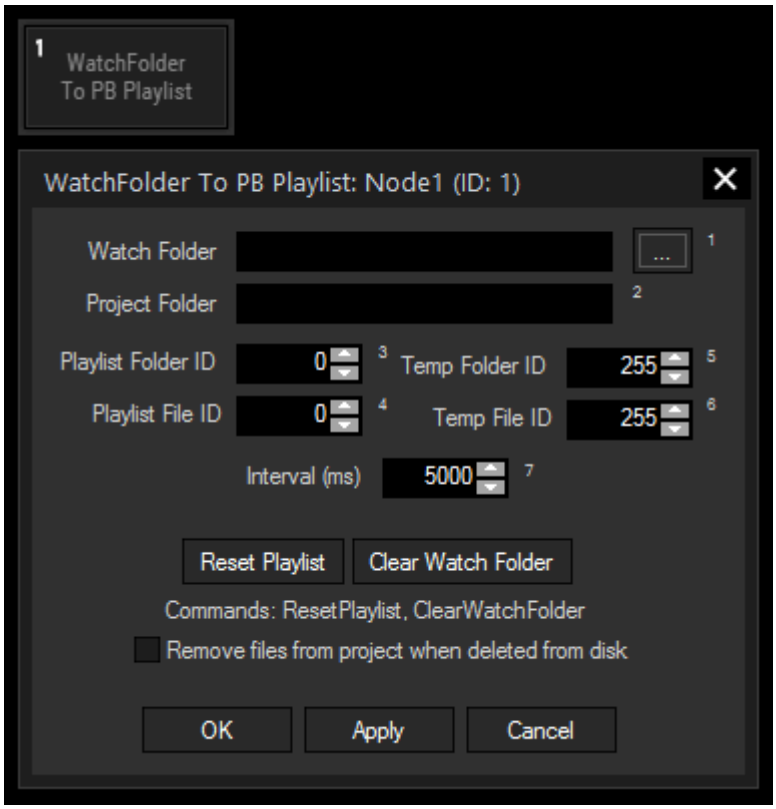
Tick the check box "Mute Node on Page Change" if the node should not process and output data when another page is active.

7.4.8.5 WatchFolder to PB Playlist

The WatchFolder to PB Playlist node enables you to create an easy to use automation for exchanging content of a Pandoras Box Playlist [239]. Content that is copied to the watch folder is automatically imported into the PB project and loaded to the prepared Playlist. Likewise, content removed from the watch folder is also removed from Playlist and project.

The files will be loaded into the Playlist with the default values set up as Playlist Initial Values in the Configuration tab [141].

This node can be found under Nodes > Interaction Nodes > WatchFolder to PB Playlist



Node Properties

Watch Folder:
Press the button on the right to select the folder to be watched from the explorer, or enter directly the file path in the text field.

Project Folder:
Enter the name of the PB folder into which the watch folder files should be loaded. The folder name without the project name is sufficient, e.g. "Playlists/NewContent".

Playlist Folder / File ID:
Enter here the file and folder IDs assigned to the prepared Playlist asset.

Temp Folder / File ID:
For automatically adding a new file to a Playlist, it needs to be assigned with a file and folder ID. This ID is then removed and assigned to the next item to be loaded and is thus used only once. Assign here a file and folder ID that will not be used by any other asset in your project.
Please note that IDs higher than 255 cannot be used in this case.

Interval:
Enter the interval time in milliseconds in which the watch folder shall be checked for changes.

Reset Playlist:
Press this button to refresh the watch folder request. This action will load the content of the watch folder into PB and refresh the Playlist content accordingly, regardless of the files already being present in the project.

Clear Watch Folder:
Press this button to permanently delete all files within the watch folder. This action will also remove the deleted files from the PB project and Playlist.

Remove files from project when deleted from disk:
Check this option to delete files also from the PB project and Playlist when they are deleted from the watch folder. When this option is not active, file entries will remain in the project and Playlist and the content cannot be played back. Note that image files are cached internally, so it might happen that deleted files are still visible in PB, however, when the project is reloaded the internally cached files are deleted when the file cannot be found anymore.

Please note:
When copying large files into the watch folder, it can occur that the file is detected and loaded into PB before the process of copying is finished, which results in an inconsistent asset.
This file must be **reloaded** (File Inspector [191] > Select file in table > press Reload) when it finished copying.
Alternatively, you can either set the node's **Interval** to a time that is sufficient for the copy process to finish and, if necessary, revert it back afterwards.
Or press the **Reset Playlist** button to import the file again.

Node control

With Widget Designer version 6, all nodes support so called node commands [1928]. Node commands access functions (i.e methods) from a node and / or set a parameter (i.e. property).

Enter "node", followed by the according ID and a dot and a list will pop up showing all available commands for the node. For instance, `Node1.TintColor.SetRGB(125,0,255)`, colors the node in purple.
The node properties with a parameter ID (the small superscript number) can be edited via the command `Node1.SetParam(ID,new Value)` or `WDNodeSetParam(NodeID,ParamID,Value)`, e.g. `Node1.SetParam(7,60000)` sets the Interval time to 1 minute.

In addition to the general commands, this nodes provides specific function commands that can replace pressing the buttons in the Configuration manually. E.g. `Node1.ClearWatchFolder` executes the clear action from the button.

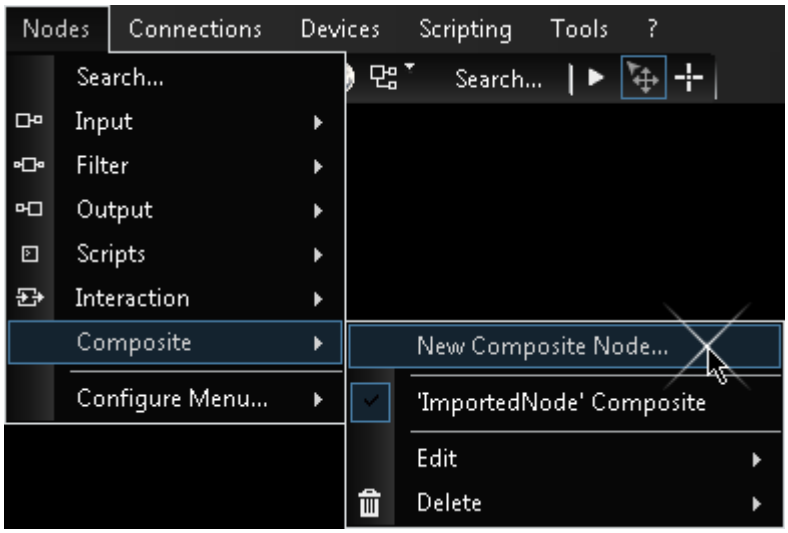
The following function commands are available for the WatchFolder to PB Playlist node:

- ClearWatchFolder
- ResetPlaylist

7.4.9 Composite Nodes

The Composite node offers you the possibility to create your own sophisticated node structure [937] and bundle it up to one compact, reusable node.

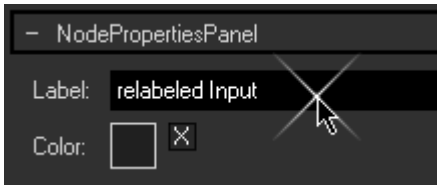
It allows you to arrange your entire node system much more clearly and to integrate larger structures easier and faster.



To construct a new Composite node, open the Nodes menu, select Composite Nodes and click on "New Composite Node...". If you have done this already or imported one, this sub menu lists available Composite nodes to create or edit them.

After you have entered a significant and valid name for your node (name may contain letters, numbers and underscore, and has to start with a letter), a new window will pop up, already containing a Comp. input and a Comp. output node.

The Comp. input and output nodes represent the interfaces to your main node system, similar to the inputs and outputs at any ordinary filter node. You can add or remove as many of them as you need, either with copy / paste of the existing ones, or by creating new ones out of the node menu.



It is highly recommended to label those input and output nodes according to their purpose, as this name will be visible at the Composite node's configuration later and thus simplifies assignment. To relabel a node, please open the node's properties either by right-clicking at the node and selecting "Item Properties", or with the keyboard shortcut [Alt + P], and enter a suitable label. Please do not change the name of the node (field at the top), unless you really need to, as this is not the one being displayed later.

The next step would be setting up your node structure as needed, for this you have all input, filter and script nodes available. You can even place widgets inside of your Composite node if necessary.

If you decide to provide your Composite node to others, please bear in mind to do a proper documentation how to setup the required environment. For example, If you have a TCP input node in your Composite node, your customer has to prepare the TCP connection [1258] correctly beforehand, or has to be informed to import this part of data from your project.

When you are done with your setup, simply close the Composite node window. A dialog asks you, whether you like to create an instance of the new node immediately.

Editing an existing Composite node is possible as follows
- via the Node menu > Composite > Edit > your Node
- right-click on the Node > Configure Node > button "Edit 'Name' Composite"

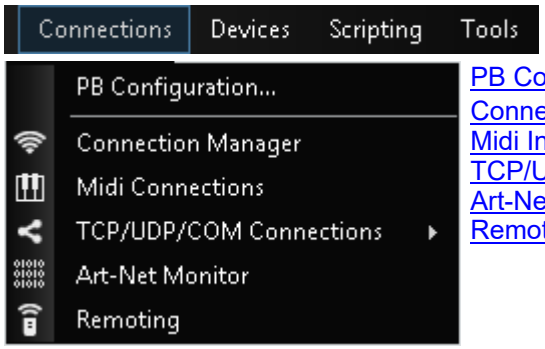
However, the later method enables you to view the concrete values handled by this instance. Opening the configuration dialogs of the nodes inside of the Composite window, displays all input and output information, an essential tool for testing and debugging your node system.

Please bear in mind that if you edit a Composite node while one or more instances are already located in the project, the changes will affect all of those. What you edit is always the master node and never a specific instance of it, despite the values shown inside of it.

7.5 Connections

The Connections menu in WD lists all available connection possibilities you can set up in Widget Designer. Other than the listed input and output protocols, you can connect to devices [1277] and other tools [1483].

Once you have chosen an entry a dialog opens with more options.

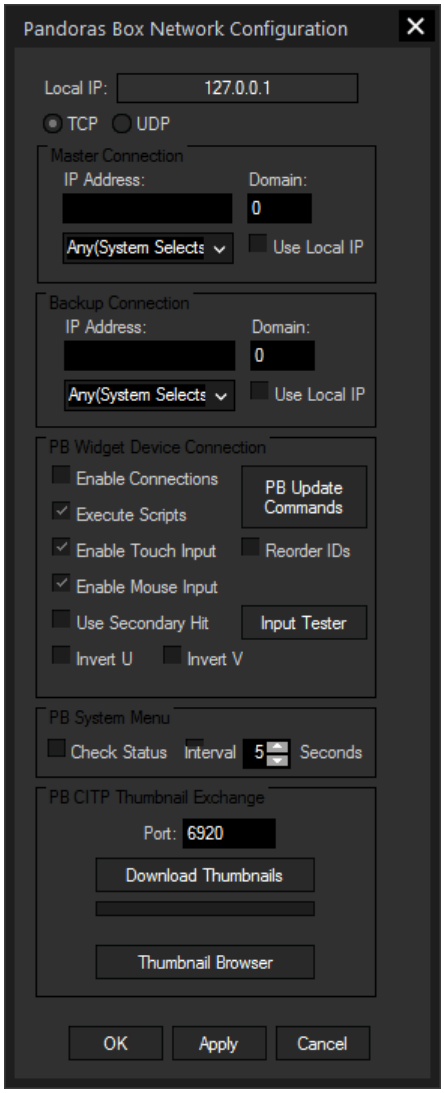


PB Configuration [1256]
Connection Manager [1258]
Midi Input [1485]
TCP/UDP/COM Connections [1272]
Art-Net Monitor [2051]
Remoting [1273]

7.5.1 PB Network Configuration

Depending on how you want to use and integrate Widget Designer (WD) to control Pandoras Box (PB) or other devices, the first thing you might want to set up is the network communication. When working in a network environment you need to make sure that the IP address, domain and network adapter are correct otherwise you might not be able to send or receive control data.

To access the PB network configuration dialog click "Connections" > "PB Configuration" in the menu bar, alternatively right-click anywhere on the empty main background window and choose the command in the context menu.



In the top section the local IP address is displayed. When the PC is not connected to a network, the local IP is 127.0.0.1. If you make any changes in the network adapter, please save your project settings and restart Widget Designer.
The next options are needed if WD should communicate with a PB system. If you are running WD on the same computer with the PB Master system that you would like to control, all that needs to match is the domain setting. Per default the domain is 0, if it is changed it can be found in the Configuration Tab [139] of Pandoras Box.

WD V6 can be used with Pandoras Box V5 and V6. For controlling older versions of Pandoras Box, please use WD V4.7 or lower.

TCP / UDP
The connection to Pandoras Box may be set up via TCP or UDP. The ports that are used are 6211 for TCP and 6212 for UDP.

TCP: this connection is bi-directional, it enables you to receive feedback messages from Pandoras Box besides sending commands only.
UDP: this connection is uni-directional. You will only be able to send commands to Pandoras Box without getting any feedback messages.

Pandoras Box Master Connection
If you would like to connect to Pandoras Box Master please enter the IP and domain of the Pandoras Box Master Device here.

In case your systems use multiple network adapters simultaneously, please always check the IP address in the asset browser tab [131] of Pandoras Box to get the right IP address of the network adapter in use.
When using the regular or Unlimited version of Widget Designer, you may define a dedicated network adapter used for transmitting this protocol. For more information about multiple network adapters please see the chapter Connection Manager [1258].

Pandoras Box Backup Connection
The network configuration dialog also allows you to set up a backup IP and domain. This setting is used when your Pandoras Box system is set up as full redundant backup with two Master systems. If you wish to send all Pandoras Box related controls to both the Master and backup system then enter all fields here.

Pandoras Box Widget Device Connection
The device type "Widget Designer" in Pandoras Box allows to execute commands directly from its Sequence in a very convenient workflow. Commands can be stored within keys. In addition to that, trigger values can be sent to the Widget Designer application. Furthermore, mouse and touch data from the PB Master or Client can be forwarded to Widget Designer either in total screen resolution or interpreted in UV data.

The check box "Enable Connections" allows Widget Designer Devices in Pandoras Box to connect to this WD application via TCP port 6213. "Execute Scripts" allows that commands programmed into a Sequence key within the Widget Designer Device can be executed by this WD application. The button "PB Update Commands" updates the available commands in PB's drop-down list, e.g. if you are using an older PB version or newer WD version. The topic "Widget Designer Device [696]" explains the workflow in more detail.

"Enable Touch" and "Enable Mouse" allow touch and mouse inputs to be transmitted. This is explained in more detail in the PB chapter about Layer Picking [252]. The button "Input Tester" shows incoming data with following

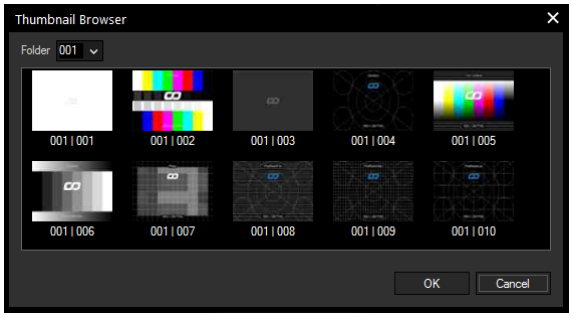
allocation: Action, IP address, Site and Device, TouchID, (Screen resolution in pixel) X and Y, (Texture coordinates in vector size) U and V.
The Site and Device ID relate to the layer that is "touched" by the touch point or mouse cursor, depending on the Input Event settings [210] of the Site in PB, i.e. whether "Route to Layers" is checked. In special applications, there might be an overlaying foreground layer with different texture coordinates. In that case you want to disregard this first layer but route the touch / mouse input to one layer further done. This is achieved by clicking the check box "Use secondary hit" in Widget Designer.
For inverting the texture coordinates (from 0→1 to 1→0) choose the according "Invert U" and / or "Invert V" option.

PB System Menu
Some nodes [936] in WD (e.g. Nodes > Input > Pandoras Box > System Menu (or search for PB Menu) connect to the PB System Menu [2097] and receive status information. Click the check box to enable the connection and choose how often it is updated.

CITP Thumbnails Exchange
Up from V4.5 of Pandoras Box you may want to download the media thumbnails for later use in button controls.

The thumbnail exchange is provided via the CITP thumbnail exchange of Pandoras Box. Before downloading any thumbnail please first activate the CITP mode in Pandoras Box configuration tab [148] and make sure that the IP domain and port matches and is applied in WD.

Please keep in mind that CITP is transmitting only indexed media. This means that any media that you would like to download needs to have a Folder and File ID [191] assigned in Pandoras Box. Once this is set up, you may press [Download Thumbnails] and once the gray progress bar is completed, all thumbnails of Pandoras Box will appear.



To get an overview of the downloaded images you may click on [Thumbnail Browser] to check if all files were successfully received and stored into WD file structure.

Please be aware that all thumbnails are stored into the WD file structure based on the given source IP address. Once you change the IP address of Pandoras Box you will have to download the files again in order to have WD find them on the local hard drive system.

7.5.2 Connection Manager

The Connection Manager is the tool [1483] to setup all main input and output communication protocols. All connections and protocols set up here are accessible to any command and user control as well as Nodes for in and output communication.

For further information please see

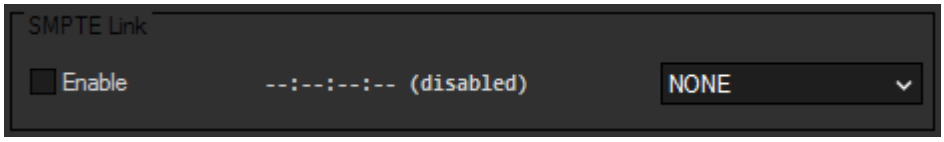
- SMPTE Link [1258]
- Art-Net [1259]
- DMX Link [1260]
- MA Net Input [1260]
- Midi [1260]
- WD Remote Touch Input/Output [1260]
- TUIO Transmitter [1261]
- TUIO Receiver [1261]
- ID Tag Tracking Server
- PSN MA Tracking Server [1262]
- OptiTrack Camera Connection Manager [1262]
- ID Tag [1263]
- IP Info & Setup [1264]
- TCP Connections [1265]
- UDP Connections [1267]
- Serial COM Connections

Working with multiple network adapters

Where applicable, a protocol can be assigned to a dedicated network adapter in Widget Designer and Widget Designer Unlimited Webclients. This is of interest when working with multiple protocols that you would like to separate from each other or when dedicated IP addresses / ranges should be supported.

Per default, the Widget Designer Free Version uses the same network adapter for each protocol. It cannot access the drop-down list to choose another adapter. If you open a new project, the drop-down list will always say "Any (System Selects)". That means that the operating system Windows decides what adapter is chosen. In general that would be the one that you have set up as the primary adapter in the (advanced) network settings.

7.5.2.1 SMPTE Link



When a SMPTE Link device is connected to the WD computer, choose if you want to send or receive SMPTE timecode.

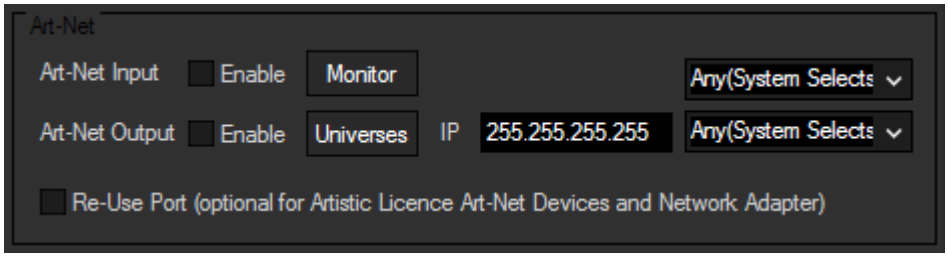
Choose the framerate from the second drop-down list if you are sending timecode.

In the text field next to the framerate you will see the current timecode. When you are in send mode, it allows you to enter the timecode start time manually.

Use the "Timecode [928]" control to display the current timecode on the WD userinterface. The SMPTE Link device can be controlled via commands [1511].

You may also route the sent / received SMPTE Timecode via the SMPTE Link Input Node [965] within the WD Node System.

7.5.2.2 Art-Net



Art-Net can be used both as input and output. Art-Net can technically transmit up to 256 DMX Universes but you should be cautious about the amount of Universes to be sent and managed from one Widget Designer System. The processor and network card limit the amount of possible Art-Net universes.

By default if you enable Art-Net output, there are no values being sent until you use either an Art-Net Output Node [1179] or Art-Net output commands [1524]. Once, a particular DMX channel is sent on a given Art-Net Subnet and Universe, Widget Designer will store that universe as an output universe and constantly update this Universe. As defined in the Art-Net specifications, as soon as one channel is transmitted with a certain value, all other channels are included in the packet, if not specified, with a value of "0". In other words, a single channel data can not be send, it is send with the entire universe.

To display the Art-Net data received by Widget Designer, click on [Art-Net Monitor [2051]]. In the dialog you may select a Subnet and universe. Then all channels are displayed with the according incoming Art-Net value. If you use WD to send Art-Net and you do so by broadcasting the values, you will be able to see these values in the Art-Net Monitor too. If you are unicasting the values, please switch to another Art-Net Monitor, e.g. the one in the PB Menu [2097].
More information [2083] about broadcast and unicast...

If you wish to stop sending specific Universes to the network you can open the Art-Net Universe List from the connection Manager and remove a Universe from the output list by right-clicking on the desired Subnet and Universe to delete the entry. Keep in mind, that a Universe is sent as long as it is listed here, even if all Art-Net output nodes were deleted. Right-click and choose "Refresh List" e.g. if you started sending a Universe again.

You may also use commands to deactivate [1523] a certain Subnet and Universe from being updated constantly anywhere where scripts are applicable in the user interface.

Reuse Port:
When reuse Port is checked Artistic License devices that require a reuse UDP port option are supported.

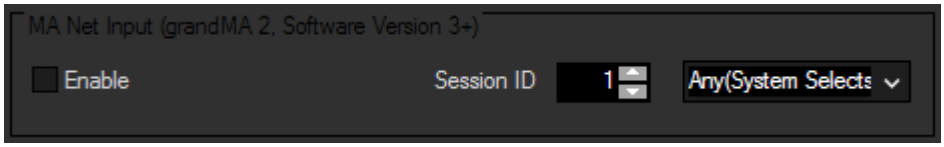
7.5.2.3 DMX Link

WD supports DMX Input data via our DMX Link IN device [1999]. To use this function, activate the check box "Enable".

You may receive and process the incoming DMX data via the DMX Link Input node.

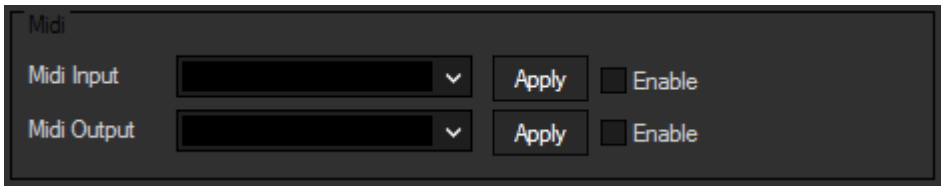
DMX Output is supported via our DMX Link OUT device. To use this function, activate the check box "Enable".

7.5.2.4 MA Net Input



Enable the MA-Net connection for the use of the MA-Net Input node and retrieve values from an MA-Lighting console.

7.5.2.5 Midi



For Midi input and output connect and install a windows compatible midi device with the Widget Designers computer first in order to access and choose the device in the Connection Manager.

The regular Widget Designer and Unlimited version support up to 8 Midi inputs and outputs that can be configured via the Connections > Midi Connections [1271] dialog.

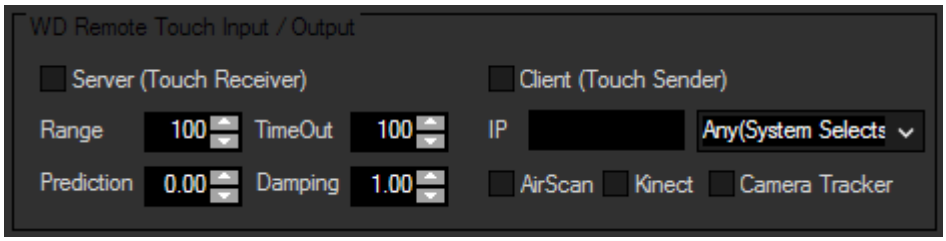
You can use MIDI connections via a tool, node or script command, the possibilities are listed below. In any case, please make sure to enter the correct ID referring to the numbering in the Midi Connections [1271] dialog. When no ID is defined, ID 1 is accessed, which is the device set in this Connection Manager dialog.

Tools > Midi Input [1485] > Midi Notes [1486], Midi Messages [1487], or one of the devices AKAI APC40 [1488] or BCF 2000 [1489]

Nodes > Input > Connections > Midi Input [959] or Midi Note Catch [960]
Nodes > Output > Connections > Midi Note On/Off [1183], Midi Raw Message [1184] or Midi Value [1185]

Most Midi commands start with MidiSend... [1557] Please open the chapter "Script language [1511]" for more information about the use of it.

7.5.2.6 WD Remote Touch



The WD Remote Touch Input/Output allows to share touch data within several Widget Designer instances (one WD Master receives the data, several WD Clients may send it). This enables you to work with multiple cameras, AirScans or Kinect devices. The touch data handling can be split to several processes or computers.

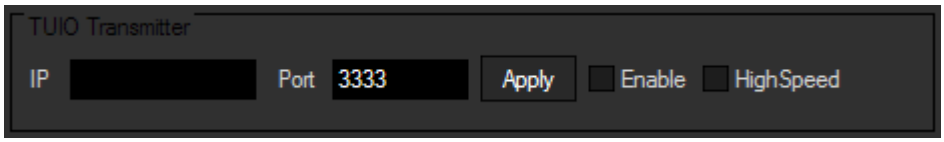
Server (Touch Receiver):
Enable the server if you want to receive touch data from other WD instances.

Client (Touch Sender):
In order to send the touch data of this WD instance to another WD (set up as Remote Touch Input Server), enter the IP address of the WD Server and tick the check box [Client (Touch Sender)]. The receiving WD may run on the same computer or on a different one.

Enable the devices that should send its touch data to the WD Server:

AirScan [1277], Camera Point Tracker [1291] and/or Kinect [1283].

7.5.2.7 TUIO Transmitter



The open source protocol TUIO allows outputting the AirScan's multi touch data to other applications

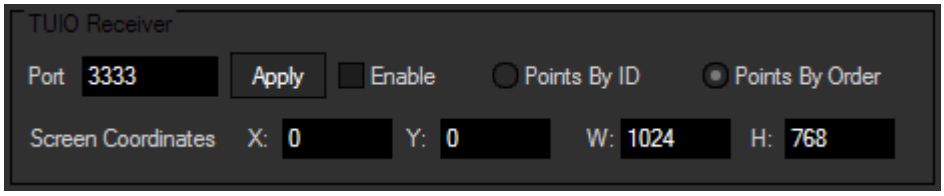
This protocol is widely used around the world by many application developers and is a commonly known way to transmit the individual touches.

See a list of compatible software and operating system frameworks here:
http://tuio.org/?software

To use the AirScan with TUIO, set up the TUIO host IP and Port here and press [Apply]. The TUIO transmitter needs to be enabled by ticking the check box [Enable].

To start sending the AirScan's touch data, open the AirScan Tool [1277] and check the option [TUIO] inside the Multi Point Mode section.

7.5.2.8 TUIO Receiver



The TUIO protocol is widely used around the world by many application developers and is a commonly known way to transmit individual touch data.

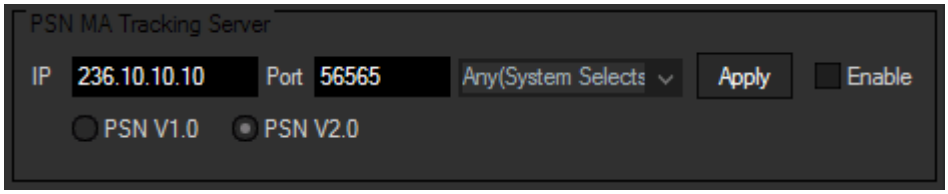
See a list of compatible software and operating system frameworks here:
http://tuio.org/?software

The received coordinates can be processed with the TUIO Input node.

You can setup your Screen Coordinates according to your TUIO touch region to generate the correct pixel values in the node. X and Y represent the X and Y offset in pixels of your TUIO region, W and H are width and height, also in pixels.

7.5.2.9 PSN MA Tracking Server

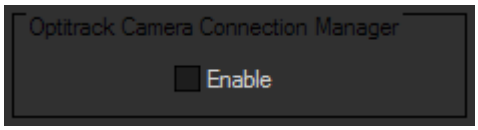
In the Connection Manager [1258] dialog, the section PSN MA Tracking Server allows to setup whether and how Widget Designer outputs PSN data. The PosiStageNet protocol is for example supported by MA Lighting consoles; more information can be found on www.posistage.net. The data itself is generated for example in the ID Tag Tracker node [985].



Please enter the IP address and port whereto the PSN data should be send to and click "Apply". For MA Lighting consoles, the default IP is 236.10.10.10 with port 56565.

Check "Enable" to activate the protocol in general and choose whether you like to send "PSN V1.0" or "PSN V2.0".

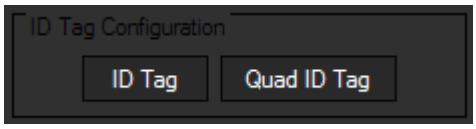
7.5.2.10 Optitrack Camera Manager



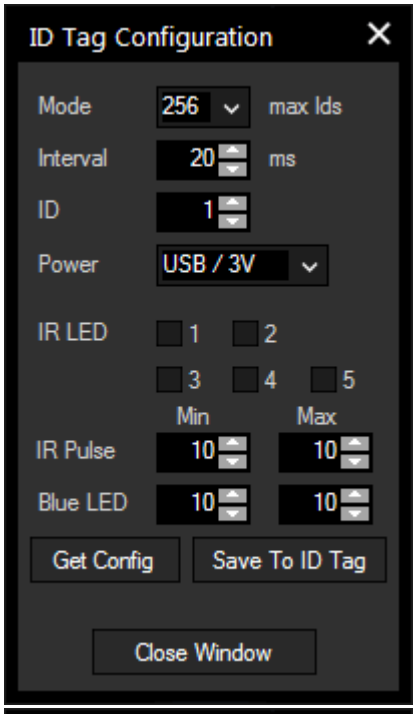
In order to communicate with Optitrack cameras, Widget Designer includes the Optitrack Camera Connection Manager. It is able to find and communicate with any Optitrack camera connected to the network without further user input.

Enable this feature for ID Tag Tracking [985].

7.5.2.11 ID Tag Configuration

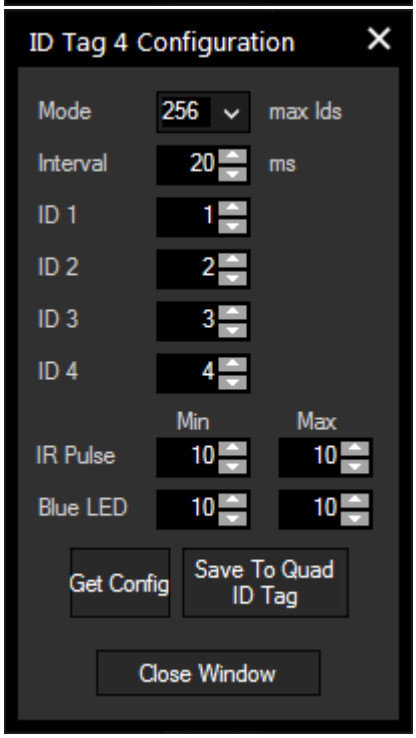


This topic describes how to configure a tag software-wise. For general information about IR ID tracking and links to other chapters please see the topic ID Tag Tracker [985].



(Single) ID Tag
The software configuration of the tag involves to set up the number and power of the IR LEDs as well as the brightness of the blue control LED. Further, the blinking mode including the ID can be set up.

The tag can be configured via the onboard Micro-USB interface. Connect the tag via the delivered USB cable (Micro-USB to Standard Type A plug) to your computer running Widget Designer. If you connect it the first time, and the driver cannot be found automatically, use the USB-driver "coousb-driver v6.0.13.0" (or a newer one) to finish the hardware installation of the device.



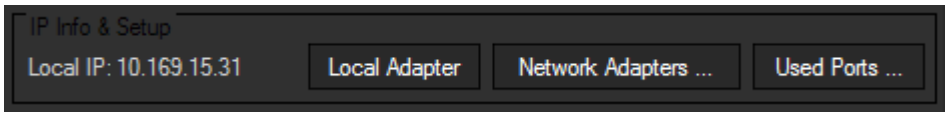
Quad ID Tag
The software configuration of the tag involves to set up the brightness of the IR LEDs and the brightness of the blue control LED. Further, the blinking mode including the ID for each port can be set up. Note that it is possible to set each port to an individual ID or to share the same ID.

The tag can be configured via the onboard Mini-USB interface. Connect the tag via the delivered USB cable (Mini-USB to Standard Type A plug) to your computer running Widget Designer. If you connect it the first time, and the driver cannot be found automatically, use the USB-driver "coousb-driver v6.0.14.0" (or a newer one) to finish the hardware installation of the device.

Mode	This influences the maximum ID a tag can hold. It is recommend to choose the smallest mode available that covers the number of tags you use. "4" allows four tags holding the ID 1,2,3 and 4. If you want to use ID 5, you need to choose a higher

	mode, e.g. 8. The tags transmit their ID with a certain blinking pattern that needs to be seen by cameras. The higher the mode is, the longer this pattern gets. Hence, the more time needs to pass in order to recognize a tag if it (re-)appears in the tracking area. Please note that it is possible to set up several ID tags with the same ID. This does not mean to increase the mode as long as the ID fits into the range. In the section "3D Tracking Settings" 1002 you can set up that "same" IDs should result in one averaged position.
Interval	This influences how fast the above described blinking pattern is transmitted. Per default it is set to 20ms. The lower the frame rate of the used cameras is, the longer an interval needs to be, the slower a tag is recognized. For a frame rate below 100-120fps, an interval of 40 ms is recommended. That applies for most USB cameras. The higher the frame rate of the used cameras is, the shorter an interval can be, the faster a tag is recognized. A frame rate of 120fps allows to lower the interval to 20ms; 150fps allows even 10ms. However, a longer interval still works with fast cameras and is more robust. In theory, a 10ms interval also works for 120fps but in practice this is very prone to error, hence an 20ms interval is better.
ID	The "ID" depends on the "Mode" chosen above and means the identification number of the connected tag.
Power	Choose the power source of your tag. Choose "USB / 3V" if you use the onboard Micro-USB interface with the delivered battery pack or your own 3V power source Choose "A675 Cell" if you use two onboard hearing cells. See more information regarding the power options in the chapter "Required Hardware" 986.
IR LED	Choose which LEDs should blink. 1 and 2 are the onboard ones. Only the "USB / 3V" power source allows to use all LEDs at the same time.
IR Pulse	Set the minimum and maximum brightness used in the blinking pattern when transmitting the tag's ID. It is recommended to set the minimum to 0. The brighter the maximum is, the higher the tracking distance gets and the more disturbing light is possible. Only the "USB / 3V" power source allows to power the LEDs with 100%.
Blue LED	Set the minimum and maximum brightness used for the blue control LED. If you do not want a blinking LED, set an equal value. The blue LED serves no other purpose than being able to see at first glance whether a tag is on or off.
Save Config	This stores the above settings to the connected ID tag. Now you can unplug it and power it (later). The settings in the dialog are not reset, hence you can plug a new tag to the PC, change only the ID and save all settings to it.

7.5.2.12 IP Info and Setup



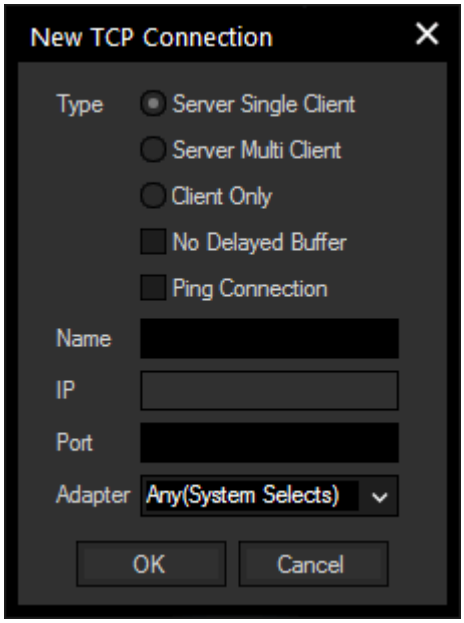
Click the button "Local Adapter" to choose one of the network interfaces from the drop-down which should be your local Widget Designer adapter. If "Any (System Selects)" is chosen, the NIC set up as primary adapter in your Windows settings will be selected automatically, if available.

"Network Adapters" offers you an overview of the available network adapters, their current IP addresses and subnet masks as well as their MAC addresses.

The button "Used ports" will open a list with all used TCP and UDP ports of the computer, not only the ports used by Widget Designer.

7.5.2.13 TCP Connections

To create a new TCP connection press [Add] in the TCP section.



In a TCP environment a computer or device (such as a router or projector) can be set up as either Client or Server.

The reason for these two types of modes is related to the way how a connection should be established between two devices. A Server waits for incoming connections while a Client can only connect to a Server.

Widget Designer offers the option to create a connection type where Widget Designer as TCP server can receive data packets from multiple Clients at the same time.

Every time a packet is send via a Server with Multi Client the data is sent to all connected Clients at the same time.

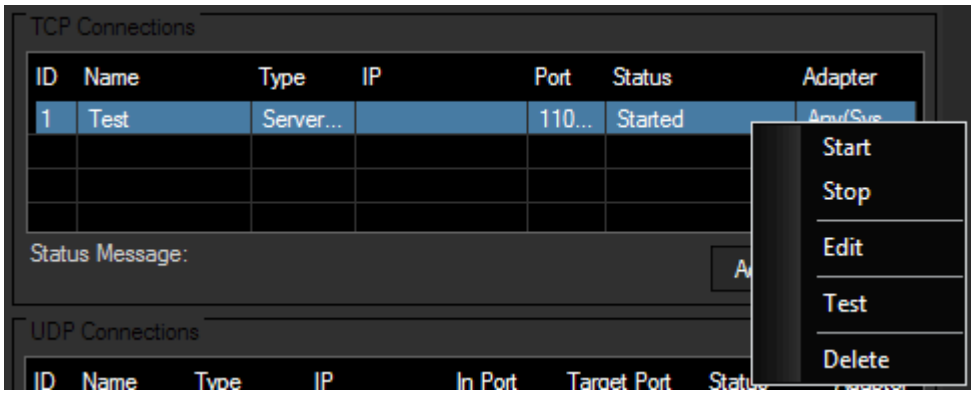
If you need to constantly stream data via TCP output nodes, it is recommended to enable No Delayed Buffer so that the packets are not concatenated and sent immediately.

When setting up a TCP Server connection you will only need to specify the listening Port.

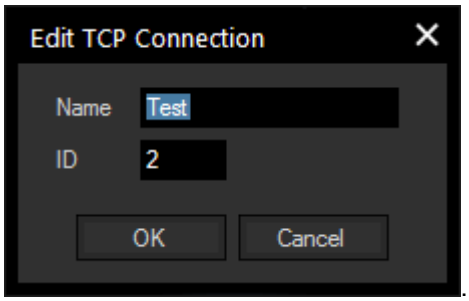
When creating a Client connection you will need to provide the TCP port and IP address of the TCP Server that you would like to connect to.

The Name property of the connection is only internal to the Connection Manager to let you label the connection with logical names for a better overview of multiple connections. Once the connection is created you will find the ID in the TCP list table. This ID will be required by commands and nodes to hook to this connection to either send or receive data.

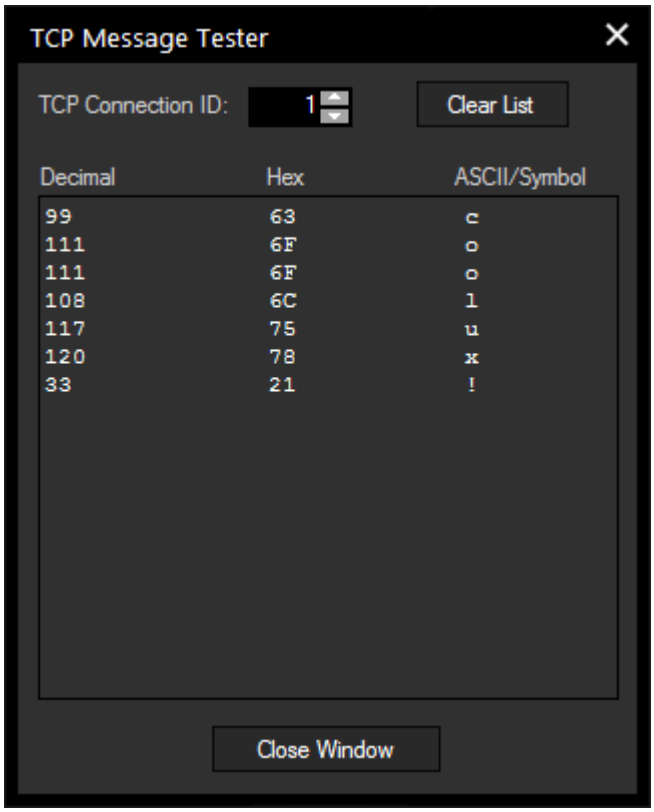
After you created a TCP connection, it will be displayed in the TCP Connection list. If you do a right-click on this TCP connection, you will be able to manually Start, Stop, Edit and Test this connection.
If you have a large number of connections, you may find a longer list enabling a better overview under Tools > TCP/UDP/COM Connections [1272] > TCP Connection.



Choose "Edit" to change the Connection ID:

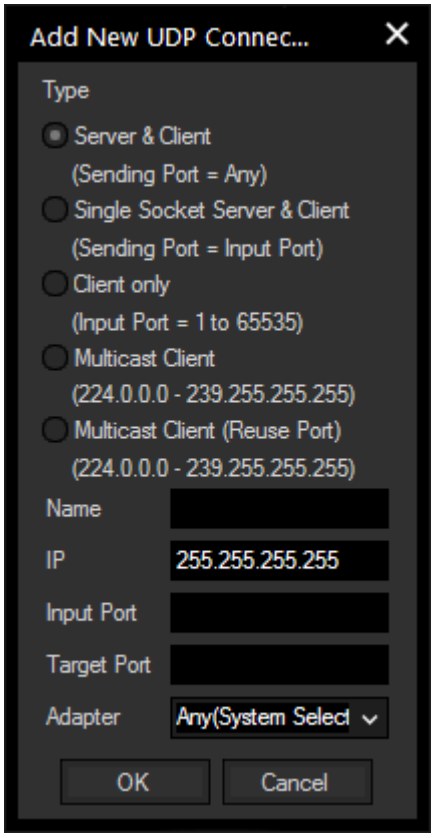


Choose "Test" and the TCP Message Tester opens. It displays the incoming UDP message as Decimal, Hex and ASCII/Symbol.



7.5.2.14 UDP Connections

To create a new UDP connection press Add in the UDP section.



UDP Broadcast
The UDP connections are by default set up as broadcast connections. A UDP connection does not require a dedicated IP address or a Server to connect to (with IP 255.255.255.255 the data will be send as broadcast into the whole network).

When sending Broadcast packets to the network, all UDP receivers that listen to the sending Port of Widgets Designers UDP connection will process the packets they receive. This way a UDP connection can be used for both sending and receiving UDP packets from the network.

Please make sure to choose different Ports for Input and Output.

UDP Unicast
You may want to send a UDP message as Unicast to a specified destination, e.g.. to the device with the IP address 192.168.5.5. To do this, keep the type "Broadcast Server Client" but change the Broadcast IP 255.255.255.255 to the desired one.

Please make sure to choose different Ports for Input and Output.

UDP Multicast
In an UDP multicast there is one Master sending messages to a specified port and multicast IP (within the Multicast IP range: 224.0.0.0 – 239.255.255.255) and any Client may listen to this multicast IP.

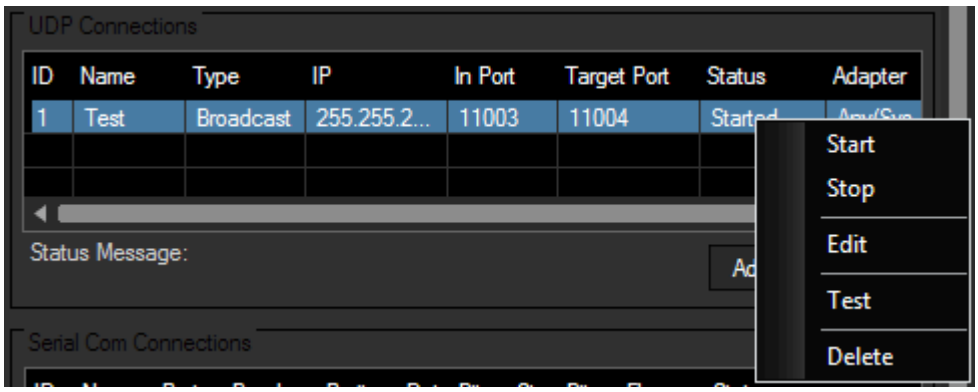
Check "Multicast Client" to listen to a multicast IP, enter the Multicast IP (within the Multicast IP range: 224.0.0.0 – 239.255.255.255) and specify the Input Port.

There is no Output Port as only a Master is able to send messages into this IP.

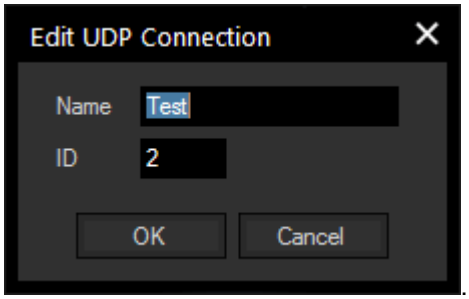
The Name property of the connection is only internal to the Connection Manager to let you label the connection with logical names for a better overview of multiple connections. Once the connection is created you will find the ID in the UDP list table. This ID will be required by commands and nodes to hook to this connection to either send or receive data.

After you created a UDP connection, it will be displayed in the UDP Connection list. If you do a right-click on this UDP connection, you will be able to manually Start, Stop, Edit or Test this connection.

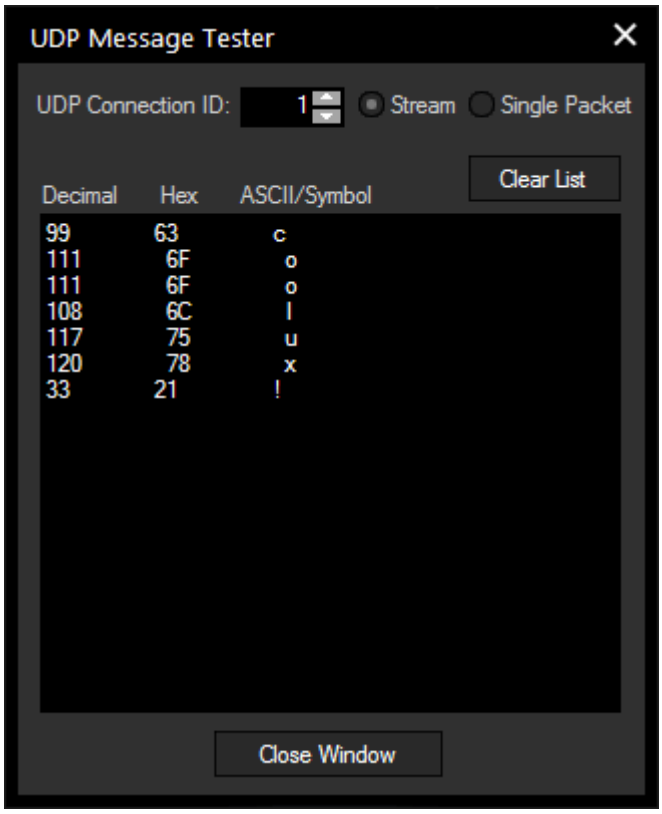
If you have a large number of connections, you may find a longer list enabling a better overview under Tools > TCP/UDP/COM Connections > UDP Connection.



Choose "Edit" to change the Connection ID:

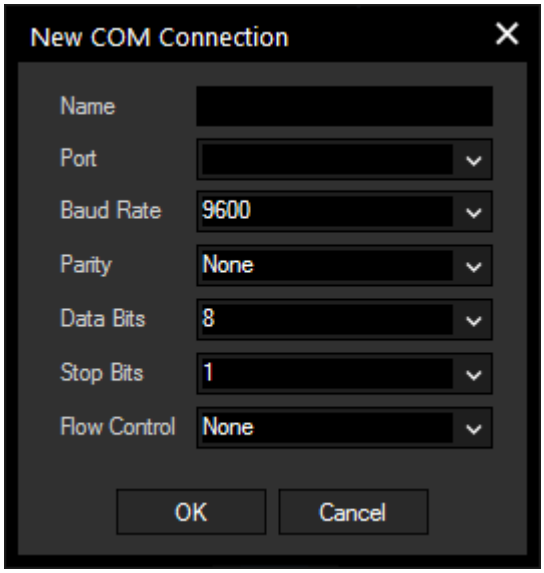


Choose "Test" and the UDP Message Tester opens. It displays the incoming UDP message as Decimal, Hex and ASCII/Symbol.



7.5.2.15 Serial COM Connections

To create a new COM Port connection press Add in the COM section.



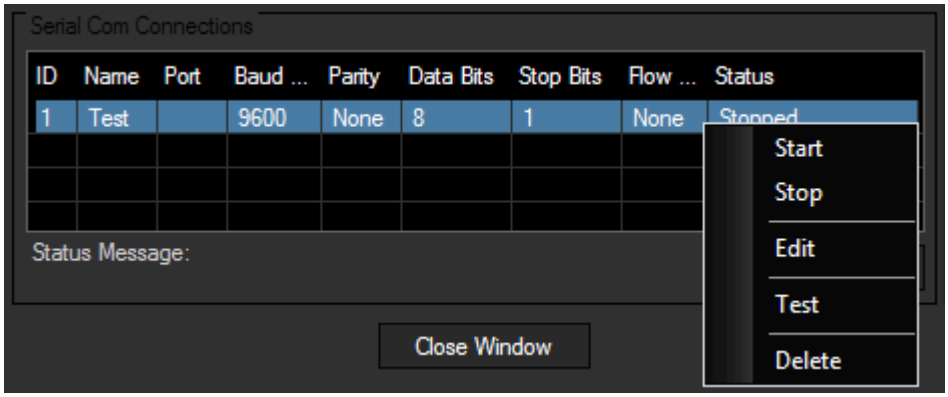
Any local Com Port will be listed in the Port drop-down list. Windows supports up to 255 comports that can be either local ports or virtual ports that are connected via network. A couple of manufacturers offer network based virtual Com Port extensions that can also be assigned here.

To setup a Com Port connection you will need to make sure that all the settings as seen above need to match the attached device settings 1:1. If the settings are not set correctly to the connected devices settings you may end up receiving corrupt or no data on this connection.

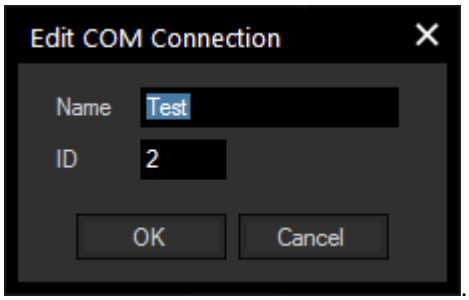
The Name property of the connection is only internal to the Connection Manager to let you label the connection with logical names for a better overview of multiple connections. Once the connection is created you will find the ID in the UDP list table. This ID will be required by commands and nodes to hook to this connection to either send or receive data.

After you created a COM connection, it will be displayed in the COM Connection list. If you do a right-click on this COM connection, you will be able to manually Start, Stop, Edit and Test this connection.

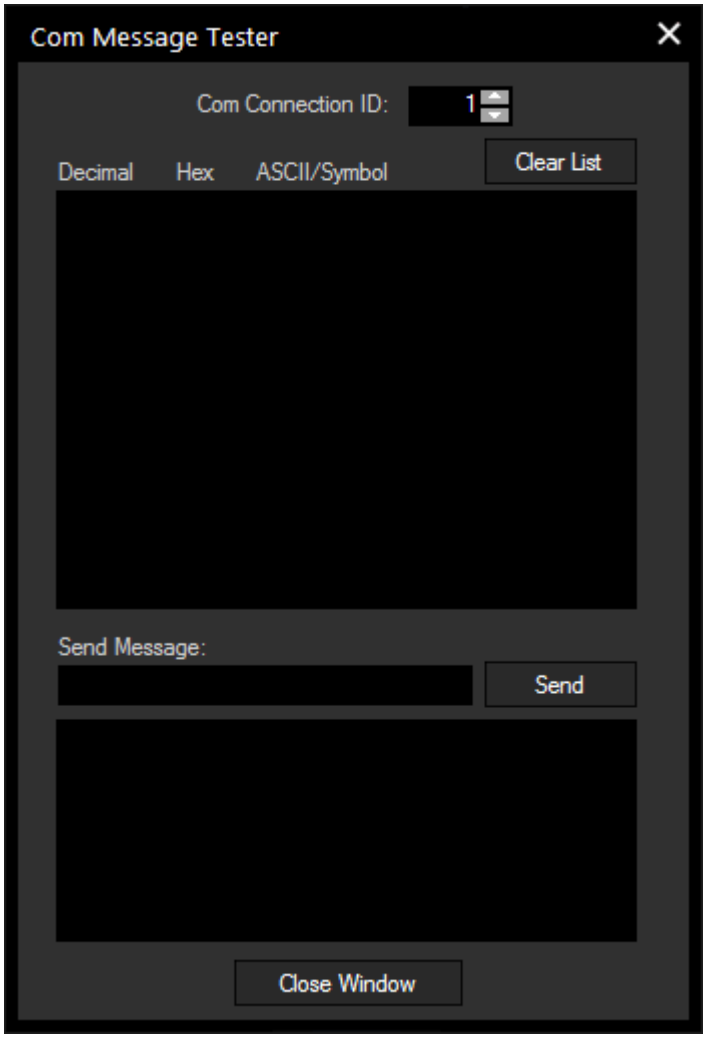
If you have a large number of connections, you may find a longer list enabling a better overview under Tools > TCP/UDP/COM Connections > COM Connection.



Choose "Edit" to change the Connection ID:



Choose "Test" and the Com Message Tester opens. It displays the incoming COM message as Decimal, Hex and ASCII/Symbol.



7.5.3 Midi Connections

Widget Designer supports up to 8 Midi inputs and outputs that can be configured via this Connections > Midi Connections dialog. Here you can select and enable different connected Midi input and output devices to be able to communicate with Widget Designer.

The Widget Designer Free edition supports only one in- and output device that can be set up in the Connection Manager [1260].

To enable a Midi device, select the respective hardware device from the drop-down, press "Apply" and check the "enable" box.

You can use MIDI connections via a tool, node or script command, the possibilities are listed below. In any case, please make sure to enter the correct ID referring to the numbering in the above dialog. When no ID is defined, ID 1 is accessed, which can also be setup in the Connection Manager dialog.

Tools > Midi Input [1485] > Midi Notes [1486], Midi Messages [1487], or one of the devices AKAI APC40 [1488] or BCF 2000 [1489]

Nodes > Input > Connections > Midi Input [959] or Midi Note Catch [960]
Nodes > Output > Connections > Midi Note On/Off [1183], Midi Raw Message [1184] or Midi Value [1185]

Most MIDI commands start with MidiSend... [1557] Please open the chapter "Script language [1511]" for more information about the use of it.

7.5.4 TCP/UDP/COM Connections

The TCP, UDP and COM Port connections are all setup in a similar way as you may want to manage multiple connections to hardware or software devices via any of the three protocols. Please refer to the Connection Manager and the sub-chapters TCP [1265], UDP [1267] & COM [1269] for more information. The Connection Manager [1258] is the tool to setup all main input and output communication protocols and includes short lists for TCP, UDP and COM connections as well.

If you like to setup or get a better overview of a larger number of connections you may open a dialog holding a longer list. Of course, all lists are synchronized.

Tools >
Connection Manager = dialog with short lists of all connections

TCP/UDP/COM Connections >
- TCP Connection = dialog with a long list of TCP connections only
- UDP Connection = dialog with a long list of UDP connections only
- COM Connection = dialog with a long list of CON connections only

7.5.5 Art-Net Monitor WD

The Art-Net Monitor can be a very useful tool if you would like to monitor Art-Net data on the network or simply find out if the DMX channels you are looking for have the right values.

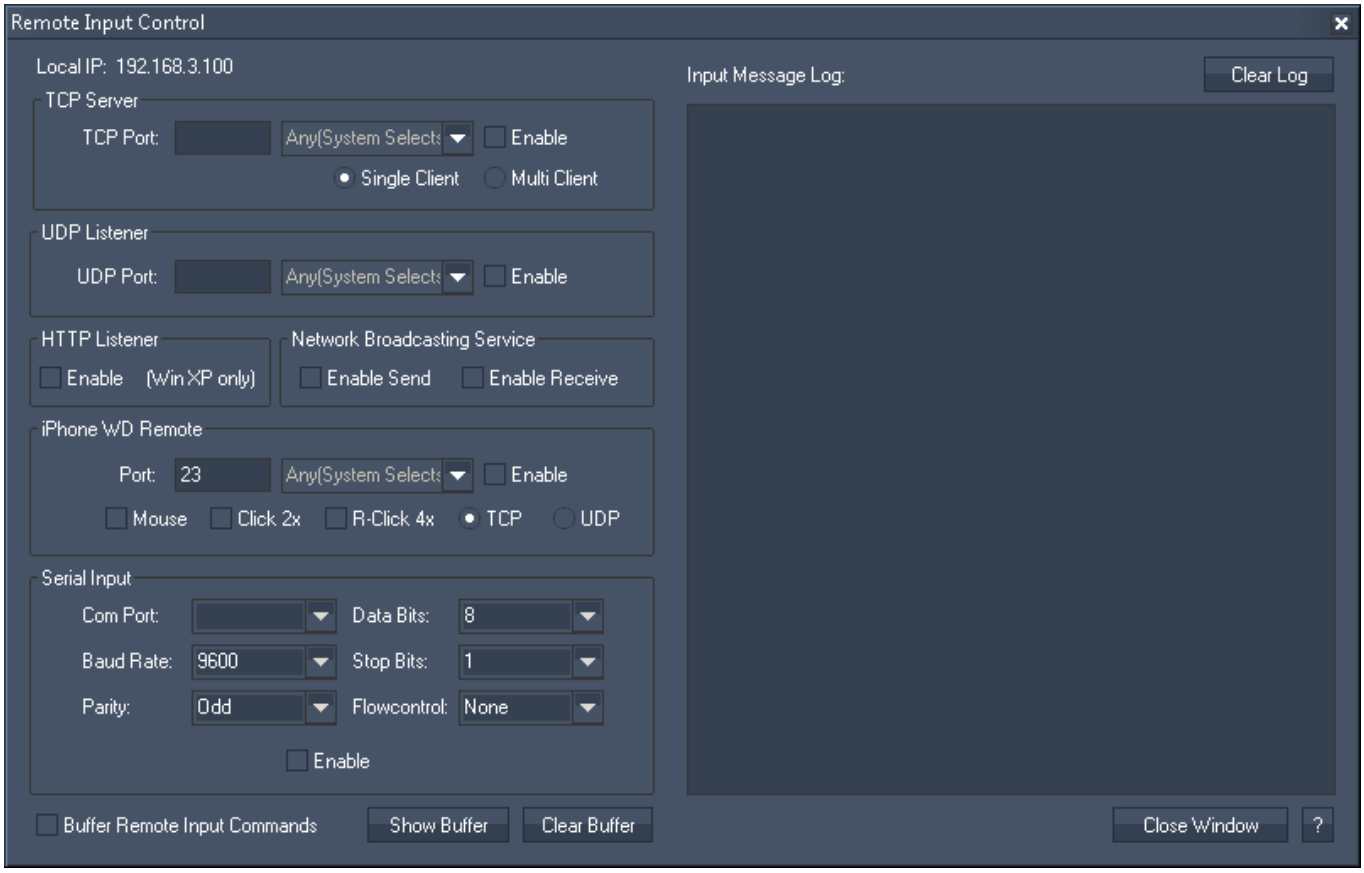
In the dialog you may select a subnet and universe. Then all channels are displayed with the according incoming Art-Net value. If you use WD to send Art-Net and you do so by broadcasting the values, you will be able to see these values in the Art-Net Monitor too. If you are unicasting the values, please switch to another Art-Net Monitor, e.g. the one in the PB Menu [2097].
More information [2083] about broadcast and unicast...

7.5.6 Remoting

The Remoting Tool allows you to setup a TCP/IP Server (single client or multi client), UDP and HTTP-Listener, the iPhone WD Remote and a Serial Input without using the Connection Manager and nodes for incoming communication.

Any incoming data will directly be executed as a command [1511] when the syntax of WD commands is used. The command must be put in curly brackets, e.g. `{WDFaderUp(1,2)}`. As a more detailed example, the command to send a remote command via the TCP Connection 1 would be:
`TCPSend(1,"{wdfaderup(301,1)}")`
Due to processing "{" and "}" cannot not be used within literals.

The messages received will be displayed in the message log.



Please see the following topics for further information:

TCP Server, [1274]
UDP Listener, [1274]
HTTP Listener, [1274]
Network Broadcasting Service
iPhone WD Remote [1275] and
Serial Input. [1277]

7.5.6.1 TCP Server

In a TCP environment a computer or device (such as a router or projector) can be set up as either Client or Server.

The reason for these two types of modes is related to the way how a connection should be established between two devices. A Server waits for incoming connections while a Client can only connect to a Server.

The Remoting Tool in Widget Designer offers the option to create a connection type where Widget Designer as TCP server can receive data packets from multiple Clients at the same time.

TCP Port:
When setting up a TCP Server connection you will need to specify the listening TCP port.

Single Client / Multi Client:
Choose if you want to establish a connection to only one client or if you want to allow multi clients to connect to your TCP server.
Press [Enable] to start the TCP Server.

7.5.6.2 UDP Listener

The UDP connection is by default set up as broadcast connection. An UDP connection does not require a dedicated IP address or a Server to connect to.

When setting up a UDP Listener connection you will only need to specify the listening Port. Press [Enable] to start the UDP Server.

7.5.6.3 HTTP Listener

The HTTP Listener allows you to use a web browser or link to send remoting commands to Widget Designer.

Syntax

Enable the HTTP Listener in the Remoting settings and use this syntax to enter your command in the browser's address bar:
http://'WD_IP_Address'/?{Command String}

Example:
http://192.168.1.20/?{WDFaderUp(2,5)}

Alternatively, you can also execute the link from any text based application that allows hyperlinks.

Please bear in mind that only **one** WD instance per PC can activate receiving commands via HTTP.

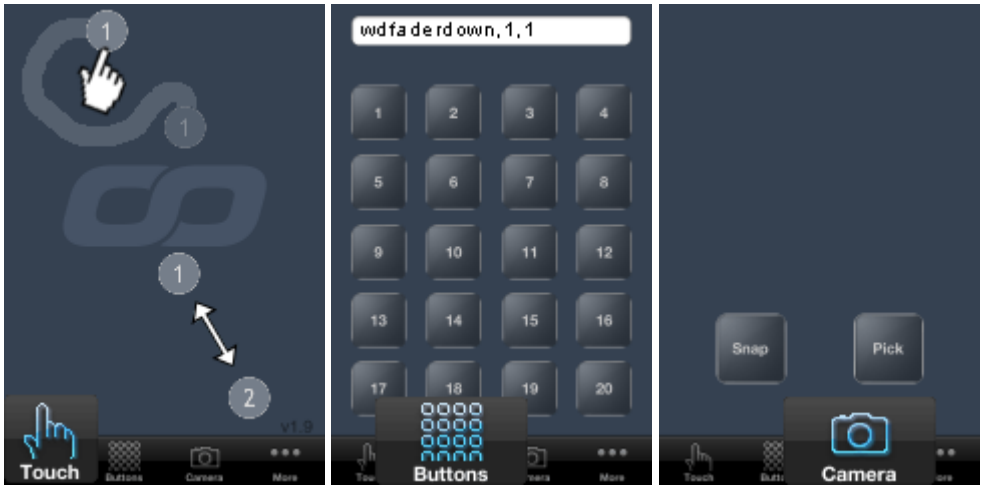
7.5.6.4 Network Broadcasting Service

This section in the Remoting dialog of Widget Designer, Network Broadcasting Service (NBS), allows to transmit and update Labels [888], Faders [874] and Custom Script Buttons [822] across multiple WD Designer instances on the network.
Please check the according options to enable/disable the Widget Designer to Send or Receive Network Broadcasting data.

7.5.6.5 IPhone WD Remote

Features of the App

The free "Widget Designer Remote" App for Apple devices such as the iPhone, the iPad or iPod allows to remote control a running Widget Designer with your mobile device. You may send
- single or multi-touch data to remote control the mouse or work directly in various controls such as the "Multi Touch Panel [899]"
- commands to execute a certain function, either by clicking a button or via a text field
- images taken with your mobile device
- acceleration data from the in-build accelerometer, e.g. to track an object



It might be of interest that Widget Designer offers another feature to be remote controlled by mobile devices. The in-build Web Server [1929] is capable to export any page as a page that can be viewed by any web browser such as Safari.

Settings within the device

After downloading and installing the App, please open the "Settings" menu. Under "Wi-Fi", make sure that you share the same wireless network as the PC Widget Designer is running on. Afterwards, scroll down the "Settings" list and search for "coolux", here you may set up the optional settings for this App.

Protocol:
Choose the connection type: TCP or UDP. For sending touch and tracking data, UDP is the better choice - for the button and command data TCP is more reliable. If you like to use the camera mode, TCP must be used. Please see this chapter [738] for detailed info about these two protocols.

IP Address:
Enter the IP address from Widget Designer. If you are not sure, have a look into the dialog "Pandoras Box Network Configuration [1256]".

Port:
Enter the listening port for the TCP or UDP connection. The same port needs to be set in Widget Designer, per default it is already set to 23.

Accelerometer:
Enable this option if you like to transmit XYZ acceleration values.

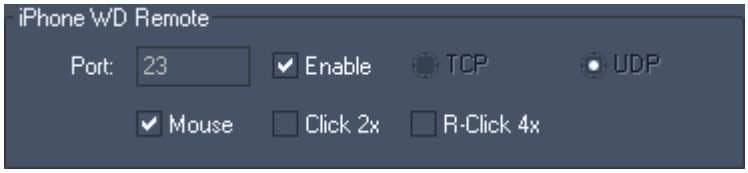
Button Settings:
Per default all buttons are labeled with the numbers 1-20. Enter a new name to change this. In the "Script" field, a command can be allocated to each button. It is mandatory to enclose the command in parentheses. It does not matter whether it is written in capitals or lower-cased. Please refer to the command list [1520] for more details.

Please note that you can enter a command in the App itself as well. The buttons screen includes a text field, when clicking into it you may enter a command without parentheses and click "Send" on the keyboard. If you like to send more than one command at once, you can either send one command that executes a macro or clicks a Custom Script Button that holds these commands. Both, the macro and the Custom Script Button need to be set up in Widget Designer.

If you are using iOS 5 and encounter problems with the App, please update to a newer operating system, e.g. iOS 6; so far no problems were reported with that version.

Settings within Widget Designer

Go to the Tools menu and open the "Remoting" dialog. The section "iPhone WD Remote" offers the following settings.



Port:
Enter the listening port for the TCP or UDP connection. The same port needs to be set in the settings for the App, per default it is already set to 23.

TCP / UDP:
Decide which protocol you like to use, again it needs to be the same as set in the App. For more details, see the above.

Enable:
Tick the check box to enable the connection.

Mouse:
This option allows to remote control the position of the mouse cursor with the touch data from the WD Remote App user interface. Regardless of that setting many controls like the "Multi Touch Panel [899]" allow to enable the input mode "iPhone" to receive the touch data.

Click 2x:
This option allows receiving a mouse click when the WD Remote App user interface is touched twice.

Click 4x:
This option allows receiving a mouse right-click when the WD Remote App user interface is touched four times in a row.

If you wish to work with the acceleration values, please create an input node: Input > Device > iPhone Remote [1008].

If you like to use the camera mode, please follow these steps. First, pick an image using the App and send it to Widget Designer. It then appears in the Image Resource Manager [1509] under Style: User, Control: User. With the help of commands, you may then display it in the Image / Picture Box, use it on a Custom Script Button or save it directly on the hard drive before adding it to your Pandoras Box project. The commands including "Recent" refer to the last image send from the App.

Some helpful commands could be:
- WDCustomScriptReleaseImageResource(ID,ResourceName) [1697]
- WDPictureBoxLoadResource(ID,Resource) [1794]
- WDPictureBoxLoadRecentResource(ID) [1794]
- WDResourceSave,Resource(Path) [1809]
- WDResourceRecentSave(Path) [1809]

7.5.6.6 Serial Input

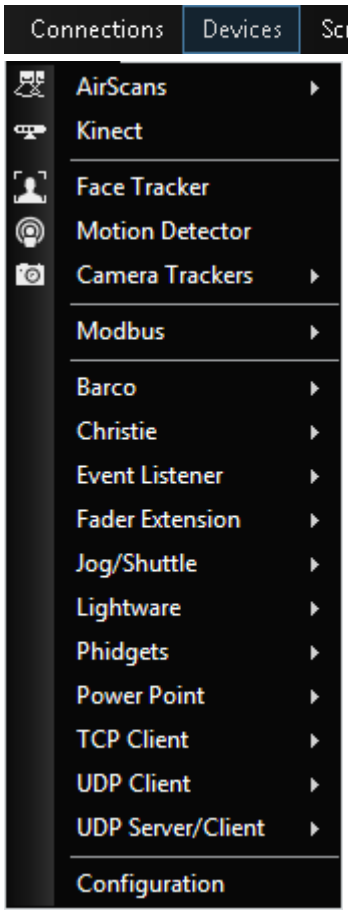
Any local Com Port will be listed in the Port drop-down list. Windows supports up to 255 comports that can be either local ports or virtual ports that are connected via network. A couple of manufacturers offer network based virtual Com Port extensions that can also be assigned here.

To setup a Com Port connection you will need to make sure that all the settings as seen above need to match the attached device settings 1:1. If the settings are not set correctly to the connected devices settings you may end up receiving corrupt or no data on this connection.

7.6 Devices

The Devices menu lists all (physical hardware) devices you can work with in Widget Designer. The licensed Widget Designer supports all listed devices whilst the Free edition includes only some of them, see below. In case your device is not available, bear in mind that it is also possible to set up a connection [1255], e.g. a TCP connection. Additionally, WD supports several Tools [1483].

Once you have chosen an entry a dialog opens with more options.



AirScans [1277]
Kinect [1283]
Face Tracker [1287]
Motion Detector [1289]
Camera Point Trackers [1291]
Modbus [1298] (also included in Free edition)
Barco [1306]
Christie [1318]
Event Listener [1332]
Fader Extension [1354] (also included in Free edition)
Jog/Shuttle [1364] (also included in Free edition)
Lightware [1378]
Phidgets [1383]
PowerPoint [1468]
TCP Client [1471]
UDP Client [1475]
UDP Server/Client [1479]
Configuration [1305]

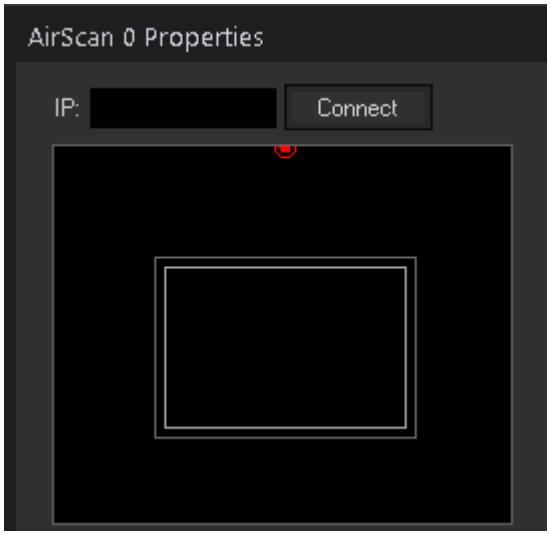
7.6.1 AirScan

The AirScan is available as an optional input device for Widget Designer, note that it is not available in the Free Version. The AirScan is a laser scanner device that measures distances in two dimensions. Thus, it allows for optical and contactless detection of objects or persons or just their hands. It can be used for any touch-less interaction with a display or a projected touch surface.

Please see Hardware and Accessories > AirScan [1988] for more information about the hardware. The chapter explains...:
- what the AirScan is, and what it can be used for
- the hardware itself including technical specifications, LEDs and connectors
- restrictions to ensure correct usage and measurement readout

In Widget Designer the AirScan can be setup using the AirScan Tool. It is possible to read up to 24 input points and use them for various applications within Widget Designer or Pandoras Box.

AirScan Tool Setup in Widget Designer



Open the AirScan tool from the Devices menu [1277]. In the AirScan setup tool you set up and connect to the IP address of the unit. Once connected you will see a yellow outline of the laser field. The input screen rectangle needs to be placed between a minimum and maximum range. All setup parameter related to the AirScan values are set in millimeters.

Once you got your touch data into the system you can map the first point to the local mouse and use one or more additional multi-touch points with a Multitouch Panel or an XY Panel.

The AirScan Tool is divided into several sections:

General Settings, [1278]
Input Point Processing, [1279]
Calibration, [1280]
Point Tracking [1281] and the
Mouse Control [1283].

AirScan node and commands

To use the AirScan for further programming, please use the AirScan Multi-Point Input Node [980]. After entering the point ID, it will offer its "Active" status and position data that you can use in following filter or output nodes.

In addition, there are many commands controlling the AirScan tool [1649]. This enables you for example to adjust the tracking during a show without the need to open the tool and look for a certain value.

7.6.1.1 General Settings

This chapter explains the general settings in the AirScan Tool in Widget Designer. For more information please see the introductory chapter [1277].

IP:
Enter the AirScan's IP address and click [Connect] to connect the AirScan.
Now you should see a yellow outline of the laser field.

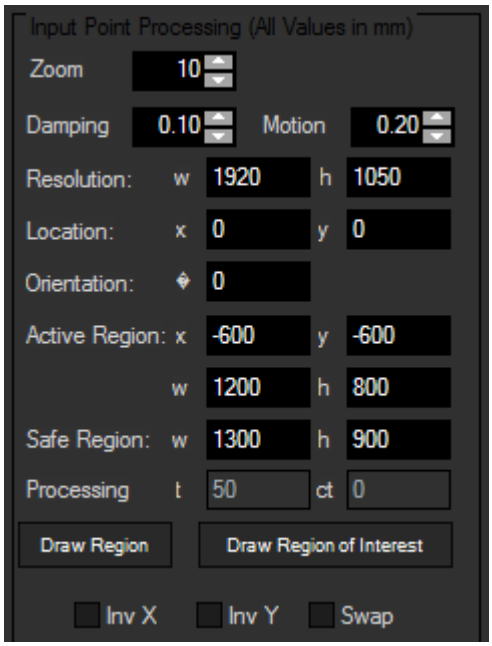
Auto Connect: (at the bottom of the AirScan Tool):
Check this option to automatically connect to the AirScan (once the connection to the AirScan is established) after a disconnect or restart of the Widget Designer.

Auto Apply:
Click this option to automatically apply all changes in the menu without the need to press the "OK" or "Apply"button.

The next chapter explains the settings for Input Point Processing [1279].

7.6.1.2 Input Point Processing

This chapter explains the Input Point Processing settings in the AirScan Tool in Widget Designer. For more information please see the introductory chapter [1277].



Zoom:
Use the zoom factor to zoom in / out the AirScan field, the preview window in Widget Designer.
Max zoom: 1 (horizontal axis ranges from left= -110mm to right= +100mm and vertical from top=0 to bottom= -185mm
When zooming further out, the 0,0 origin point, which is the intersection of the horizontal and vertical axis, will always be located in the middle of the top border.

Damping:
This value is important as the IR readout of the AirScan can have noticeable noise that is filtered out with this prediction based damping.

The damping factor is set by default to 0.10. You can change this setting from 0.1 = maximum damping up to 1 = no damping.

Motion:
The Dynamic Motion Damping improves the stability of the touch point in motion and stabilizes the points when they are not moving. It is set by default to 0.2. Its value ranges from 0 (max. smoothness) to 1 (no motion filtering).

Please note, that there is also the option to predict the motion under "Point Tracking">"Motion Prediction" [1281].

Resolution:
Enter here the resolution of your Widget Designer GUI (width and height). The Active Region will later map to this resolution.

Location x and y, Orientation:
Enter the location (in mm) of the AirScan hardware and orientation in degree. The location is of special interest when working with multiple AirScans.

Active Region:
The coordinates x and y define the position of the Active Region in relation to the AirScan's position in mm. Please enter the size of the region for w (width) and h (height) in mm. Alternatively you can use the "Draw Region (of Interest)" button or Calibration [1280].
This region is now mapped to the Resolution you entered above. So first, the AirScan device reports a total distance in mm for an active point that entered the AirScan scan field. Now it is checked whether it is inside the Active Region. And only if that is the case, it is translated to a pixel value according to the given "Width" and "Height" of the Active Region and the "Resolution".

Safe Region:
The Safe Region makes the detection of points on the border of the Active Region more robust.
Enter values in mm for the width and height. Ideally the Safe Region is a bit bigger than the Active Region, e.g. by 10%.

Draw Region:
Click this button and drag a rectangle in the AirScan preview field starting with the upper left corner (when looking at this window!) and ending with the bottom right corner. The drawn rectangle overwrites the values for the Active Region including Save Region, explained above.

Draw Region of Interest:
Click this button to create a region of interest if that is not a rectangle but a shape with x corners. A regular Active Region has to be set up beforehand, the Region of Interest needs to lie within it to generate active points. The X and Y position data of points in the Region of Interest is also calculated referring to the space defined by the Active Region. Use this feature e.g to exclude elevated structures or noisy areas on your touch surface.

To do so, first click the button, then once into the window which creates (an invisible) corner for your shape. When you now move the mouse and click a second time, a line is created and a second corner is defined. Each click will now create a line that started at the previous click location and ends at the current one. To "close" the shape, i.e. connect the last click connection to the very first one, simply click the button again. A third button click will let you draw a new region.

Inv X / Inv Y / Swap:
This inverts the horizontal (X) or vertical (Y) min/max v alues or swaps X with Y. Use these options if the AirScan is not located above the active field.
If the AirScan position is e.g. below the active field, check "Inv Y" to receive correct Y Position data.

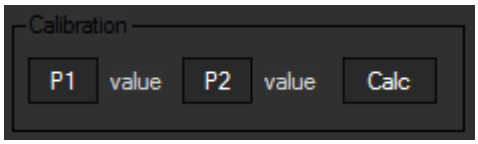
The next chapter explains the settings for Calibration [1280], which allows an intuitive calibration of the Active Region.

7.6.1.3 Calibration

This chapter explains the Calibration settings in the AirScan Tool in Widget Designer, which allows an intuitive calibration of the Active Region. For more information please see the introductory chapter [1277].

The Auto-Calibration method is an alternative to the "Draw Region (of Interest)" option or manual input of the position and size of the Active Region which is explained in the last chapter [1279].

This is a step-by-step description how to use the Auto-Calibration:



1. Draw an approximate Active region around your desired touch area. The points used for the Auto-Calibration have to be recognized as active points.

2. Interrupt the AirScan field at the position of the first corner. The corner should be the top left one when looking at the preview window of the AirScan tool in WD, so in real and depending on your orientation of the AirScan it might be another one.

3. In WD, press [P1]. This captures the coordinates of the upper left corner and you will see the corner's position (x/y in px, relative to the coordinate system created by the current Active Region) next to the button.

4. Now interrupt the AirScan field at the position of the second corner, the lower right one when looking at the preview window of the AirScan tool in WD.

5. Press [P2]. This captures the coordinates of the lower right corner and you will see the corner's position (x/y in px) next to the button.

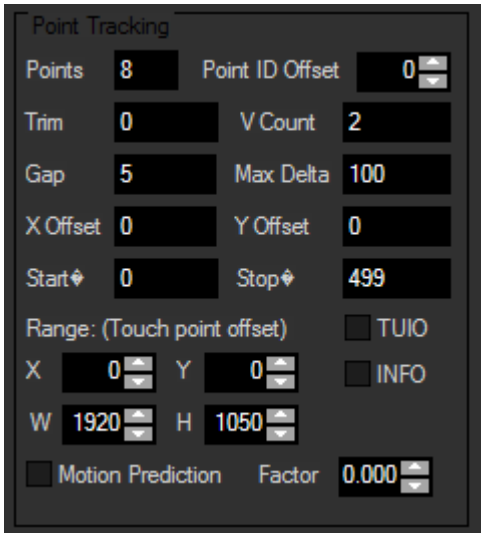
6. When this is done, press [Calc] and the position values will be transferred into the Active Region Settings.
Please note: the Safe Region has to be adjusted manually, e.g. with a size 10% larger than the Active Region.

The next chapter explains the settings for Point Tracking [1281].

7.6.1.4 Point Tracking

This chapter explains the Point Tracking settings in the AirScan Tool in Widget Designer. For more information please see the introductory chapter [1277].

Almost all settings in this section can be scripted using the available "WDAirScan..." commands.



Points:
Enter here the amount of points that should be detected and provided for further processing.

Point ID Offset:
The Point ID Offset should be used when you are sending the multi-touch data from this AirScan via the Remote Input [1260] to another Widget Designer. It avoids that the multi-touch points from several data inputs get the same IDs causing ID conflicts.

Trim:
Trim allows to minimize errors on Point enter, with this value a number of scans will be ignored until the point motion processing is activated. The default value for Trim is 0.

VCount (Validation Count):
The Validation Count defines the amount of rays that have to be interrupted in order to validate a touch point. The default value is 2.

Gap:
The default value for the Gap is 5. This means that within 5 rays distance of a detected point all other interrupts will be discarded.

Max Delta:
The default value for Max Delta is 100. The Max Delta function determines the maximum speed for Multi touch Point Tracking.

X and Y Offset:
These offsets allow to run the Widget Designer on a secondary screen. If you're doing so please enter here the pixel offset for x and y.
Example: resolution of primary and secondary screen is 1024x768, WD runs on the secondary screen, enter x=1025 and y=0.

Start (Ray):
The AirScan sends out 500 Rays over 180° clockwise. If you want to discard a certain amount of rays up from the start, enter the new value here (0-499).

Stop (Ray):
The AirScan sends out 500 Rays over 180° clockwise. If you want to discard a certain amount of rays before the actual end, enter the new value here (0-499).

Range: (Touch point offset):
The touch point offset for the Multi-Point mode allows running the WD on a secondary screen.

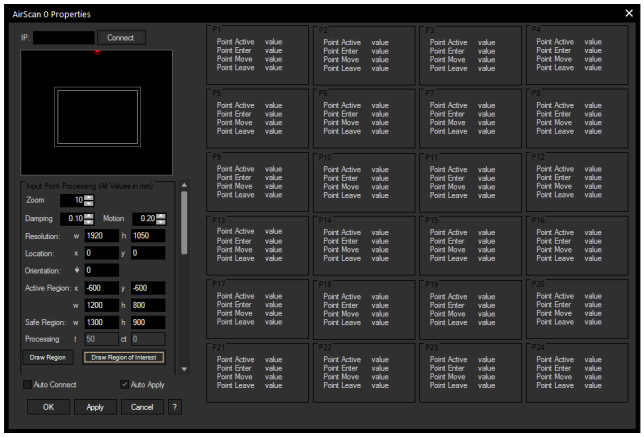
Motion Prediction and Factor:
In addition to the options to damp the incoming data, you can also predict its motion. First, enable the "Motion Prediction" then set the factor. With a higher factor you can reduce the latency that occurs when the tracked points move quite fast. However, sudden movements should be avoided then. For example, if your prediction value is very high and the movement stops suddenly, WD "overshoots" as the movement before the stop is predicted to also happen in the future. Hence, the motion prediction must be fine-tuned to achieve a good compromise between latency and overshooting. In addition, it is important to set up a good damping value in subsequent nodes. For example, with no damping, there is quite some noise and this jittering movement is of course also predicted which results in more jitter.

TUIO:
In order to output multi-touch data to other applications this option allows to send the AirScan's multi-touch data via the open source protocol TUIO.

This protocol is widely used around the world by many application developers and is a commonly known way to transmit the individual touches.
To use AirScan with TUIO, set up the TUIO host IP and Port in the Connection Manager [1258].

Info

Enable the "Info" check box to show the Point Read-Out Dialog with the data of all 24 multi-touch points.



Generally the first ray interrupt will be displayed as P1, second one as P2 and so on.
Each point will be tracked until it is removed out of the AirScan Field. So if there are P1 and P2 detected and P1 is removed, P2 stays P2 and does not changes its ID to be P1.

Point Active (true or false):
A point turns active as soon as there is a ray interrupt started inside the Active Region [1279] of the AirScan.
As long as the interrupt stays within the Safe Region [1279], the point stays active.
If the interrupt started out of the Active Region or if there is no interrupt, the Point's status turns to False.

Point Enter:
If there is an interrupt detected inside the Safe Region (whether it is active or not), the value for the X & Y position of a Point where it entered the AirScan field will be displayed here.

Point Move:
As long as the value for Point Enter is true (1), the X and Y position of this Point will be displayed here.

Point Leave:
The last valid X & Y position of a Point while leaving the AirScan field will be displayed here.

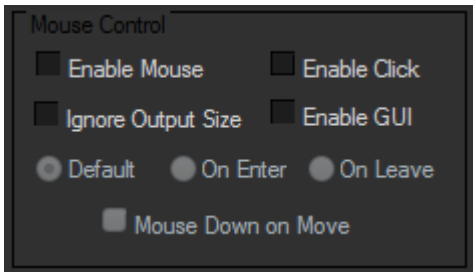
To use this point read-out for further processing, please use the AirScan Multi-Point Input Node [980].

The next chapter explains the settings for Mouse Control [1283].

7.6.1.5 Mouse Control

This chapter explains the Mouse Control settings in the AirScan Tool in Widget Designer. For more information please see the introductory chapter [1277].

Almost all settings in this section can be scripted using the available "WDAirScanMouse..." commands [1650].



Enable Mouse:
Check this option if the Point1 input data should control the mouse movement of your WD computer.

Enable Click:
If this option is checked the mouse can generate clicks. Use the tick boxes below to define when the click should be executed.

Default, On Enter, On Leave:
These options are available when the check box "Enable Click" is activated and define when a mouse click is generated.

Default: generates mouse down on enter and mouse up on leave.
On Enter: the click is generated on enter.
On Leave: the click is generated on leave.

Another option to generate a click on a Custom Script Button [822] is not to use "Enable Click" but to use the buttons timeout settings. When positioning the mouse cursor over a button for e.g. 500 ms a mouse click is executed automatically.

Ignore Output Size:
If your Active Region is larger than your computer's output resolution, WD automatically is only resopnsive within the output boundaries. I.e.: If you set up your Active Region with a resolution of 3840x2160px, but the output resolution of your monitor is only 1920x1080px, the mouse cursor would only be active within the 1920x1080px area. To use the cursor e.g. on a second monitor, check this box to ignore the output boundaries.

Enable GUI:
Check this box if you only want to operate your WD interface, i.e. buttons, Faders, etc., with the AirScan. A blue dot then acts as mouse cursor, but it doesn't affect other applications.

Mouse Down on Move:
If this option is checked, left mouse down on move is always active. Its only available when the check box "Enable Click" is activated.

7.6.2 Kinect

The Kinect hardware interface is available as optional input device for Widget Designer.
It can be used for capturing up to 8 filtered and tracked points/regions of interest.

Use the Kinect Tool to control PB via the Multitouch Panel [899], to control the mouse of your WD computer in order to e.g. proceed button click or process the data you get out of the Kinect Input Node [1011].

Originally designed for Xbox the Kinect Sensor features a Color Camera and a so called Depth Camera. The Depth Sensor (640x480@30fps) delivers a distance value per pixel (0-2047 units). It has an angular field of view of 57° horizontally and of 43° vertically.

The Depth Camera is used to determine motion at specific distances; it can read distance data between 1.2 - 3.5m distance quite well.

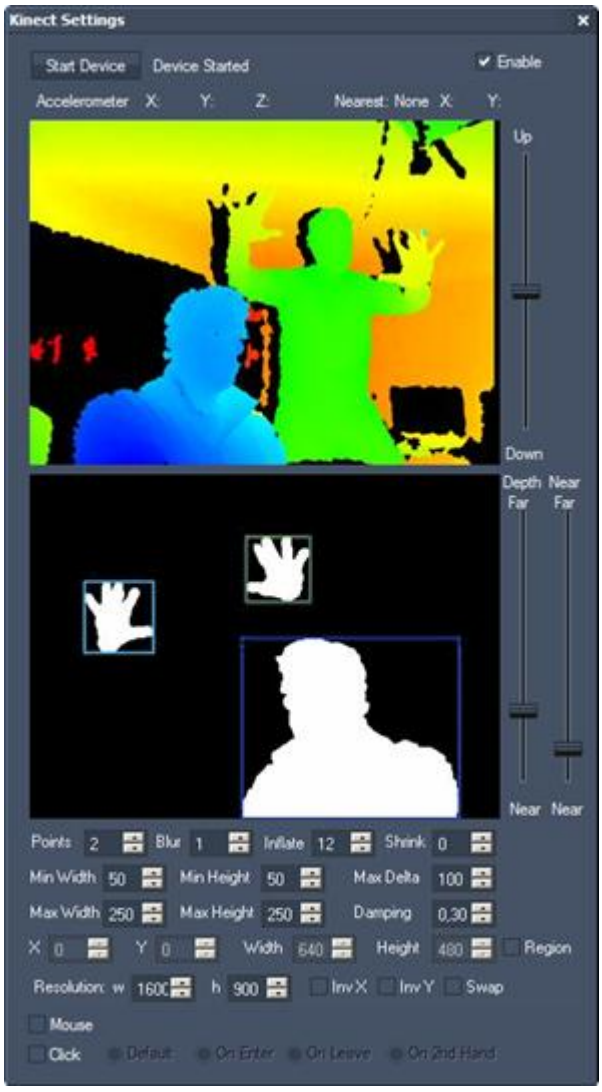
A motorized pivot is capable of tilting the sensor as much as 27° either up or down.

Important Notice Regarding Kinect Device usage:
Kinect driver from codelaboratories is still an early version and seems to run stable on WIN7 only.
When the Kinect Device is connected to a Win7 system there might be a driver conflict when using Logitec USB cameras at the same time.

Please install the Platform driver first in order to use Kinect within Widget.
The Kinect must be plugged in for the Driver to work other wise it will terminate the WD application when no device is attached.

Current Kinect Driver for Win 7 64bit Systems:
http://codelaboratories.com/downloads/
This driver MUST be used in order to work with the Kinect device.

Please note:
Do not use the Kinect Dialog in Show mode (the rendering of the Depth Camera requires too much system performance).



Starting and adjusting the device

To start the Kinect device, tick the [enable]-check box and press the button [Start Device]. Now you should see the colored camera image in the upper image field.

Depth camera image:
The different colors symbolize different distances from the sensor (e.g. blue=very near, orange=far).

Motorized Pivot:
Use the fader next to the upper camera image to bring the Kinect's sensor in the right position by tilting it up and down.

Binary camera image:
In this image every item that is detected by the camera in between a certain distance range will be shown as white parts on a black background. To define which distance range should be taken, use the faders "Depth" and "Near" next to the binary camera image.

Depth and Near Faders:
With these faders please define the distance range from which the points should be detected.

Move the Depth-Fader up or down to define the farthest point of the distance range.
Move the Near-Fader up or down to define the nearest point of the distance range.
The Near-Fader always has to be lower than the Depth-Fader.

Points:
By default two points will be detected. Change this value to decrease it down to one point or increase it to get up to eight points.

If you choose Points=0, a Fullscreen Mask in Kinect Dialog is shown - currently built-in for evaluation purposes.

Image processing settings

Blur:
Use this option to smooth the detected areas, so that rough pixels become smooth spots (0 = minimum blur, 15 = maximum blur).

Please be aware that this takes more performance!

Inflate:
This option inflates the detected areas (0 = minimum inflation, 15 = maximum inflation).

Shrink:
The result of the (eventually blurred and inflated) areas will be shrunk when using this option (0 = minimum shrinking, 15 = maximum shrinking).

Please note:
There are no recommendations which values would be best to enter for Blur, Inflate and Shrink. It depends on your setup and especially the lighting conditions.

Points Detection

Min Width / Max Width:
Define which minimum and maximum width (in pixels) a spot should have to be detected as point.
The pixel sizes are related to the Kinect's camera resolution of 640x480 px.

Min Height / Max Height:
Define which minimum and maximum height (in pixels) a spot should have to be detected as point.
The pixel sizes are related to the Kinect's camera resolution of 640x480 px.

Max Delta:
The Max Delta value defines how far a point is allowed to move (in px, related to the Kinect's resolution of 640x480 px) between two frames in order to still be detected as point.

Damping:
The damping allows to reduce noisy input values. The damping factor is set by default to 0,3. This setting can be changed from 0,1 = maximum damping up to 1 = no damping.

Region:
If it is necessary for a good point detection to use only a certain region of the Kinect's point of view, please check the option "Region". Every movement outside this region will not be executed.

Width / Height:
Choose the region's size by entering the values (px) for Width and Height. Please note that due to the Kinect's Camera resolution the maximum region size is 640x480 px.

X/Y:
The region's top left pixel is located in 0/0 (X/Y) be default. To move the region to a different position please enter the according X/Y position here.

Resolution:
Please enter here the resolution of your monitor. This is needed to adopt the positions of the detected points to fit to your screen.

Inv X:
Check "Inv X" if you need to invert the X axis.

Inv Y:
Check "Inv Y" if you need to invert the Y axis.

Swap:
Check "Swap" if you need to swap the X and Y axis, e.g. when the Kinect is turned about 90°.

Mouse Control

Mouse:
Enable the mouse if Point1 input data should control the mouse of your WD computer.

Click:
If this option is checked the mouse cursor generates clicks. Use the tick boxes to define when the click should be executed.
- Default generates mouse down on enter and mouse up on leave.
- On Enter: the click is generated on enter.
- On Leave: the click is generated on leave.
- On 2nd Hand: the click is generated as soon as a 2nd point (through the 2nd hand) is detected.

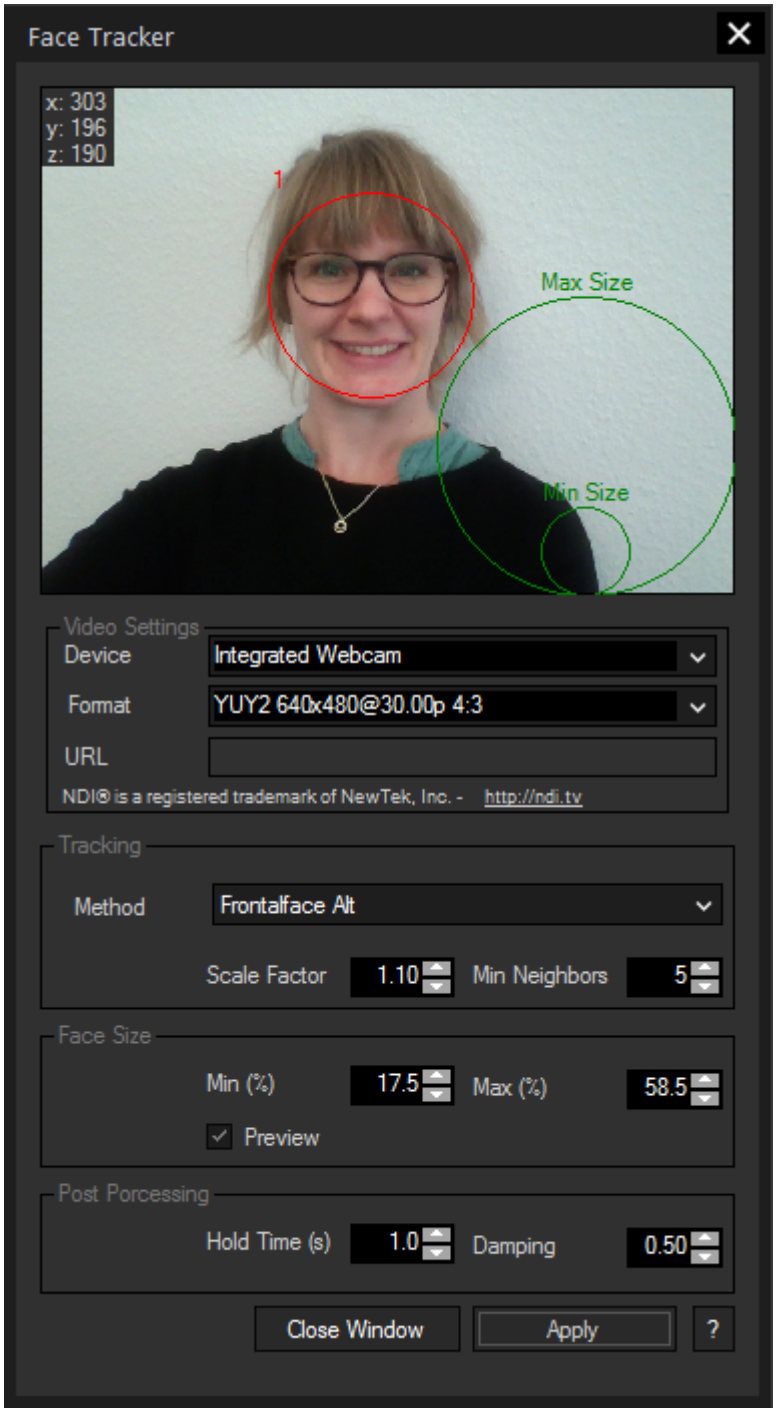
Another option to generate a click on a Custom Script Button [822] is not to use "Enable Click" but to use the buttons timeout settings. When positioning the mouse cursor over a button for e.g. 500 ms a mouse click is executed automatically.

To change the Kinect settings from e.g. a Custom Script Button, these commands [1549] are available.

7.6.3 Face Tracker

The Face Tracker Tool tracks a detected face/ body or eyes and outputs its position data via the Facetracking Input Node [1070] for further processing.

Open the Face Tracker Tool from the Devices menu [1277]. In the Face Tracker Tool you set up and connect to the unit/stream. Once connected you see the video input displayed in the preview and you can start setting up your tracking.



Video Settings

Device
Choose your DirectShow Camera Device, NDI- or RTSP stream from the drop-down list.

Format
Select the format (resolution@frame rate, aspect ratio) in the drop-down list. Please note that the format is automatically set when using NDI- or RTSP streams.

URL
Enter an URL in case you chose "RTSP" from the Device list. This can be an address from a web cam within your local network for example or a link to a public stream.

Tracking

Method
Choose the Tracking Method from the drop-down list. The following modes are available:
Eye
Eye Tree Eyeglasses
Frontal Face Alt
Frontal Face Alt2
Frontal Face Alt_Tree
Full Body
Lower Body
Profile Face
Upper Body

Scale Factor
This parameter specifies how much the image size is reduced at each image scale.

Min Neighbors
This parameter specifies how many neighbors each candidate circle should have to retain it. This parameter will affect the quality of the detected parts. Higher value results in less detections but with higher quality.

Face Size

Min%
Enter the minimum possible object size in percent. Objects smaller than are ignored.

Max%
Enter the maximum possible object size in percent. Objects larger than are ignored.

Preview
Activates/ Deactivates the preview for displaying the minimum and maximum tracking point circles (green).

Post Processing

Hold Time (s)
Enter the time in seconds for how long the active tracking points (red circle) should be displayed in the preview.

Damping
The damping allows reducing noisy input values. The damping factor is set by default to 0,5. This setting can be changed from 0 = maximum damping up to 1 = no damping.

As soon as the Face Tracker detects a face / body or eyes, it will be surrounded by a red circle.
The position data of this circle (x,y,z) is displayed top left of the Window.

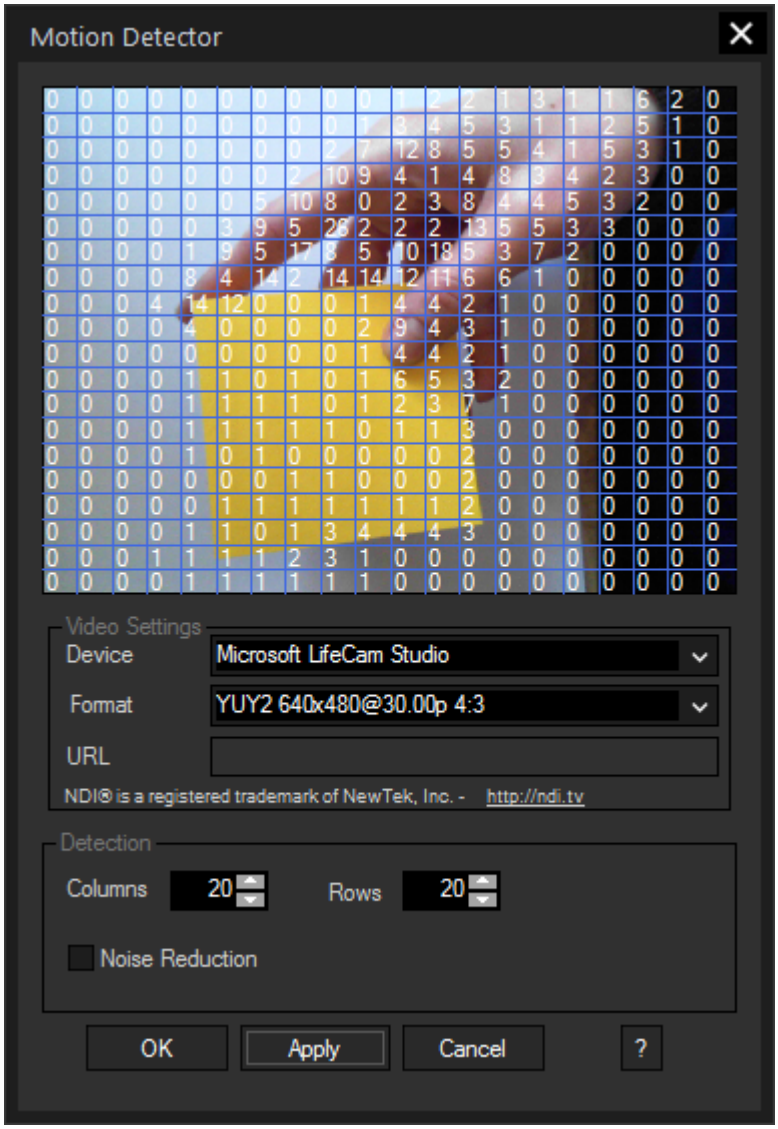
To use this data for further processing, please refer to Facetracking Input Node [1070].

7.6.4 Motion Detector

The Motion Detector Tool lets you divide a video input into any amount of columns and rows. It displays the color difference per segment caused by moving objects and outputs this data via the Motion Detector Input Node [1071] for further processing.

Open the Motion Detector Tool from the Devices menu [1277]. In the Motion Detector Tool you set up and connect to the unit/stream. Once connected you see the video input displayed in the preview and you can start setting up your tracking.

Please note:
This tool as well as the other video processing tools inside WD need a lot of RAM and CPU power!



Video Settings

Device
Choose your DirectShow Camera Device, NDI- or RTSP stream from the drop-down list.

Format
Select the format (resolution, frame rate, aspect ratio) in the drop-down list. Please note that the format is automatically set when using NDI- or RTSP streams.

URL
Enter an URL in case you chose "RTSP" from the Device list. This can be an address from a web cam within your local network for example or a link to a public stream.

Detection

Columns
Enter here the amount of columns the video input should be divided in.

Rows
Enter here the amount of rows the video input should be divided in.

Noise Reduction
Check this option to apply a noise reduction on the video input.

The values inside the single segments display the color difference caused by moving objects.

To use these values for further processing, please refer to Motion Detector Input Node [1071].

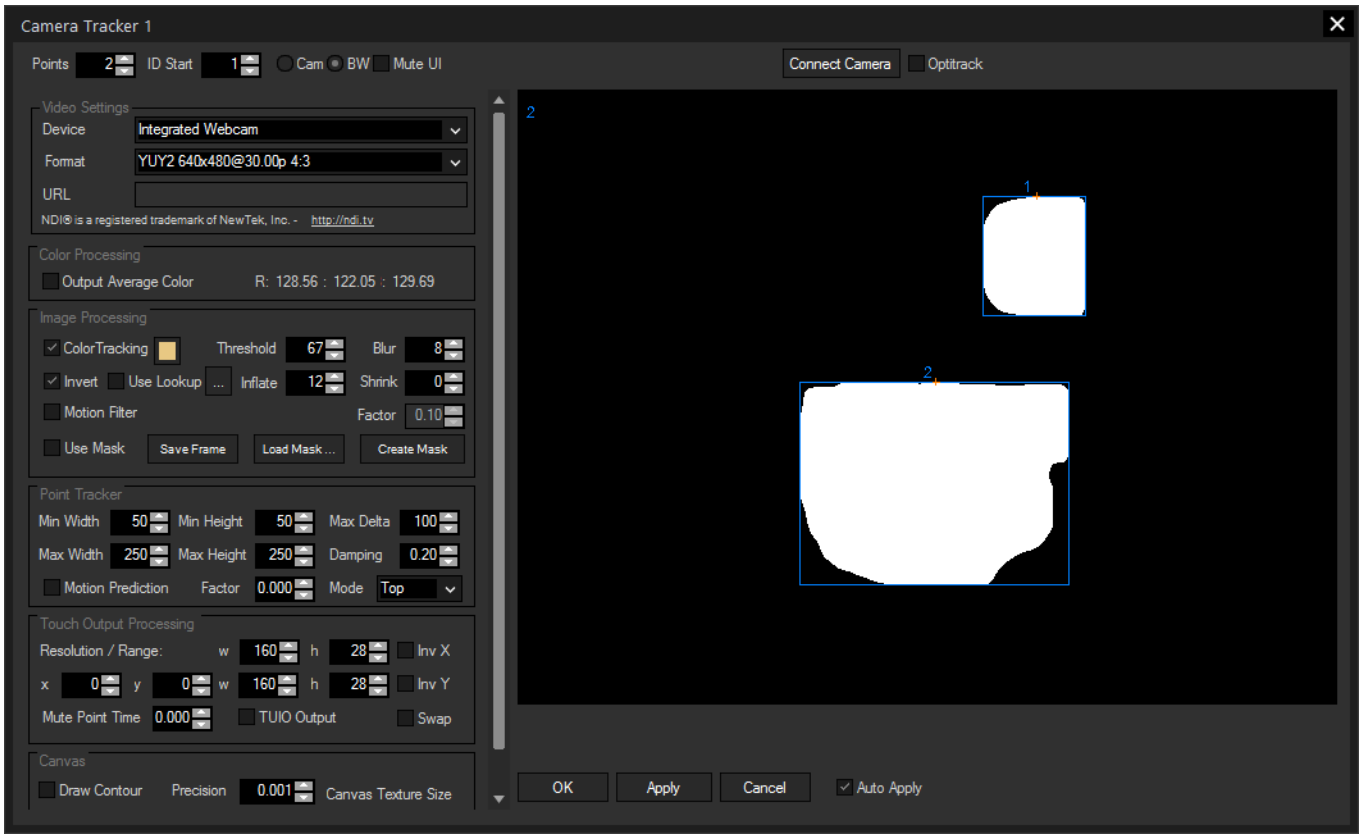
7.6.5 Camera Tracker

The Camera Tracker lets you track up to 99 objects and outputs via an attached DirectShow compatible camera, NDI- or RTSP stream. The number of cameras/streams is limited to 8.

The tracking data is available via the Camera Tracking Input Node [1067] for further processing. The touch points may be used in e.g. the Multitouch Panel as well.

Open the Camera Tracker Tool from the Devices menu [1277]. In the Camera Tracker Tool you set up and connect to the unit/stream. Once connected you see the video input displayed in the preview and you can start setting up your tracking.

Please note:
This tool as well as the other video processing tools inside WD need a lot of RAM and CPU power!



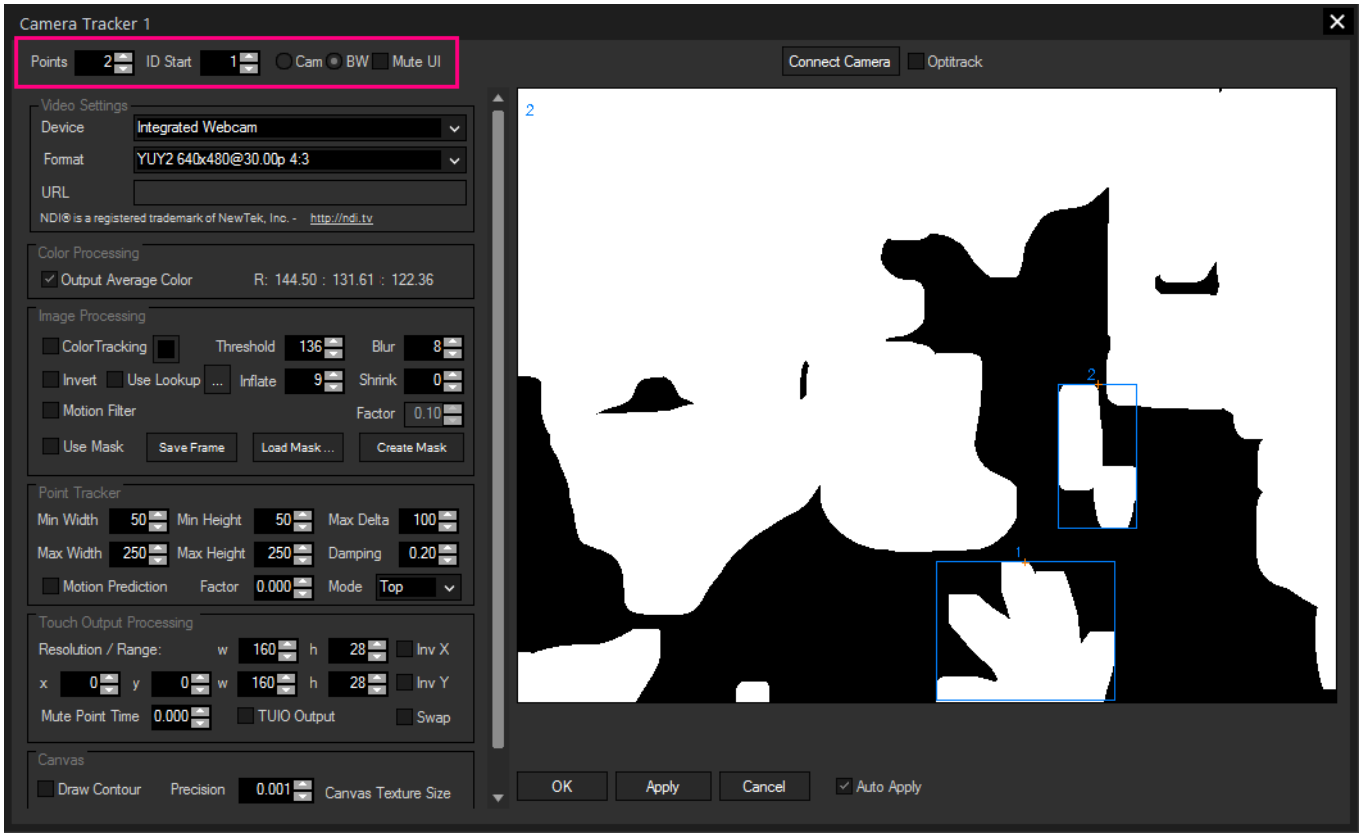
The device window is divided into several sections:

General settings, [1292]
Video Settings, [1293]
Color Processing [1294],
Image Processing [1294],
Point Tracker [1296]
Touch Output Processing [1297] and
Canvas. [1297]

At the bottom of the dialog you see that the "Auto Apply" option is ticked per default. This means that all changes are applied automatically without the need to press the "OK" or "Apply" button.

7.6.5.1 General Settings

This chapter explains the general settings from the Camera Tracker [1291] in Widget Designer.



Points
If you want to track more than 2 points (default setting) you may increase this value here to be able to track up to 99 points.

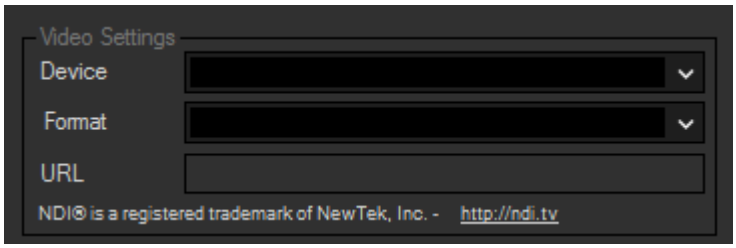
ID Start
The generated touch points may get an ID offset that you can set up here. This offset can be helpful for the source identification if you have several touch inputs in a multi-touch panel coming from different touch point sources within the Widget Designer or from several Widget Designer instances.

Cam / BW
When "Cam" is chosen the camera image will be displayed in the camera window above. This mode is helpful to set up the camera as well to choose a tracking color, see section Image Processing [1294].
When "BW" is chosen the Image Processing [1294] and Point Tracker [1296] settings will be applied to the camera input, the result is shown in black and white.

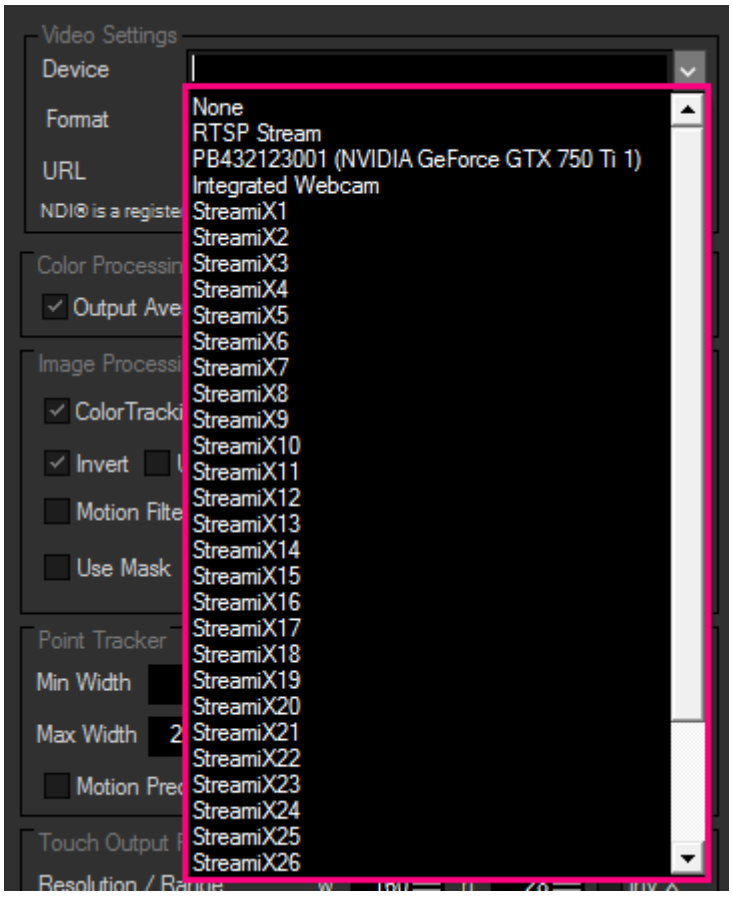
Mute UI
Mutes the Camera Tracker UI and shows the last frame.

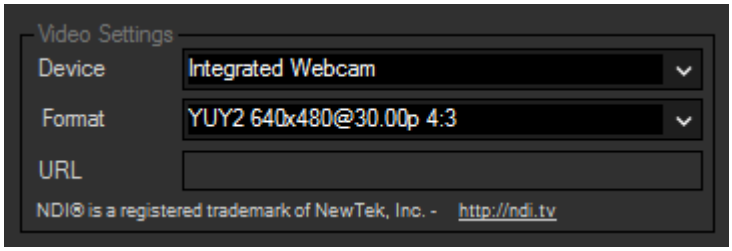
7.6.5.2 Video Settings

This chapter explains how to choose a hardware device or network stream for the Camera Tracker [1291] in Widget Designer.



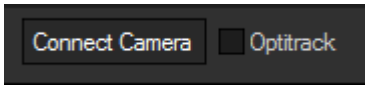
Device
Choose your DirectShow Camera Device, NDI- or RTSP stream from the drop-down list.





Format
Select the format (resolution, frame rate, aspect ratio) in the drop-down list. Please note that the format is automatically set when using NDI- or RTSP streams.

URL
Enter an URL in case you chose "RTSP" from the Device list. This can be an address from a web cam within your local network for example or a link to a public stream.

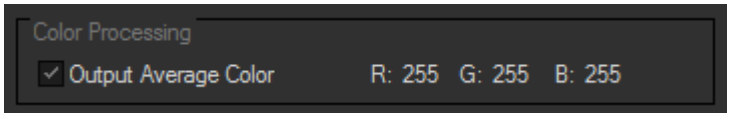


Connect Camera
Connects to the active input. Press this button once the video input is selected.

OptiTrack
Tick the check box to communicate to the OptiTrack camera.

7.6.5.3 Color Processing

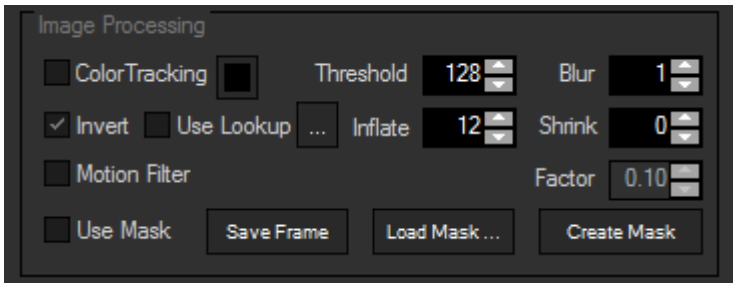
This chapter explains the "Color Processing" settings in the Camera Tracker [1291] in Widget Designer.



Output Average Color
When this option is checked the Camera Tracker Input Node [1067] will additionally output the average color of the camera input, separated in the colors red, green and blue.

7.6.5.4 Image Processing

This chapter explains the "Image Processing" settings in the Camera Tracker [1291] in Widget Designer.



The image processing section allows adjusting the camera BW image to get the best point detection, depending on your lighting conditions and your setup. Therefore there are no recommended settings; you will always have to modify different parameters to get the best result.

Please note:
The point detection will take effect for the white parts of the black & white image, not for the black parts.

Color Tracking
The tracking can be done with two different methods:
1. Based on all colors in the camera image,
2. Based on a color you picked.

The default setting is based on all colors in the camera image. If you want to use the Color Tracking, please tick the check box "ColorTracking".

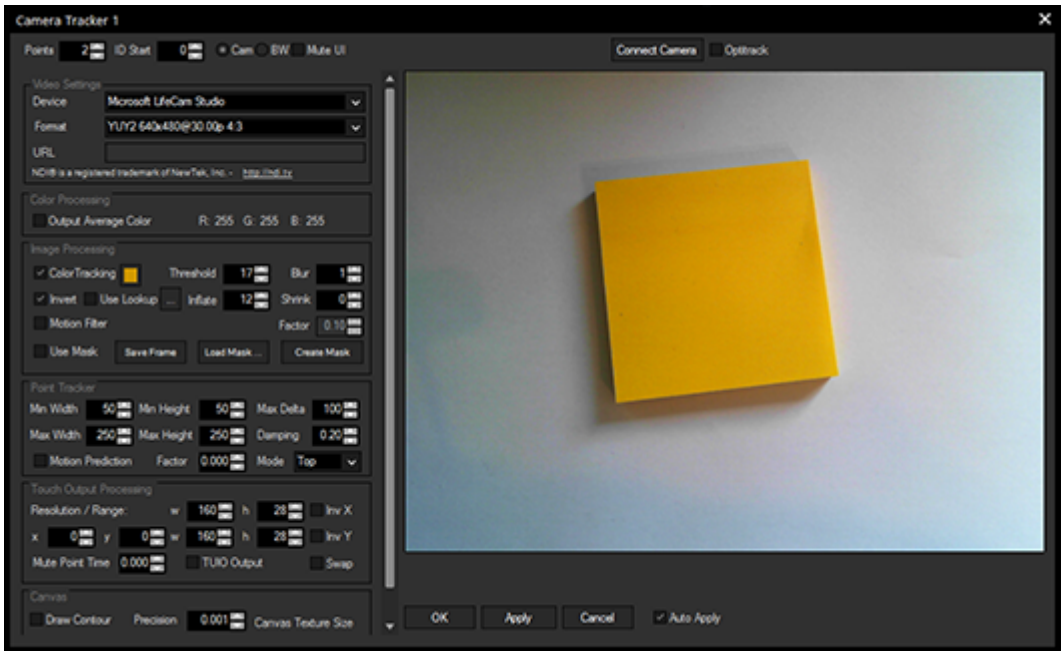
To set a color that should be tracked, activate the "CAM"-Mode in the camera settings section so that you see the colored camera image above.

Picking a color directly from the camera image:
Proceed a mouse-click on the desired color in the camera image. The box next to the ColorTracking check box will overtake this color.

Picking a color from the color picker box:
Click on the colored box next to the ColorTracking check box. Now you may choose a color from the list.

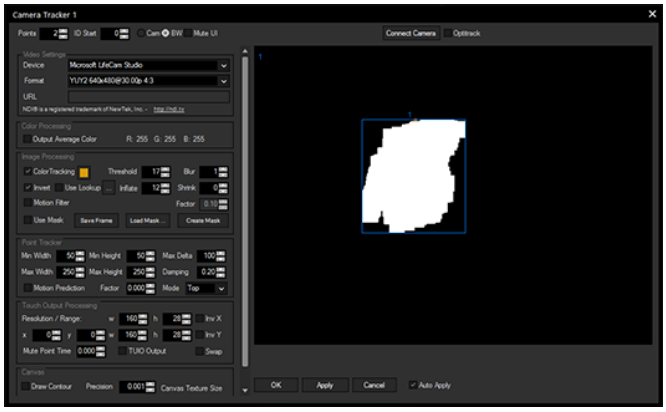
Threshold
Set up here the value for the threshold. The threshold defines which parts of the camera image will be converted into white and which ones into black parts when turning on the BW-Mode. It is related to the camera images luminance value by default. When Color Tracking is enabled, it is related to the special color chosen.

Example

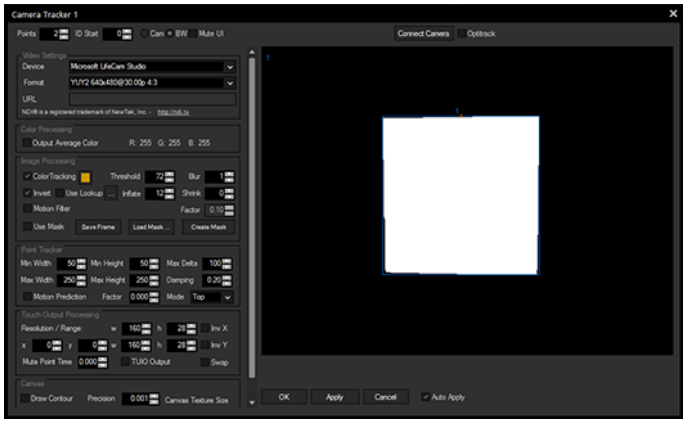


In the setup above the Color Tracking is enabled. The color was picked from the darker yellow plane inside the camera image.

The two pictures below show the tracking result depending on the threshold value.



Left: The threshold value is very small (17), only the darker yellow plane is tracked.



Right: The threshold value is increased to 72. The lighter yellow plane is now tracked as well.

Blur
Use this option to smooth the detected areas, so that rough pixels become smooth spots (0 = minimum blur, 15 = maximum blur).

Please be aware that increasing the blur value takes more performance!

Invert
The inverted mode is enabled by default. In BW-Mode all lighter parts of the camera image will be shown in black, the darker parts will be shown in white, depending on the Threshold value. To disable the color inversion please remove the check.
As already mentioned above: The point detection will take effect for the white parts of the black & white image, not for the black parts.

Use Lookup
Loads the Camera Lookup file with the type .dat.

Inflate
This option inflates the detected areas (0 = minimum inflation, 15 = maximum inflation).

Shrink
The result of the (eventually blurred and inflated) areas will be shrunk when using this option (0 = minimum shrinking, 15 = maximum shrinking).

Please note:
There are no recommendations which values would be best to enter for Blur, Inflate and Shrink. It depends on your setup and especially the lighting conditions.

Motion Filter
Tracks only the movement when this option is enabled. Static objects will be ignored and have no influence on tracking.

Use Mask
This enables the option to load a BW mask. The black parts cover the active regions in the video input and the white parts remain active.

Save Frame
Opens the file browser under the path C:\Christie\content\projects to save the image as PNG with the same pixel dimensions as the chosen video format.

Load Mask…
Opens the file browser under the path C:\Christie\content\projects to load a mask file. The types Bitmap, Jpeg, PNG or GIF can be used. Please note that the loaded mask must match the size of your chosen format.

Create Mask
Captures the current frame and inverts this image. The file browser under C:\Christie\content\projects opens automatically to save the mask as PNG.

7.6.5.5 Point Tracker

This chapter explains the "Point Tracker" settings in the Camera Tracker [1291] in Widget Designer.

Min Width / Max Width
Define which minimum and maximum width (in pixels) a spot should have to be detected as point.
The pixel sizes are related to the camera's resolution.

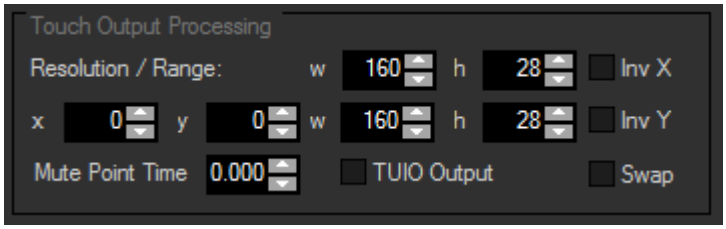
Min Height / Max Height
Define which minimum and maximum height (in pixels) a spot should have to be detected as point.
The pixel sizes are related to the camera's resolution.

Max Delta
The Max Delta value defines how far a point is allowed to move (in px, related to the camera's resolution) between two image frames in order to still be detected as point.

Damping
The damping allows reducing noisy input values. The damping factor is set by default to 0,3. This setting can be changed from 0,1 = maximum damping up to 1 = no damping.

7.6.5.6 Touch Output Processing

This chapter explains the "Touch Output Processing" settings in the Camera Tracker [1291] in Widget Designer.



Resolution/ Range (w/h)
Please enter here the resolution of your monitor (w=width, h=height) where the WD runs on. This is needed to adopt the positions of the detected touch points to fit to your screen.

Inv X
Check "Inv X" if you need to invert the X axis.

x/y/w/h (Range Conversion for Touch points)
The range conversion for touch points can be used if the camera point tracker should only affect a specific screen region, or if multiple Widgets are used with multiple camera point trackers together.
Enter the position of the region's starting pixel (left top pixel) inside the text fields for x and y.
The regions size will be defined by entering its width (w) and height (h).

Inv Y
Check "Inv Y" if you need to invert the Y axis.

TUIO Output
In order to output multi touch data to other applications this option allows to send the Camera Point Tracker's multi touch data via the open source protocol TUIO.
This protocol is widely used around the world by many application developers and is a commonly known way to transmit the individual touches.
To use Camera Point Tracker with TUIO, set up the TUIO host IP and Port in the Connection Manager.

Swap
Check "Swap" if you need to swap the X and Y axis, e.g. when the camera is turned about 90°.

7.6.5.7 Canvas

This chapter explains the "Canvas" settings in the Camera Tracker [1291] in Widget Designer.

Draw Contour
Tick the check box to activate the drawing contour of your video input. If the check box is ticked then Pandoras Box will output the drawing contour on the respective Canvas asset.

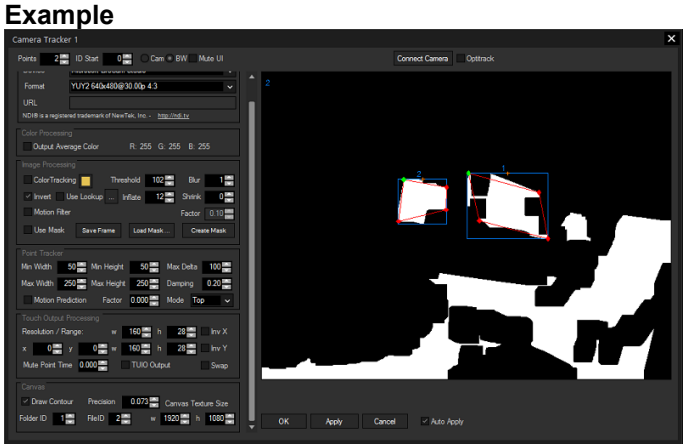
Precision
Enter the precision value of the drawing contour. The default value is 0.001 and can be increased to 1.

Folder ID
Enter the Folder ID of your Canvas asset in Pandoras Box.

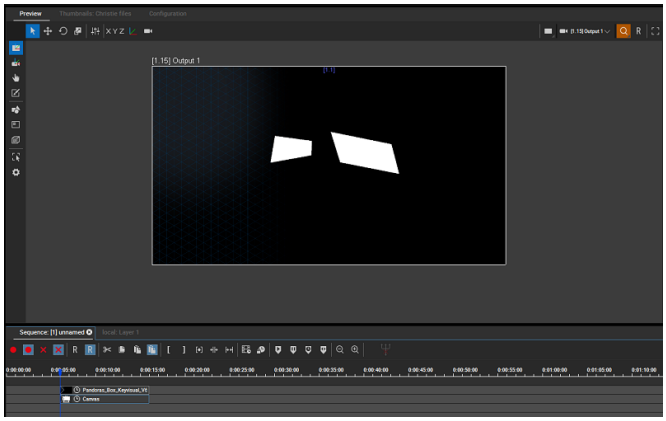
File ID
Enter the File ID of your Canvas asset in Pandoras Box.

Canvas Texture Size w/h
Enter the pixel size [width,height] of your Canvas asset [280] in Pandoras Box.

Example



Left: The drawing contour of the two tracking points are displayed in red (precision value = 0.073).



Right: The red contour results in a white shape that is displayed on the Pandoras Box canvas asset.

7.6.6 Modbus

Modbus is a serial communications protocol, designed for connecting industrial electronic devices. It was developed with industrial implementations in mind, but is already used for a wide variety of other applications. One advantage of this protocol is its simplicity, it is openly published and royalty-free, easy to deploy and maintain, and simply moves raw bits or words without placing many restrictions.

Modbus enables communication among many devices connected to the same network, for example a system that measures temperature and humidity and sends the results to a computer. For more information about Modbus and a general overview over the protocol, please refer to the respective Wikipedia article.

Widget Designer enables you to connect to a network of Modbus devices and retrieve or set data for certain devices. Input [1015] and output [1196] nodes are available, as well as options for scripting [1302].

You can find all tools for managing your Modbus connections in the Devices menu [810]. Please note that the Modbus device is also supported by the Widget Designer Free edition.

For getting started, click on "Add Master" and the Modbus Master Configuration dialog [1299] will open.

For managing existing Masters, click on "Master List" to open the Modbus Devices dialog [1304] providing an overview over all set up devices and their connection status.

As soon as you add Masters, they will be listed in this menu for quick access.

More information about patches within a Master can be found in the topic Modbus Patch Configuration [1300]. The chapter Using Modbus [1302] explains how to utilize values with nodes and scripting.

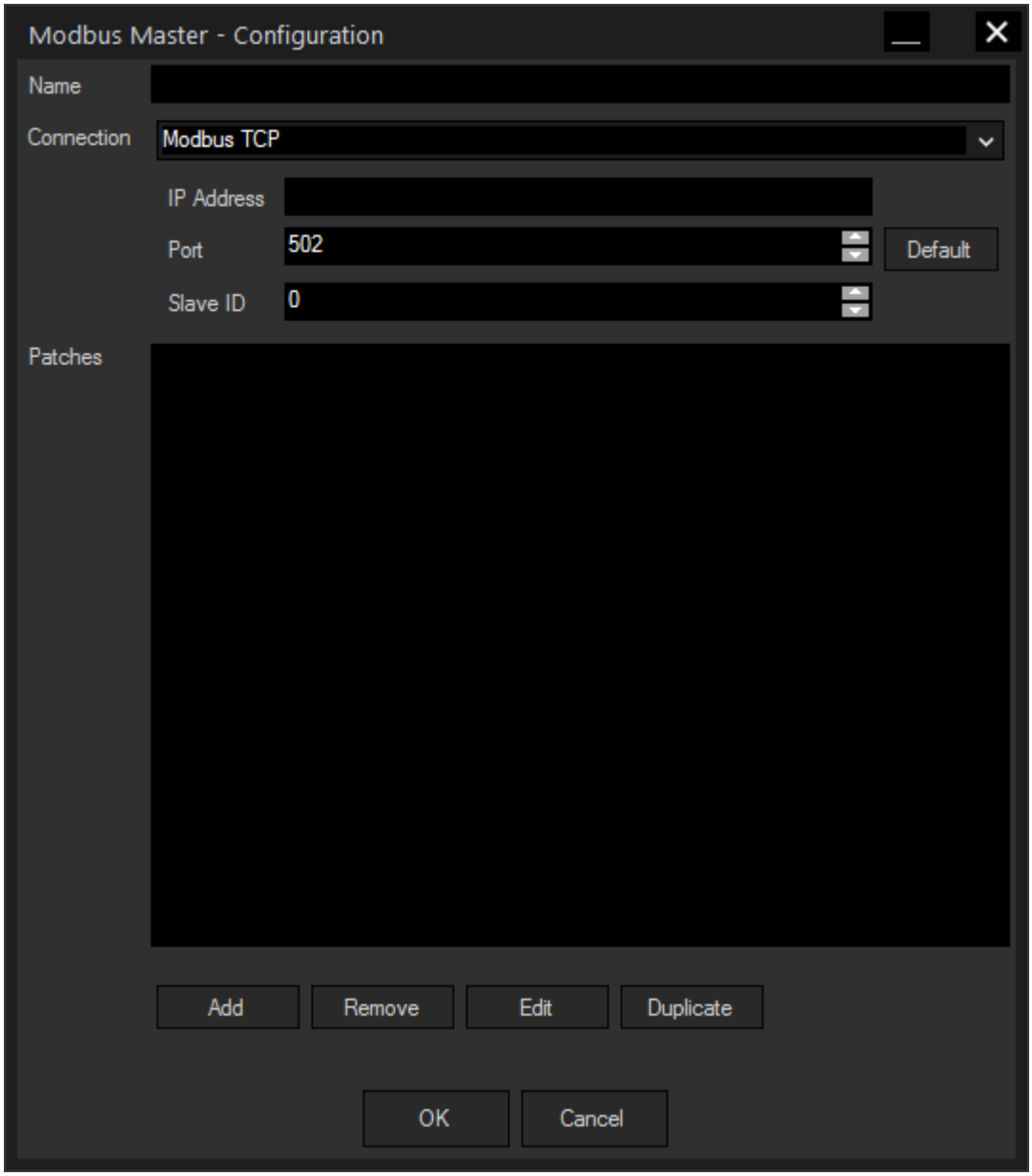
7.6.6.1 Modbus Master Configuration

The Modbus Master Configuration dialog enables you to establish a connection between WD and a Modbus control device and manage the connected infrastructure. If you are interested in setting up a patch within a Master, please refer to the next topic, Patch Configuration [1300]. For more information about how to utilize Modbus values in WD, please have a look at Using Modbus [1302].

If you want to add a new Modbus Master, go to Devices > Modbus > Add Master, or press the "Add" button in the Modbus Devices dialog [1304] (Master List).

Already existing Master configurations can be accessed with the list in the Devices > Modbus menu, or via the "Edit" button in the Master List.

Setting up a Master



Name:
Enter a name for your Modbus Master. It may consist of letters (lowercase or capital) and numbers, as well as underscore "_", the first character must be a letter. A valid name would be e.g. "Modbus_Motordevices1". The name has to be unique in WD.

Scripting members [1904] for this Master will be available when typing this name in a scripting field. This is explained in the chapter Using Modbus [1302].
When using nodes (e.g. the Modbus Input [1015]) the name is available in the respective drop-down menu.

Variables [1900]

Connection:
Currently, only TCP is available for connecting a Modbus device. A serial connection will be added in future versions.

IP Address:
Enter here the IP address of your Modbus control device, e.g. a bus coupler.

Port:
The default port for Modbus TCP communication is 502. If your setup requires another port, you can enter it here. The **Default** button reverts the port to 502.

Slave ID:
If you have several control units on the same bus, you can address specific units with the Slave ID. An ID of 0 is used for addressing all connected devices at once. As TCP is already a connection-oriented protocol, only one device is directly connected by definition and the Slave ID is not crucial.

Patches:
As soon as you define patches [1300] for controlling your hardware in- and outputs, they will appear in the list.

For adding a new patch, press the **Add** button which opens the Modbus Patch Configuration dialog. To remove an existing patch, select the entry in the list and press the **Remove** button. The **Edit** button opens the Modbus Patch Configuration dialog for the selected list entry, the **Duplicate** button creates a new patch with the same settings as the selected patch.

Press **OK** for submitting all changes or **Cancel** to revert to the former state.

After creating or editing a Master, please check the connection state of the device in the Master List [1304] and initiate a "Connect" or "Reconnect" if necessary.

7.6.6.2 Modbus Patch Configuration

This chapter explains how to set up a patch within a Modbus Master device. To learn how to set up a Master, please refer to the previous chapter Modbus Master Configuration [1299]. If you would like to learn how to use values with Modbus, please refer to the next topic Using Modbus [1302].

To open the Modbus Patch Configuration dialog, click "Add" or "Edit" at the bottom from the Modbus Master Configuration dialog [1299].

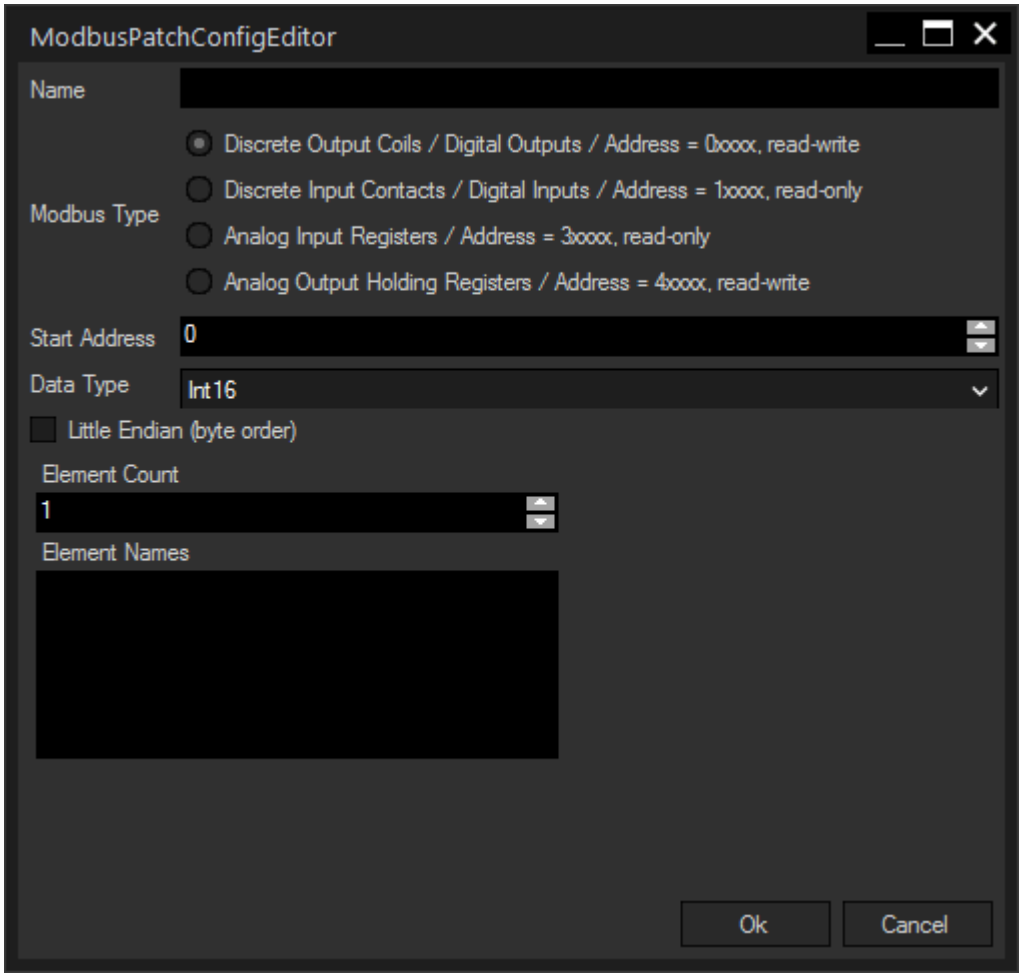
Setting up a Patch

Modbus specifies four different input/output device types:

- Discrete Output Coils (digital output, 1bit)
- Discrete Input Contacts (digital input, 1bit)
- Analog Output Holding Registers (analog output, 16bit)
- Analog Input Register (analog input, 16bit)

Note that both output options are "read-write" allowing to get and set data whilst the input options are "read-only" allowing only to get values.

Widget Designer also supports the possibility to swap the byte order of sent and received data packets, as well as a 32bit data size.



Name:
Enter a name for your Modbus patch. It may consist of letters (lowercase or capital) and numbers, as well as underscore "_", the first character must be a letter. A valid name would be e.g. "Motor_123". The name has to be unique in WD.

Scripting members [1904] for this Patch will be available when typing this name in a scripting field after the Master's name followed by a dot. This is explained in the chapter Using Modbus [1302].
When using nodes (e.g. the Modbus Input [1015]) the name is available in the respective drop-down menu.

Modbus Type:
Select here what kind of Modbus device you want to address with this patch.

Start Address:
Enter here the start address of your patch. Please bear in mind that some hardware devices require specific addresses.

Data Type:
Select how the incoming data should be interpreted. Int16 and UInt16 are the default configuration for one data word. Int16: values -32768 to 32767, UInt16: value 0 to 65535. Int32 and UInt32 require two data words (achievable e.g. with using two combined 16bit registers).
The discrete in- and outputs only contain a 1bit value (0 or 1), which are still transmitted as one 16bit word.

The data type FHPP, the Festo handling and positioning profile, is also available. For more information about how to use this, please refer to the chapter Using Modbus [1302].

Little Endian (byte order):
Check this box to switch the order of bits within the two bytes forming one data word of 16bit. Per default, this check box is not checked as Big Endian is the standard for most network devices.

Little Endian (word order):
Only available for Int32 or UInt32. Check this box to swap the order of the two 16bit words forming the 32bit value.

Element Count:
Enter here how many elements of the selected data type should be addressed with this patch.

Element Names:
Optionally, you can enter a name for addressing each of the elements via scripting. Element names are separated by a new line.
If you don't enter any names, the elements can still be accessed by their index number. For example:
`Mastername.Patchname.Get(Elementname)` or `Mastername.Patchname.Get(Indexnumber)`.
When using nodes (e.g. the Modbus Input [1015]) the elements can be accessed by their name or index number.

Press **OK** for submitting all changes or **Cancel** to revert to the former state.

Refresh Delay:
Only available for the data type Fhpp. Enter a delay for refreshing the value in milliseconds.

7.6.6.3 Using Modbus

This chapter shows how to use Modbus values in Widget Designer. To follow the examples, please set up a Master, a patch, or a patch element as explained in the previous chapters Modbus Master Configuration [1299] and Modbus Patch Configuration [1300].

The Modbus data can be sent and received via two independent approaches: scripting and nodes. Both ways can be used at the same time, it depends on the application what suits your needs best.

How to work with Modbus

For detailed information about the setup of the Modbus Input [1015] and Modbus Output [1196] node, please refer to the respective chapters.

The scripting members [1904] for each Master and its respective patches are available as soon as the Master setup is confirmed. Simply start writing the Master name and let the Script Assistant complete the text for you.

There are functional members available for connecting, reconnecting and disconnecting a complete Master to or from its hardware. Please see the below list for all available members and examples.

Furthermore, each patch can be addressed independently. All used patch names will be offered by the script assist after typing a dot behind the master name. Likewise, additional members for the patch can be accessed by setting another dot behind the patch name.

If you defined element names in the patch configuration, they will also be available here for retrieving or assigning values directly. Please note that all Modbus members are offered for all patch types, though the "Set" commands do not have any effect on patches configured as Input Contacts or Input Registers. Same goes for setting a value with an equal sign "=".

Using Fhpp with Modbus

FHPP, Festo handling and positioning profile, is a data profile designed for handling and positioning tasks, e.g. executed by robot arms.

For more information about the profile itself and an exact description of the used byte structure, please visit Festo's website and take a closer look at the specifications:
https://www.festo.com/net/SupportPortal/Files/403824/CMM_-FHPP_2010-06a_555696g1.pdf

WD uses solely the **Record Selection** mode, which means that the flags CCON_OPM1 = 0 and CCON_OPM2 = 0 in the control byte need to be set. The same applies for the operation acknowledgment flags in the status byte, SCON_OPM1 and SCON_OPM2.

The Modbus Fhpp input node [1013] allows you to read out each flag, set for incoming control and status bytes, as well as the actual position value and input / output record number.

If you decide to send a value in Fhpp format, you can set (but also read out) each flag individually by accessing them via the patch members. An example is shown in the table below.

Modbus Members

Please find here a table with all members available for Modbus. The commands can be entered in any scripting field.

Object	Master Members	Patch Members	Example

MasterName	.Connect
	.Reconnect
	.Disconnect
			`M1.Connect`
			`M1.Reconnect`
			`M1.Disconnect`

Select one of those members to connect, reconnect or disconnect the specified Master without having to open the Master List.			
	.PatchName		`M1.DigitalOut`
To access the members of a specific patch, select the respective patch name from the list.			
		.Get(ElementIndex or ElementName)	`Label1.Text = M1.DigitalOut.Get(0)` `Label2.Text = M1.DigitalIn.Get("A")`
Use the "Get" member to retrieve the current value of any single element of this patch, may it be configured as input or output. You can use the element's index as integer or its name as string if you have assigned one.			

		.Set(ElementIndex or ElementName)	`M1.DigitalOut.Set(1,0)`
			`M1.DigitalOut.Set("A",false)`
			`M1.AnalogOut.Set(0,11001)`

Use the "Set" member to set the value of one single element of this patch, it has to be configured as output. You can use the element's index as integer or its name as string if you have assigned one. Digital Output Coils can receive their output state either the integer "0" and "1" or as Boolean expression "true" and "false"			
		.SetMany(ElementIndex or ElementName, list of values)	`M1.DigitalOut.SetMany(0,[1,1,0])` `M1.AnalogOut.SetMany("C", [13579,0,1001])`
The "SetMany" member allows you setting the output value of several Coils or Registers of one patch at once. Specify the starting element and enter a list of values. The values will then be applied with each list element corresponding to a patch element consecutively following the start element.			

		.ElementName	`M1.DigitalOut.A = true`
			`M1.DigitalOut.B = 0`
			`M1.AnalogOut.C = 556`
			`Label1.Text = M1.AnalogIn.D`
			`Debugmessage(M1.DigitalIn.E.ToBoolean)`

Set or retrieve the current value of any single element of this patch, may it be configured as input or output. Setting a value effects only patches configured as output. To use this member, element names must have been set beforehand in the patch configuration. This member returns an **object** data type, so further data type specific members [1914] can be used.			

		.ccon_enable
		.cpos_jogp
		.out_recordset
		.position
		.scon_warn
		.spos_still
		...
			`M1.FhppOut.ccon_reset = true`
			`M1.FhppOut.spos_halt = 0`
			`M1.FhppOut.in_recordset = 23`
			`Label1.Text = M1.FhppIn.position`

Only available for Fhpp type patches. Set and retrieve each single flag of the control and status bytes of a message received in Fhpp format.			

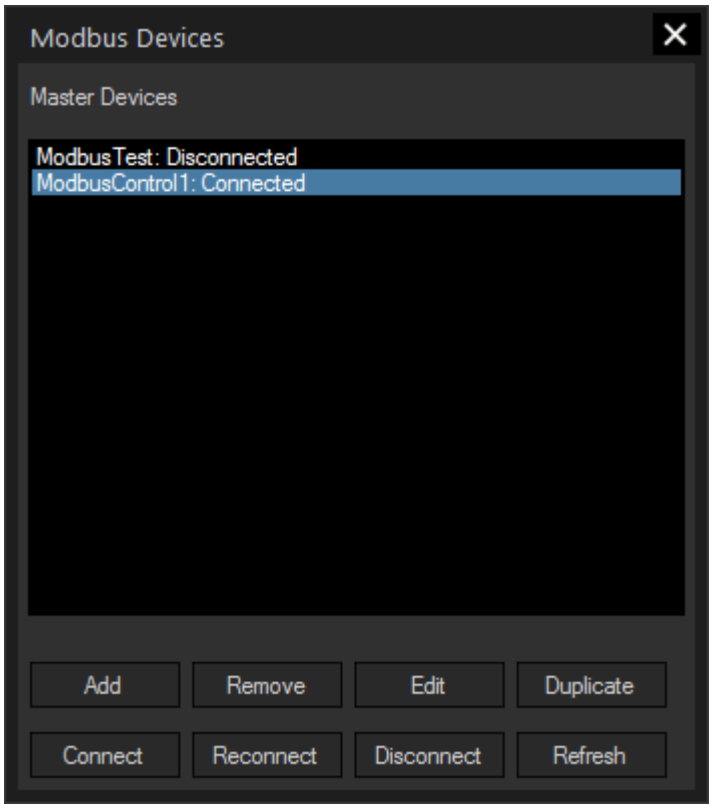
You can do so by either using "0" and "1" as well as Boolean values.
Also available are the input and output record set numbers (as byte value, 0 - 255) and the current position as 32bit integer value.

7.6.6.4 Modbus Master List

The Master List, or Modbus Devices dialog, enables you to quickly overview all set up Modbus devices.

It shows the current connection state of all devices and provides you with additional editing functionality.

To open it, choose Devices menu > Modbus > Master List.



Add:
Press Add to open the Modbus Master Configuration [1299] dialog and create a new Master.

Remove:
To delete a Master, select it in the list and press Remove.

Edit:
Pressing Edit will open the Modbus Master Configuration dialog for the selected Master.

Duplicate:
This option creates a new Master with the exact same configuration as the selected one from the list.

Connect / Reconnect / Disconnect:
Press one of those buttons to perform the respective action on the selected Master. The commands are also available as members of the Master in case you like to script that. Examples can be found in the previous chapter Using Modbus [1302].

Refresh:
For initiating a manual status update for all devices in the list, press Refresh. It might be necessary to press Refresh once at the beginning when loading an existing project with Modbus devices already set up.

7.6.7 Configuration

With the Configuration dialog WD version 6.1.1 introduces a new way of handling device communication and triggering.
This enables you to set and retrieve values from devices such as Phidgets or Christie Terra, without the necessity to manually implement ways of interfacing with them.
Actions and parameter changes of those devices can be monitored with an Event Listener and used as triggers for scripts.

Over time, more and more devices will be added to the Configuration dialog and allow the user to easily access them in a uniform and highly automated process.

New devices or Event Listeners are added as WD objects and full functionality is provided for scripting via Object and Member Notation [1904].

For managing your Devices and Event Listeners, open the Configuration dialog via Devices > Configuration. If you created them already, you can also directly click on one in the Devices menu (e.g. Devices > Christie > Terra > Christie_Terra1) as this opens the chosen device in the Configuration dialog. The licensed Widget Designer supports all listed devices whilst the Free edition includes only some of them.

In the dialog itself, you can configure existing devices, delete them or create new ones.

When you select one of the existing Devices or Listeners in the list, the respective editable properties are displayed.

Currently, you can configure the following devices. They are listed when you select the "All" or "Devices" tab in the Configuration dialog:
Barco [1306]
Christie [1318]
Fader Extension [1354]
Jog/Shuttle [1364]
Lightware [1378]
Phidgets [1383]
PowerPoint [1468]
TCP Client [1471]
UDP Client [1475]
UDP Server&Client [1479]

Currently, you can configure the following scripting objects. They are listed when you select the "All" or "Scriptings" tab in the Configuration dialog:
Event Listeners [1353]
Phidgets IRCodeTable [1418]

7.6.8 Barco

You can add two different Barco devices to the Configuration dialog:
- Barco ImagePro2 [1306]
- Barco MatrixPro2 [1314]

7.6.8.1 Barco ImagePro2

The Barco ImagePro2 Device in the Configuration dialog allows to easily remote control different functions of a Barco ImagePro2 device. Note that the device is only available in the licensed Widget Designer edition, not the Free version.

Adding a New Barco ImagePro2 Device

To add a Barco ImagePro2 device, open the Devices menu and select "Barco > ImagePro2 > Create ImagePro2 Device". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side you see several options:

The **Name** is the unique identifier for this Barco object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Barco device is a filled blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A filled gray circle indicates a disabled device.

Now, please enter the **IP address** of the Barco device. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

If the connection is successful, the **Product Type** shows to which device you are connected, e.g. "ImagePro-II Version 03.61.01".

You can close the dialog now. The newly created device will also be added to the Devices menu > Barco > ImagePro2 and can be opened from here or with Devices menu > Configuration.

Using the Device in Regular Scripting

After creating a device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Barco_ImagePro2_1"), Script Assist will offer you a list of all Barco ImagePro2 Members [1307].

You can for example save the current system state:
```
Barco_ImagePro2_1.SaveSystemState
```

You can also retrieve specific information via scripting:
```
vstring = Barco_ImagePro2_1.LastMessageReceived
```

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and Barco ImagePro2 Events [1307] for an overview (with examples and description) over the events raised by this device.

7.6.8.1.1 Barco ImagePro2 Events

This chapter gives an overview of the events that are raised by a Barco ImagePro2 device and that can be used in the Event Listener [1353].
Please read the chapters "Barco ImagePro2" [1306] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Barco ImagePro2 Members" [1307].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the device dis-/connects.

MessageReceived

Example:
Label1.Text = MessageReceived

This event is raised whenever the connected device sends a message.
The event returns one parameter to WD which is a string value with the name "MessageReceived" and holds the message.
If you select this event and copy the example into the scripting field of the Event Listener, it will write the message into the Label [888] with ID 1 whenever the device sends one.

7.6.8.1.2 Barco ImagePro2 Members

This chapter gives an overview of the members available for the Barco ImagePro2 device.
Please read the chapter "Barco ImagePro2" [1306] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

CaptureLogo

Barco_ImagePro2_ID.CaptureLogo

Example:
Barco_ImagePro2_1.CaptureLogo

This captures the logo which can be called via the member SetInput("Logo") [1313]. Please refer to the manual of your ImagePro2 device for a documentation of the Logo feature

CustomCommand

Barco_ImagePro2_ID.CustomCommand(command)

Example:
Barco_ImagePro2_1.CustomCommand(String)

This sends a custom command to the device "Barco_ImagePro2_1". Please check the manual of your ImagePro2 device for a documentation which commands it supports.

Disable

Barco_ImagePro2_ID.Disable

Example:
Barco_ImagePro2_1.Disable

This disables the device "Barco_ImagePro2_1" in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1308] member to reverse this command.

Enable

Barco_ImagePro2_ID.Enable

Example:
Barco_ImagePro2_1.Enable

This enables the device "Barco_ImagePro2_1" in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1308] member to reverse this command.

FadeFromBlack

Barco_ImagePro2_ID.FadeFromBlack(time)

Example:
Barco_ImagePro2_1.FadeFromBlack(1.5)

This fades the current black output within 1.5seconds to the normal output again when in was faded out before, e.g. using the member `FadeToBlack` [1308]. This is not to be confused with the input named "Black".

FadeToBlack

Barco_ImagePro2_ID.FadeToBlack(time)

Example:
Barco_ImagePro2_1.FadeToBlack(1.5)

This fades the current output within 1.5seconds to a black output. You can fade back in using the member `FadeFromBlack` [1308]. This is not to be confused with the input named "Black".

Freeze

Barco_ImagePro2_ID.Freeze(value)

Example:
vstring = Barco_ImagePro2_1.Freeze

This member sets and returns the "Freeze" state as a Boolean value. The result is "True" if the video image on the display is freezed, and "False" if it is not.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.Freeze`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.Freeze)`
An alternative member is: `GetFreeze` [1309]

Example2:
Barco_ImagePro2_1.Freeze(True)

The second example shows, how to freeze the current output, in other words the video image on the display. You can reverse this by setting the parameter to "False" or by switching to another inpute, e.g. using the member `SetInput` [1313].

An alternative member is: `SetFreeze` [1313]

GetFreeze

Barco_ImagePro2_ID.GetFreeze

Example:
vstring = Barco_ImagePro2_1.GetFreeze

This returns the "Freeze" state as a Boolean value. The result is "True" if the video image on the display is freezed, and "False" if it is not.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.GetFreeze`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.GetFreeze)`
An alternative member is: `Freeze` [1308]

GetInput

Barco_ImagePro2_ID.GetInput

Example:
vstring = Barco_ImagePro2_1.GetInput

This returns the input channel that is routed. The result is a string value, e.g. "DVI".
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.GetInput`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.GetInput)`
An alternative member is: `Input` [1310]

GetLastMessageReceived

Barco_ImagePro2_ID.GetLastMessageReceived

Example:
vstring = Barco_ImagePro2_1.GetLastMessageReceived

This returns the last message that was received from the device "Barco_ImagePro2_1" as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.GetLastMessageReceived`
or simply display it in the Debug Logger [812]:
`DebugMessage(Barco_ImagePro2_1.GetLastMessageReceived)`

An alternative command is: `LastMessageReceived` [1311]

GetLockFrontPanel

Barco_ImagePro2_ID.GetLockFrontPanel

Example:
vstring = Barco_ImagePro2_1.GetLockFrontPanel

This returns the "FrontPanelLocked" state as a Boolean value. The result is "True" if the front panel of the ImagePro2 device is locked and "False" if it is not.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.GetLockFrontPanel`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.GetLockFrontPanel)`
An alternative member is: `LockFrontPanel` [1312]

GetProductType

Barco_ImagePro2_ID.GetProductType

Example:
vstring = Barco_ImagePro2_1.GetProductType

This returns the product type from the device "Barco_ImagePro2_1" as a string value. The result could look as follows: Barco ImagePRO-II Version 03.61.01
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.GetProductType`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.GetProductType)`

An alternative command is: `ProductType` [1312]

Input

Barco_ImagePro2_ID.Input

Example:
vstring = Barco_ImagePro2_1.Input

This member sets and returns the input channel that is routed. The result is a string value, e.g. "DVI".
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.Input`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.Input)`
An alternative member is: `GetInput` [1309]

Example2:
Barco_ImagePro2_1.SetInput("SDI1")

The second example shows, how to select and route the according input to be processed. You can choose eight inputs: "DVI", "HD15", "HDMI", "DisplayPort", "SDI1", "SDI2", "Black", "Logo"

An alternative member is: `SetInput` [1313]

IpAddress

Barco_ImagePro2_ID.IpAddress

Example:
vstring = Barco_ImagePro2_1.IpAddress

This returns the IP address of the device "Barco_ImagePro2_1" as a string. The result could look as follows: 10.169.80.10
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.IpAddress`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.IpAddress)`

In return, you can also assign an IP address to the connected ImagePro2 device. WD automatically connects to it afterwards.
`Barco_ImagePro2_1.IpAddress = "10.169.80.10"`

IsConnected

Barco_ImagePro2_ID.IsConnected

Example:
vstring = Barco_ImagePro2_ID.IsConnected

This member returns the connection status for the device "Barco_ImagePro2_1" as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_ID.IsConnected`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_ID.IsConnected)`

IsEnabled

Barco_ImagePro2_ID.IsEnabled

Example:
vstring = Barco_ImagePro2_ID.IsEnabled

This member returns the status of the check box "Enable" for the device "Barco_ImagePro2_1" in the Configuration dialog [1305] as a string or Boolean value. The result is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_ID.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_ID.IsEnabled)`

LastMessageReceived

Barco_ImagePro2_ID.LastMessageReceived

Example:
vstring = Barco_ImagePro2_1.LastMessageReceived

This returns the last message that was received from the device "Barco_ImagePro2_ID" as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.LastMessageReceived`
or simply display it in the Debug Logger [812]:
`DebugMessage(Barco_ImagePro2_1.LastMessageReceived)`

An alternative command is: GetLastMessageReceived [1309]

LockFrontPanel

Barco_ImagePro2_ID.LockFrontPanel(value)

Example:
vstring = Barco_ImagePro2_1.LockFrontPanel

This member sets and returns the "FrontPanelLocked" state as a Boolean value. The result is "True" if the front panel of the ImagePro2 device is locked and "False" if it is not.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.LockFrontPanel`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.LockFrontPanel)`
An alternative member is: GetLockFrontPanel [1310]

Example2:
Barco_ImagePro2_1.LockFrontPanel(True)

The second example shows, how to lock the front panel of the ImagePro2 device. When the front panel is locked, button presses have no effect. In the locked mode:
1) All button presses and all turns of the ADJUST knob are ignored.
2) All Ethernet and serial communications commands function normally.
3)All button lights continue to reflect the correct state of the inputs.

An alternative member is: SetLockFrontPanel [1313]

ProductType

Barco_ImagePro2_ID.ProductType

Example:
vstring = Barco_ImagePro2_1.ProductType

This returns the product type from the device "Barco_ImagePro2_1" as a string value. The result could look as follows: Barco ImagePRO-II Version 03.61.01
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_ImagePro2_1.ProductType`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_ImagePro2_1.ProductType)`

An alternative command is: GetProductType [1310]

SaveSystemState

Barco_ImagePro2_ID.SaveSystemState

Example:
Barco_ImagePro2_1.SaveSystemState

This saves the System State to non-volatile memory.

SetFreeze

Barco_ImagePro2_ID.SetFreeze(value)

Example:
Barco_ImagePro2_1.SetFreeze(True)

This freezes the current output, in other words the video image on the display. You can reverse this by setting the parameter to "False" or by switching to another inpute, e.g. using the member SetInput[1313].

An alternative member is: Freeze[1308]

SetInput

Barco_ImagePro2_ID.SetInput(value)

Example:
Barco_ImagePro2_1.SetInput("SDI1")

This selects and routes the according input to be processed. You can choose eight inputs: "DVI", "HD15", "HDMI", "DisplayPort", "SDI1", "SDI2", "Black", "Logo"

An alternative member is: Input[1310]

SetLockFrontPanel

Barco_ImagePro2_ID.SetLockFrontPanel(value)

Example:
Barco_ImagePro2_1.SetLockFrontPanel(True)

This locks the front panel of the ImagePro2 device. When the front panel is locked, button presses have no effect. In the locked mode:
1) All button presses and all turns of the ADJUST knob are ignored.
2) All Ethernet and serial communications commands function normally.
3)All button lights continue to reflect the correct state of the inputs.

An alternative member is: LockFrontPanel[1312]

Testpattern

Barco_ImagePro2_ID.Testpattern(patternType,motionMode)

Example:
Barco_ImagePro2_1.Testpattern("ColorBars_100P",True)

This calls the test pattern "ColorBars_100P" and turns on diagonal motion. You can choose 14 different test pattern, the names are offered in the Script Assistant.

7.6.8.2 Barco MatrixPro2

The Barco MatrixPro2 Device in the Configuration dialog allows to easily remote control different functions of a Barco MatrixPro2 device, e.g a MatrixPro-II 16x16 DVI Router. Note that the device is only available in the licensed Widget Designer edition, not the Free version.

Adding a New Barco MatrixPro2 Device

To add a Barco MatrixPro2 device, open the Devices menu and select "Barco > MatrixPro2 > Create MatrixPro2 Device". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side you see several options:

The **Name** is the unique identifier for this Barco object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Barco device is a filled blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A filled gray circle indicates a disabled device.

Now, please enter the **IP address** of the Barco device. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

If the connection is successful, the **Product Type** shows to which device you are connected.

You can close the dialog now. The newly created device will also be added to the Devices menu > Barco > MatrixPro2 and can be opened from here or with Devices menu > Configuration.

Using the Device in Regular Scripting

After creating a device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Barco_MatrixPro2_1"), Script Assist will offer you a list of all Barco MatrixPro2 Members [1315].

You can for example route a certain input to an output:

```
Barco_MatrixPro2_1.Route(2,1)
```

You can also retrieve specific information via scripting:

```
vstring = Barco_MatrixPro2_1.LastMessageReceived
```

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and Barco MatrixPro2 Events [1315] for an overview (with examples and description) over the events raised by this device.

7.6.8.2.1 Barco MatrixPro2 Events

This chapter gives an overview of the events that are raised by a Barco MatrixPro2 device and that can be used in the Event Listener [1353].
Please read the chapters "Barco MatrixPro2" [1314] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Barco MatrixPro2 Members" [1315].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the device dis-/connects.

MessageReceived

Example:
Label1.Text = MessageReceived

This event is raised whenever the connected device sends a message.
The event returns one parameter to WD which is a string value with the name "MessageReceived" and holds the message.
If you select this event and copy the example into the scripting field of the Event Listener, it will write the message into the Label [888] with ID 1 whenever the device sends one.

7.6.8.2.2 Barco MatrixPro2 Members

This chapter gives an overview of the members available for the MatrixPro2 device.
Please read the chapter "Barco MatrixPro2" [1314] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

CustomCommand

Barco_MatrixPro2_ID.CustomCommand(command)

Example:
Barco_MatrixPro2_1.CustomCommand("String")

This sends a custom command to the device "Barco_MatrixPro2_1". Please check the manual of your Barco device for a documentation which commands it supports.

Disable

Barco_MatrixPro2_ID.Disable

Example:
Barco_MatrixPro2_1.Disable

This disables the device "Barco_MatrixPro2_1" in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1316] member to reverse this command.

Enable

Barco_MatrixPro2_ID.Enable

Example:
Barco_MatrixPro2_1.Enable

This enables the device "Barco_MatrixPro2_1" in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1315] member to reverse this command.

GetLastMessageReceived

Barco_MatrixPro2_ID.GetLastMessageReceived

Example:
vstring = Barco_MatrixPro2_1.GetLastMessageReceived

This returns the last message that was received from the device "Barco_MatrixPro2_1" as a string value. The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_MatrixPro2_1.GetLastMessageReceived`
or simply display it in the Debug Logger [812]:
`DebugMessage(Barco_MatrixPro2_1.GetLastMessageReceived)`

An alternative command is: `LastMessageReceived` [1317]

GetProductType

Barco_MatrixPro2_ID.GetProductType

Example:
vstring = Barco_MatrixPro2_1.GetProductType

This returns the product type from the device "Barco_MatrixPro2_1" as a string value. The result could look as follows: Barco Matrix Pro II
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_MatrixPro2_1.GetProductType`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_MatrixPro2_1.GetProductType)`

An alternative command is: `ProductType` [1317]

IpAddress

Barco_MatrixPro2_ID.IpAddress

Example:
vstring = Barco_MatrixPro2_1.IpAddress

This returns the IP address of the device "Barco_MatrixPro2_1" as a string. The result could look as follows: 10.169.80.10
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_MatrixPro2_1.IpAddress`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_MatrixPro2_1.IpAddress)`

In return, you can also assign an IP address to the connected Barco device. WD automatically connects to it afterwards.
`Barco_MatrixPro2_1.IpAddress = "10.169.80.10"`

IsConnected

Barco_MatrixPro2_ID.IsConnected

Example:
vstring = Barco_MatrixPro2_ID.IsConnected

This member returns the connection status for the device "Barco_MatrixPro2_1" as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_MatrixPro2_ID.IsConnected`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_MatrixPro2_ID.IsConnected)`

IsEnabled

Barco_MatrixPro2_ID.IsEnabled

Example:
vstring = Barco_MatrixPro2_ID.IsEnabled

This member returns the status of the check box "Enable" for the device "Barco_MatrixPro2_1" in the Configuration dialog [1305] as a string or Boolean value. The result is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_MatrixPro2_ID.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_MatrixPro2_ID.IsEnabled)`

LastMessageReceived

Barco_MatrixPro2_ID.LastMessageReceived

Example:
vstring = Barco_MatrixPro2_1.LastMessageReceived

This returns the last message that was received from the device "Barco_MatrixPro2_ID" as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_MatrixPro2_1.LastMessageReceived`
or simply display it in the Debug Logger [812]:
`DebugMessage(Barco_MatrixPro2_1.LastMessageReceived)`

An alternative command is: `GetLastMessageReceived` [1316]

ProductType

Barco_MatrixPro2_ID.ProductType

Example:
vstring = Barco_MatrixPro2_1.ProductType

This returns the product type from the device "Barco_MatrixPro2_1" as a string value. The result could look as follows: Barco Matrix Pro II
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Barco_MatrixPro2_1.ProductType`
or simply display it in the Debug Logger [812]: `DebugMessage(Barco_MatrixPro2_1.ProductType)`

An alternative command is: `GetProductType` [1316]

Route

Barco_MatrixPro2_ID.Route(input,output)

Example:
Barco_MatrixPro2_1.Route(3,2)

This sends a command to the device "Barco_MatrixPro2_ID" which routes the input 3 to output 2.

7.6.9 Christie

You can add three different Christie devices to the Configuration dialog:
- Phoenix [1319]
- Spyder [1322]
- Terra [1332]

7.6.9.1 Phoenix

The Phoenix Device in the Configuration dialog allows to easily monitor and remote control different functions of a Christie Phoenix controller. Note that the device is only available in the licensed Widget Designer edition, not the Free version.

Adding a New Phoenix Device

To add a Phoenix device, open the Devices menu and select "Christie > Phoenix > Create Phoenix". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side, you see several options:

The **Name** is the unique identifier for this Phoenix object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your Phoenix device is a filled blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A filled gray circle indicates a disabled device.

Now, please enter the **IP address** of the Phoenix controller. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

You can close the dialog now. The newly created device will also be added to the Devices menu > Christie > Phoenix and can be opened from here or with Devices menu > Configuration.

Using the Device in Regular Scripting

After creating a device, you can access its commands and parameters easily via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Christie_Phoenix1"), Script Assist will offer you a list of all Phoenix Members [1321].
Some of them are commands to perform actions on the device, e.g. applying a layout:

```
Christie_Phoenix1.ApplyLayout("FullView")
```

You can also retrieve specific values via scripting in the same way as from other widgets (e.g. a Fader value):

```
vlist = Christie_Phoenix1.LayoutNames
```

This member returns all layout names as a list that can be displayed in the Debug Logger [812] or assigned to a variable [1900], DropDown List [868], etc.

The Phoenix tutorial [1345] shows how to build a small interface to interact with a Phoenix system and uses the most important commands and Event Listeners.

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and Phoenix Events [1320] for an overview (with examples and description) over the events raised by a Phoenix device.

7.6.9.1.1 Phoenix Events

This chapter gives an overview of the events that are raised by a Phoenix device and that can be used in the Event Listener [1353].
Please read the chapters "Phoenix" [1319] and "Event Listener" [1353] if you like to know how to add and use both. There is also a "Terra Tutorial" [1345] available which explains how to use Event Listeners.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the device dis-/connects.

LayoutNames

Example:
DropDownList1.SetItemsFromArray("layoutNames")

This event is raised whenever a layout is changed (i.e. edited, created or deleted) in the connected Phoenix system.
The event returns one parameter to WD which is a list with the name "layoutNames" and holds all layout names.
If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List [868] with ID 1 with all current layout names whenever a layout is edited, created or deleted.
You can also directly access the layout names via the member Christie_PhoenixID.LayoutNames [1341] within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.

SourceNames

Example:
DropDownList1.SetItemsFromArray("sourceNames")

This event is raised whenever a source is changed in the connected Phoenix system.
The event returns one parameter to WD which is a list with the name "sourceNames" and holds all source names.
If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List [868] with ID 1 with all current source names whenever a source is changed.
You can also directly access the device names via the member Christie_PhoenixID.SourceNames [1338] within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.

WallNames

Example:
DropDownList1.SetItemsFromArray("wallNames")

This event is raised whenever a wall is changed (i.e. edited, created or deleted) in the connected Phoenix system.
The event returns one parameter to WD which is a list with the name "wallNames" and holds all wall names.
If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List [868] with ID 1 with all current wall names whenever a wall is edited, created or deleted.
You can also directly access the receiver names via the member Christie_PhoenixID.WallNames [1342] within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.

LayoutRecalled

Example:
Label1.Text = "The layout named `"+layoutName+"` (ID:"+layoutId+") was applied on the wall named `"+wallName +"` (ID:" +wallId+ ")."

This event is raised whenever a layout is recalled in the connected Phoenix system.
The event returns five parameters to WD that can be used as variables in the below scripting field.
If you select this event and copy the example into the scripting field of the Event Listener, it will write the following in the Label with ID 1, according to your IDs and names of course:
The layout named `mixed` (ID:1) was applied on the wall named `OUT` (ID:2).

7.6.9.1.2 Phoenix Members

This chapter gives an overview of the members available for the Phoenix device.
Please read the chapter "Phoenix" [1319] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

7.6.9.2 Spyder

The Spyder Device in the Configuration dialog allows to easily remote control different functions of a Christie Spyder X80 device. You can also connect to a Spyder X20, with the difference that few commands (members of the Spyder device [1323]) are not supported there, as the protocol differs slightly. Note that the device is only available in the licensed Widget Designer edition, not the Free version.

Adding a New Spyder Device

To add a Spyder device, open the Devices menu and select "Christie > Spyder > Create Spyder". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.
Widget Designer

On the right side, you see several options:

The **Name** is the unique identifier for this Spyder object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only and for connecting the Spyder output node [1204]. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Spyder device is a filled blue circle.. A filled gray circle indicates a disabled device. As the connection is based on the UDP protocol [738], there is no status whether the Spyder is connected or not.

Now, please enter the **IP address** of the Spyder device. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Christie > Spyder and can be opened from here or with Devices menu > Configuration.

Using the Device in Regular Scripting

After creating a device, you can use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Christie_Spyder1"), Script Assist will offer you a list of all Spyder Members [1323].
Most commands assign or call a specific programming, e.g.:

```
Christie_Spyder1.ApplyTreatment(1,5)
```

You can also retrieve specific values via scripting:

```
vstring = Christie_Spyder1.IpAddress
```

If the needed functionality is not available with these ready-made members, you can use the following member to send a custom command:

```
Spyder1.SendCustomCommand("LSP 1 150 300 1920 5")
```

Using the Device in the Node System

After creating a device, you can also use it in the node system as an output node to send scaling and positioning values to certain layers. You can connect other input or filter nodes to it to receive their values, which allows interactive control. It is for example possible to create a Fader in Widget Designer which controls the position of a layer in Spyder. Please see the chapter "Spyder Output" [1204] for more information about the node itself or the chapter "Tutorial: Nodes" [939] for information about the node system.

7.6.9.2.1 Spyder Members

This chapter gives an overview of the members available for the Spyder device.
Please read the chapter "Spyder" [1322] if you like to know how to add and use a Spyder device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

ApplyTreatment

Christie_SpyderID.ApplyTreatment(layerId,treatmentId)

Example:
Christie_Spyder1.ApplyTreatment(1,5)

This member is based on the Spyder command "KTR–Treatment Recall". It recalls a treatment to a layer.

ClearLayerImage

Christie_SpyderID.ClearLayerImage(layerId)

Example:
Christie_Spyder1.ClearLayerImage(1)

This member is based on the Spyder command "SCL–Clear Still on Layer". It clears any loaded still images from a layer.

ClearOutputImage

Christie_SpyderID.ClearOutputImage(outputId)

Example:
Christie_Spyder1.ClearOutputImage(8)

This member is based on the Spyder command "CSO–Clear Still on Output". It clears an image currently loaded on a specified output ID.

Disable

Christie_SpyderID.Disable

Example:
Christie_Spyder1.Disable

This disables the Spyder device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1323] member to reverse this command.

Enable

Christie_SpyderID.Enable

Example:
Christie_Spyder1.Enable

This enables the Spyder device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1323] member to reverse this command.

FadeLayerIn

Christie_SpyderID.FadeLayerIn(layerId,duration)

Example:
Christie_Spyder1.FadeLayerIn(1,10)

This member is based on the Spyder command "TRN–Transition Layers". It transitions a layer on its currently assigned pixelspace.

The "duration" argument sets the transition duration in number of frames per second. The member FadeLayerOut [1324] transitions the layer off of the pixelspace.

FadeLayerOut

Christie_SpyderID.FadeLayerOut(layerId,duration)

Example:
Christie_Spyder1.FadeLayerOut(1,10)

This member is based on the Spyder command "TRN–Transition Layers". It transitions a layer off of its currently assigned pixelspace.

The "duration" argument sets the transition duration in number of frames per second. The member FadeLayerIn [1324] transitions the layer on the pixelspace.

FreezeLayer

Christie_SpyderID.FreezeLayer(layerId)

Example:
Christie_Spyder1.FreezeLayer(1)

This member is based on the Spyder command "FRZ–Freeze Layer". It freezes a layer.

Use the `.UnfreezeLayer` [1331] member to reverse this command.

FreezeOutput

Christie_SpyderID.FreezeOutput(outputId)

Example:
Christie_Spyder1.FreezeOutput(8)

This member is based on the Spyder command "OFZ–Freeze Output". It freezes or unfreezes one or more outputs. This command is only compatible with universal outputs. DX4 outputs do not support individual freeze or unfreeze functionality.

Use the `.UnfreezeOutput` [1331] member to reverse this command.

FunctionKeyRecallByFunctionKeyId

Christie_SpyderID.FunctionKeyRecallByFunctionKeyId(layerId,functionKeyId)

Example:
Christie_Spyder1.FunctionKeyRecallByFunctionKeyId(1,5)

This member is based on the Spyder command "FKR–Function Key Recall". It recalls a single function key defined in Spyder.
Enter the ID specified within the function ID settings. The example recalls Function Key ID 5, no matter in which register or page it is placed or if it is moved.

There is also the member FunctionKeyRecallByRegisterId [1325].

FunctionKeyRecallByRegisterId

Christie_SpyderID.FunctionKeyRecallByRegisterId(layerId,pageId,registerId)

Example:
Christie_Spyder1.FunctionKeyRecallByRegisterId(1,2,5)

This member is based on the Spyder command "FKR–Function Key Recall". It recalls a single function key defined in Spyder.
Enter the Register ID, meaning its slot within the UI. The example recalls register 5 on page 2, no matter which Function Key has been placed in that spot.

There is also the member FunctionKeyRecallByFunctionKeyId [1324].

IpAddress

Christie_SpyderID.IpAddress

Example:
vstring = Christie_Spyder1.IpAddress

This returns the IP address of the Spyder device as a string. The result could look as follows: 10.169.80.10
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Spyder1.IpAddress`
or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Spyder1.IpAddress)`

In return, you can also assign an IP address to the connected Spyder device. WD automatically connects to it afterwards.
`Christie_Spyder1.IpAddress = "10.169.80.10"`

IsEnabled

Christie_SpyderID.IsEnabled

Example:
vstring = Christie_Spyder1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog as a string or Boolean value. The result is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Spyder1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Spyder1.IsEnabled)`

LoadBackgroundImage

Christie_SpyderID.LoadBackgroundImage(pixelSpaceId,filename)

Example:
Christie_Spyder1.LoadBackgroundImage(3,testpattern)

This member is based on the Spyder command "BLD–Load Still in Background". It loads a file as the background on the current layer.

The argument "filename" specifies the name of the file to load. If the image is not the same size as the pixelspace it is being loaded on, the image is automatically scaled to fit into it.

Use the member LoadBackgroundImageNext 1326 to load it to the next layer.

LoadBackgroundImageNext

Christie_SpyderID.LoadBackgroundImageNext(pixelSpaceId,filename)

Example:
Christie_Spyder1.LoadBackgroundImageNext(3,testpattern)

This member is based on the Spyder command "BLD–Load Still in Background". It loads a file as the background on the current layer.

The argument "filename" specifies the name of the file to load. If the image is not the same size as the pixelspace it is being loaded on, the image is automatically scaled to fit into it.

Use the member LoadBackgroundImage 1325 to load it to the current layer.

LoadLayerImage

Christie_SpyderID.LoadLayerImage(layerId,filename)

Example:
Christie_Spyder1.LoadLayerImage(1,testpattern)

This member is based on the Spyder command "SLD–Load Still on Layer". It loads a file onto a layer.

LoadOutputImage

Christie_SpyderID.LoadOutputImage(outputId,filename)

Example:
Christie_Spyder1.LoadOutputImage(8,testpattern)

This member is based on the Spyder command "LSO–Load Still on Output". It loads an unscaled still image directly onto a Spyder output for the purpose of loading custom test patterns.

The argument "filename" sets the file of the image to load. The filename must exist in the Stills directory on the Spyder server, which can be accessed using FTP or the Spyder Studio software.

MoveAndResizeLayer

Christie_SpyderID.MoveAndResizeLayer(layerId,positionX,positionY,width)

Example:
Christie_Spyder1.MoveAndResizeLayer(1,960,540,960)

The example resizes and places the layer 1 to the absolute position x=960 and y=540 by scaling it to 960px width by keeping its ratio. On an HD screen, the layer is placed into the bottom right quadrant.

This member is based on the Spyder command "LSP–Layer Size and Position Change". It sets the size and horizontal and vertical position of a layer as an absolute position setting. In cases where the size and position are being modified simultaneously for an application, Christie recommends using this command instead of the individual layer size and position commands (KPS and KSZ).

The argument "width" sets the horizontal size, in pixels, of the specified layer.

The member MoveAndResizeLayerRelative [1327] changes the position relatively.

MoveAndResizeLayerRelative

Christie_SpyderID.MoveAndResizeLayerRelative(layerId,positionX,positionY,width)

Example:
Christie_Spyder1.MoveAndResizeLayerRelative(1,-200,500,960)

The example moves and resizes layer 1 relatively from its original position. It results into moving the layer 200 pixels to the left and 500 pixels down. It is also scaled to 960px width by keeping its ratio.

This member is based on the Spyder command "LSP–Layer Size and Position Change". It sets the size and horizontal and vertical position of a layer relative to the layer's current position. In cases where the size and position are being modified simultaneously for an application, Christie recommends using this command instead of the individual layer size and position commands (KPS and KSZ).

The argument "width" sets the horizontal size, in pixels, of the specified layer.

The member MoveAndResizeLayer [1326] applies an absolute position setting.

MoveLayer

Christie_SpyderID.MoveLayer(layerId,positionX,positionY)

Example:
Christie_Spyder1.MoveLayer(1,960,540)

The example places the layer 1 to the absolute position x=960 and y=540.

This member is based on the Spyder command "KPS–Layer Position Change". It sets the horizontal and vertical position of a layer as an absolute position setting. Note: Positions are mapped in pixels, relative to the top-left pixelspace corner associated with the layer.

The member MoveLayerRelative [1327] changes the position relatively.

MoveLayerRelative

Christie_SpyderID.MoveLayerRelative(layerId,positionX,positionY)

Example:
Christie_Spyder1.MoveLayerRelative(1,-200,500)

The example moves layer 1 relatively from its original position. It results into moving the layer 200 pixels to the left and 500 pixels down.

This member is based on the Spyder command "KPS–Layer Position Change". It sets the horizontal and vertical position of a layer relative to the layer's current position. Note: Positions are mapped in pixels, relative to the top-left pixelspace corner associated with the layer.

The member MoveLayer [1327] applies an absolute position setting.

MoveLayerToPixelSpace

Christie_SpyderID.MoveLayerToPixelSpace(layerId,pixelSpaceId,visibility)

Example:
Christie_Spyder1.MoveLayerToPixelSpace(1,3,0)

This member is based on the Spyder command "LAP–Layer Assign pixelspace". It associates a layer with a particular pixelspace.

Set the "visibility" argument to 0 in order to hide the layer(s) and to 1 in order to show the layer(s) at the time this command is applied. Making the layer visible allows additional commands to be sent to configure the layer before it is transitioned onto the screen with the TRN command which is the FadeLayerIn 1324 member.

RecallScriptCueByScriptId

Christie_SpyderID.RecallScriptCueByScriptId(scriptId,cueId)

Example:
Christie_Spyder1.RecallScriptCueByScriptId(13,2)

This member is based on the Spyder command "RSC–Recall Script Cue". It recalls an existing script at a specific cue. To build scripts, use Spyder Studio.

Enter the ID specified within the script ID settings. The example recalls Script ID 13, no matter in which register or page it is placed or if it is moved.
Note: CueIDs are zero based so (13,2) recalls the third cue of script 13.
Tip: Cue 0 often prepares the Cue in Preview, while Cue1 pushes it to the output.

There is also the member RecallScriptCueRegisterId 1328.

RecallScriptCueRegisterId

Christie_SpyderID.RecallScriptCueRegisterId(pageId,registerId,cueId)

Example:
Christie_Spyder1.RecallScriptCueRegisterId(2,5,2)

This member is based on the Spyder command "RSC–Recall Script Cue". It recalls an existing script at a specific cue. To build scripts, use Spyder Studio.

Enter the Register ID, meaning its slot within the UI. The example recalls register 5 on page 2, no matter which Function Key has been placed in that spot.
Note: CueIDs are zero based so (2) recalls the third cue.
Tip: Cue 0 often prepares the Cue in Preview, while Cue1 pushes it to the output.

There is also the member RecallScriptCueByScriptId 1328.

ResizeLayer

Christie_SpyderID.ResizeLayer(layerId,width)

Example:
Christie_Spyder1.ResizeLayer(1,960)

This member is based on the Spyder command "KSZ–Layer Size Change". It sets the horizontal size of the specified layer.

The argument "width" sets the horizontal size, in pixels, of the specified layer. The vertical size adjusts automatically to ensure the layer's aspect ratio.

RestartSpyderServer

Christie_SpyderID.RestartSpyderServer

Example:
Christie_Spyder1.RestartSpyderServer

This member is based on the Spyder command "SDN–Restart Spyder Server". It restarts the Spyder server application remotely.

The member ShutdownSpyderServer 1330 would power it off.

SendCustomCommand

Christie_SpyderID.SendCustomCommand(command)

Example:
Christie_Spyder1.SendCustomCommand("LSP 1 150 300 1920 5")

This allows to send a custom command to the Spyder device which gives access to all supported Serial API commands. Please refer to the advanced manual of your Christie Spyder device in the section "External Control Protocol to learn the external control protocol acronyms of the specific command you want to use as well as the requested arguments. You DO NOT need to include the header or trailing parameters. Just enter the command and its parameters. Note that arguments are delimited with a space character.

The example command sends the relative position +150/+300 and width 1920px to Layer 5.

SetBackgroundTransitionDuration

Christie_SpyderID.SetBackgroundTransitionDuration(duration)

Example:
Christie_Spyder1.SetBackgroundTransitionDuration(10)

This member is based on the Spyder command "BTR–Transition Background". It transitions the background layers across all pixelspaces. Pixelspace backgrounds cannot be transitioned individually.

The "duration" argument determines how long, in frames per second, a background layer takes to transition.

SetLayerAspectRatio

Christie_SpyderID.SetLayerAspectRatio(layerId,aspectRatio)

Example:
Christie_Spyder1.SetLayerAspectRatio(1,1.33)

This member is based on the Spyder command "ARO–Aspect Ratio Offset". It sets the aspect ratio for a layer. It keeps its width and changes the height accordingly.

SetMixerTransitionDuration

Christie_SpyderID.SetMixerTransitionDuration(duration,deviceIndex)

Example:
Christie_Spyder1.SetMixerTransitionDuration(60,1)

This member is based on the Spyder command "DMT–Device Mixer Transition". It sets the automatic transition for a mixer one or more devices. This command is intended for devices configured as mixers.

The "duration" argument sets the transition duration to be applied, in frames per second; "1" forces a cut transition.
The "deviceIndex" argument indicates the target device indexes and is zero-based.

The example results into a 1 second fade when set to 60fps.

SetOutputModeToNormal

Christie_SpyderID.SetOutputModeToNormal(outputId,positionX,positionY)

Example:
Christie_Spyder1.SetOutputModeToNormal(8,0,0)

This member is based on the Spyder command "OCM–Output Configuration Mode". It applies the output mode "Normal" and sets the horizontal and vertical starting position for an output.
The example shows the image on output 8 beginning in the top left corner.

There is also the member SetOutputModeToScaled [1330].

SetOutputModeToScaled

Christie_SpyderID.SetOutputModeToScaled(outputId,pixelSpaceId)

Example:
Christie_Spyder1.SetOutputModeToScaled(8,3)

This member is based on the Spyder command "OCM–Output Configuration Mode". It applies the output mode "Scaled" and sets the program pixelspace to focus Scaled on. If a preview ID is supplied, its associated program pixelspace is applied.

There is also the member SetOutputModeToNormal [1330].

ShutdownSpyderServer

Christie_SpyderID.ShutdownSpyderServer

Example:
Christie_Spyder1.ShutdownSpyderServer

This member is based on the Spyder command "SDN–Restart Spyder Server". It powers off the Spyder server application remotely.

The member RestartSpyderServer [1329] would restart it.

SourceApplyToLayer

Christie_SpyderID.SourceApplyToLayer(layerId,sourceName)

Example:
Christie_Spyder1.SourceApplyToLayer(1,PandorasBox)

This member is based on the Spyder command "SRA–Source Apply". It applies an existing source to the specified layer. A source is defined as an input configuration and a router input. To create sources and define connected routers, use Spyder Studio.

The example assigns the input "PandorasBox" to layer 1.

SwapLayers

Christie_SpyderID.SwapLayers(firstLayerId,secondLayerId)

Example:
Christie_Spyder1.SwapLayers(1,2)

This member is based on the Spyder command "SWA–Swap Layers". It swaps all properties between two specified layers, including PixelSpace, content, and keyframe. Note: both layers must be visible on screen when the command is executed

UnfreezeLayer

Christie_SpyderID.UnfreezeLayer(layerId)

Example:
Christie_Spyder1.UnfreezeLayer(1)

This member is based on the Spyder command "FRZ–Freeze Layer". It unfreezes a layer.

Use the `.FreezeLayer` [1324] member to reverse this command.

UnfreezeOutput

Christie_SpyderID.UnfreezeOutput(outputId)

Example:
Christie_Spyder1.UnfreezeOutput(8)

This member is based on the Spyder command "OFZ–Freeze Output". It unfreezes one or more outputs. This command is only compatible with universal outputs. DX4 outputs do not support individual freeze or unfreeze functionality.

Use the `.FreezeOutput` [1324] member to reverse this command.

7.6.9.3 Terra

The Terra Device in the Configuration dialog allows to easily monitor and remote control different functions of a Christie Terra controller. Note that the device is only available in the licensed Widget Designer edition, not the Free version.

Adding a New Terra Device

To add a Terra device, open the Devices menu and select "Christie > Terra > Create Terra". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side you see several options:

The **Name** is the unique identifier for this Terra object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your Terra device is a filled blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A filled gray circle indicates a disabled device.
In case, you cannot connect to the Terra controller, make sure that the remote option is turned on in the Terra manager. Open the "Global" page and look under "External Control" for the option "JSON API" and enable it.

Now, please enter the **IP address** of the Terra controller. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

You can close the dialog now. The newly created device will also be added to the Devices menu > Christie > Terra and can be opened from here or with Devices menu > Configuration.

Using the Device in Regular Scripting

After creating a device, you can access its commands and parameters easily via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Christie_Terra1"), Script Assist will offer you a list of all members [1335].
Some of them are commands to perform actions on the device, e.g. applying a layout:

```
Christie_Terra1.ApplyLayout("FullView")
```

You can also retrieve specific values via scripting in the same way as from other widgets (e.g. a Fader value):

```
vlist = Christie_Terra1.LayoutNames
```

This member returns all layout names as a list that can be displayed in the Debug Logger [812] or assigned to a variable [1900], DropDown List [868], etc.

The Terra tutorial [1345] shows how to build a small interface to interact with a Terra system and uses the most important commands and Event Listeners.

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and Terra Events [1333] for an overview (with examples and description) over the events raised by a Terra device.

Using the Device as a Widget

There is also a Terra Display Array [863] widget available which shows the display arrangement of a Terra device.

7.6.9.3.1 Terra Events

This chapter gives an overview of the events that are raised by a Terra device and that can be used in the Event Listener [1353].
Please read the chapters "Terra" [1332] and "Event Listener" [1353] if you like to know how to add and use a Terra device and Event Listener. There is also a "Terra Tutorial" [1345] available which explains how to use Event Listeners. The available Terra members that can be used in our scripting language are explained in the chapter "Terra Device Members" [1335].

IpAddress

Example:
Label1.Text = ipAddress

This event is raised whenever the IP address of the connected Terra system is changed from anywhere.
The event returns one parameter to WD which is a string with the name "ipAddress" and holds the current IP address.
If you select this event and copy the example into the scripting field of the Event Listener, it will write the IP address into the Label [888] with ID 1 whenever the IP address is changed.

LayoutNames

Example:
DropDownList1.SetItemsFromArray("layoutNames")

This event is raised whenever a layout is changed (i.e. edited, created or deleted) in the connected Terra system.
The event returns one parameter to WD which is a list with the name "layoutNames" and holds all layout names.
If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List [868] with ID 1 with all current layout names whenever a layout is edited, created or deleted.
You can also directly access the layout names via the member Christie_TerraID.LayoutNames [1341] within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.
The member Christie_TerraID.DisplayArrayLayoutName [1339] returns the currently applied layout per display array.

DeviceNames

Example:
DropDownList1.SetItemsFromArray("deviceNames")

This event is raised whenever a device (i.e. transmitter or receiver) is changed (i.e. edited, created or deleted) in the connected Terra system.
The event returns one parameter to WD which is a list with the name "deviceNames" and holds all device names.
If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List [868] with ID 1 with all current device names whenever a device is edited, created or deleted.
You can also directly access the device names via the member Christie_TerraID.DeviceNames [1338] within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.

ReceiverNames

Example:
DropDownList1.SetItemsFromArray("receiverNames")

This event is raised whenever a receiver is changed (i.e. edited, created or deleted) in the connected Terra system.
The event returns one parameter to WD which is a list with the name "receiverNames" and holds all receiver names.

If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List 868 with ID 1 with all current receiver names whenever a receiver is edited, created or deleted. You can also directly access the receiver names via the member Christie_TerraID.ReceiverNames 1342 within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.

TransmitterNames

Example:
DropDownList1.SetItemsFromArray("transmitterNames")

This event is raised whenever a transmitter is changed (i.e. edited, created or deleted) in the connected Terra system.
The event returns one parameter to WD which is a list with the name "transmitterNames" and holds all transmitter names.
If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List 868 with ID 1 with all current transmitter names whenever a transmitter is edited, created or deleted.
You can also directly access the transmitter names via the member Christie_TerraID.TransmitterNames 1344 within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.

DisplayArrayNames

Example:
DropDownList1.SetItemsFromArray("displayArrayNames")

This event is raised whenever a display array is changed (i.e. edited, created or deleted) in the connected Terra system.
The event returns one parameter to WD which is a list with the name "displayArrayNames" and holds all names of the display arrays.
If you select this event and copy the example into the scripting field of the Event Listener, it will populate the Drop Down List 868 with ID 1 with all current display array names whenever a display array is edited, created or deleted.
You can also directly access the display array names via the member Christie_TerraID.DisplayArrayNames 1339 within the regular scripting. The Event Listener just keeps your system automatically updated as long as WD is turned on.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa.
The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label 888 with ID 1 whenever the device dis-/connects.

SystemShutDown

Example:
Label1.Text = "YourMessage"

This event is raised whenever the connected Terra system receives a "shut down" command from anywhere.
If you select this event and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label 888 with ID 1 whenever Terra is shut down.

SystemRestart

Example:
Label1.Text = "YourMessage"

This event is raised whenever the connected Terra system receives a "restart" command from anywhere.
If you select this event and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever Terra is restarted.

SystemUpdating

Example:
Label1.Text = "YourMessage"

This event is raised whenever the connected Terra system is updating its firmware.
If you select this event and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever Terra is updating.

LayoutApplied

Example:
Label1.Text = "Current layout: " + layout

This event is raised whenever a layout is applied in the connected Terra system.
The event returns two parameters to WD. The first is a string with the name "layout" and holds the new layout name. The second is a string with the name "displayArray" and holds the name of the display array the layout was applied to.
If you select this event and copy the example into the scripting field of the Event Listener, it will write into the Label [888] with ID 1 "Current layout: " and the layout name behind it, so for example "Current layout: PIP".

7.6.9.3.2 Terra Members

This chapter gives an overview of the members available for the Terra device.
Please read the chapter "Terra" [1332] if you like to know how to add and use a Terra device. There is also a "Terra Tutorial" [1345] available which explains the most important steps.
The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

ApplyLayout

Christie_TerraID.ApplyLayout(layoutName)

Example:
Christie_Terra1.ApplyLayout("PIP")

This applies the layout named "PIP" to the display array it was created for.

ApplyVideoSource

Christie_TerraID.ApplyVideoSource(displayArrayName,layoutName,viewIndex,transmitterName)

Example:
Christie_Terra1.ApplyVideoSource("4LCDs","PIP",2,"Pandora_Dual_1")

This applies the video source of the Transmitter named "Pandora_Dual_1" to the window with the view index 2 within the layout named "PIP" on the display array named "4LCDs". Bare in mind that this change of the layout is not saved automatically. To save the layout, use the member "SaveLayout" [1342].

ClearDisplayArray

Christie_TerraID.ClearDisplayArray(displayArrayName)

Example:
Christie_Terra1.ClearDisplayArray("4LCDs")

This clears the current layout and all sources from the display array named "4LCDs".

DeleteLayout

Christie_TerraID.DeleteLayout(layoutName)

Example:
Christie_Terra1.DeleteLayout("PIP")

This removes the layout named "PIP" from the Terra Manager so that it cannot be applied anymore.

DeviceInfo

Christie_TerraID.DeviceInfo(deviceName)

Example:
Christie_Terra1.DeviceInfo("xy").Fps
Christie_Terra1.DeviceInfo("xy").Height
etc.

Use this member with its further members, e.g.: Fps,Height,Id,...

You can either use:
- local variables and the Debug Logger [812]: `var temp = Christie_Terra1.DeviceInfo("xy")` followed by `DebugMessage(temp.Fps)`
or directly `DebugMessage(Christie_Terra1.DeviceInfo("xy").Fps)`
- existing global variables [1900]: `vint = Christie_Terra1.DeviceInfo("xy").Fps`
- or Widgets like labels [888]: `Label1.Text = Christie_Terra1.DeviceInfo("xy").Fps`

DeviceInfo().Id

Christie_TerraID.DeviceInfo(deviceName).Id

Example:
vstring = Christie_Terra1.DeviceInfo("Top_Left").Id

This member returns the ID of the receiver (or transmitter) named "Top_Left". Which, by the way, is its MAC address. The result could look as follows: 000948dd0118
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Top_Left").Id`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Top_Left").Id)`

DeviceInfo().Name

Christie_TerraID.DeviceInfo(deviceName).Name

Example:
vstring = Christie_Terra1.DeviceInfo("Top_Left").Name

This member returns the name of the receiver (or transmitter) named "Top_Left". The result could look as follows: Top_Left (or 000948DD0118 if you did not change the default name)
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Top_Left").Name`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Top_Left").Name)`

DeviceInfo().Width

Christie_TerraID.DeviceInfo(deviceName).Width

Example:
vstring = Christie_Terra1.DeviceInfo("Top_Left").Width

This member returns the resolution width of the receiver (or transmitter) named "Top_Left". The result could look as follows: 1920
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Top_Left").Width`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Top_Left").Width)`

DeviceInfo().Height

Christie_TerraID.DeviceInfo(deviceName).Height

Example:
vstring = Christie_Terra1.DeviceInfo("Top_Left").Height

This member returns the resolution height of the receiver (or transmitter) named "Top_Left". The result could look as follows: 1080
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Top_Left").Height`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Top_Left").Height)`

DeviceInfo().Fps

Christie_TerraID.DeviceInfo(deviceName).Fps

Example:
vstring = Christie_Terra1.DeviceInfo("Top_Left").Fps

This member returns the Framerate of the receiver (or transmitter) named "Top_Left". The result could look as follows: 60
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Top_Left").Fps`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Top_Left").Fps)`

DeviceInfo().IsReceiver

Christie_TerraID.DeviceInfo(deviceName).IsReceiver

Example:
vstring = Christie_Terra1.DeviceInfo("Top_Left").IsReceiver

This member returns the answer whether the receiver (or transmitter) named "Top_Left" is a receiver. The result is "true" or "false".
The first example shows how to use it with an existing (global) string variable [1900] named "vstring" (you can of course also use a Boolean variable).

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Top_Left").IsReceiver`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Top_Left").IsReceiver)`

DeviceInfo().IsTransmitter

Christie_TerraID.DeviceInfo(deviceName).IsTransmitter

Example:
vstring = Christie_Terra1.DeviceInfo("Top_Left").IsTransmitter

This member returns the answer whether the receiver (or transmitter) named "Top_Left" is a transmitter. The result is "true" or "false".
The first example shows how to use it with an existing (global) string variable [1900] named "vstring" (you can of course also use a Boolean variable).

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Top_Left").IsTransmitter`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Top_Left").IsTransmitter)`

DeviceInfo().InputConnector

Christie_TerraID.DeviceInfo(deviceName).InputConnector

Example:
vstring = Christie_Terra1.DeviceInfo("Pandora_Dual_1").InputConnector

This member returns the connected inputs of the transmitter (!) named "Pandora_Dual_1". The result could look as follows: HDMI (or "DisplayPort" or in case of a receiver "none")
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DeviceInfo("Pandora_Dual_1").InputConnector`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DeviceInfo("Pandora_Dual_1").InputConnector)`

If you like to change the input connector, use the command
`Christie_Terra1.SwitchInputConnector("Pandora_Dual_1","HDMI")`

DeviceNames

Christie_TerraID.DeviceNames

Example:
vlist = Christie_Terra1.DeviceNames

This member returns all device names (i.e. running transmitters and receivers) as a list. The result could look as follows: [Pandora_Dual_1, Pandora_Dual_2, Top_Left, Top_Right, Bottom_Left, Bottom_Right] (or [000948AA006A, 000948AA0698, 000948DD0118, 000948DD0227, 000948DD0336, 000948DD0445] if you did not change the default names).
The first example shows how to use it with an existing (global) list variable [1900] named "vlist".

You could also write it into a Widget, e.g. a DropDown List [868] via the script and a local variable [1903] "list":
```
var list = Christie_Terra1.DeviceNames
DropDownList1.SetItemsFromArray("list")
```

or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Terra1.DeviceNames)`

Disable

Christie_TerraID.Disable

Example:
Christie_Terra1.Disable

This disables the Terra device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1340] member to reverse this command.

DisplayArrayLayoutName

Christie_TerraID.DisplayArrayLayoutName(displayArrayName)

Example:
vstring = Christie_Terra1.DisplayArrayLayoutName("4LCDs")

This member returns the name of the layout currently applied to the display array named "4LCDs". The result could look as follows: PIP
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DisplayArrayLayoutName("4LCDs")`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DisplayArrayLayoutName("4LCDs"))`

DisplayArrayNames

Christie_TerraID.DisplayArrayNames

Example:
vlist = Christie_Terra1.DisplayArrayNames

This member returns all display array names as a list. The result could look as follows: [4LCDs,Projectorblend,MicroTiles]
The first example shows how to use it with an existing (global) list variable [1900] named "vlist".

You could also write it into a Widget, e.g. a DropDown List [868] via the script and a local variable [1903] "list":
```
var list = Christie_Terra1.DisplayArrayNames
DropDownList1.SetItemsFromArray("list")
```

or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Terra1.DisplayArrayNames)`

DisplayArrayWindowCount

Christie_TerraID.DisplayArrayWindowCount(displayArrayName)

Example:
vint = Christie_Terra1.DisplayArrayWindowCount("4LCDs")

This member returns the current window count for the display array named "4LCDs" as an integer number. Note that all windows have a view index which always starts with [1] and is numbered through depending on the window's position, so the next one has [2]. The result could look as follows: 4

The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DisplayArrayWindowCount("4LCDs")`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DisplayArrayWindowCount("4LCDs"))`

DisplayArrayWindowSourceName

Christie_TerraID.DisplayArrayWindowSourceName(displayArrayName,viewIndex)

Example:
vstring = Christie_Terra1.DisplayArrayWindowSourceName("4LCDs",1)

This member returns the name of the source (= transmitter) for the window with view index [1] within the display array named "4LCDs". The result could look as follows: Pandora_Dual_1 (or 000948AA006A if you did not change the default name)
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.DisplayArrayWindowSourceName("4LCDs",1)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Christie_Terra1.DisplayArrayWindowSourceName("4LCDs",1))`

Enable

Christie_TerraID.Enable

Example:
Christie_Terra1.Enable

This enables the Terra device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1339] member to reverse this command.

IpAddress

Christie_TerraID.IpAddress

Example:
vstring = Christie_Terra1.IpAddress

This returns the IP address of the Terra device as a string. The result could look as follows: 10.169.80.10
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.IpAddress`
or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Terra1.IpAddress)`

In return, you can also assign an IP address to the connected Terra controller. WD automatically connects to it afterwards.
`Christie_Terra1.IpAddress = "10.169.80.10"`

IsConnected

Christie_TerraID.IsConnected

Example:
vstring = Christie_Terra1.IsConnected

This member returns the connection status for the Terra controller as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.IsConnected`
or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Terra1.IsConnected)`

IsEnabled

Christie_TerraID.IsEnabled

Example:
vstring = Christie_Terra1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog as a string or Boolean value. The result is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Christie_Terra1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Terra1.IsEnabled)`

LayoutNames

Christie_TerraID.LayoutNames

Example:
vlist = Christie_Terra1.LayoutNames

This member returns all layout names as a list. The result could look as follows: [PIP,Fullscreen]
The first example shows how to use it with an existing (global) list variable [1900] named "vlist".

You could also write it into a Widget, e.g. a DropDown List [868] via the script and a local variable [1903] "list":

```
var list = Christie_Terra1.LayoutNames
DropDownList1.SetItemsFromArray("list")
```

or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Terra1.LayoutNames)`

LayoutNamesByDisplayArray

Christie_TerraID.LayoutNamesByDisplayArray(displayArrayName)

Example:
vlist = Christie_Terra1.LayoutNamesByDisplayArray("4LCDs")

This member returns all layout names as a list that are saved with the display array named "4LCDs". The result could look as follows: [PIP,Fullscreen]
The first example shows how to use it with an existing (global) list variable [1900] named "vlist".

You could also write it into a Widget, e.g. a DropDown List [868] via the script and a local variable [1903] "list":
```
var list = Christie_Terra1.LayoutNamesByDisplayArray("4LCDs")
DropDownList1.SetItemsFromArray("list")
```

or simply display it in the Debug Logger [812]:
```
DebugMessage(Christie_Terra1.LayoutNamesByDisplayArray("4LCDs"))
```

ReceiverNames

Christie_TerraID.ReceiverNames

Example:
vlist = Christie_Terra1.ReceiverNames

This member returns the names of all running receiver devices as a list. The result could look as follows: [Top_Left, Top_Right, Bottom_Left, Bottom_Right] (or [000948DD0118,000948DD0227,000948DD0336,000948DD0445] if you did not change the default names). The first example shows how to use it with an existing (global) list variable [1900] named "vlist".

You could also write it into a Widget, e.g. a DropDown List [868] via the script and a local variable [1903] "list":
```
var list = Christie_Terra1.ReceiverNames
DropDownList1.SetItemsFromArray("list")
```

or simply display it in the Debug Logger [812]: `DebugMessage(Christie_Terra1.ReceiverNames)`

RestartSystem

Christie_TerraID.RestartSystem

Example:
Christie_Terra1.RestartSystem

This shuts down the Terra controller and restarts it.

SaveLayout

Christie_TerraID.SaveLayout(displayArrayName,newLayoutName,overwrite)

Example:
Christie_Terra1.SaveLayout("4LCDs","Fullscreen","true")

This saves a (new) layout with the name "Fullscreen" for the display array named "4LCDs". As the "overwrite" parameter is set to "true" the current state of the layout will be overwritten. Set it to "false" in case you never want to overwrite existing layouts but only save completely new ones.

SendRS232

Christie_TerraID.SendRS232(deviceName,targetGroup,port,data)

Example:
Christie_Terra1.SendRS232("Top_Left","NONE",0,"foo")

This creates the data package "foo" and sends it to the receiver named "Top_Left" to output it to a connected device like a monitor or projector. If you set the parameter "targetGroup" to another of the suggested values (ALL, ALL_RX, ALL_TX)you can access more than one device.

StopDevice

Christie_TerraID.StopDevice(deviceName)

Example:
Christie_Terra1.StopDevice("Pandora_Dual_1")

This tells the transmitter named "Pandora_Dual_1" to stop sending any video streams. Use the SwitchVideo[1344] member to start sending a stream again.

SwitchAudio

Christie_TerraID.SwitchAudio(fromTransmitterName,toReceiverName,type)

Example:
Christie_Terra1.SwitchAudio("Pandora_Dual_1","Top_Left","HDMI_AUDIO")

This switches the audio source from the transmitter named "Pandora_Dual_1" to the receiver named "Top_Left" With the "type" parameter you can choose between the analog or HDMI audio source. Note that this does not show up in the Terra Manager as it directly routes the signal and has nothing to do with Layouts or Display Arrays.

SwitchInputConnector

Christie_TerraID.SwitchInputConnector(transmitterName,connector)

Example:
Christie_Terra1.SwitchInputConnector("Pandora_Dual_1","HDMI")

This chooses the HDMI input to be the source for the transmitter named "Pandora_Dual_1" e.g. if you have two inputs connected to one transmitter. You can also switch it to the input "DisplayPort".

SwitchRS232

Christie_TerraID.SwitchRS232(fromTransmitterName,toReceiverName)

Example:
Christie_Terra1.SwitchRS232("Pandora_Dual_1",Top_Left)´

This switches the RS232 source from the transmitter named "Pandora_Dual_1" to the receiver named "Top_Left". Note that this does not show up in the Terra Manager as it directly routes the signal and has nothing to do with Layouts or Display Arrays.

SwitchUSB

Christie_TerraID.SwitchUSB(fromReceiverName,toTransmitterName)

Example:
Christie_Terra1.SwitchUSB("Top_Left","Pandora_Dual_1")

This switches the USB source from the receiver (!) named "Top_Left" to the transmitter (!) named "Pandora_Dual_1". Note that this does not show up in the Terra Manager as it directly routes the signal and has nothing to do with Layouts or Display Arrays.

SwitchVideo

Christie_TerraID.SwitchVideo(fromTransmitterName,toReceiverName)

Example:
Christie_Terra1.SwitchVideo("Pandora_Dual_1","Top_Left")

This switches the video source from the transmitter named "Pandora_Dual_1" to the receiver named "Top_Left". Note that this does not show up in the Terra Manager as it directly routes the signal and has nothing to do with Layouts or Display Arrays. Use the member "SwitchVideoWithEmbeddedAudio 1344" if you like to switch video together with its embedded audio information. Use the member "StopDevice 1343" to stop sending any stream.

SwitchVideoWithEmbeddedAudio

Christie_TerraID.SwitchVideoWithEmbeddedAudio(fromTransmitterName,toReceiverName)

Example:
Christie_Terra1.SwitchVideoWithEmbeddedAudio(Pandora_Dual_1,"Top_Left")

This switches the video (with embedded audio) source from the transmitter named "Pandora_Dual_1" to the receiver named "Top_Left". Note that this does not show up in the Terra Manager as it directly routes the signal and has nothing to do with Layouts or Display Arrays.

TransmitterNames

Christie_TerraID.TransmitterNames

Example:
vlist = Christie_Terra1.TransmitterNames

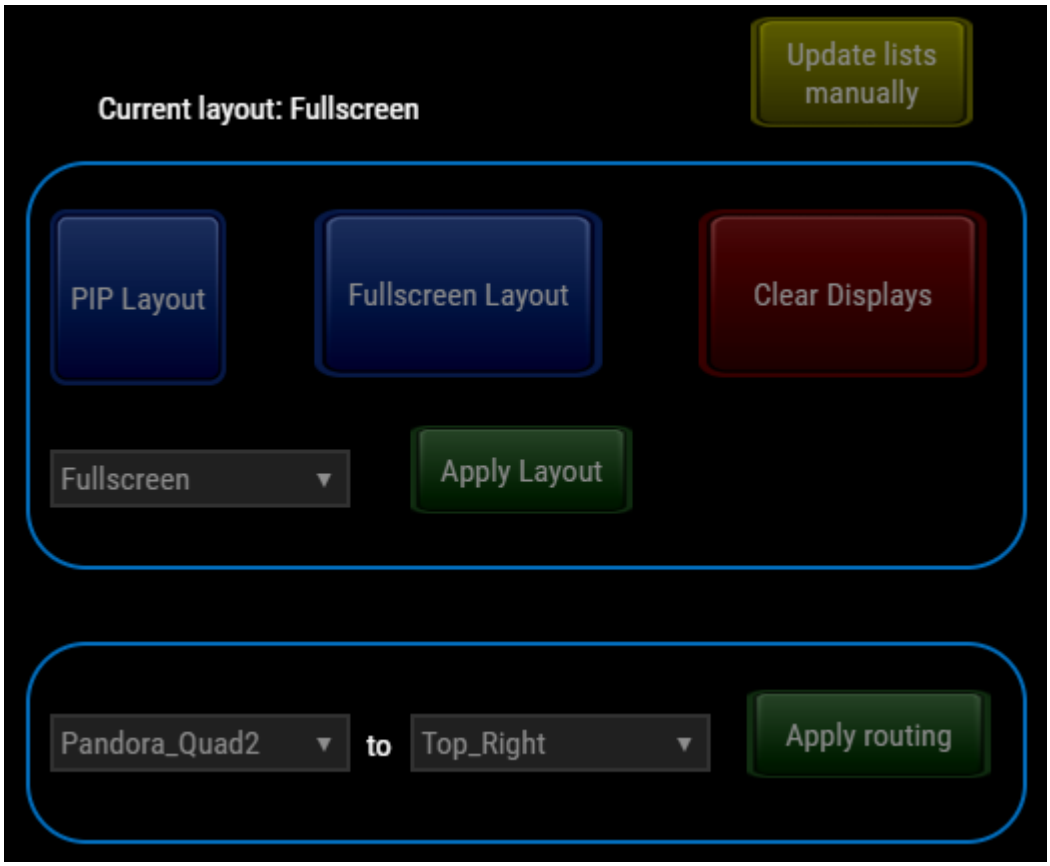
This member returns the names of all running transmitter devices as a list. The result could look as follows: [Pandora_Dual_1, Pandora_Dual_2] (or [000948AA006A,000948AA0698] if you did not change the default names).
The first example shows how to use it with an existing (global) list variable 1900 named "vlist".

You could also write it into a Widget, e.g. a DropDown List 868 via the script and a local variable 1903 "list":

```
var list = Christie_Terra1.TransmitterNames
DropDownList1.SetItemsFromArray("list")
```

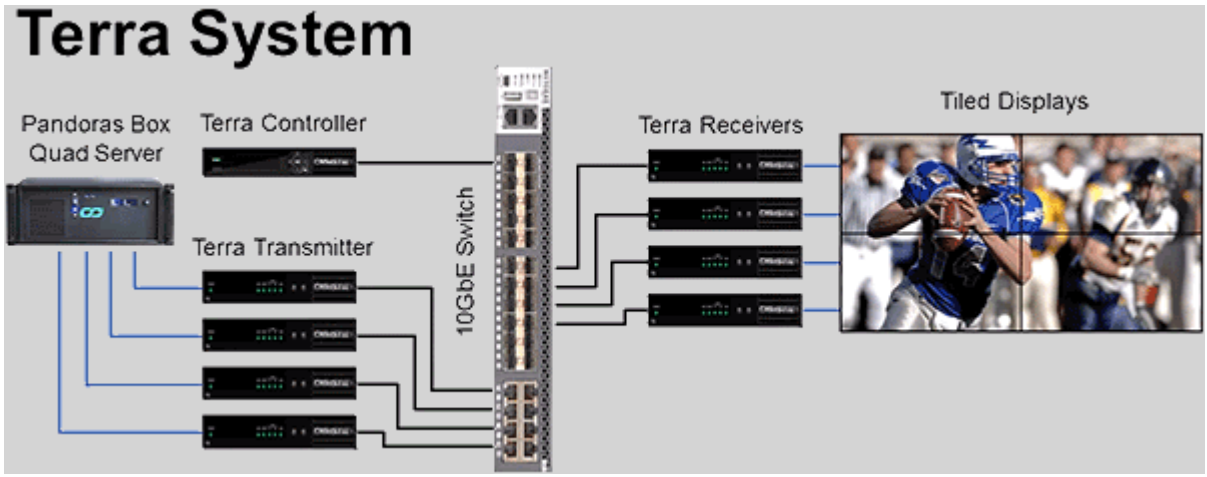
or simply display it in the Debug Logger 812: `DebugMessage(Christie_Terra1.TransmitterNames)`

7.6.9.3.3 Terra Tutorial



This tutorial shows you how to build a simple interface in Widget Designer that includes the most important commands for a Terra system so that everybody can control it intuitively, even an end customer without technical knowledge.

First we will create and configure the Terra device in Widget Designer. Then we build buttons to apply and clear layouts and buttons to switch single sources from A to B. Afterwards we will do a bit more complex programming to react on system events and update our interface accordingly.



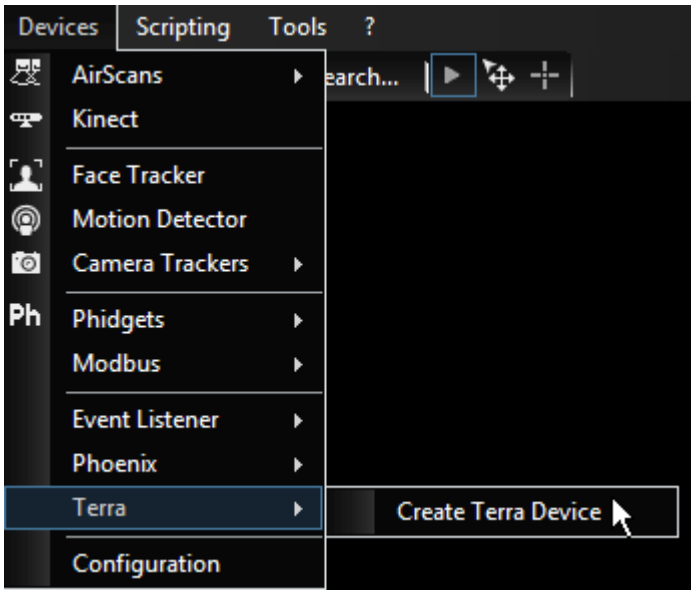
We assume that the Terra system is set up already and is running. This example is based on a very simple Terra network with a Terra controller, four Terra transmitters (which are connected to a Pandoras Box Server with four outputs) and four Terra receivers that are connected to LCD displays in a 2x2 matrix. With the Terra manager, one display array was created and two layouts were programmed. But of course, you can adopt this tutorial to any Terra system.

There are also two video tutorials available this tutorial is based on:
- Widget Designer control for Terra - Commands and examples: https://youtu.be/0k-KsdsKKkU
- Widget Designer control for Terra - EventListener and Automation: https://youtu.be/N0zNmB1AK6A

Creating a Terra Device

To create a Terra device, go to the top menu and choose "Devices > Christie > Terra > Create Terra"

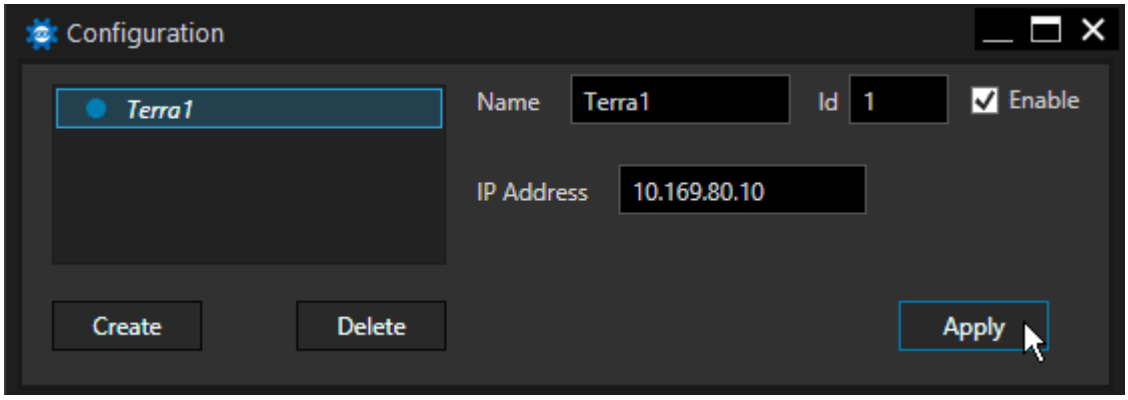
This opens the Configuration dialog.



Alternatively, you can click the -"+" button in the Configuration dialog and choose "Christie > Terra".

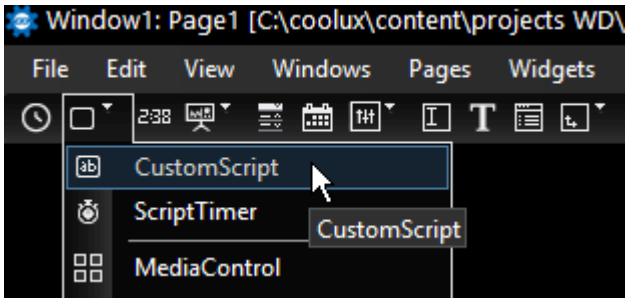
Configuring a Terra Device

To configure a Terra device, simply enter the IP address of the Terra controller in the Configuration dialog and hit the "Apply" button or press [Ctrl + Enter]. (In case the dialog is not open, go the top menu and choose "Devices > Christie > Terra > Christie_Terra1")
On the left you should see that the icon in front of your Terra device is a filled out blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A filled gray circle indicates a disabled device. In our tutorial we will work with a Terra device with the default name "Christie_Terra1". You can change it to a more descriptive name if you like.
You can close the dialog now.



Creating an ApplyLayout Button

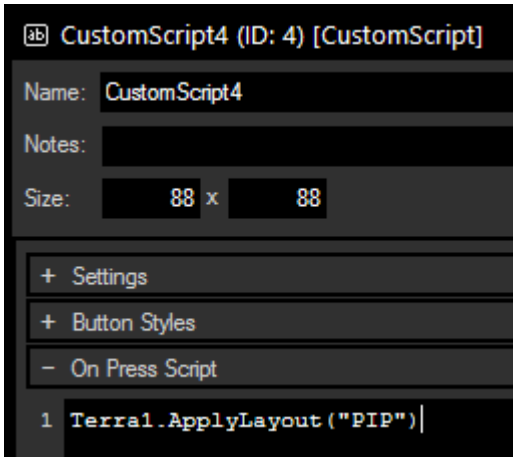
To create a button that calls the command to apply a certain layout in Terra we need a Widget control called "Custom Script button". Click the second icon in the icon bar and choose "CustomScript". Or choose "CustomScript" from the top menu "Widgets".



The cursor changes to a cross-hair icon, indicating the Create mode. Position the mouse cursor and click once to create a Custom Script button. Further clicks would create further buttons but we will continue with one for the moment.

Right-click the button and choose the first entry "CustomScript Properties" which opens a new dialog.
Open the section "On Press Script" and enter the following command. It will be executed as soon as the button is pressed and applies the layout named "PIP" to the display array the layout was saved for in the Terra Manager. Bare in mind, that any WD command is case insensitive, but the layout name must accord exactly to your Terra system.

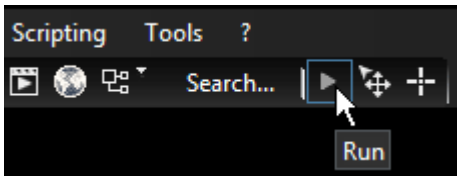
```
Christie_Terra1.ApplyLayout("PIP")
```



You will notice that the Script Assistant suggest entries whilst typing. If you like, you can select something by either clicking on it, or using the [arrow keys] and the [Enter] or [Tab] key.
The chapter "Terra Members" [1335] explains all Terra commands.

Optionally, open the section "Button Style" and enter a descriptive button name in the field "Label", e.g. "PIP Layout".
Optionally, open the section "Button Style" and click the "Tint" color field from which you can choose a new color for the CustomScript button.
Optionally, use the "Size" (width and height) fields (third row in the Properties dialog) to resize the CustomScript button. Alternatively, you can drag the bottom right corner of the button (with the three diagonal, blue lines). Dragging another area of the CustomScript button, will move the button itself.
The chapter "Custom Script Button [822]" explains all properties.

Click "OK" to close the Properties dialog. To test the button we need to enter the Run mode. Either, press [F8], [Alt + R] or click the third button from the right in the icon bar. The other modes are called "Edit/Move" (to resize or move the button; [F9] or [Alt + M]) and "Create" (when choosing something from the Create menu as we have done before).



If you like to apply other layouts, you could either work with more buttons or look at the solution further down that uses a drop-down list [1349].
To create more buttons, you could either redo the above steps, or copy the existing button. Enter the Edit/Move mode [F9], select the "PIP Layout" button and press [Ctrl + C] to copy it. Then move your mouse to the position of the top left corner of the new button and press [Ctrl + V] to paste it there. To open the properties, you can also double-click the button (when being in the Move/Edit mode). Now, change the "On Press Script" to call another layout and do the according with the button label in the section "Button Style" if you followed the optional step above and named the button.

Creating a ClearDisplay Button

In this step we will create a button to clear the display array (i.e. remove any applied layout). First, create a CustomScript button as explained above [1346].

Open the section "On Press Script" and enter the following command. It will be executed as soon as the button is pressed and will clear the current layout and all sources from the display array named "4LCDs".

```
Christie_Terra1.ClearDisplayArray("4LCDs")
```

Creating an XY-Switch Button

In this step we will create a button to switch an input to an output (i.e. route a signal directly). First, create a CustomScript button as explained above [1346].

Open the section "On Press Script" and enter the following command. It will be executed as soon as the button is pressed and will switch the video source from the transmitter named "Pandora_Quad_1" to the receiver named "Top_Left". Note that this does not show up in the Terra Manager as it directly routes the signal and has nothing to do with Layouts or Display Arrays. Use the member "SwitchVideoWithEmbeddedAudio [1344]" if you like to switch video together with audio.

```
Christie_Terra1.SwitchVideo("Pandora_Quad_1","Top_Left")
```

Creating an ApplyVideoSource Button

In this step we will create a button to apply a certain video source within a given layout. First, create a CustomScript button as explained above [1346].

Open the section "On Press Script" and enter the following command. It will be executed as soon as the button is pressed and will apply the video source of the Transmitter named "Pandora_Quad_1" to the window with the view index 2 within the layout named "PIP" on the display array named "4LCDs".

```
Christie_Terra1.ApplyVideoSource("4LCDs","PIP",2,"Pandora_Quad_1")
```

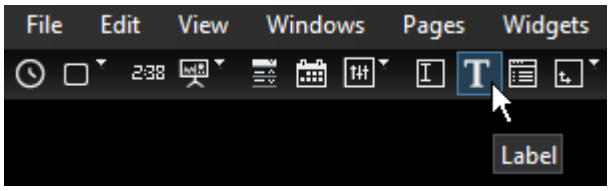
Bare in mind that this change of the layout is not saved automatically. If you like to save it, we need to enter a second line of code. This command saves the layout with the name "PIP" for the display array named "4LCDs" and overwrites the current layout setting.

```
Christie_Terra1.SaveLayout("4LCDs","PIP","true")
```

Creating an Event Listener to Update a Label

In this step we will create for the first time an Event Listener, which gives a trigger as soon as a chosen event happens in the system. In this case, we want to display the current Layout name in a Label and update the text whenever the layout changes.

To create a Label, click the "T"-icon in the icon bar. Or choose "Label" from the top menu "Widgets".



Again, the cursor changes to a cross-hair icon, indicating the Create mode. Position the mouse cursor and click once to create a Label. Enter the run mode [F8].

To create an Event Listener, go to the top menu and choose "Devices > Event Listener > Create Event Listener". This opens the Configuration dialog where the Event Listener is added to the Terra device.
Alternatively, choose "Devices > Configuration" and hit the "+" button to choose "Event Listener".

Now, open the "Device" drop-down list and choose "Christie_Terra1", then the "Event" list and choose "LayoutApplied". This triggers when a layout is applied from anywhere in the Terra system. The event returns two parameters to WD. The first is a string with the name "layout" and holds the new layout name. The second is a string with the name "displayArray" and holds the name of the display array the layout was applied to.

Next, we add the command that is triggered for that event. Enter the following command into the scripting field to write the current layout name into our Label:

```
Label1.Text = layout
```

Press the "Apply" button to save the changes we have done so far. To test the Event Listener, simply press one of the buttons that applies a layout. The label should display the layout name. The nice thing is that the Label will always display the current layout name even if it was not programmed to any WD button or if the layout was renamed using the Terra manager.

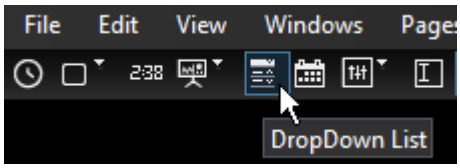
Optionally, add some text in front of the layout name, so that another person knows what it is. To do so, change the Event Listener's script to: `Label1.Text = "Current layout: " + layout`
Optionally, give the Event Listener a descriptive name, e.g. "CurrentLayout". As we will add further Event Listeners, this will help to keep track of what they are doing.

The chapter "Terra Events" [1333] explains all Terra events that can be monitored.

Creating a Layout Selection with a DropDown List

In this step we will create a DropDown List widget that lists all current layout names. A button applies the layout that is chosen in the DropDown List. An Event Listener keeps it up to date and populates it with all new names whenever a the layout is changed, created or deleted in the Terra system.

To create a DropDown List, click the fifth icon in the icon bar. Or choose "DropDown List" from the top menu "Widgets".



Again, the cursor changes to a cross-hair icon, indicating the Create mode. Position the mouse cursor and click once to create a DropDown List. Enter the run mode [F8].

Create a second Event Listener, as explained above [1348]. Optionally, give the Event Listener a descriptive name, e.g. "LayoutsUpdated".
Choose the event "LayoutNames" which gives us a trigger when a layout is changed, created or deleted. The event returns one parameter to WD which is a list with the name "value" and holds all layout names.

To populate our DropDown List with those layout names, enter the following command into the scripting field.

```
DropDownList1.SetItemsFromArray("value")
```

Press the "Apply" button to save the changes we have done so far. To test the Event Listener, create a new layout in the Terra Manager or make use of the command Christie_Terra1.SaveLayout [1342] (e.g. `Christie_Terra1.SaveLayout("4LCDs","testLayout","true")`). To delete "testLayout" execute the command `Christie_Terra1.DeleteLayout("testLayout")`.
By the way, you can execute commands by typing them into any scripting field, highlight it and choose "Test Selected Lines" from the right-click menu, or press [Ctrl + Shift + T]. If something goes wrong with the command, a Debug Logger [812] opens and gives you feedback.

If you now click on the DropDown List, you will see a list of all your layouts. Please select one.
If you are interested in changing the DropDown List, head to the chapter "DropDown List" [868].

In a final step, we create a button to apply this selection. Create a CustomScript button as explained in the beginning [1346] (or copy the "PIP Layout" button) and insert the command:

```
Christie_Terra1.ApplyLayout(Dropdownlist1.Text)
```

Optionally, build a CustomScript button that populates the DropDown List with the layout names "manually", i.e. without an Event Listener. This might be of interest when the WD project was not running at the time when layout changes were done. The fastest script is a two- liner that uses a local variable [1900]:

```
var list = Christie_Terra1.LayoutNames
DropDownList1.SetItemsFromArray("list")
```

Creating a flexible XY-switch

In this step we will create two DropDown Lists where one shows all Terra transmitter names and the other all Terra receiver names. A button routes the signal directly from the chosen TX to RX. Two Event Listeners populate all new device names whenever a device is changed, added or removed from the Terra system.

First, create two DropDown Lists and a CustomScript button. Then we need two Event Listeners. All this was explained in the previous steps.

For the first Event Listener, choose the event "TransmitterNames" which is triggered when a Transmitter is changed, added or removed from the Terra system. The event returns all Transmitter names as a list.
To populate the second DropDown List with those names, enter the following command into the scripting field.
`DropDownList2.SetItemsFromArray("value")`
Optionally, give the Event Listener a descriptive name, e.g. "Tx_names".

For the second Event Listener, choose the event "ReceiverNames" which is triggered when a Receiver is changed, added or removed from the Terra system. The event returns all Receiver names as a list.
To populate the third DropDown List with those names, enter the following command into the scripting field.
`DropDownList3.SetItemsFromArray("value")`
Optionally, give the Event Listener a descriptive name, e.g. "Rx_names".

The command for the CustomScript button is:

```
Christie_Terra1.SwitchVideo(DropDownList2.Text,DropDownList3.Text)
```

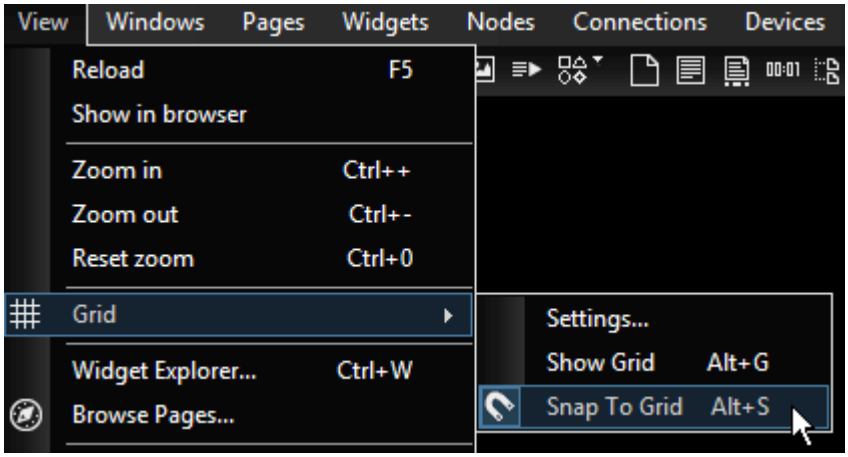
Optionally, add the commands to populates both DropDown Lists with the Transmitter and Receiver names "manually", i.e. without an Event Listener. The solution follows the principle from last time:

```
var list2 = Christie_Terra1.TransmitterNames
DropDownList2.SetItemsFromArray("list2")
var list3 = Christie_Terra1.ReceiverNames
DropDownList3.SetItemsFromArray("list3")
```

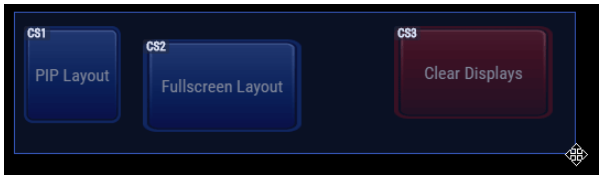
Aligning and Grouping Controls

The more buttons and other controls you create, the more you might like to align them for the sake of good order. There are two workflows. Lastly, we will add Shape widgets that group our buttons visually which makes our interface nicer and easier to understand.

First, you can work with a grid and use the "Snap To Grid" option so that the control borders snap to the grid lines when a control is being moved or resized. Note that this is possible even when the "Show Grid" option is not chosen. The "Grid Settings" dialog allows to alter the grid size and appearance.
Alternatively, you can also select a control in the Move/Edit mode and use the [arrow keys] to move it one pixel. If you use the shortcut [Ctrl + arrow key] it will snap to the next widget's border.

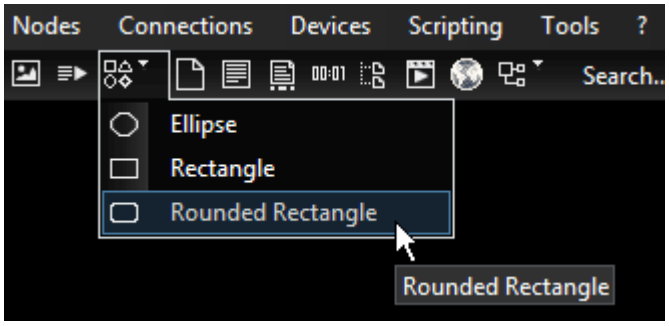


Second, you can align controls manually after selecting multiple controls in the Move/Edit mode. You can either hold the [Ctrl] key and click another button to add it to the selection or you can draw a selection box with the left mouse button.



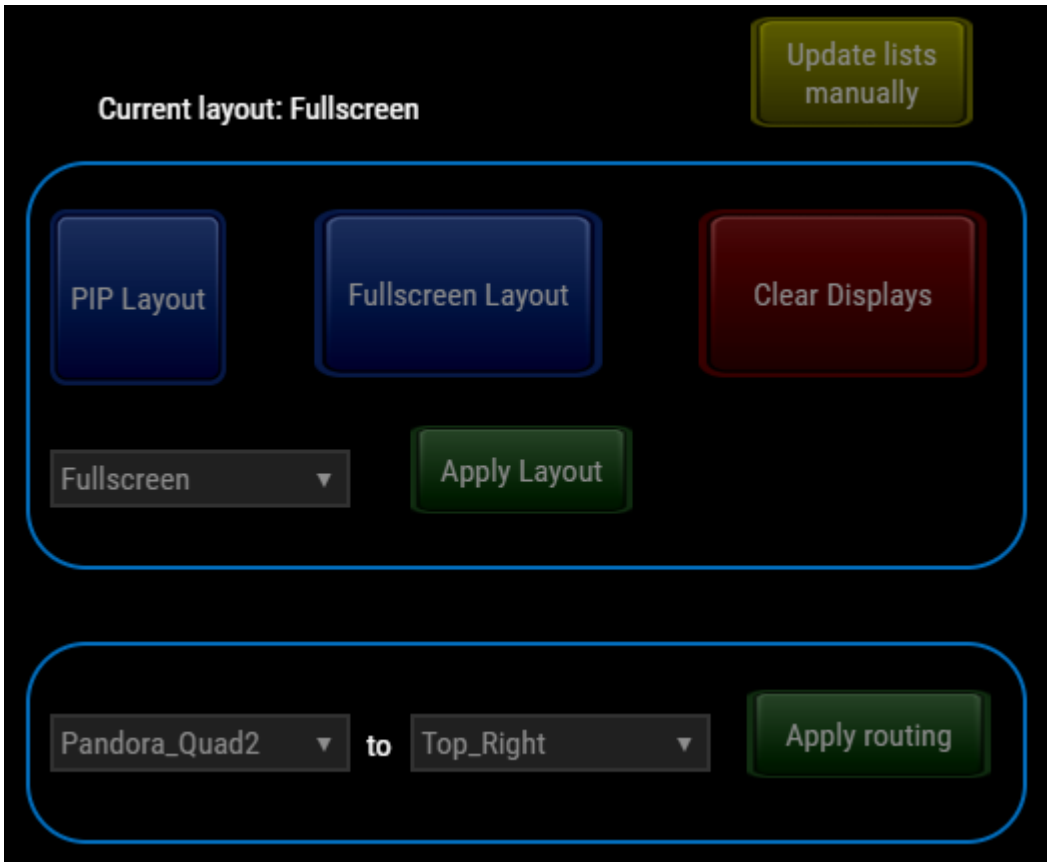
Now, from the top menu, choose "Edit > Align Selection" and for example "Top" to align all top borders to the highest position. You can also right-click the selection and choose "Align Selection" from the context menu.

To add some shapes to your interface , click the shape icon in the middle of the icon bar. Or choose "Shapes > Rounded Rectangle" from the top menu "Widgets".



Then, click once where the top left corner should be and a second time for the bottom right one. As always, there is a Properties dialog that offers more options to adjust the widget's appearance (and behavior).

Below, you see an example how your interface could look like if you followed all the steps in this tutorial.



If you like to, you can now also create a TerraDisplayArray [863] widget which provides a nice overview of a selected Display Array.

7.6.10 Event Listener

The Configuration dialog also provides Event Listeners which offer a simple way of listening to triggers from the connected devices.
Very similar to the Action Script [1237] node, you are able to define a trigger for specific actions or events from the respective device. The number and type of events depend on the device and range from simple value changes to shutdown events.
Note that this is only available in the licensed Widget Designer edition, not the Free version.

The Event Listener shows only events from devices that were added to the Configuration dialog before. The following devices offer events:
Barco devices [1306]
Christie devices [1318]
Fader Extension Events [1355]
Jog/Shuttle Events [1365]
Lightware Events [1379]
Phidgets devices [1383]
PowerPoint Events [1470]
TCP Client Events [1472]
UDP Client Events [1476]
UDP Server&Client Events [1480]

How to Add an Event Listener

To add an Event Listener, open the Devices menu and select "Event Listener > Create Event Listener". This opens the Configuration dialog [1305]. Alternatively, you can add a new Event Listener in the Configuration dialog with the "+" button.

How to Configure an Event Listener

On the right side, you can enter the event's name, device, event and script. Click the "**Apply**" button to save any changes done in the Configuration dialog. You can also use the shortcut [Ctrl + Enter] to do so.

The **Name** is the unique identifier for this Event Listener. It helps to keep track what you have programmed where. The general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).

The "**Enable**" check box is ticked per default. Uncheck it to stop the Event Listener (temporarily). You can also program this, e.g. to a CustomScript button, as described further down.

The "**Device**" drop-down lists all devices that were added to the Configuration and offer events. Choose one and the "Event" drop-down will offer all events referring to that device.

The "**Event**" specifies the trigger, i.e. when to execute the following script. Select one from the drop-down list.

The **scripting field** can be filled with commands that should be executed when the event happens. You could for example write: Label1.Text = "yourMessage"

Some events offer "**Parameters**". When choosing the event "LayoutApplied" from the device Christie_Terra1 for example, you will see that it lists some parameter names on top of the scripting field, such as "layout (String)" and "displayArray (String)". This means that this specific event provides these values as local variables [1903] to be used in the script. For instance, you can write "`Label1.Text = layout`" and the Label will instantly show the name of the Layout which was just applied, whenever this occurs.

How to Use Event Listeners in Regular Scripting

Each Event Listener can also be addressed via scripting for en- and disabling the listener.

`EventListener1.Enable`
This enables the Event Listener with the name "EventListener1", i.e. the "Enable" check box in the Configuration dialog is checked.

```
EventListener1.Disable
```
This disables the Event Listener with the name "EventListener1", i.e. the "Enable" check box in the Configuration dialog is not checked.

```
DebugMessage(EventListener1.IsEnabled)
or
v_bool = EventListener1.IsEnabled
```
This returns "true" if the Event Listener with the name "EventListener1" is enabled, i.e. the "Enable" check box in the Configuration dialog is checked and "false" if it is not.

7.6.11 Fader Extension

The Fader Extension [1993] is a hardware device with faders and buttons and acts as extension for the Jog/Shuttle Controller [1991]. When adding the device to the Configuration [1305] dialog you can assign custom scripts to buttons and the Sequence opacity or other Parameters to faders. The Fader Extension device is available in all Widget Designer editions, including the Free version.

Adding a New Fader Extension Device

To add a Fader Extension device, open the Devices menu and select "Fader Extension > Create Fader Extension Device". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

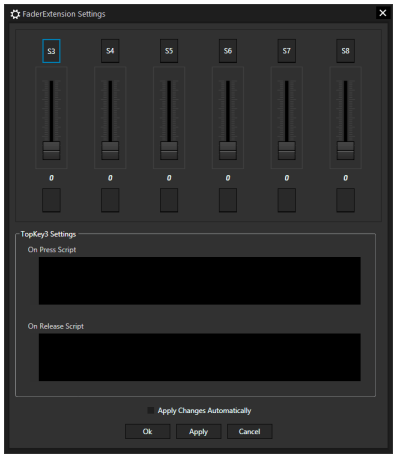
On the right side, you can name, dis-/enable the device or configure it with the "**Open Settings**" button as explained in the next paragraph.

The **Name** is the unique identifier for this Fader Extension object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your Fader Extension device is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Fader Extension and can be opened from here or with Devices menu > Configuration.

The Fader Extension Settings



This dialog allows to assign scripts to buttons of the Fader Extension control hardware and parameter control to its faders. This offers you more flexibility compared to the usage of the device with only the predetermined functionality when it is connected to Pandoras Box.

Press one of the buttons and enter a script which should be executed when the button is pressed or released. Please see the chapter Script Language [1511] for more information how to enter commands.

Select a fader to edit which PB Sequence or which PB Device parameter should be controlled with it. Enter the Site and Device ID from the Layer in Pandoras Box, which you would like to control, e.g. "2.1" or "2.1 3.1" in case you like to control the first Layer of Site 2 and 3 at the same time. If you like to send the inverted fader value (i.e. "0" when fader says "255") to a parameter, fill out the two fields "Device (INV)" and "Parameter (INV)".

Press "OK" or "Apply" to save your changes or check the option "Apply Changes Automatically".

Using the Device in Regular Scripting

After creating a device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "FaderExtension"), Script Assist will offer you a list of all Fader Extension Events [1355].
There are commands to set the dimmer level of specific LEDs:

```
FaderExtension.SetTopButtonBlueLedBrightness(3,255)
```

You can also retrieve specific values via scripting in the same way as from other widgets (e.g. a Fader value):

```
vint = FaderExtension.Fader3.GetValue
```

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and Fader Extension Events [1355] for an overview (with examples and description) over the events raised by a Fader Extension [1993] device.

Using the Device in the Node System

After creating a device, you can also use it in the node system as an input node. Input nodes allow to retrieve information from the associated device and send it to other nodes. Please see the chapter "FaderExtension Input" [983] for more information about the node itself or the chapter "Tutorial: Nodes" [939] for information about the node system.

7.6.11.1 Fader Extension Events

This chapter gives an overview of the events that are raised by a Fader Extension device and that can be used in the Event Listener [1353].
Please read the chapters "Fader Extension" [1354] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Fader Extension Members" [1356].

FaderID.Value

Example:
Fader4.Value = value

This event is raised whenever the according Fader changes its value.
The event returns one parameter to WD which is an integer with the name "value" and holds the current Fader value.
If you select the event "Fader4.Value" and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value from the Fader Extension fader whenever it is moved.

ButtonID.Pressed

Example:
Label1.Text = "YourMessage"

This event is raised whenever the according button is pressed down.
If you select the event "TopKey1.Pressed" and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever the first button on the top of the Fader Extension is pressed down.

ButtonID.Released

Example:
Label1.Text = "YourMessage"

This event is raised whenever the according button is released.
If you select the event "TopKey1.Released" and copy the example into the scripting field of the Event Listener,

it will write "YourMessage" into the Label [888] with ID 1 whenever the first button on the top of the Fader Extension is released.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the device dis-/connects.

7.6.11.2 Fader Extension Members

This chapter gives an overview of the members available for the Fader Extension device.
Please read the chapter "Fader Extension" [1364] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

BottomButtonID.GetState

FaderExtension.BottomButtonID.GetState

Example:
vstring = FaderExtension.BottomButton3.GetState

This member returns the state of BottomButton3 as a Boolean value (True = pressed state or False = released state). BottomButton3 is located below Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.BottomButton3.GetState`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.BottomButton3.GetState)`

This member is available for "BottomButton3" - "BottomButton8".
Alternative commands are: `FaderExtension.GetBottomButtonState(id)` [1357] and `FaderExtension.BottomButtonID.State` [1356]

BottomButtonID.State

FaderExtension.BottomButtonID.State

Example:
vstring = FaderExtension.BottomButton3.State

This member returns the state of BottomButton3 as a Boolean value (True = pressed state or False = released state). BottomButton3 is located below Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.BottomButton3.State`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.BottomButton3.State)`

This member is available for "BottomButton3" - "BottomButton8".
Alternative commands are: `FaderExtension.GetBottomButtonState(id)` [1357] and `FaderExtension.BottomButtonID.GetState` [1356]

Disable

FaderExtension.Disable

Example:
FaderExtension.Disable

This disables the Fader Extension device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1357] member to reverse this command.

Enable

FaderExtension.Enable

Example:
FaderExtension.Enable

This enables the Fader Extension device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1357] member to reverse this command.

FaderID.GetValue

FaderExtension.FaderID.GetValue

Example:
vint = FaderExtension.Fader3.GetValue

This member returns the raw fader position Fader3 as an integer value (0-65535). Fader3 is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.Fader3.GetValue`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.Fader3.GetValue)`
This member is available for "Fader3" - "Fader8".
Alternative commands are: FaderExtension.GetFaderValue(id) [1358] and FaderExtension.FaderID.Value [1357]

FaderID.Value

FaderExtension.FaderID.Value

Example:
vint = FaderExtension.Fader3.Value

This member returns the raw fader position Fader3 as an integer value (0-65535). Fader3 is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.Fader3.Value`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.Fader3.Value)`
This member is available for "Fader3" - "Fader8".
Alternative commands are: FaderExtension.GetFaderValue(id) [1358] and FaderExtension.FaderID.GetValue [1357]

GetBottomButtonState

FaderExtension.GetBottomButtonState(id)

Example:
vstring = FaderExtension.GetBottomButtonState(3)

This member returns the state of BottomButton3 as a Boolean value (True = pressed state or False = released state). BottomButton3 is located below Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.GetBottomButtonState(3)`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.GetBottomButtonState(3))`
Alternative commands are: `FaderExtension.BottomButtonID.GetState` [1356] and `FaderExtension.BottomButtonID.State` [1356]

GetFaderValue

FaderExtension.GetFaderValue(id)

Example:
vint = FaderExtension.GetFaderValue(3)

This member returns the raw fader position Fader3 as an integer value (0-65535). Fader3 is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.GetFaderValue(3)`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.GetFaderValue(3))`

Alternative commands are: FaderExtension.FaderID.GetValue [1357] and FaderExtension.FaderID.Value [1357]

GetTopButtonBlueLedBrightness

FaderExtension.GetTopButtonBlueLedBrightness(id)

Example:
vint = FaderExtension.GetTopButtonBlueLedBrightness(3)

This member returns the dimmer level of the blue LED of TopButton3 as an integer value (0-255). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.GetTopButtonBlueLedBrightness(3)`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.GetTopButtonBlueLedBrightness(3))`

An alternative command is: `FaderExtension.TopButtonID.GetBlueLedBrightness` [1361] and `FaderExtension.TopButtonID.BlueLedBrightness` [1360]

GetTopButtonRedLedBrightness

FaderExtension.GetTopButtonRedLedBrightness(id)

Example:
vint = FaderExtension.GetTopButtonRedLedBrightness(3)

This member returns the dimmer level of the red LED of TopButton3 (the button located above Fader3) as an integer value (0-255). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.GetTopButtonRedLedBrightness(3)`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.GetTopButtonRedLedBrightness(3))`

An alternative command is: `FaderExtension.TopButtonID.GetRedLedBrightness` [1361] and `FaderExtension.TopButtonID.RedLedBrightness` [1362]

GetTopButtonState

FaderExtension.GetTopButtonState(id)

Example:
vstring = FaderExtension.GetTopButtonState(3)

This member returns the state of TopButton3 as a Boolean value (True = pressed state or False = released state). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.GetTopButtonState(3)`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.GetTopButtonState(3))`
Alternative commands are: `FaderExtension.TopButtonID.GetState` [1361] and `FaderExtension.TopButtonID.State` [1363]

IsConnected

FaderExtension.IsConnected

Example:
vstring = FaderExtension.IsConnected

This member returns the connection status for the Fader Extension device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.IsConnected`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.IsConnected)`

IsEnabled

FaderExtension.IsEnabled

Example:
vstring = FaderExtension.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog as a string or Boolean value. The result is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.IsEnabled)`

SetTopButtonBlueLedBrightness

FaderExtension.SetTopButtonBlueLedBrightness(id,value)

Example:
FaderExtension.SetTopButtonBlueLedBrightness(3,125)

This command sets the dimmer level for the blue LED of TopButton3 to 125. Values between 0 (off) and 255 (fully on) are possible. TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.

An alternative command is: `FaderExtension.TopButtonID.SetBlueLedBrightness` [1362] and `FaderExtension.TopButtonID.BlueLedBrightness` [1360]

SetTopButtonRedLedBrightness

FaderExtension.SetTopButtonRedLedBrightness(id,value)

Example:
FaderExtension.SetTopButtonRedLedBrightness(3,125)

This command sets the dimmer level for the red LED of TopButton3 to 125. Values between 0 (off) and 255 (fully on) are possible. TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.

An alternative command is: `FaderExtension.TopButtonID.SetRedLedBrightness` [1362] and `FaderExtension.TopButtonID.RedLedBrightness` [1362]

TopButtonID.BlueLedBrightness

FaderExtension.TopButtonID.BlueLedBrightness

Example:
vint = FaderExtension.TopButton3.BlueLedBrightness

This member sets and returns the dimmer level of the blue LED of TopButton3 as an integer value (0-255). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.TopButton3.BlueLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.TopButton3.BlueLedBrightness)`
Alternative commands are: `FaderExtension.GetTopButtonBlueLedBrightness(id,value)` [1358] and `FaderExtension.TopButtonID.GetBlueLedBrightness` [1361]

Example2:
FaderExtension.TopButton3.BlueLedBrightness = 125

The second example shows, how to set the dimmer level for the blue LED of TopButton3 to a value of 125. Values between 0 (off) and 255 (fully on) are possible.
An alternative command is: `FaderExtension.SetTopButtonBlueLedBrightness(id,value)` [1360] and `FaderExtension.TopButtonID.SetBlueLedBrightness` [1362]

This member is also available for "TopButton3" - "TopButton8".

TopButtonID.GetBlueLedBrightness

FaderExtension.TopButtonID.GetBlueLedBrightness

Example:
vint = FaderExtension.TopButton3.GetBlueLedBrightness

This member returns the dimmer level of the blue LED of TopButton3 as an integer value (0-255). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.TopButton3.GetBlueLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.TopButton3.GetBlueLedBrightness)`

This member is also available for "TopButton3" - "TopButton8".
An alternative command is: `FaderExtension.GetTopButtonBlueLedBrightness(id)` [1358] and `FaderExtension.TopButtonID.BlueLedBrightness` [1360]

TopButtonID.GetRedLedBrightness

FaderExtension.TopButtonID.GetRedLedBrightness

Example:
vint = FaderExtension.TopButton3.GetRedLedBrightness

This member returns the dimmer level of the red LED of TopButton3 (the button located above Fader3) as an integer value (0-255). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.TopButton3.GetRedLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.TopButton3.GetRedLedBrightness)`

This member is also available for "TopButton3" - "TopButton8".
An alternative command is: `FaderExtension.GetTopButtonRedLedBrightness(id)` [1358] and `FaderExtension.TopButtonID.RedLedBrightness` [1362]

TopButtonID.GetState

FaderExtension.TopButtonID.GetState

Example:
vstring = FaderExtension.TopButton3.GetState

This member returns the state of TopButton3 as a Boolean value (True = pressed state or False = released state). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.TopButton3.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.TopButton3.GetState)`

This member is available for "TopButton3" - "TopButton8".

Alternative commands are: `FaderExtension.GetTopButtonState(id)` [1359] and `FaderExtension.TopButtonID.State` [1363]

TopButtonID.RedLedBrightness

FaderExtension.TopButtonID.RedLedBrightness

Example:
vint = FaderExtension.TopButton3.RedLedBrightness

This member sets and returns the dimmer level of the red LED of TopButton3 as an integer value (0-255). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.TopButton3.RedLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(FaderExtension.TopButton3.RedLedBrightness)`
Alternative commands are: `FaderExtension.GetTopButtonRedLedBrightness(id,value)` [1358] and `FaderExtension.TopButtonID.GetRedLedBrightness` [1361]

Example2:
FaderExtension.TopButton3.RedLedBrightness = 125

The second example shows, how to set the dimmer level for the red LED of TopButton3 to a value of 125. Values between 0 (off) and 255 (fully on) are possible.
An alternative command is: `FaderExtension.SetTopButtonRedLedBrightness(id,value)` [1360] and `FaderExtension.TopButtonID.SetRedLedBrightness` [1362]

This member is also available for "TopButton3" - "TopButton8".

TopButtonID.SetBlueLedBrightness

FaderExtension.TopButtonID.SetBlueLedBrightness(value)

Example:
FaderExtension.TopButton3.SetBlueLedBrightness(125)

This command sets the dimmer level for the blue LED of TopButton3 to 125. Values between 0 (off) and 255 (fully on) are possible. TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.

This member is also available for "TopButton3" - "TopButton8".
Alternative commands are: `FaderExtension.SetTopButtonBlueLedBrightness(id,value)` [1360] and `FaderExtension.TopButtonID.BlueLedBrightness` [1360]

TopButtonID.SetRedLedBrightness

FaderExtension.TopButtonID.SetRedLedBrightness(value)

Example:
FaderExtension.TopButton3.SetRedLedBrightness(125)

This command sets the dimmer level for the red LED of TopButton3 to 125. Values between 0 (off) and 255 (fully on) are possible. TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.

This member is also available for "TopButton3" - "TopButton8".
An alternative command is: `FaderExtension.SetTopButtonRedLedBrightness(id,value)` [1360] and `FaderExtension.TopButtonID.RedLedBrightness` [1362]

TopButtonID.State

FaderExtension.TopButtonID.State

Example:
vstring = FaderExtension.TopButton3.State

This member returns the state of TopButton3 as a Boolean value (True = pressed state or False = released state). TopButton3 is located above Fader3, which is the first fader on the Fader Extension as the Jog/Shuttle [1364] has two faders.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = FaderExtension.TopButton3.State`
or simply display it in the Debug Logger [812]: `DebugMessage(FaderExtension.TopButton3.State)`
This member is available for "TopButton3" - "TopButton8".

Alternative commands are: `FaderExtension.GetTopButtonState(id)` [1359] and `FaderExtension.TopButtonID.GetState` [1361]

7.6.12 Jog/Shuttle

The Jog/Shuttle [1991] is a hardware device with faders and buttons and can be used together with the Fader Extension [1993]. When adding the device to the Configuration [1305] dialog you can assign custom scripts to buttons and the Sequence opacity or other Parameters to faders. The Jog/Shuttle device is available in all Widget Designer editions, including the Free version.

Adding a New Jog/Shuttle Device

To add a Jog/Shuttle device, open the Devices menu and select "Jog/Shuttle > Create Jog/Shuttle Device". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

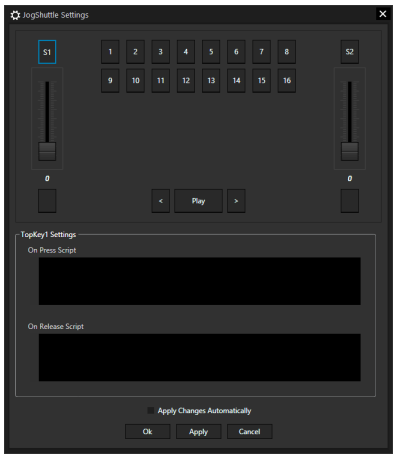
On the right side, you can name, dis-/enable the device or configure it with the "**Open Settings**" button as explained in the next paragraph.

The **Name** is the unique identifier for this Jog/Shuttle object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your Jog/Shuttle device is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Jog/Shuttle and can be opened from here or with Devices menu > Configuration.

The Jog/Shuttle Settings



This dialog allows to assign scripts to buttons of the Jog/Shuttle control hardware and parameter control to its faders. This offers you more flexibility compared to the usage of the device with only the predetermined functionality when it is connected to Pandoras Box.

Press one of the buttons (physically or in the UI) and enter a script which should be executed when the button is pressed or released. Please see the chapter Script Language [1511] for more information how to enter commands.

Select a fader to edit which PB Sequence or which PB Device parameter should be controlled with it. Enter the Site and Device ID from the Layer in Pandoras Box, which you would like to control, e.g. "2.1" or "2.1 3.1" in case you like to control the first Layer of Site 2 and 3 at the same time. If you like to send the inverted fader value (i.e. "0" when fader says "255") to a parameter, fill out the two fields "Device (INV)" and "Parameter (INV)".

Press "OK" or "Apply" to save your changes or check the option "Apply Changes Automatically".

Using the Device in Regular Scripting

After creating a device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "JogShuttle"), Script Assist will offer you a list of all Jog/Shuttle Members [1366].

There are commands to set the dimmer level of specific LEDs:

```
JogShuttle.SetTopButtonBlueLedBrightness(1,255)
```

You can also retrieve specific values via scripting in the same way as from other widgets (e.g. a Fader value):

```
vint = JogShuttle.Fader1.GetValue
```

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and Jog/Shuttle Events [1365] for an overview (with examples and description) over the events raised by a Jog/Shuttle [1991] device.

Using the Device in the Node System

After creating a device, you can also use it in the node system as an input node. Input nodes allow to retrieve information from the associated device and send it to other nodes. Please see the chapter "JogShuttle Input" [1009] for more information about the node itself or the chapter "Tutorial: Nodes" [939] for information about the node system.

7.6.12.1 Jog/Shuttle Events

This chapter gives an overview of the events that are raised by a Jog/Shuttle device and that can be used in the Event Listener [1353].
Please read the chapters "Jog/Shuttle" [1364] and "Event Listener" [1353] if you like to know how to add and use both.
The available members that can be used in our scripting language are explained in the chapter "Jog/Shuttle Members" [1366].

FaderID.Value

Example:
Fader2.Value = value

This event is raised whenever the according Fader changes its value.
The event returns one parameter to WD which is an integer with the name "value" and holds the current Fader value.
If you select the event "Fader2.Value" and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 2 will be assigned with the same value from the Jog/Shuttle fader whenever it is moved.

ButtonID.Pressed

Example:
Label1.Text = "YourMessage"

This event is raised whenever the according button is pressed down.
If you select the event "Button1.Pressed" and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever Button1 on the Jog/Shuttle is pressed down.

ButtonID.Released

Example:
Label1.Text = "YourMessage"

This event is raised whenever the according button is released.
If you select the event "Button1.Released" and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever Button1 on the Jog/Shuttle is released.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the device dis-/connects.

7.6.12.2 Jog/Shuttle Members

This chapter gives an overview of the members available for the Jog/Shuttle device.
Please read the chapter "Jog/Shuttle" [1364] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

BottomButtonID.GetState

JogShuttle.BottomButtonID.GetState

Example:
vstring = JogShuttle.BottomButton1.GetState

This member returns the state of BottomButton1 (the button located below Fader1) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.BottomButton1.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.BottomButton1.GetState)`
This member is also available for "BottomButton2".
Alternative commands are: `JogShuttle.GetBottomButtonState(id)` [1369] and `JogShuttle.BottomButtonID.State` [1366]

BottomButtonID.State

JogShuttle.BottomButtonID.State

Example:
vstring = JogShuttle.BottomButton1.State

This member returns the state of BottomButton1 (the button located below Fader1) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.BottomButton1.State`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.BottomButton1.State)`
This member is also available for "BottomButton2".
Alternative commands are: `JogShuttle.GetBottomButtonState(id)` [1369] and `JogShuttle.BottomButtonID.GetState` [1366]

CueButtonID.GetLedBrightness

JogShuttle.CueButtonID.GetLedBrightness

Example:
vint = JogShuttle.CueButton1.GetLedBrightness

This member returns the dimmer level of the LED of CueButton1 as an integer value (0-255).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.CueButton1.GetLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(JogShuttle.CueButton1.GetLedBrightness)`

This member is also available for "CueButton2" - "CueButton16".
Alternative commands are: `JogShuttle.GetCueButtonLedBrightness(id)` [1370] and `JogShuttle.CueButton1.LedBrightness` [1367]

CueButtonID.GetState

JogShuttle.CueButtonID.GetState

Example:
vstring = JogShuttle.CueButton1.GetState

This member returns the state of Button1 as a Boolean value (True = pressed state or False = released state). The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.CueButton1.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.CueButton1.GetState)`

This member is also available for "CueButton2" - "CueButton16".
An alternative command is: `JogShuttle.GetCueButtonState(id)` [1369] and `JogShuttle.CueButtonID.State` [1368]

CueButtonID.LedBrightness

JogShuttle.CueButtonID.LedBrightness

Example:
vint = JogShuttle.CueButton1.LedBrightness

This member sets and returns the dimmer level of the LED of CueButton1 as an integer value (0-255). The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.CueButton1.LedBrightness`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.CueButton1.LedBrightness)`
Alternative commands are: `JogShuttle.GetCueButtonLedBrightness(id,value)` [1370] and `JogShuttle.CueButton1.GetLedBrightness` [1366]

Example2:
JogShuttle.CueButton1.LedBrightness = 125

The second example shows, how to set the dimmer level for the LED of CueButton1 to a value of 125. Values between 0 (off) and 255 (fully on) are possible.
Alternative commands are: `JogShuttle.SetCueButtonLedBrightness(id,value)` [1373] and `JogShuttle.CueButton1.SetLedBrightness` [1367]

This member is also available for "CueButton2" - "CueButton16".

CueButtonID.SetLedBrightness

JogShuttle.CueButtonID.SetLedBrightness(value)

Example:
JogShuttle.CueButton1.SetLedBrightness(125)

This command sets the dimmer level for the LED of CueButton1 to a value of 125. Values between 0 (off) and 255 (fully on) are possible.

This member is also available for "CueButton2" - "CueButton16".
Alternative commands are: `JogShuttle.SetCueButtonLedBrightness(id,value)` [1373] and `JogShuttle.CueButton1.LedBrightness` [1367]

CueButtonID.State

JogShuttle.CueButtonID.State

Example:
vstring = JogShuttle.CueButton1.State

This member returns the state of Button1 as a Boolean value (True = pressed state or False = released state). The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.CueButton1.State`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.CueButton1.State)`
This member is also available for "CueButton2" - "CueButton16".

Alternative commands are: `JogShuttle.GetCueButtonState(id)` [1369] and `JogShuttle.CueButtonID.GetState` [1367]

Disable

JogShuttle.Disable

Example:
JogShuttle.Disable

This disables the Jog/Shuttle device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1368] member to reverse this command.

Enable

JogShuttle.Enable

Example:
JogShuttle.Enable

This enables the Jog/Shuttle device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1368] member to reverse this command.

FaderID.GetValue

JogShuttle.FaderID.GetValue

Example:
vint = JogShuttle.Fader1.GetValue

This member returns the raw fader position Fader1 as an integer value (0-65535).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.Fader1.GetValue`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.Fader1.GetValue)`

This member is also available for "Fader2".
Alternative commands are: JogShuttle.GetFaderValue(id) [1370] and JogShuttle.FaderID.Value [1369]

FaderID.Value

JogShuttle.FaderID.Value

Example:
vint = JogShuttle.Fader1.Value

This member returns the raw fader position Fader1 as an integer value (0-65535).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.Fader1.Value`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.Fader1.Value)`
This member is also available for "Fader2".
Alternative commands are: JogShuttle.GetFaderValue(id) [1370] and JogShuttle.FaderID.GetValue [1368]

GetBottomButtonState

JogShuttle.GetBottomButtonState(id)

Example:
vstring = JogShuttle.GetBottomButtonState(1)

This member returns the state of BottomButton1 (the button located below Fader1) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.GetBottomButtonState(1)`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.GetBottomButtonState(1))`
Alternative commands are: `JogShuttle.BottomButtonID.GetState` [1366] and `JogShuttle.BottomButtonID.State` [1366]

GetCueButtonState

JogShuttle.GetCueButtonState(id)

Example:
vstring = JogShuttle.GetCueButtonState(1)

This member returns the state of CueButton1 as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.GetCueButtonState(1)`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.GetCueButtonState(1))`
This member is also available for CueButton ID 2-16.

An alternative command is: `JogShuttle.CueButtonID.GetState` [1367] and `JogShuttle.CueButtonID.State` [1368]

GetCueButtonLedBrightness

JogShuttle.GetCueButtonLedBrightness(id)

Example:
vint = JogShuttle.GetCueButtonLedBrightness(1)

This member returns the dimmer level of the LED of (Cue) Button1 as an integer value (0-255).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.GetCueButtonLedBrightness(1)`
or simply display it in the Debug Logger [812]:
`DebugMessage(JogShuttle.GetCueButtonLedBrightness(1))`
Alternative commands are: `JogShuttle.CueButtonID.GetLedBrightness` [1366] and `JogShuttle.CueButtonID.LedBrightness` [1367]

GetFaderValue

JogShuttle.GetFaderValue(id)

Example:
vint = JogShuttle.GetFaderValue(1)

This member returns the raw fader position Fader1 as an integer value (0-65535).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.GetFaderValue(1)`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.GetFaderValue(1))`

Alternative commands are: JogShuttle.FaderID.GetValue [1368] and JogShuttle.FaderID.Value [1369]

GetTopButtonBlueLedBrightness

JogShuttle.GetTopButtonBlueLedBrightness(id)

Example:
vint = JogShuttle.GetTopButtonBlueLedBrightness(1)

This member returns the dimmer level of the blue LED of TopButton1 (the button located above Fader1) as an integer value (0-255).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.GetTopButtonBlueLedBrightness(1)`
or simply display it in the Debug Logger [812]:
`DebugMessage(JogShuttle.GetTopButtonBlueLedBrightness(1))`

An alternative command is: `JogShuttle.TopButtonID.GetBlueLedBrightness` [1374] and `JogShuttle.TopButtonID.BlueLedBrightness` [1374]

GetTopButtonRedLedBrightness

JogShuttle.GetTopButtonRedLedBrightness(id)

Example:
vint = JogShuttle.GetTopButtonRedLedBrightness(1)

This member returns the dimmer level of the red LED of TopButton1 (the button located above Fader1) as an integer value (0-255).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.GetTopButtonRedLedBrightness(1)`
or simply display it in the Debug Logger [812]:
`DebugMessage(JogShuttle.GetTopButtonRedLedBrightness(1))`

An alternative command is: `JogShuttle.TopButtonID.GetRedLedBrightness` [1375] and `JogShuttle.TopButtonID.RedLedBrightness` [1375]

GetTopButtonState

JogShuttle.GetTopButtonState(id)

Example:
vstring = JogShuttle.GetTopButtonState(1)

This member returns the state of TopButton1 (the button located above Fader1) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.GetTopButtonState(1)`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.GetTopButtonState(1))`
Alternative commands are: `JogShuttle.TopButtonID.GetState` [1375] and `JogShuttle.TopButtonID.State` [1376]

IsConnected

JogShuttle.IsConnected

Example:
vstring = JogShuttle.IsConnected

This member returns the connection status for the Jog/Shuttle device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.IsConnected`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.IsConnected)`

IsEnabled

JogShuttle.IsEnabled

Example:
vstring = JogShuttle.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog as a string or Boolean value. The result is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.IsEnabled)`

LeftButton.GetState

JogShuttle.LeftButton.GetState

Example:
vstring = JogShuttle.LeftButton.GetState

This member returns the state of the left Button (next to the Play button) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.LeftButton.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.LeftButton.GetState)`
An alternative command is: `JogShuttle.LeftButton.State` [1372]

LeftButton.State

JogShuttle.LeftButton.State

Example:
vstring = JogShuttle.LeftButton.State

This member returns the state of the left Button (next to the Play button) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.LeftButton.State`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.LeftButton.State)`
An alternative command is: `JogShuttle.LeftButton.GetState` [1372]

PlayButton.GetState

JogShuttle.PlayButton.GetState

Example:
vstring = JogShuttle.PlayButton.GetState

This member returns the state of the Play Button as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.PlayButton.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.PlayButton.GetState)`
An alternative command is: `JogShuttle.PlayButton.State` [1372]

PlayButton.State

JogShuttle.PlayButton.State

Example:
vstring = JogShuttle.PlayButton.State

This member returns the state of the Play Button as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.PlayButton.State`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.PlayButton.State)`
An alternative command is: `JogShuttle.PlayButton.GetState` [1372]

RightButton.GetState

JogShuttle.RightButton.GetState

Example:
vstring = JogShuttle.RightButton.GetState

This member returns the state of the right button (next to the Play button) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.RightButton.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.RightButton.GetState)`
An alternative command is: `JogShuttle.RightButton.State` [1373]

RightButton.State

JogShuttle.RightButton.State

Example:
vstring = JogShuttle.RightButton.State

This member returns the state of the right button (next to the Play button) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.RightButton.State`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.RightButton.State)`
An alternative command is: `JogShuttle.RightButton.GetState` [1373]

SetCueButtonLedBrightness

JogShuttle.SetCueButtonLedBrightness(id,value)

Example:
JogShuttle.SetCueButtonLedBrightness(1,125)

This member sets the dimmer level of the LED of (Cue) Button1 to a value of 125. Values between 0 (off) and 255 (fully on) are possible.

Alternative commands are: `JogShuttle.CueButtonID.SetLedBrightness` [1367] and `JogShuttle.CueButtonID.LedBrightness` [1367]

SetTopButtonBlueLedBrightness

JogShuttle.SetTopButtonBlueLedBrightness(id,value)

Example:
JogShuttle.SetTopButtonBlueLedBrightness(1,125)

This member sets the dimmer level of the blue LED of TopButton1 (the button located above Fader1) to a value of 125. Values between 0 (off) and 255 (fully on) are possible.

An alternative command is: `JogShuttle.TopButtonID.SetBlueLedBrightness` 1376 and `JogShuttle.TopButtonID.BlueLedBrightness` 1374

SetTopButtonRedLedBrightness

JogShuttle.SetTopButtonRedLedBrightness(id,value)

Example:
JogShuttle.SetTopButtonRedLedBrightness(1,125)

This member sets the dimmer level of the red LED of TopButton1 (the button located above Fader1) to a value of 125. Values between 0 (off) and 255 (fully on) are possible.

An alternative command is: `JogShuttle.TopButtonID.SetRedLedBrightness` 1376 and `JogShuttle.TopButtonID.RedLedBrightness` 1375

TopButtonID.BlueLedBrightness

JogShuttle.TopButtonID.BlueLedBrightness

Example:
vint = JogShuttle.TopButton1.BlueLedBrightness

This command sets and returns the dimmer level for the blue LED of TopButton1 (the button located above Fader1) as an integer value (0-255).
The first example shows how to return the value to an existing (global) integer variable 1900 named "vint".

You could also write it into a Widget, e.g. a Label 888 via the script: `Label1.Text = JogShuttle.TopButton1.BlueLedBrightness`
or simply display it in the Debug Logger 812:
`DebugMessage(JogShuttle.TopButton1.BlueLedBrightness)`
Alternative commands are: `JogShuttle.GetTopButtonBlueLedBrightness(id,value)` 1370 and `JogShuttle.TopButtonID.GetBlueLedBrightness` 1374

Example2:
JogShuttle.TopButton1.BlueLedBrightness = 125

The second example shows, how to set the dimmer level for the blue LED of TopButton1 to a value of 125. Values between 0 (off) and 255 (fully on) are possible.
An alternative command is: `JogShuttle.SetTopButtonBlueLedBrightness(id,value)` 1373 and `JogShuttle.TopButtonID.SetBlueLedBrightness` 1376

This member is also available for "TopButton2".

TopButtonID.GetBlueLedBrightness

JogShuttle.TopButtonID.GetBlueLedBrightness

Example:
vint = JogShuttle.TopButton1.GetBlueLedBrightness

This member returns the dimmer level of the blue LED of TopButton1 (the button located above Fader1) as an integer value (0-255).
The first example shows how to use it with an existing (global) integer variable 1900 named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.TopButton1.GetBlueLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(JogShuttle.TopButton1.GetBlueLedBrightness)`

This member is also available for "TopButton2".
An alternative command is: `JogShuttle.GetTopButtonBlueLedBrightness(id)` [1370] and `JogShuttle.TopButtonID.BlueLedBrightness` [1374]

TopButtonID.GetRedLedBrightness

JogShuttle.TopButtonID.GetRedLedBrightness

Example:
vint = JogShuttle.TopButton1.GetRedLedBrightness

This member returns the dimmer level of the red LED of TopButton1 (the button located above Fader1) as an integer value (0-255).
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.TopButton1.GetRedLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(JogShuttle.TopButton1.GetRedLedBrightness)`

This member is also available for "TopButton2".
An alternative command is: `JogShuttle.GetTopButtonRedLedBrightness(id)` [1370] and `JogShuttle.TopButtonID.RedLedBrightness` [1375]

TopButtonID.GetState

JogShuttle.TopButtonID.GetState

Example:
vstring = JogShuttle.TopButton1.GetState

This member returns the state of TopButton1 (the button located above Fader1) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.TopButton1.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.TopButton1.GetState)`
This member is also available for "TopButton2".

Alternative commands are: `JogShuttle.GetTopButtonState(id)` [1371] and `JogShuttle.TopButtonID.State` [1376]

TopButtonID.RedLedBrightness

JogShuttle.TopButtonID.RedLedBrightness

Example:
vint = JogShuttle.TopButton1.RedLedBrightness

This command sets and returns the dimmer level for the red LED of TopButton1 (the button located above Fader1) as an integer value (0-255).
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.TopButton1.RedLedBrightness`
or simply display it in the Debug Logger [812]:
`DebugMessage(JogShuttle.TopButton1.RedLedBrightness)`
Alternative commands are: `JogShuttle.GetTopButtonRedLedBrightness(id,value)` [1370] and `JogShuttle.TopButtonID.GetRedLedBrightness` [1375]

Example2:
JogShuttle.TopButton1.RedLedBrightness = 125

The second example shows, how to set the dimmer level for the red LED of TopButton1 to a value of 125. Values between 0 (off) and 255 (fully on) are possible.
An alternative command is: `JogShuttle.SetTopButtonRedLedBrightness(id,value)` [1374] and `JogShuttle.TopButtonID.SetRedLedBrightness` [1376]

This member is also available for "TopButton2".

TopButtonID.SetBlueLedBrightness

JogShuttle.TopButtonID.SetBlueLedBrightness(value)

Example:
JogShuttle.TopButton1.SetBlueLedBrightness(255)

This command sets the dimmer level for the blue LED of TopButton1 (the button located above Fader1) to 255. Values between 0 (off) and 255 (fully on) are possible.

This member is also available for "TopButton2".
Alternative commands are: `JogShuttle.SetTopButtonBlueLedBrightness(id,value)` [1373] and `JogShuttle.TopButtonID.BlueLedBrightness` [1374]

TopButtonID.SetRedLedBrightness

JogShuttle.TopButtonID.SetRedLedBrightness(value)

Example:
JogShuttle.TopButton1.SetRedLedBrightness(255)

This command sets the dimmer level for the red LED of TopButton1 (the button located above Fader1) to 255. Values between 0 (off) and 255 (fully on) are possible.

This member is also available for "TopButton2".
An alternative command is: `JogShuttle.SetTopButtonRedLedBrightness(id,value)` [1374] and `JogShuttle.TopButtonID.RedLedBrightness(value)` [1375]

TopButtonID.State

JogShuttle.TopButtonID.State

Example:
vstring = JogShuttle.TopButton1.State

This member returns the state of TopButton1 (the button located above Fader1) as a Boolean value (True = pressed state or False = released state).
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = JogShuttle.TopButton1.State`
or simply display it in the Debug Logger [812]: `DebugMessage(JogShuttle.TopButton1.State)`

This member is also available for "TopButton2".

Alternative commands are: `JogShuttle.GetTopButtonState(id)` [1371] and `JogShuttle.TopButtonID.GetState` [1375]

7.6.13 Lightware Matrix LW2

The Lightware LW2 Device in the Configuration dialog allows to easily remote control different functions of a Lightware device e.g. a matrix. Widget Designer uses the LW2 protocol which was developed by Lightware Visual Engineering and is present in their video signal management equipment per default. Note that the device is only available in the licensed Widget Designer edition, not the Free version.

Adding a New Lightware LW2 Device

To add a Lightware LW2 device, open the Devices menu and select "Lightware > Matrix LW2 > Create Matrix LW2". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side you see several options:

The **Name** is the unique identifier for this Lightware object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your Lightware device is a filled blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A filled gray circle indicates a disabled device.

Now, please enter the **IP address** of the Lightware device. If needed, adjust the **Port**. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

If the connection is successful, the **Product Type** shows to which device you are connected, e.g. "MX8x8DVI Frame".

You can close the dialog now. The newly created device will also be added to the Devices menu > Lightware > Matrix LW2 and can be opened from here or with Devices menu > Configuration.

Using the Device in Regular Scripting

After creating a device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Lightware_MatrixLW2_1"), Script Assist will offer you a list of all Lightware Matrix LW2 Members [1379].

You can for example route a source to a target signal:

```
Lightware_MatrixLW2_1.Route(2,1)
```

You can also retrieve information via scripting:

```
vstring = Lightware_MatrixLW2_1.GetLastMessageReceived
```

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and Jog/Shuttle Events [1365] for an overview (with examples and description) over the events raised by this device.

7.6.13.1 Lightware Matrix LW2 Events

This chapter gives an overview of the events that are raised by a Lightware LW2 device and that can be used in the Event Listener [1353].
Please read the chapters "Lightware Matrix LW2" [1378] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Lightware Matrix LW2 Members" [1379].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the device dis-/connects.

MessageReceived

Example:
Label1.Text = MessageReceived

This event is raised whenever the connected device sends a message.
The event returns one parameter to WD which is a string value with the name "MessageReceived" and holds the message.
If you select this event and copy the example into the scripting field of the Event Listener, it will write the message into the Label [888] with ID 1 whenever the device sends one.

7.6.13.2 Lightware Matrix LW2 Members

This chapter gives an overview of the members available for the Lightware Matrix LW2 device.
Please read the chapter "Lightware Matrix LW2" [1378] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

CustomCommand

Lightware_MatrixLW2_ID.CustomCommand(command)

Example:
Lightware_MatrixLW2_1.CustomCommand("String")

This sends a custom command to the Lightware device. Please check the manual of your Lightware device for a documentation which LW2 commands it supports.

Disable

Lightware_MatrixLW2_ID.Disable

Example:
Lightware_MatrixLW2_1.Disable

This disables the Lightware device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1380] member to reverse this command.

Enable

Lightware_MatrixLW2_ID.Enable

Example:
Lightware_MatrixLW2_1.Enable

This enables the Lightware device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1379] member to reverse this command.

GetLastMessageReceived

Lightware_MatrixLW2_ID.GetLastMessageReceived

Example:
vstring = Lightware_MatrixLW2_1.GetLastMessageReceived

This returns the last message that was received from the Lightware device as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_1.GetLastMessageReceived`
or simply display it in the Debug Logger [812]:
`DebugMessage(Lightware_MatrixLW2_1.GetLastMessageReceived)`

An alternative command is: `LastMessageReceived` [1381]

GetProductType

Lightware_MatrixLW2_ID.GetProductType

Example:
vstring = Lightware_MatrixLW2_1.GetProductType

This returns the product type from the Lightware device as a string value. The result could look as follows: MX8x8DVI Frame
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_1.GetProductType`
or simply display it in the Debug Logger [812]:
`DebugMessage(Lightware_MatrixLW2_1.GetProductType)`

An alternative command is: `ProductType` [1382]

IpAddress

Lightware_MatrixLW2_ID.IpAddress

Example:
vstring = Lightware_MatrixLW2_1.IpAddress

This returns the IP address of the Lightware device as a string. The result could look as follows: 10.169.80.10
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_1.IpAddress`
or simply display it in the Debug Logger [812]: `DebugMessage(Lightware_MatrixLW2_1.IpAddress)`

In return, you can also assign an IP address to the connected Lightware device. WD automatically connects to it afterwards.
`Lightware_MatrixLW2_1.IpAddress = "10.169.80.10"`

IsConnected

Lightware_MatrixLW2_ID.IsConnected

Example:
vstring = Lightware_MatrixLW2_ID.IsConnected

This member returns the connection status for the Lightware device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_ID.IsConnected`
or simply display it in the Debug Logger [812]: `DebugMessage(Lightware_MatrixLW2_ID.IsConnected)`

IsEnabled

Lightware_MatrixLW2_ID.IsEnabled

Example:
vstring = Lightware_MatrixLW2_ID.IsEnabled

This member returns the status of the check box "Enable" for the Lightware device in the Configuration dialog [1305] as a string or Boolean value. The result is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_ID.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Lightware_MatrixLW2_ID.IsEnabled)`

LastMessageReceived

Lightware_MatrixLW2_ID.LastMessageReceived

Example:
vstring = Lightware_MatrixLW2_1.LastMessageReceived

This returns the last message that was received from the Lightware device as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_1.LastMessageReceived`
or simply display it in the Debug Logger [812]:
`DebugMessage(Lightware_MatrixLW2_1.LastMessageReceived)`

An alternative command is: `GetLastMessageReceived` [1380]

LoadPreset

Lightware_MatrixLW2_ID.LoadPreset(presetID)

Example:
Lightware_MatrixLW2_1.LoadPreset(2)

This sends a command to the Lightware device which loads the preset with ID 2.

Port

Lightware_MatrixLW2_ID.Port

Example:
vint = Lightware_MatrixLW2_1.Port

This returns the port number of the Lightware device in the Configuration dialog [1305] as an integer value. The result could look as follows: 10001
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_1.Port`
or simply display it in the Debug Logger [812]: `DebugMessage(Lightware_MatrixLW2_1.Port)`

In return, you can also assign another port to the connected Lightware device. WD automatically connects to it afterwards.
`Lightware_MatrixLW2_1.Port = 10002`

ProductType

Lightware_MatrixLW2_ID.ProductType

Example:
vstring = Lightware_MatrixLW2_1.ProductType

This returns the product type from the Lightware device as a string value. The result could look as follows: MX8x8DVI Frame
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Lightware_MatrixLW2_1.ProductType`
or simply display it in the Debug Logger [812]: `DebugMessage(Lightware_MatrixLW2_1.ProductType)`

An alternative command is: `GetProductType` [1380]

Route

Lightware_MatrixLW2_ID.Route(source,target)

Example:
Lightware_MatrixLW2_1.Route(3,2)

This sends a command to the Lightware device which routes the source 3 to target 2.

7.6.14 Phidgets

Phidgets are a set of "plug and play" building blocks for low cost USB sensing and controlling which can be read out and controlled via Widget Designer. Note that the devices are only available in the licensed Widget Designer edition, not the Free version.
The new implementation of Phidgets since version 6.1.2 offers a large range of available devices for all kinds of scenarios.

For learning more about the hardware devices themselves, please refer to the official Phidgets website.

These Phidgets are currently implemented in WD:

- Digital Input 1388
- Digital Output 1393
- InterfaceKit 0/0/4 1399
- InterfaceKit 0/16/16 1399
- InterfaceKit 8/8/8 1399
- IR Code Table 1418
- PhidgetIR 1414
- PhidgetRFID 1422
- Server 1425
- Servo 1428
- Sound Phidget 1431
- Spatial 0/0/3 1434
- Spatial 3/3/3 1436
- Stepper 1439
- Temperature Phidget 1450
- Voltage Input 1452
- Voltage Ratio Input 1460

Setting up the Hardware

Before being able to use Phidget devices, you need to install the Phidgets driver on your computer.
You can find the latest driver for download in the Learn section of the Phidgets website, you only need this one driver for all devices.

The Network Server feature allows you to access Phidgets connected to any other computer in your network.
If you intend to use network Phidgets, please make sure to use the **same driver version on every machine**.
Different driver versions can result in various communication problems.

After you have connected all Phidgets you require for your project, use the Phidget Control Panel 1385 to check if they were all recognized and are ready to use.

Adding a New Phidgets Device

To add a Phidget device, open the Devices menu and select the respective device in the "Phidgets Device" section. This opens the Configuration dialog 1305. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

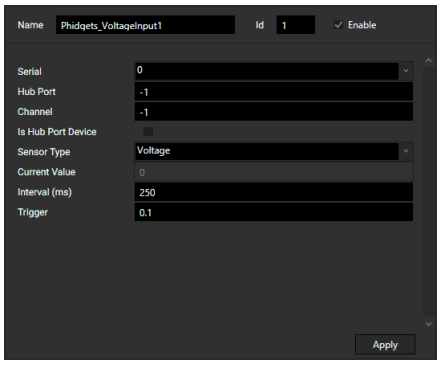
The **Name** is the unique identifier for this Phidget object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).

If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidget device is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > specific device and can be opened from here or with Devices menu > Configuration.

The Phidgets Settings



This dialog allows to set up a Phidget device, for example a Voltage Input device.
If the device is physically connected to your computer or the Phidget network, select the respective serial number of the device from the drop-down list. If it is connected to a hub, the hub's serial is needed. If necessary, enter the settings for **Hub Port**, **Channel** and **Is Hub Port Device** as seen in the Phidget Control Panel's addressing information [1386] window.

Select the device's product number from the drop-down if required, for example "PN_1142" for an "Indoor Light Sensor", which is a Voltage Input device usually connected to either a VINT Hub or an InterfaceKit.
Depending on the device type, you can also adjust parameters like Trigger (sensitivity) and value request Interval time.
Each device type has its own set of parameters.

Press "Apply" to save your changes and connect to the device

Using the Device in Regular Scripting

After adding a device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_InterfaceKit_8_8_8_1"), Script Assist will offer you a list of all available members.

The InterfaceKit [1399] in this example enables you to set one of the eight digital outputs to "True" (on state) and "False" (off state):

```
Phidgets_InterfaceKit_8_8_8_1.SetOutputState(0,true)
```

You can also retrieve specific values via scripting in the same way as from other widgets (e.g. a Digital Input value):

```
v_bool = Phidgets_InterfaceKit_8_8_8_1.Input3.State
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:

```
Phidgets_InterfaceKit_8_8_8_1.Output0.State = true
Label5.Text = "State Input 3: " + Phidgets_InterfaceKit_8_8_8_1.Input3.State
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:

```
Phidgets_InterfaceKit_8_8_8_1.Output0.SetState(true)
Label5.Text = "State Input 3: " + Phidgets_InterfaceKit_8_8_8_1.Input3.GetState
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. Digital Input values, and do not provide a "Set..." member.
Furthermore, the higher-level properties of a device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For more information about how to script specific attributes, please refer to the respective device's "Member" chapter.

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

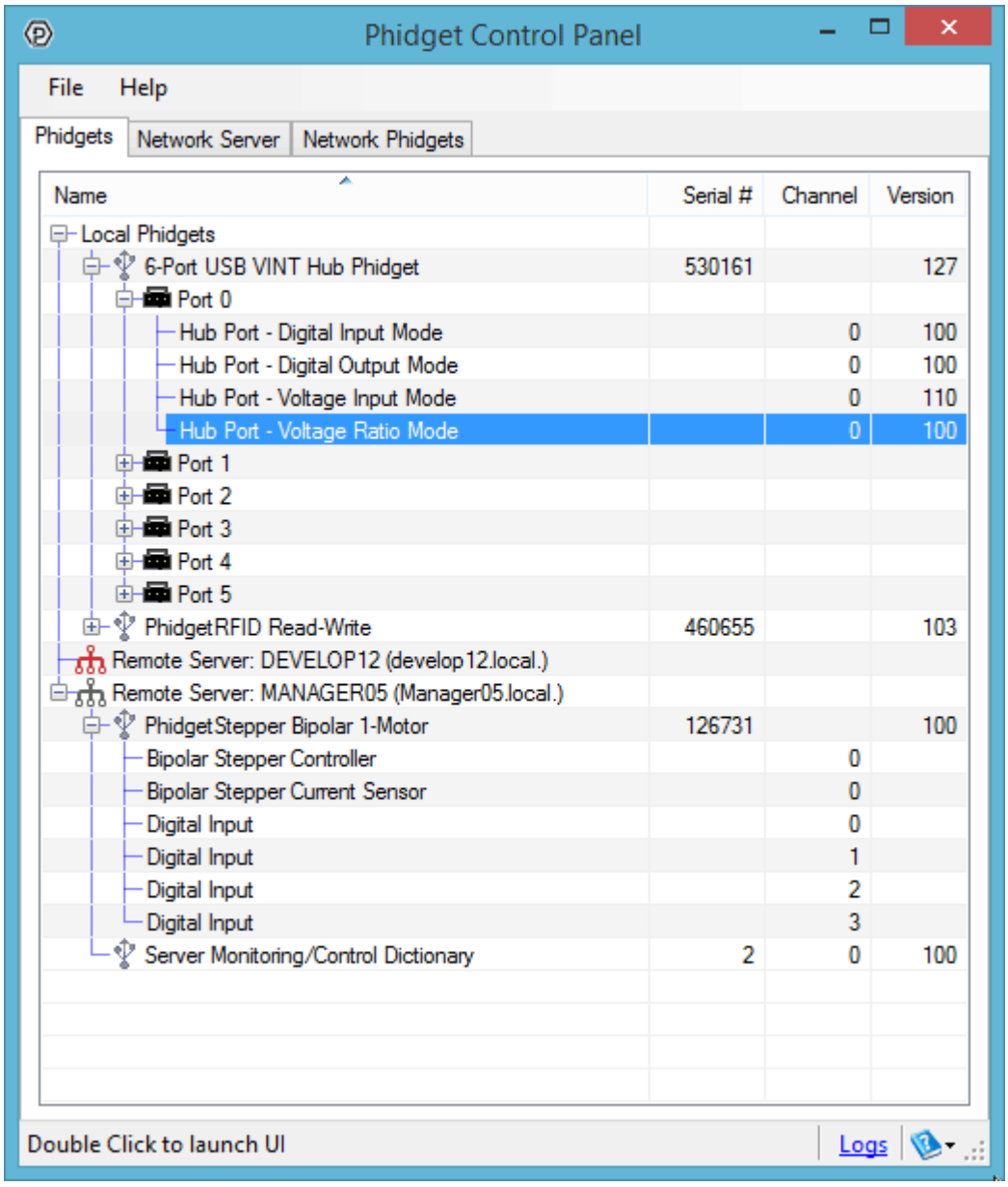
Please see the chapter Event Listener [1353] for a detailed description of its functionality and the respective Phidget device's "Event" chapter for an overview (with examples and description) over the possible events raised by each device.

7.6.14.1 Phidget Control Panel

Before connecting to any Phidget device, please install the Phidget driver first.

This driver also installs the Phidget Control Panel, which you can use to check connectivity, status and parameters of the devices.
This application also enables you to send an receive data of Phidgets on the network.

Overview

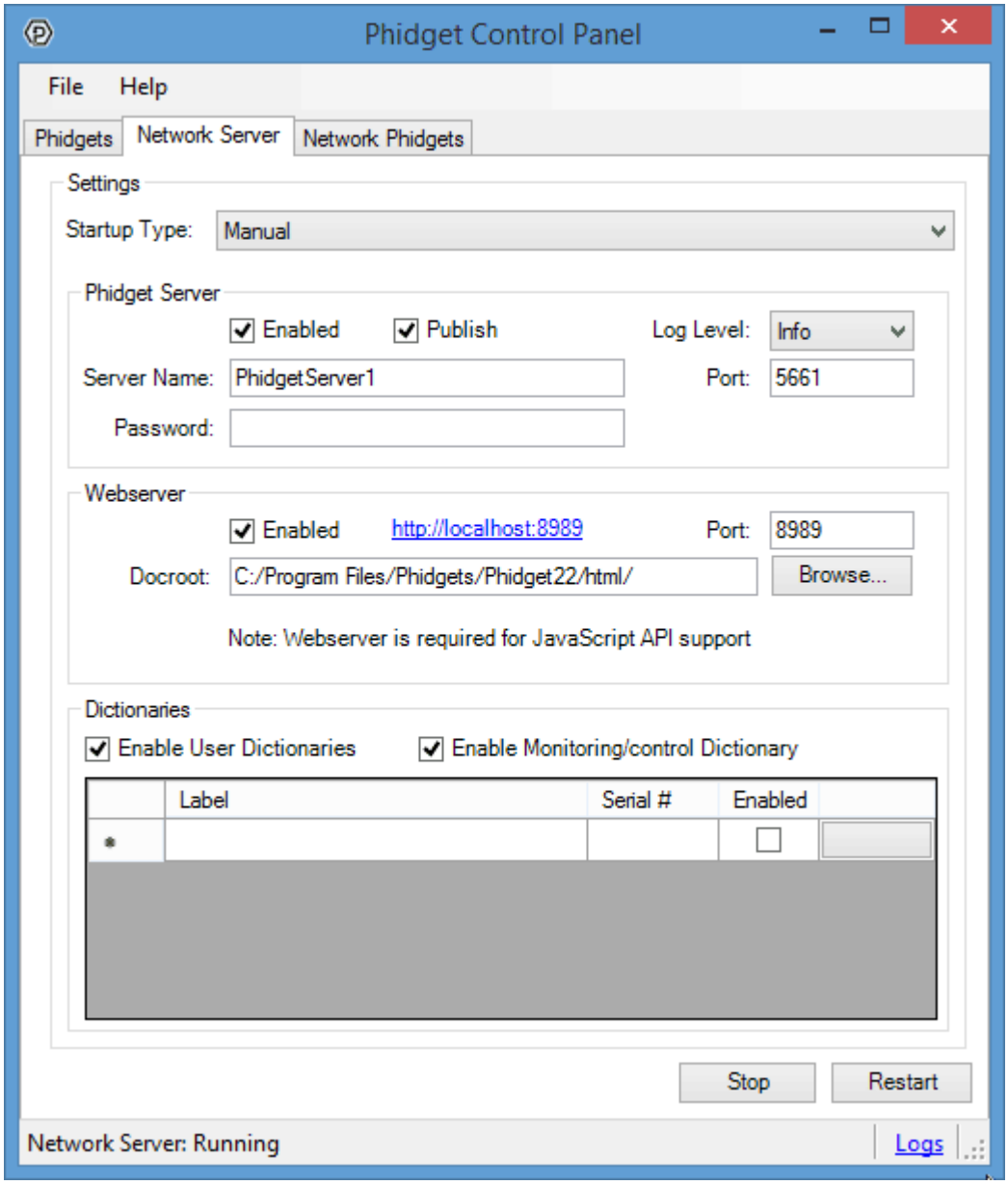


This dialog displays all locally connected devices, their type, as well as their serial number.
In this example, one VINT Hub module and one RFID Read-Write module are connected via USB to the local computer.
Each port can be expanded for more detailed information about the incoming or outgoing values.
Simply double-click on the correct mode, like "Voltage Ratio Mode", to open the port's properties and retrieve for example the addressing information.

Phidgets connected via network are listed below their respective Server device. In this example, the Server "MANAGER05" provides access to a Stepper Motor.

A second Server device, "DEVELOP12", is password protected and marked with a red icon, it will not display any further information until the user credentials were entered.

Network Server



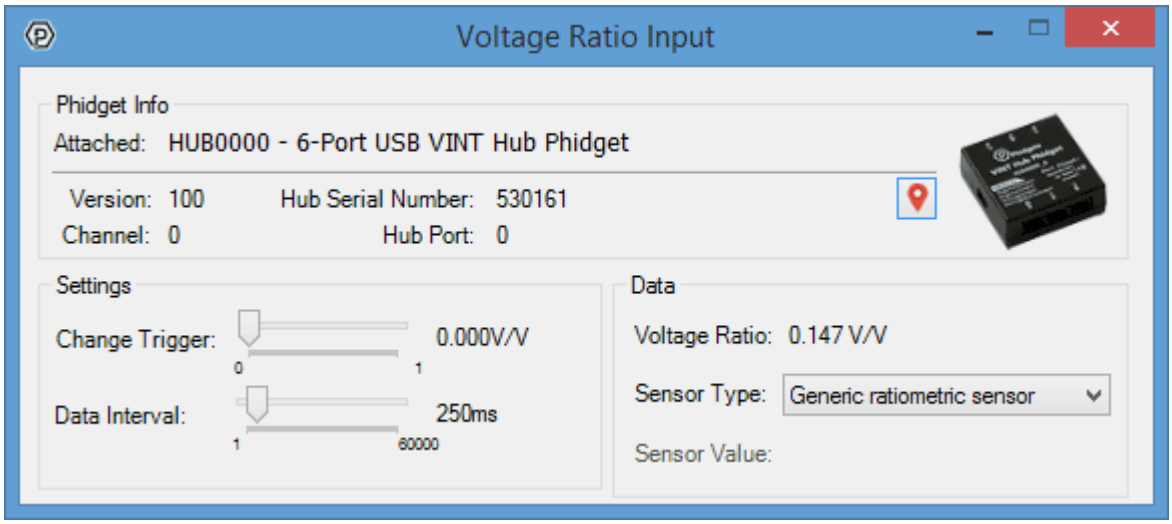
If you want to use Phidgets connected to a remote machine, the Network Server of this machine needs to be enabled.

To achieve this, go to the Network Server tab in the Control Panel and press the "Start" button at the bottom.
You can edit settings like the Server name or the port.

If you want to automatically start the Network Server when the Control Panel is started, choose a different option from the "Startup Type" drop-down.

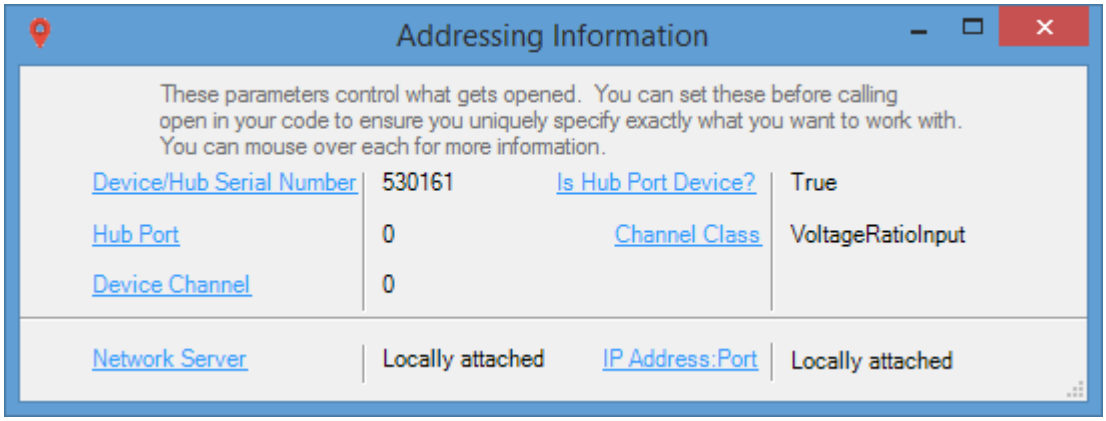
Addressing Information

Accessing certain devices, such as Voltage Input or Digital Output devices, with WD requires the additional parameters "Hub Port", "Channel" and the check box "Is Hub Port Device".
If you are not certain which values to enter here, go to the Phidget Control Panel and double-click on the device or the used port, like indicated in the example picture above on "Voltage Ratio Mode".



This configuration dialog can look different for different Phidgets models and enables you to set and retrieve certain parameters.

To retrieve the correct address information, press the red location icon to open the Addressing Information dialog.



Now simply transfer the "Hub Port", "Device Channel" and "Is Hub Port Device?" information to the respective equivalents in WD.

Important Note:

A Phidget Device can only be either connected in WD or opened and configured in the Phidgets Control Panel.

As long as the configuration dialog of the Phidget Control Panel is open, WD will not be able to connect to this device.

To enable the connectivity, close the configuration dialog.

7.6.14.2 Phidgets - Digital Input

Example: 4x Digital Input Phidget - P/N DAQ1200_0
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

A Phidgets Digital Input connects to a digital input (e.g. of an InterfaceKit) [1399] or VINT Hub port.

The depicted 4x Digital Input Phidget for example can be connected to a VINT Hub.

Adding a New Phidgets Digital Input

To add a Phidgets Digital Input device, open the Devices menu and select Phidgets Device > Digital Input Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

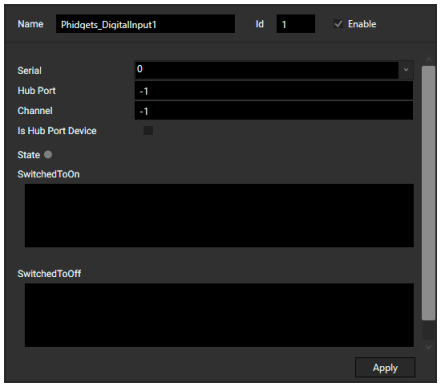
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Digital Input object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter). If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Digital Input is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Digital Input Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Digital Input Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub or InterfaceKit, use their serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub or InterfaceKit) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.
The current **State** is indicated by a circle, blue for "on", gray means "off".
If you want to execute special scripts every time when the input state changes, you can enter the commands in the **SwitchedToOn** and **SwitchedToOff** fields.

Press **Apply** to confirm your settings.

Using the Device in Regular Scripting

After adding a Phidgets Digital Input, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_DigitalInput1"), Script Assist will offer you a list of all available members.

You can for example disable the device:
```
Phidgets_DigitalInput1.Disable
```

You can also retrieve specific values via scripting in the same way as from other widgets:
```
vbool = Phidgets_DigitalInput1.DeviceSerial
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Label5.Text = "Current Input State: " + Phidgets_DigitalInput1.State
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Get..." member:
```
Label5.Text = "Current Input State: " + Phidgets_DigitalInput1.GetState
```

Please keep in mind that some read-only properties, such as the "State". Its value can only be retrieved but not set.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Digital Input Members [1390].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Digital Input Events [1389] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.2.1 Phidgets - Digital Input Events

This chapter gives an overview of the events that are raised by a Phidgets Digital Input device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Digital Input" [1388] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Digital Input Members" [1390].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa.
The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

State

Example:
Label1.Text = state

This event is raised whenever the state of the Phidgets_DigitalInput device changes.
The event returns one parameter to WD which is a Boolean value with the name "state" and holds either "True" ("On" state) or "False" ("Off" state).
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device state changes.

SwitchedToOff

Example:
Label1.Text = "YourMessage"

This event is raised whenever the state of the Phidgets_DigitalInput device changes to "Off".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever state of the Phidgets_DigitalInput1 device change to "Off".

SwitchedToOn

Example:
Label1.Text = "YourMessage"

This event is raised whenever the state of the Phidgets_DigitalInput device changes to "On".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever state of the Phidgets_DigitalInput1 device change to "On".

7.6.14.2.2 Phidgets - Digital Input Members

This chapter gives an overview of the members available for the Phidgets Digital Input device.
Please read the chapter "Phidgets - Digital Input" [1388] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Channel

Phidgets_DigitalInputID.Channel

Example:
vint = Phidgets_DigitalInput1.Channel

This member sets and returns the channel which is displayed next to "Channel" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalInput1.Channel`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalInput1.Channel)`
Example2:
Phidgets_DigitalInput1.Channel = 2

The second example shows, how to set the channel to a value of 2.

DeviceSerial

Phidgets_DigitalInputID.DeviceSerial

Example:
vint = Phidgets_DigitalInput1.DeviceSerial

This member sets and returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalInput1.DeviceSerial`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalInput1.DeviceSerial)`

Example2:
Phidgets_DigitalInput1.DeviceSerial = 530161

The second example shows, how to set the channel of the digital output 1 to a value of 530161.

Disable

Phidgets_DigitalInputID.Disable

Example:
Phidgets_DigitalInput1.Disable

This disables the Phidgets_DigitalInput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1391] member to reverse this command.

Enable

Phidgets_DigitalInputID.Enable

Example:
Phidgets_DigitalInput1.Enable

This enables the Phidgets_DigitalInput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1391] member to reverse this command.

GetState

Phidgets_DigitalInputID.Phidgets_DigitalInputID.GetState

Example:
vbool = Phidgets_DigitalInput1.GetState

This member returns the state which is displayed as an icon in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalInput1.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalInput1.GetState)`
An alternative command is: `Phidgets_DigitalInput1.State` [1392]

HubPort

Phidgets_DigitalInputID.HubPort

Example:
vint = Phidgets_DigitalInput1.HubPort

This member sets and returns the hub port number which is displayed next to "Hub Port" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalInput1.HubPort`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalInput1.HubPort)`
Example2:
Phidgets_DigitalInput1.HubPort = 2

The second example shows, how to set the hub port to a value of 2.

IsConnected

Phidgets_DigitalInputID.IsConnected

Example:
vbool = Phidgets_DigitalInput1.IsConnected

This member returns the connection status for the Phidgets_DigitalInput1 device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_DigitalInput1.IsConnected`
or simply display it in the Debug Logger: `DebugMessage(Phidgets_DigitalInput1.IsConnected)`

IsEnabled

Phidgets_DigitalInputID.IsEnabled

Example:
vbool = Phidgets_DigitalInput1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalInput1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalInput1.IsEnabled)`

IsHubPortDevice

Phidgets_DigitalInputID.IsHubPortDevice

Example:
vbool = Phidgets_DigitalInput1.IsHubPortDevice

This member returns the status of the check box "Is Hub Port Device" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is a hub port device and "False" if it is not.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalInput1.IsHubPortDevice`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_DigitalInput1.IsHubPortDevice)`

State

Phidgets_DigitalInputID.State

Example:
vbool = Phidgets_DigitalInput1.State

This member returns the state which is displayed as an icon in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalInput1.State`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalInput1.State)`
An alternative command is: `Phidgets_DigitalInput1.GetState` [1391]

7.6.14.3 Phidgets - Digital Output

Example: 4x Digital Output Phidget - P/N OUT1100_0
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

A Phidgets Digital Output connects to a digital output (e.g. of an InterfaceKit) [1399] or VINT Hub port.

The depicted 4x Digital Output Phidget for example can be connected to a VINT Hub and provides four 5V PWM-enabled digital outputs.

Adding a New Phidgets Digital Output

To add a Phidgets Digital Output device, open the Devices menu and select Phidgets Device > Digital Output Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

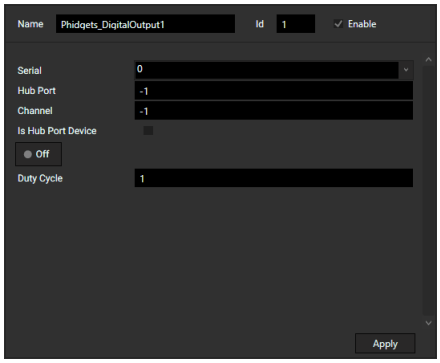
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Digital Output object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter). If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Digital Output is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Digital Output Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Digital Output Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub or InterfaceKit, use their serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub or InterfaceKit) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.
The **On / Off** button can be used to manually set the on / off state of the

Digital Output, a blue circle indicates that the current state is on, gray means off.
If you are using a device which is capable of pulse width modulation (PWM), you can also adjust the **Duty Cycle**. Enter a value between 0 and 1 to represent 0% to 100% duty cycle.

Press **Apply** to confirm your settings.

Using the Device in Regular Scripting

After adding a Phidgets Digital Output, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_DigitalOutput1"), Script Assist will offer you a list of all available members.

You can for example set the duty cycle to 75%:
```
Phidgets_DigitalOutput1.SetDutyCycle(0.75)
```

You can also retrieve specific values via scripting in the same way as from other widgets:
```
vint = Phidgets_DigitalOutput1.DeviceSerial
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Phidgets_DigitalOutput1.State = True
Label5.Text = "Current Output State: " + Phidgets_DigitalOutput1.State
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Phidgets_DigitalOutput1.SetState(True)
Label5.Text = "Current Output State: " + Phidgets_DigitalOutput1.GetState
```

Both approaches can be used interchangeably.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Digital Output Members [1395].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Digital Output Events [1394] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.3.1 Phidgets - Digital Output Events

This chapter gives an overview of the events that are raised by a Phidgets_DigitalOutput device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Digital Output" [1393] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Digital Output Members" [1395].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

7.6.14.3.2 Phidgets - Digital Output Members

This chapter gives an overview of the members available for the Phidgets Digital Output device.
Please read the chapter "Phidgets - Digital Output" [1393] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Channel

Phidgets_DigitalOutputID.Channel

Example:
vint = Phidgets_DigitalOutput1.Channel

This member sets and returns the channel which is displayed next to "Channel" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.Channel`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalOutput1.Channel)`

Example2:
Phidgets_DigitalOutput1.Channel = 2

The second example shows, how to set the channel to a value of 2.

DeviceSerial

Phidgets_DigitalOutputID.DeviceSerial

Example:
vint = Phidgets_DigitalOutput1.DeviceSerial

This member sets and returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.DeviceSerial`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_DigitalOutput1.DeviceSerial)`

Example2:
Phidgets_DigitalOutput1.DeviceSerial = 530161

The second example shows, how to set the channel of the digital output 1 to a value of 530161.

Disable

Phidgets_DigitalOutputID.Disable

Example:
Phidgets_DigitalOutput1.Disable

This disables the Phidgets_DigitalOutput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1396] member to reverse this command.

DutyCycle

Phidgets_DigitalOutputID.DutyCycle

Example:
vdouble = Phidgets_DigitalOutput1.DutyCycle

This member sets and returns the duty cycle value which is displayed next to "Duty Cycle" in the Configuration menu when selecting the according Phidget device. The value range is 0 - 1 (for 0-100%).
The first example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.DutyCycle`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalOutput1.DutyCycle)`
An alternative command is: `Phidgets_DigitalOutput1.GetDutyCycle` [1396]

Example2:
Phidgets_DigitalOutput1.DutyCycle = 0.2

The second example shows, how to set the duty cycle to a value of 0.2.

An alternative command is: `Phidgets_DigitalOutput1.SetDutyCycle` [1398]

Enable

Phidgets_DigitalOutputID.Enable

Example:
Phidgets_DigitalOutput1.Enable

This enables the Phidgets_DigitalOutput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1395] member to reverse this command.

GetDutyCycle

Phidgets_DigitalOutputID.GetDutyCycle

Example:
vdouble = Phidgets_DigitalOutput1.GetDutyCycle

This member returns the duty cycle value which is displayed next to "Duty Cycle" in the Configuration menu when selecting the according Phidget device. The value range is 0 - 1 (for 0-100%).
The first example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.GetDutyCycle`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_DigitalOutput1.GetDutyCycle)`
An alternative command is: `Phidgets_DigitalOutput1.DutyCycle` [1396]

GetState

Phidgets_DigitalOutputID.Phidgets_DigitalOutputID.GetState

Example:
vbool = Phidgets_DigitalOutput1.GetState

This member returns the state which is displayed as the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.GetState`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalOutput1.GetState)`
An alternative command is: `Phidgets_DigitalOutput1.State` [1398]

HubPort

Phidgets_DigitalOutputID.HubPort

Example:
vint = Phidgets_DigitalOutput1.HubPort

This member sets and returns the hub port number which is displayed next to "Hub Port" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.HubPort`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalOutput1.HubPort)`
Example2:
Phidgets_DigitalOutput1.HubPort = 2

The second example shows, how to set the hub port to a value of 2.

IsConnected

Phidgets_DigitalOutputID.IsConnected

Example:
vbool = Phidgets_DigitalOutput1.IsConnected

This member returns the connection status for the Phidgets_DigitalOutput1 device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_DigitalOutput1.IsConnected`
or simply display it in the Debug Logger: `DebugMessage(Phidgets_DigitalOutput1.IsConnected)`

IsEnabled

Phidgets_DigitalOutputID.IsEnabled

Example:
vbool = Phidgets_DigitalOutput1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalOutput1.IsEnabled)`

IsHubPortDevice

Phidgets_DigitalOutputID.IsHubPortDevice

Example:
vbool = Phidgets_DigitalOutput1.IsHubPortDevice

This member returns the status of the check box "Is Hub Port Device" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is a hub port device and "False" if it is not.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.IsHubPortDevice`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_DigitalOutput1.IsHubPortDevice)`

SetDutyCycle

Phidgets_DigitalOutputID.SetDutyCycle(Value)

Example:
Phidgets_DigitalOutput1.SetDutyCycle(0.2)

This member sets the duty cycle which is displayed next to "Duty Cycle" in the Configuration menu when selecting the according Phidget device. The value range is 0-1 (for 0-100%).
The first example shows how to how to set the duty cycle to a value of 0.2.

An alternative command is: `Phidgets DigitalOutput1.DutyCycle` [1396]

SetState

Phidgets_DigitalOutputID.SetState(value)

Example:
Phidgets_DigitalOutput1.SetState(False)

This member sets the state which is displayed as the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.

An alternative command is: `Phidgets DigitalOutput1.State` [1398]

State

Phidgets_DigitalOutputID.State

Example:
vbool = Phidgets_DigitalOutput1.State

This member sets and returns the state which is displayed as the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

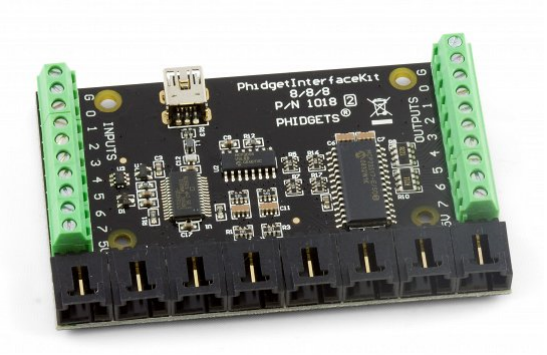
You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_DigitalOutput1.State`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_DigitalOutput1.State)`
An alternative command is: `Phidgets DigitalOutput1.GetState` [1396]

Example2:
Phidgets_DigitalOutput1.State = False

The second example shows, how to set the state to False (which is the "Off" state).

An alternative command is: Phidgets_DigitalOutput1.SetState [1398]

7.6.14.4 Phidgets - InterfaceKits



Example: Interface Kit 8/8/8 - P/N 1018_2
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

Phidgets InterfaceKits provide Digital Inputs, Digital Outputs and Voltage Channels (Analog Inputs) in varying numbers.

WD currently has three InterfaceKits natively implemented:
- InterfaceKit 8/8/8
- InterfaceKit 0/16/16
- InterfaceKit 0/0/4

If you would like to use other InterfaceKits, you can still add each In- or Output individually as single Digital Inputs [1388], Digital Outputs [1393], Voltage Inputs [1452], or Voltage Ratio Inputs [1460]. Keep in mind to enter the respective in- or output number as Channel parameter.

Adding a New Phidgets InterfaceKit

To add a Phidgets InterfaceKit device, open the Devices menu and select Phidgets Device > InterfaceKit X/X/X Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

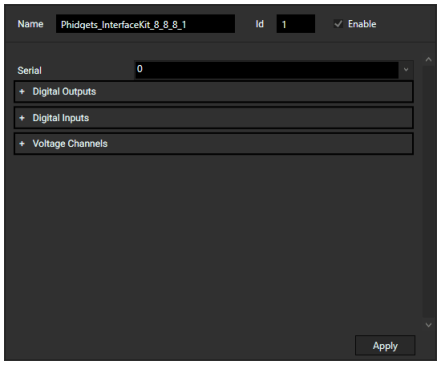
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets InterfaceKit object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter). If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets InterfaceKit is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > InterfaceKit X/X/X Device and can be opened from here or with Devices menu > Configuration.

The Phidgets InterfaceKit Settings



Select the **Serial** number of the connected device from the drop-down. If the correct serial does not show in the drop-down, make sure that the device (or Hub or InterfaceKit) is shown in the Phidget Control Panel [1385]. To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

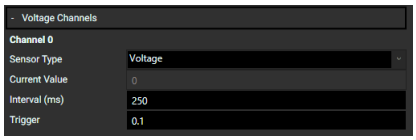
Depending device connected, you also have access to the Digital Outputs, Digital Inputs and / or Voltage Channel configuration.

Digital Outputs:
The **On / Off** button can be used to manually set the on / off state of all digital

output channels. A blue circle indicates that the current state is on, gray means off.
This option is available for all three implemented InterfaceKits, the number depends on the model used.

Digital Inputs:
The current **State** of each digital input channel is indicated by a circle, blue for "on", gray means "off".
If you want to execute special scripts every time when the input state of a channel changes, you can enter the commands in the **SwitchedToOn** and **SwitchedToOff** fields.

This option is only available for the InterfaceKits 8/8/8 and 0/16/16, the number depends on the model used.



Voltage Channels:
Select the product number of the each connected sensor from the **Sensor Type** drop-down.
You can also adjust update **Interval** time and a **Trigger** value (for sensitivity) according to your needs for every single channel.

After the settings are applied, the **Current Value** field will update as soon as a change was registered on each channel and also display the correct value units.

This option is only available for the InterfaceKit 8/8/8.

Press **Apply** to confirm your settings.

Using the Device in Regular Scripting

After adding a Phidgets InterfaceKit, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_InterfaceKit_8_8_8_1"), Script Assist will offer you a list of all available members.

You can for example set the digital output with ID 3 to its "on" state:
```
Phidgets_InterfaceKit_8_8_8_1.SetOutputState(3,True)
```

You can also retrieve specific values via scripting in the same way as from other widgets:
```
vint = Phidgets_InterfaceKit_8_8_8_1.DeviceSerial
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Phidgets_InterfaceKit_8_8_8_1.Output3.State = True
Label5.Text = "Output 3 Current State: " +
Phidgets_InterfaceKit_8_8_8_1.Output3.State
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Phidgets_InterfaceKit_8_8_8_1.Output3.SetState(True)
Label5.Text = "Output 3 Current State: " +
Phidgets_Phidgets_InterfaceKit_8_8_8_1.Output3.GetState
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. "Input.State" or "Voltage.Value", and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets InterfaceKits Members [1402].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets InterfaceKits Events [1401] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.4.1 Phidgets - InterfaceKits Events

This chapter gives an overview of the events that are raised by a Phidgets InterfaceKit device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - InterfaceKits" [1399] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - InterfaceKits Members" [1402].

AnyInputChanged

available for Phidgets_InterfaceKit_0_16_16 and Phidgets_InterfaceKit_8_8_8

Example:
Label1.Text = "Input" + AnyInputChanged + " changed to " + value2

This event is raised whenever the state of any digital input of your Phidgets InterfaceKit changes.
The event returns two parameters to WD. The first is an integer value and holds the number of the digital input that changed its state. The first input returns "0" and the last either "7" or "15" depending on the InterfaceKit.
The second parameter is a Boolean value with the name "value2" and holds either "True" ("On" state) or "False" ("Off" state).

If you select this event and copy the example into the scripting field of the Event Listener, it will write new text into the Label [888] with ID 1 whenever a digital input state changes. The Label will for example say: Input2 changed to False

IsConnected

available for all InterfaceKits, Phidgets_InterfaceKit_0_16_16 and Phidgets_InterfaceKit_8_8_8 and Phidgets_InterfaceKit_0_0_4

Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa.
The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

State

available for Phidgets_InterfaceKit_0_16_16 and Phidgets_InterfaceKit_8_8_8

Example:
Label1.Text = state

This event is raised whenever the state of the according digital input changes.
The event returns one parameter to WD which is a Boolean value with the name "state" and holds either "True" ("On" state) or "False" ("Off" state).

If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the digital input state changes.

SwitchedToOff

available for Phidgets_InterfaceKit_0_16_16 and Phidgets_InterfaceKit_8_8_8

Example:
Label1.Text = "YourMessage"

This event is raised whenever the state of the according digital input of the Phidgets InterfaceKit changes to "Off".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever state of the according digital input changes to "Off".

SwitchedToOn

available for Phidgets_InterfaceKit_0_16_16 and Phidgets_InterfaceKit_8_8_8

Example:
Label1.Text = "YourMessage"

This event is raised whenever the state of the according digital input of the Phidgets InterfaceKit changes to "On".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "YourMessage" into the Label [888] with ID 1 whenever state of the according digital input changes to "On".

Value

available for Phidgets_InterfaceKit_8_8_8

Example:
Fader4.Value = value

This event is raised whenever the according voltage input of the InterfaceKit 8/8/8 changes its value.
The event returns one parameter to WD which is a double with the name "value" and holds the current input value.
If you select the event and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value as from the according voltage input whenever it changes.

7.6.14.4.2 Phidgets - InterfaceKits Members

This chapter gives an overview of the members available for the Phidgets InterfaceKit 8/8/8. The members of the InterfaceKits 0/0/4 and 0/16/16 are named in the same way and therefore they are not listed separately. The only difference is, that there are fewer members because these Kits have less functionality. Voltage inputs, for example, are only offered by the InterfaceKit 8/8/8 Kit which therefore has members referring to them whilst the others do not offer them

Please read the chapter "Phidgets InterfaceKits" [1399] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

DeviceSerial

Phidgets_InterfaceKit_8_8_8_ID.DeviceSerial

Example:
vint = Phidgets_InterfaceKit_8_8_8_1.DeviceSerial

This member sets and returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidgets device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.DeviceSerial`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.DeviceSerial)`

Example2:
Phidgets_InterfaceKit_8_8_8_1.DeviceSerial = 530161

The second example shows, how to set the serial number of the device in the Configuration menu to 530161.

Disable

Phidgets_InterfaceKit_8_8_8_ID.Disable

Example:
Phidgets_InterfaceKit_8_8_8_1.Disable

This disables the Phidgets device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1403] member to reverse this command.

Enable

Phidgets_InterfaceKit_8_8_8_ID.Enable

Example:
Phidgets_InterfaceKit_8_8_8_1.Enable

This enables the Phidgets device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1403] member to reverse this command.

GetDeviceSerial

Phidgets_InterfaceKit_8_8_8_ID.GetDeviceSerial

Example:
vint = Phidgets_InterfaceKit_8_8_8_1.GetDeviceSerial

This member returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidgets device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetDeviceSerial`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetDeviceSerial)`

GetInputState

Phidgets_InterfaceKit_8_8_8_ID.GetInputState(inputIndex)

Example:
vbool = Phidgets_InterfaceKit_8_8_8_1.GetInputState(0)

This member returns the state of the first input (which starts with index 0!). The state is also displayed as an icon in the Configuration menu when selecting the according Phidget device. True = "On" state and False =

"Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetInputState(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetInputState(0))`

GetOutputState

Phidgets_InterfaceKit_8_8_8_ID.GetOutputState(outputIndex)

Example:
vbool = Phidgets_InterfaceKit_8_8_8_1.GetOutputState(0)

This member returns the state of the first output (which starts with index 0!). The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetOutputState(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetOutputState(0))`

GetVoltageInterval

Phidgets_InterfaceKit_8_8_8_ID.GetVoltageInterval(voltageIndex)

Example:
vint = Phidgets_InterfaceKit_8_8_8_1.GetVoltageInterval(0)

This member returns the interval time of the first voltage input (which starts with index 0!). This is also displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetVoltageInterval(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetVoltageInterval(0))`

GetVoltageSensorType

Phidgets_InterfaceKit_8_8_8_ID.GetVoltageSensorType(voltageIndex)

Example:
vstring = Phidgets_InterfaceKit_8_8_8_1.GetVoltageSensorType(0)

This member returns the sensor type of the first voltage input (which starts with index 0!). This is also displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetVoltageSensorType(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetVoltageSensorType(0))`

GetVoltageTrigger

Phidgets_InterfaceKit_8_8_8_ID.GetVoltageTrigger(voltageIndex)

Example:
vdouble= Phidgets_InterfaceKit_8_8_8_1.GetVoltageTrigger(0)

This member returns the trigger value of the first voltage input (which starts with index 0!). This is also displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetVoltageTrigger(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetVoltageTrigger(0))`

GetVoltageUnit

Phidgets_InterfaceKit_8_8_8_ID.GetVoltageUnit(voltageIndex)

Example:
vstring = Phidgets_InterfaceKit_8_8_8_1.GetVoltageUnit(0)

This member returns the unit, e.g. "v" for "Volts" of the first voltage input (which starts with index 0!).
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetVoltageUnit(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetVoltageUnit(0))`

GetVoltageValue

Phidgets_InterfaceKit_8_8_8_ID.GetVoltageValue(voltageIndex)

Example:
vdouble = Phidgets_InterfaceKit_8_8_8_1.GetVoltageValue(0)

This member returns the value of the first voltage input (which starts with index 0!). This is also displayed next to "Current Value" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetVoltageValue(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetVoltageValue(0))`

Input0.GetState

Phidgets_InterfaceKit_8_8_8_ID.InputID.GetState

Example:
vbool = Phidgets_InterfaceKit_8_8_8_1.Input0.GetState

This member returns the state of the first input (which starts with index 0!). The state is also displayed as an icon in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Input0.GetState`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Input0.GetState)`
An alternative command is `Phidgets_InterfaceKit_8_8_8_ID.GetInputState(inputIndex)` [1403] which allows to address the input dynamically.

Input0.State

Phidgets_InterfaceKit_8_8_8_ID.InputID.State

Example:
vbool = Phidgets_InterfaceKit_8_8_8_1.Input0.State

This member returns the state of the first input (which starts with index 0!). The state is also displayed as an icon in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Input0.State`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Input0.State)`
An alternative command is: `Phidgets_InterfaceKit_8_8_8_ID.GetInputState(inputIndex)` [1403] which allows to address the input dynamically.

isConnected

Phidgets_InterfaceKit_8_8_8_ID.isConnected

Example:
vbool = Phidgets_InterfaceKit_8_8_8_1.isConnected

This member returns the connection status for the Phidgets device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.IsConnected`
or simply display it in the Debug Logger:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.IsConnected)`

isEnabled

Phidgets_InterfaceKit_8_8_8_ID.isEnabled

Example:
Phidgets_InterfaceKit_8_8_8_1.isEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidgets device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.IsEnabled`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.IsEnabled)`

Output0.GetState

Phidgets_InterfaceKit_8_8_8_ID.OutputID.GetState

Example:
vbool = Phidgets_InterfaceKit_8_8_8_1.Output0.GetState

This member returns the state of the first output (which starts with index 0!). The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Output0.GetState`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Output0.GetState)`
An alternative command is `Phidgets_InterfaceKit_8_8_8_ID.GetOutputState(outputIndex)` [1404] which allows to address the output dynamically.

Output0.SetState

Phidgets_InterfaceKit_8_8_8_ID.OutputID.SetState(value)

Example:
Phidgets_InterfaceKit_8_8_8_1.Output0.SetState(True)

This member sets the state of the first output (which starts with index 0!). The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.

An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.SetOutputState(outputIndex,value)` [1408] which allows to address the output dynamically.

Output0.State

Phidgets_InterfaceKit_8_8_8_ID.OutputID.State

Example:
vbool = Phidgets_InterfaceKit_8_8_8_1.Output0.State

This member sets and returns the state of the first output (which starts with index 0!). The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.
The first example shows how to return its result to an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Output0.State`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Output0.State)`
An alternative command is `Phidgets_InterfaceKit_8_8_8_ID.GetOutputState(outputIndex)` [1404] which allows to address the output dynamically.

Example2:
Phidgets_InterfaceKit_8_8_8_1.Output0.State = False

The second example shows, how to set the state to False (which is the "Off" state).

An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.SetOutputState(outputIndex,value)` [1408] which allows to address the output dynamically.

SetAllOutputsOff

Phidgets_InterfaceKit_8_8_8_ID.SetAllOutputsOff

Example:
Phidgets_InterfaceKit_8_8_8_1.SetAllOutputsOff

This member sets the state of all outputs to off. The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.

SetAllOutputsOn

Phidgets_InterfaceKit_8_8_8_ID.SetAllOutputsOn

Example:
Phidgets_InterfaceKit_8_8_8_1.SetAllOutputsOn

This member sets the state of all outputs to on. The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.

SetAllOutputStates

Phidgets_InterfaceKit_8_8_8_ID.SetAllOutputStates(state)

Example:
Phidgets_InterfaceKit_8_8_8_1.SetAllOutputStates(True)

This member sets the state of all outputs. The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.

SetDeviceSerial

Phidgets_InterfaceKit_8_8_8_ID.SetDeviceSerial(value)

Example:
Phidgets_InterfaceKit_8_8_8_1.SetDeviceSerial(530161)

This member sets the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidgets device.

SetOutputState

Phidgets_InterfaceKit_8_8_8_ID.SetOutputState(outputIndex,value)

Example:
Phidgets_InterfaceKit_8_8_8_1.SetOutputState(0,True)

This member sets the state of the first output (which starts with index 0!). The state is displayed with the On/Off button in the Configuration menu when selecting the according Phidget device. True = "On" state and False = "Off" state.

SetVoltageInterval

Phidgets_InterfaceKit_8_8_8_ID.SetVoltageInterval(voltageIndex,value)

Example:
Phidgets_InterfaceKit_8_8_8_1.SetVoltageInterval(0,500)

This member sets the interval time of the first voltage input (which starts with index 0!) to 500ms. This is also displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.

SetVoltageSensorType

Phidgets_InterfaceKit_8_8_8_ID.SetVoltageSensorType(voltageIndex,value)

Example:
Phidgets_InterfaceKit_8_8_8_1.SetVoltageSensorType(0,"PN_1102")

This member sets the sensor type of the first voltage input (which starts with index 0!) to "PN_1102". This is also displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.

SetVoltageTrigger

Phidgets_InterfaceKit_8_8_8_ID.SetVoltageTrigger(voltageIndex,value)

Example:
Phidgets_InterfaceKit_8_8_8_1.SetVoltageTrigger(0,0.2)

This member sets the trigger value of the first voltage input (which starts with index 0!) to 0.2. This is also displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.

Voltage0.GetInterval

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.GetInterval

Example:
vint = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetInterval

This member returns the interval time of the first voltage input (which starts with index 0!). This is also displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_ID.VoltageID.GetInterval`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_ID.VoltageID.GetInterval)`
An alternative command is `Phidgets_InterfaceKit_8_8_8_ID.GetVoltageInterval(voltageIndex)` [1404] which allows to address the output dynamically.

Voltage0.GetSensorType

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.GetSensorType

Example:
vstring = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetSensorType

This member returns the sensor type of the first voltage input (which starts with index 0!). This is also displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetSensorType`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetSensorType)`

An alternative command is
Phidgets_InterfaceKit_8_8_8_ID.GetVoltageSensorType(voltageIndex) [1404] which allows to address the output dynamically.

VoltageID.GetTrigger

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.GetTrigger

Example:
vdouble = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetTrigger

This member returns the trigger value of the first voltage input (which starts with index 0!). This is also displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetTrigger`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetTrigger)`
An alternative command is
Phidgets_InterfaceKit_8_8_8_ID.GetVoltageTrigger(voltageIndex) [1404] which allows to address the output dynamically.

VoltageID.GetUnit

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.GetUnit

Example:
vstring = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetUnit

This member returns the unit, e.g. "v" for "Volts" of the first voltage input (which starts with index 0!).
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.GetVoltageUnit(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.GetVoltageUnit(0))`
An alternative command is Phidgets_InterfaceKit_8_8_8_ID.GetVoltageUnit(voltageIndex) [1405] which allows to address the output dynamically.

VoltageID.GetValue

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.GetValue

Example:
vdouble = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetValue

This member returns the value of the first voltage input (which starts with index 0!). This is also displayed next to "Current Value" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetValue`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Voltage0.GetValue)`
An alternative command is
Phidgets_InterfaceKit_8_8_8_ID.GetVoltageValue(voltageIndex) [1405] which allows to address the output dynamically.

VoltageID.Interval

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.Interval

Example:
vint = Phidgets_InterfaceKit_8_8_8_1.Voltage0.Interval

This member sets and returns the interval time of the first voltage input (which starts with index 0!). This is also displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_ID.VoltageID.Interval`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_ID.VoltageID.Interval)`
An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.GetVoltageInterval(voltageIndex)` [1404] which allows to address the output dynamically.

Example2:
Phidgets_InterfaceKit_8_8_8_1.Voltage0.Interval = 500

The second example sets the interval time to 500ms.

An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.SetVoltageInterval(voltageIndex,value)` [1408] which allows to address the output dynamically.

VoltageID.SensorType

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.SensorType

Example:
vstring = Phidgets_InterfaceKit_8_8_8_1.Voltage0.SensorType

This member sets and returns the sensor type of the first voltage input (which starts with index 0!). This is also displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Voltage0.SensorType`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Voltage0.SensorType)`
An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.GetVoltageSensorType(voltageIndex)` [1404] which allows to address the output dynamically.

Example2:
Phidgets_InterfaceKit_8_8_8_1.Voltage0.SensorType = "PN_1102"

The second example sets the sensor type to "PN_1102".

An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.SetVoltageSensorType(voltageIndex,value)` [1409] which allows to address the output dynamically.

VoltageID.SetInterval

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.SetInterval(value)

Example:
Phidgets_InterfaceKit_8_8_8_1.Voltage0.SetInterval(500)

This member sets the interval time of the first voltage input (which starts with index 0!) to 500ms. This is also displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.

An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.SetVoltageInterval(voltageIndex,value)` [1408] which allows to address the output dynamically.

VoltageID.SetSensorType

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.SetSensorType

Example:
Phidgets_InterfaceKit_8_8_8_1.Voltage0.SetSensorType("PN_1102")

This member sets the sensor type of the first voltage input (which starts with index 0!) to "PN_1102". This is also displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.

An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.SetVoltageSensorType(voltageIndex,value)` [1409] which allows to address the output dynamically.

VoltageID.SetTrigger

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.SetTrigger

Example:
Phidgets_InterfaceKit_8_8_8_1.Voltage0.SetTrigger(0.2)

This member sets the trigger value of the first voltage input (which starts with index 0!) to 0.2. This is also displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.

An alternative command is
`Phidgets_InterfaceKit_8_8_8_ID.SetVoltageTrigger(voltageIndex,value)` [1409] which allows to address the output dynamically.

VoltageID.Trigger

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.Trigger

Example:
vdouble = Phidgets_InterfaceKit_8_8_8_1.Voltage0.Trigger

This member sets and returns the trigger value of the first voltage input (which starts with index 0!). This is also displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Voltage0.Trigger`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Voltage0.Trigger)`

An alternative command is
Phidgets_InterfaceKit_8_8_8_ID.GetVoltageTrigger(voltageIndex) [1404] which allows to address the output dynamically.

Example2:
Phidgets_InterfaceKit_8_8_8_1.Voltage0.Trigger = 0.2

The second example sets the trigger value to 0.2.

An alternative command is
Phidgets_InterfaceKit_8_8_8_ID.SetVoltageTrigger(voltageIndex,value) [1409] which allows to address the output dynamically.

VoltageID.Unit

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.Unit

Example:
vstring = Phidgets_InterfaceKit_8_8_8_1.Voltage0.Unit

This member returns the unit, e.g. "v" for "Volts" of the first voltage input (which starts with index 0!).
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.VoltageUnit(0)`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.VoltageUnit(0))`
An alternative command is Phidgets_InterfaceKit_8_8_8_ID.GetVoltageUnit(voltageIndex) [1405] which allows to address the output dynamically.

VoltageID.Value

Phidgets_InterfaceKit_8_8_8_ID.VoltageID.Value

Example:
vdouble = Phidgets_InterfaceKit_8_8_8_1.Voltage0.Value

This member returns the value of the first voltage input (which starts with index 0!). This is also displayed next to "Current Value" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_InterfaceKit_8_8_8_1.Voltage0.Value`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_InterfaceKit_8_8_8_1.Voltage0.Value)`
An alternative command is
Phidgets_InterfaceKit_8_8_8_ID.GetVoltageValue(voltageIndex) [1405] which allows to address the output dynamically.

7.6.14.5 Phidgets - PhidgetIR

PhidgetIR - P/N 1055_0B
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

The Phidgets IR device is capable of receiving as well as sending IR code, like for example emitted by remote controls.

It can be used to either program actions triggered by external IR signals, or to access functions of IR controllable devices such as projectors or BlueRay players.

Adding a New Phidgets IR

To add a Phidgets IR device, open the Devices menu and select Phidgets > PhidgetIR. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

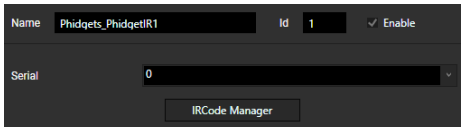
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this PhidgetIR object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your PhidgetIR is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets > PhidgetIR and can be opened from here or with Devices menu > Configuration.

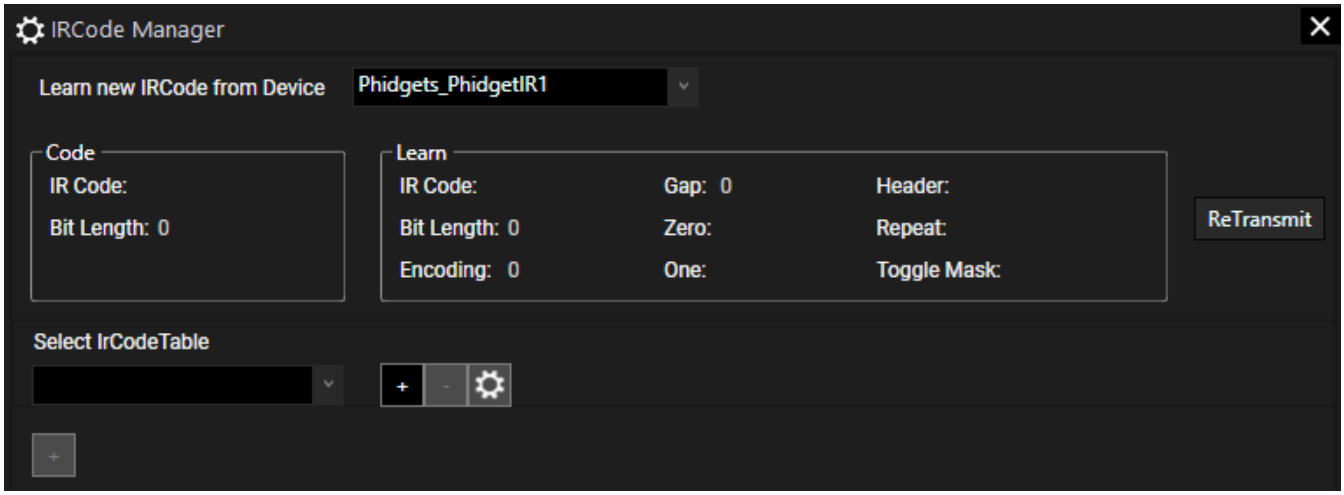
The Phidgets IR Settings



Select the **Serial** number of the connected device from the drop-down. If the correct serial does not show in the drop-down, make sure that the device is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Press **Apply** to connect your device and open the IRCode Manager for reading and storing received IR signals.

The IRCode Manager



If you have multiple IR devices set up, you can select your current device from the **topmost drop-down**.

As soon as a signal is received by the device, its IR code word and bit length are displayed in the **Code** section. The **Learn** section displays the IR code, broken down by its components.
To make sure that the data of the latest code word is received completely, please make sure to send the signal continuously (e.g. by keeping the remote control button pressed) until the **IR Codes** of the Code and the Learn section match.

If you want to test the last received IR code, press the ReTransmit button to send it from the device's IR diodes.

The next step is to add the learned code to an IR Code Table [1418].

Use the drop-down to select an existing code table or press the "**+**" button on the right side to create a new table. If you want to delete the currently selected code table, press the "**-**" button, if you want to rename it, press the **gear icon**.

Pressing the "**+**" button below the drop-down will add the current code to the selected table.
You can assign a descriptive **Alias** to this code, like "Play" or "Shutter".
The Alias must abide by the general naming rules for objects (only letters, numbers and underscore are allowed, the first symbol must be a letter) and must be unique within this specific code table.

Remove learned codes from the table by pressing the "**-**" button to the left of the Alias.

Using the Device in Regular Scripting

After adding a PhidgetIR device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_PhidgetIR1"), Script Assist will offer you a list of all available members.

An important member of this device is the Send method, which is used to send a specific code from the device's IR diodes:

```
Phidgets_PhidgetIR1.Send("Phidgets_IRCodeTable1","Play",10)
```

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets IR Members [1416].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets IR Events [1416] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.5.1 Phidgets - PhidgetIR Events

This chapter gives an overview of the events that are raised by a PhidgetIR device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - PhidgetIR" [1414] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - PhidgetIR Members" [1416].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

CodeReceived

Example:
Label1.Text = CodeReceived

This event is raised whenever the Phidgets device receives an IR code. If the IRCode Manager dialog [1415] is open, it would displays the code with all parameters.
The event returns one parameter to WD which is a string with the name "CodeReceived" and holds the received code, e.g. .04fbd827
If you select the event and copy the example into the scripting field of the Event Listener, it will write the code into the Label [888] with ID 1 whenever a new one is received.

7.6.14.5.2 Phidgets - PhidgetIR Members

This chapter gives an overview of the members available for the PhidgetIR device.
Please read the chapter "Phidgets - PhidgetIR" [1414] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

DeviceSerial

Phidgets_PhidgetIRID.DeviceSerial

Example:
vint = Phidgets_PhidgetIR1.DeviceSerial

This member sets and returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidgets device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_PhidgetIR1.DeviceSerial`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_PhidgetIR1.DeviceSerial)`
Example2:
Phidgets_PhidgetIR1.DeviceSerial = 530161

The second example shows, how to set the serial number of the device in the Configuration menu to 530161.

Disable

Phidgets_PhidgetIRID.Disable

Example:
Phidgets_PhidgetIR1.Disable

This disables the Phidgets device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1417] member to reverse this command.

Enable

Phidgets_PhidgetIRID.Enable

Example:
Phidgets_PhidgetIR1.Enable

This enables the Phidgets device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1417] member to reverse this command.

IsConnected

Phidgets_PhidgetIRID.IsConnected

Example:
vbool = Phidgets_PhidgetIR1.IsConnected

This member returns the connection status for the Phidgets device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_PhidgetIR1.IsConnected`
or simply display it in the Debug Logger: `DebugMessage(Phidgets_PhidgetIR1.IsConnected)`

IsEnabled

Phidgets_PhidgetIRID.IsEnabled

Example:
vbool = Phidgets_PhidgetIR1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidgets device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_PhidgetIR1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_PhidgetIR1.IsEnabled)`

Send

Phidgets_PhidgetIRID.Send(table,Alias,repeatCount)

Example:
Phidgets_PhidgetIR1.Send("Phidgets_IRCodeTable1","Play",10)

The Send member is used to send a learned code out via the device's IR diodes. The example refers to the IR Code Table [1418] named "Phidgets_IRCodeTable1" and a code named "Play" which is saved there. Both names are suggested by the ScriptAssistant. According to the repeatCount, the code is sent out 10 times.

If you like to send an IR code that has not been stored in a code table, you can use the `Phidgets PhidgetIR1.SendRaw` [1418] command.

Another possibility to send a learned code is using the member of the table: `Phidgets IRCodeTable1.Send` [1422]

SendRaw

Phidgets_PhidgetIRID.SendRaw(data,carrierFrequency,dutyCycle,gap)

Example:
Phidgets_PhidgetIR1.SendRaw([list with integer values],int,double,int)

The SendRaw member is used to send a raw code out via the device's IR diodes. Please refer to the Phidgets SDK for more information, e.g. PhidgetIR - 1055_0B

Note that the alternative member `Phidgets PhidgetIR1.Send` [1418] refers to an IR Code Table [1418] and previously learned codes, hence there is no need to enter the code parameters manually.

7.6.14.6 Phidgets - IR Code Table

The Phidgets IR Code Table is a specialized **Scripting** object, similar to an Event Listener [1353], used in combination with the PhidgetIR [1414] device.

A Code Table contains a selection of stored IR codes and allows managing these codes, as well as adding scripts to be triggered when a specific code is received.

Codes can be learned from a selected IR device and stored in a Code Table from the IRCode Manager [1415].

Adding a New Phidgets IR Code Table

To add a Phidgets IR Code Table, open the Devices menu and select Phidgets > IR Code Table. This opens the Configuration dialog [1305]. Alternatively, you can add a new table in the Configuration dialog with the "+" button.

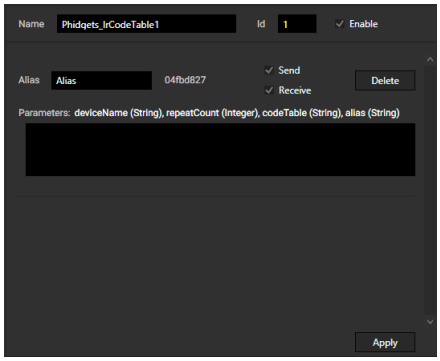
On the right side, you can name, dis-/enable the table or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets IR Code Table object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter). If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets IR Code Table is a filled blue circle, which means that the table is enabled. A filled gray circle indicates a disabled table.

You can close the dialog at any time. The newly created table will also be added to the Devices menu > Phidgets > IR Code Table and can be opened from here or with Devices menu > Configuration.

The Phidgets IR Code Table Settings



Each learned IR code is displayed in this dialog with its Alias and the code word.
You can change the **Alias**, but it needs to be unique within the selected Code Table.

Use the **Delete** button to remove the code word from this table.

The **scripting field** can be filled with commands that should be executed when this code is received.
You can also use the offered **Parameters** as local variables, e.g. to indicate from where the command was triggered: `Label1.Text = alias + " command issued from: " + deviceName`

If you need to, you can also determine, whether this code can be only **Sent** or only **Received**.
Unchecking the Receive box for example will prevent the script from being executed when the code is received.

Press **Apply** to confirm your settings.

Using the IR Code Table in Regular Scripting

After adding a Phidgets IR Code Table, you can also use it via scripting which allows to perform actions on the Code Table object as well as retrieve information from it.

To use the table as scripting object, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the table's identifier name into the script field (per default that is e.g. "Phidgets_IRCodeTable1"), Script Assist will offer you a list of all available members.

An important member of this object is the Send method, which is used to send a code stored in this table with a specified Phidgets IR device's IR diodes:
```
Phidgets_IRCodeTable1.Send("Play","Phidgets_Ir1",10)
```

The Send and Receive check boxes can be ticked or unticked via scripting. This can be especially useful when the execution is supposed to be blocked due to a certain condition:
```
Phidgets_IRCodeTable1.DisableReceive("Play")
```

A complete list of all stored codes can be retrieved with the GetAliasNames member, for example to be displayed in a DropDownList [868]:
```
DropDownList1.SetItemsFromArray(Phidgets_IRCodeTable1.GetAliasNames)
```

The IR code word associated with a certain alias can be read out as well:
```
vstring = Phidgets_IRCodeTable1.IRCode("Play")
```

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets IR Members [1416].

7.6.14.6.1 Phidgets - IR Code Table members

This chapter gives an overview of the members available for the Phidgets IR Code Table.
Please read the chapter "Phidgets - IR Code Table" [1418] if you like to know how to add and use it. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Disable

Phidgets_IRCodeTableID.Disable

Example:
Phidgets_IRCodeTable1.Disable

This disables the according Phidgets IR Code Table in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1420] member to reverse this command.

DisableReceive

Phidgets_IRCodeTableID.DisableReceive(alias)

Example:
Phidgets_IRCodeTable1.DisableReceive("Play")

This refers to the code named "Play" in the according Phidgets IR Code Table and disables the execution from the commands that were entered in the scripting field in the Configuration dialog [1305]. Accordingly, the "Receive" option is unchecked. Use the `.EnableReceive` [1420] member to reverse this command.

DisableSend

Phidgets_IRCodeTableID.DisableSend(alias)

Example:
Phidgets_IRCodeTable1.DisableSend("Play")

This refers to the code named "Play" in the according Phidgets IR Code Table and disables the option to send this code out. Accordingly, the "Send" option in the Configuration dialog [1305] is unchecked. Use the `.EnableSend` [1420] member to reverse this command.

Enable

Phidgets_IRCodeTableID.Enable

Example:
Phidgets_IRCodeTable1.Enable

This enables the according Phidgets IR Code Table in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1419] member to reverse this command.

EnableReceive

Phidgets_IRCodeTableID.EnableReceive(alias)

Example:
Phidgets_IRCodeTable1.EnableReceive("Play")

This refers to the code named "Play" in the according Phidgets IR Code Table and enables the execution from the commands that were entered in the scripting field in the Configuration dialog [1305]. Accordingly, the "Receive" option is checked. Use the `.DisableReceive` [1420] member to reverse this command.

EnableSend

Phidgets_IRCodeTableID.EnableSend(alias)

Example:
Phidgets_IRCodeTable1.EnableSend("Play")

This refers to the code named "Play" in the according Phidgets IR Code Table and enables the option to send this code out. Accordingly, the "Send" option in the Configuration dialog [1305] is checked. Use the `.DisableSend` [1420] member to reverse this command.

GetAliasNames

Phidgets_IRCodeTableID.GetAliasNames

Example:
vlist = Phidgets_IRCodeTable1.GetAliasNames

This member returns the alias names of all learned codes from the according Phidgets IR Code Table as a list. The result could look as follows: ["Play", "Pause", "Stop", "Home"]
The first example shows how to use it with an existing (global) list variable [1900] named "vlist".

You could also write it into a Widget, e.g. a DropDown List [868] via the script and a local variable [1903] "list":
`DropDownList1.SetItemsFromArray(Phidgets_IRCodeTable1.GetAliasNames)`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_IRCodeTable1.GetAliasNames)`

IRCode

Phidgets_IRCodeTableID.IRCode(alias)

Example:
vstring = Phidgets_IRCodeTable1.IRCode("Play")

This returns the IR code word associated with the learned code named "Play" from the according Phidgets IR Code Table.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_IRCodeTable1.IRCode("Play")`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_IRCodeTable1.IRCode("Play"))`

IsEnabled

Phidgets_IRCodeTableID.IsEnabled

Example:
vbool = Phidgets_IRCodeTable1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according IR Code Table. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_IRCodeTable1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_IRCodeTable1.IsEnabled)`

isReceiveEnabled

Phidgets_IRCodeTableID.isReceiveEnabled(alias)

Example:
vbool = Phidgets_IRCodeTable1.isReceiveEnabled("Play")

This refers to the code named "Play" in the according Phidgets IR Code Table and returns the status of the check box "Receive". The result, a Boolean value (or string), is "True" if the option is enabled and "False" if it is currently not enabled which means that the commands that were entered in the scripting field in the Configuration dialog [1305] would not be executed.

The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_IRCodeTable1.isReceiveEnabled("Play")`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_IRCodeTable1.isReceiveEnabled("Play"))`

isSendEnabled

Phidgets_IRCodeTableID.isSendEnabled(alias)

Example:
vbool = Phidgets_IRCodeTable1.isSendEnabled(keyword)

This refers to the code named "Play" in the according Phidgets IR Code Table and returns the status of the check box "Send". The result, a Boolean value (or string), is "True" if the option is enabled and "False" if it is currently not enabled which means that the code cannot be sent out.

The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_IRCodeTable1.isSendEnabled("Play")`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_IRCodeTable1.isSendEnabled("Play"))`

Send

Phidgets_IRCodeTableID.Send(alias,device,repeatCount)

Example:
Phidgets_IRCodeTable1.Send("Play","Phidgets_PhidgetIR1",10)

This refers to the learned code named "Play" which is stored in the according Phidgets IR Code Table. The device named "Phidgets_PhidgetIR1" sends the code 10 times out via its IR diodes

Alternatively, you can also send a code via the command: `Phidgets PhidgetIR1.Send` [1417]

7.6.14.7 Phidgets - PhidgetRFID

PhidgetRFID Read-Write - P/N 1024_0B
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

The Phidgets RFID device reads RFID tags that are brought in close proximity to the reader and returns the tag identification number. Writing data to T5577 tags is also supported.

Adding a New Phidgets RFID

To add a Phidgets RFID device, open the Devices menu and select Phidgets Device > RFID Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

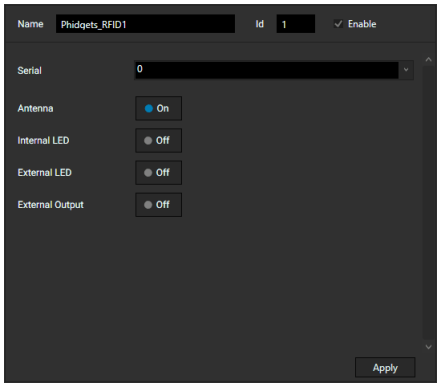
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets RFID object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets RFID is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > RFID Device and can be opened from here or with Devices menu > Configuration.

The Phidgets RFID Settings



Select the **Serial** number of the connected device from the drop-down.
If the correct serial does not show in the drop-down, make sure that the device is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

There are four **On / Off** buttons to manually set the on / off state of the **Antenna**, **Internal LED**, **External LED** and **External digital Output** of the RFID reader.
A blue circle indicates that the current state is on, gray means off.

Press **Apply** to confirm your settings.

Using the Device in Regular Scripting

After adding a Phidgets RFID, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_RFID1"), Script Assist will offer you a list of all available members.

The RFID Phidget can for example also write strings to tags (if they are rewritable):

```
Phidgets_RFID1.WriteTag("test","PhidgetTAG")
```

When you write a new string on a tag, keep in mind to follow the tag string requirements of the protocol you are using.

You can also retrieve specific values via scripting in the same way as from other widgets:

```
vstring = Phidgets_RFID1.TagString
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:

```
Phidgets_RFID1.ExternalLedEnabled = True
Label5.Text = "Current LED State: " + Phidgets_RFID1.ExternalLedEnabled
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:

```
Phidgets_RFID1.SetExternalLedEnabled(True)
Label5.Text = "Current LED State: " + Phidgets_RFID1.GetExternalLedEnabled
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. "TagString", and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets RFID Members [1425].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets RFID Events [1424] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.7.1 Phidgets - PhidgetRFID Events

This chapter gives an overview of the events that are raised by a PhidgetRFID device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - RFID" [1422] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - RFID Members" [1425].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

TagEnter

Example:
Label1.Text = tag

This event is raised whenever the Phidgets device registers a tag entering its RF filed.
The event returns two parameters to WD. The first, "tag", contains the identifier code written on the tag itself as string. As the tags can be configured with different protocols, EM4100, ISO11785 FDX-B and PhidgetTag, the second parameter "protocol" returns the name of the protocol the current tag is encoded with.
If you select the event and copy the example into the scripting field of the Event Listener, it will write the tag code into the Label [888] with ID 1 whenever a new one is detected.

TagLeave

Example:
Label1.Text = tag

This event is raised whenever the Phidgets device registers a tag leaving its RF filed.
The event returns two parameters to WD. The first, "tag", contains the identifier code written on the tag itself as string. As the tags can be configured with different protocols, EM4100, ISO11785 FDX-B and PhidgetTag, the second parameter "protocol" returns the name of the protocol the current tag is encoded with.
If you select the event and copy the example into the scripting field of the Event Listener, it will write the tag code into the Label [888] with ID 1 whenever a leaving tag is detected.

7.6.14.7.2 Phidgets - PhidgetRFID Members

This page is still under construction.

Please check out our Online Manual for updates!

7.6.14.8 Phidgets - Server

The Phidgets Server enables you to access Phidget devices that are connected not to your local machine, but to a machine in the network.
If you want to make use of Phidgets via network, the respective remote machine needs to be added to your project as Phidget Server device .

The Network Server [1386] must be enabled in the Phidget Control Panel of the remote machine, please make sure that it is reachable via your local Phidget Control Panel.

Adding a New Phidgets Server

To add a Phidget Server device, open the Devices menu and select Phidgets Device > Server Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

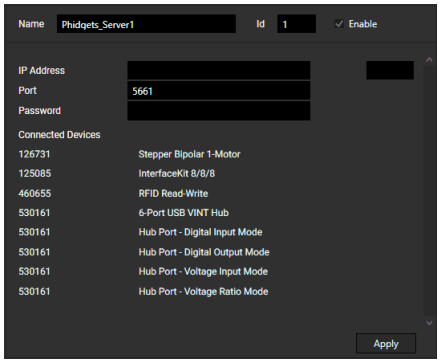
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Server object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Server is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Server Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Server Settings



To connect to a Phidgets Server, simply enter the remote **IP Address** and correct **Port**. There is no need to pick a specific network adapter as WD assigns one automatically based on the IP.
The standard port is 5661, but you may select any free port in the Network Server settings of the remote machine.

If you have set up a **Password** to access the remote Phidgets, you need to enter it here as well.
Press **Apply** to confirm your settings.

After the connection is established, all Phidget devices connected to the remote machine are listed in this dialog.
The list is updated automatically as soon as a device is physically connected or disconnected.

Using the Device in Regular Scripting

After adding a Phidget Server, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_Server1"), Script Assist will offer you a list of all available members.

You can for example en- or disable the device via scripting:

```
Phidgets_Server1.Enable
```

It also offers you the possibility to retrieve a list of the serial numbers of the connected device and write this to a global or local list variable:
`vlist = Phidgets_Server1.GetSerialList`
This enables you for example to check whether new devices were added or if one was disconnected.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Server Members [1426].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Server Events [1426] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.8.1 Phidgets - Server Events

This chapter gives an overview of the events that are raised by a Phidgets Server device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Server" [1425] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Server Members" [1426].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

7.6.14.8.2 Phidgets - Server Members

This chapter gives an overview of the members available for the Phidgets Server device.
Please read the chapter "Phidgets - Server" [1425] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Disable

Phidgets_ServerID.Disable

Example:
Phidgets_Server1.Disable

This disables the Phidgets_Server1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1426] member to reverse this command.

Enable

Phidgets_ServerID.Enable

Example:
Phidgets_Server1.Enable

This enables the Phidgets_Server1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1426] member to reverse this command.

GetSerialList

Phidgets_ServerID.GetSerialList

Example:
vlist = Phidgets_Server1.GetSerialList

Returns the Device Serials as a list

IpAddress

Phidgets_ServerID.IpAddress

Example:
vstring = Phidgets_Server1.IpAddress

Returns the "IpAddress", also displayed in Configuration dialog

IsConnected

Phidgets_ServerID.IsConnected

Example:
vbool = Phidgets_Server1.IsConnected

This member returns the connection status for the Phidgets_Server1 device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_Server1.IsConnected`
or simply display it in the Debug Logger: `DebugMessage(Phidgets_Server1.IsConnected)`

IsEnabled

Phidgets_ServerID.IsEnabled

Example:
vbool = Phidgets_Server1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable 1900 named "vbool".

You could also write it into a Widget, e.g. a Label 888 via the script: `Label1.Text = Phidgets_Server1.IsEnabled`
or simply display it in the Debug Logger 812: `DebugMessage(Phidgets_Server1.IsEnabled)`

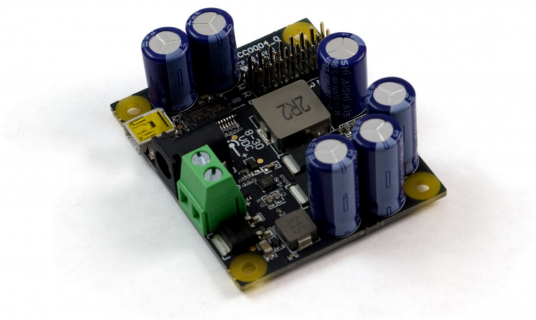
Port

Phidgets_ServerID.Port

Example:
vint = Phidgets_Server1.Port

Returns the "Port", also displayed in Configuration dialog

7.6.14.9 Phidgets - Servo



PhidgetAdvancedServo 8-Motor - P/N RCC0004_0
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

The Servo Phidgets are controller boards for interfacing with RC servo motors or linear actuators.

Currently available are boards to with one, eight or sixteen slots.
If you are using more than one servo slot per board, you need to create a second Phidgets Servo device for each additional motor and assign the respective channel number.

Adding a New Phidgets Servo

To add a Phidgets Servo device, open the Devices menu and select Phidgets Device > Servo Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

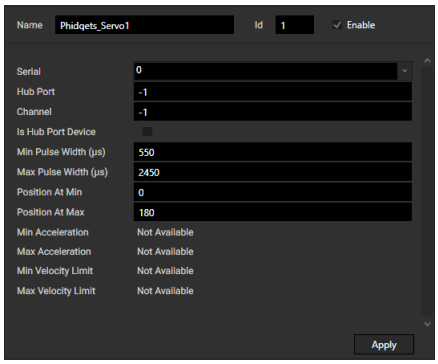
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Servo object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Servo is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Phidgets Servo Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Servo Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub, use its serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.

The **Min** and **Max Pulse Width** need to be set according to the connected servo motor's specifications. Please make sure you have the correct values here to avoid damaging your motor.
Adjust the **Min** and **Max Position** values as well if necessary.

Press **Apply** to confirm your settings.
After the connection is established and the pulse widths are set, the **Acceleration** and **Velocity** fields will update according to the current setting.

Important Note:
Before a Servo motor can be used, it must be set to "Engaged", using the respective Script described below!
Also, before a Servo motor can be engaged, a position value must have been sent to the device in

advance!
This can happen either via scripting or via node.

Using the Device in Regular Scripting

After adding a Phidgets Servo, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_Servo1"), Script Assist will offer you a list of all available members.

You can for example set a start position and engage the device using scripting:

```
Phidgets__Servo1.MoveToPosition(0)
Phidgets__Servo1.Engage
```

You can also read out the minimum acceleration value that can be achieved with the current settings:

```
vdouble = Phidgets_Servo1.MinAcceleration
```

Or use this value to move the motor into a certain position as slowly as possible:

```
Phidgets_Servo1.MoveToPositionWithRamp(180,vdouble,Phidgets_Servo1.MinVelocityLimit)
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:

```
Label5.Text = "Current Position: " + Phidgets_Servo1.Position
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:

```
Label5.Text = "Current Position: " + Phidgets_Servo1.GetPosition
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. the acceleration and velocity limit values, and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Servo Members [1430].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Servo Events [1430] chapter for a list (with examples and description) of the possible events raised by this device type.

Using the Device in the Node System

After creating a Phidgets Servo device, you can also use it in the node system as an input node to retrieve position, acceleration and velocity information.
There also is an output node for the Servo device, enabling you to move the motor to a certain position and influence the movement parameters.
It is for example possible to create a Fader in Widget Designer which controls the position of the servo motor.

Please see the chapters "Phidgets Servo Input [1020]" and "Phidgets Servo Output [1201]" for more information about the nodes themselves or the chapter "Tutorial: Nodes" [939] for information about the node system.

7.6.14.9.1 Phidgets - Servo Events

This chapter gives an overview of the events that are raised by a Phidgets Servo device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Servo" [1428] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Servo Members" [1430].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa.
The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

IsMoving

Example:
Label1.Text = isMoving

This event is raised whenever the device either starts or stops moving.
The event returns one parameter to WD which is a Boolean value with the name "isMoving" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device starts/stops moving.

Position

Example:
Fader4.Value = position

This event is raised whenever the position of the Phidgets_Servo1 device changes.
The event returns one parameter to WD which is a double value with the name "position".
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value as the position value from the Phidgets device.

7.6.14.9.2 Phidgets - Servo Members

This page is still under construction.

Please check out our Online Manual for updates!

7.6.14.10 Phidgets - Sound Phidget

Sound Phidget - P/N SND1000_0
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

The Phidgets Sound Phidget (SND1000_0) is a sound sensor that measures ambient sound pressure levels from 34 to 102 dB.

If you are using the Sound Sensor Phidget (PN_1133), please add it as a Voltage Input [1452] device.

Adding a New Sound Phidget

To add a Sound Phidget device, open the Devices menu and select Phidgets Device > Sound Phidget Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

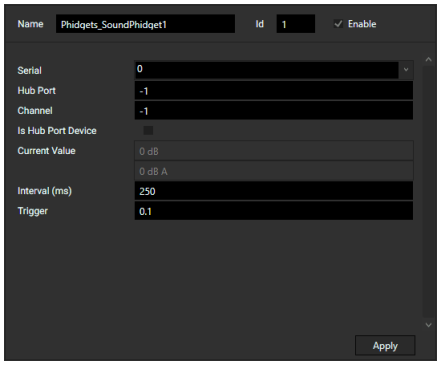
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Sound Phidget object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Sound Phidget is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Sound Phidget Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Sound Phidget Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub, use its serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.
Select an update **Interval** time and a **Trigger** value (for sensitivity) according to your needs.

Press **Apply** to confirm your settings.

After the connection is established, the **Current Value** fields will update as soon as a change is registered.

Using the Device in Regular Scripting

After adding a Sound Phidget, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_SoundPhidget1"), Script Assist will offer you a list of all available members.

You can for example enable the device using scripting:
```
Phidgets_SoundPhidget1.Enable
```

You can also retrieve the current measurement in dB A:
```
vdouble = Phidgets_SoundPhidget1.Dba
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Phidgets_SoundPhidget1.Interval = 100
Label5.Text = "Current Update Interval: " + Phidgets_SoundPhidget1.Interval
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Phidgets_SoundPhidget1.SetInterval(100)
Label5.Text = "Current Update Interval: " + Phidgets_SoundPhidget1.GetInterval
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. the measured decibel values or "Octaves", and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Sound Phidget Members [1433].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Sound Phidget Events [1432] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.10.1 Phidgets - Sound Phidget Events

This chapter gives an overview of the events that are raised by a Phidgets Servo device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Sound Phidget" [1431] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Sound Phidget Members" [1433].

Db

Example:
BarGraph4.Value = db

This event is raised whenever the registered sound pressure level in dB changes.
The event returns one parameter to WD which is a double value with the name "db".
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer BarGraph [852] with ID 4 will be assigned with the same value as the level value from the Phidgets device and acts as a display for sound pressure.

Dba

Example:
BarGraph4.Value = dba

This event is raised whenever the registered sound pressure level in dBa changes. dBA is a weighted measurement, designed to respond similarly to the human ear.
The event returns one parameter to WD which is a double value with the name "dba".
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer BarGraph 852 with ID 4 will be assigned with the same value as the level value from the Phidgets device and acts as a display for sound pressure.

Dbc

Example:
BarGraph4.Value = dbc

This event is raised whenever the registered sound pressure level in dBc changes. dBC is a weighted measurement, commonly used in sound engineering and testing.
The event returns one parameter to WD which is a double value with the name "dbc".
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer BarGraph 852 with ID 4 will be assigned with the same value as the level value from the Phidgets device and acts as a display for sound pressure.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa.
The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label 888 with ID 1 whenever the device dis-/connects.

Octaves

Example:
Label1.Text = "1kHz: " + octaves[5] + "dB"

The Sound Phidget is capable of measuring the distribution of sound across the frequency spectrum, split in ten sub-bands.
If the intensity value within at least one of those frequency bands changes, this event is raised.
The event returns one parameter to WD which is a List with the name "octaves" and holds ten sound pressure levels corresponding to the ten sub-bands (32Hz, 63Hz, 125Hz, 250Hz, 500Hz, 1kHz, 2kHz, 4kHz, 8kHz, 16kHz).
If you select this event and copy the example into the scripting field of the Event Listener, it will write the current sound pressure level of the 1kHz frequency band into the Label 888 with ID 1 whenever a change in any band is registered.

7.6.14.10.2 Phidgets - Sound Phidget Members

This page is still under construction.

Please check out our Online Manual for updates!

7.6.14.11 Phidgets - Spatial 0/0/3

PhidgetSpatial Precision 0/0/3 - P/N 1043_1B
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

The Spatial 0/0/3 is a sensor family capable of measuring accelerations of ±8g on three axes.

WD supports the Basic and the Precision model of the USB connected PhidgetSpatial 0/0/3, as well as the VINT connected Accelerometer Phidget.

Additionally, any Spatial 3/3/3 Phidget [1022] can be set up as a 0/0/3 device, it will then only return the acceleration values.

Adding a New Phidgets Spatial 0/0/3

To add a Phidgets Spatial 0/0/3 device, open the Devices menu and select Phidgets Device > Spatial 0/0/3 Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

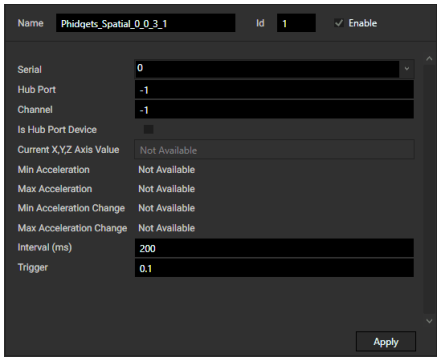
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Spatial 0/0/3 object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter). If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Spatial 0/0/3 is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Phidgets Spatial 0/0/3 Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Spatial 0/0/3 Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub, use its serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.
Select an update **Interval** time and a **Trigger** value (for sensitivity) according to your needs.

Press **Apply** to confirm your settings.

After the connection is established, the **Value** and **Acceleration** fields will update as soon as a change is registered.

Using the Device in Regular Scripting

After adding a Phidgets Spatial 0/0/3 device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_Spatial_0_0_3_1"), Script Assist will offer you a list of all available members.

You can for example enable the device using scripting:
```
Phidgets__Spatial_0_0_3_1.Enable
```

You can also read out the minimum acceleration value (in g) which triggers a response of the device:
```
vint = Phidgets_Spatial_0_0_3_1.MinAccelerationChangeTrigger
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Label5.Text = "Current X acceleration: " + Phidgets_Spatial_0_0_3_1.XAxis + "g"
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Label5.Text = "Current X acceleration: " + Phidgets_Spatial_0_0_3_1.GetXAxis + "g"
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. the measured acceleration values for the three axes, and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Spatial 0/0/3 Members [1436].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Spatial 0/0/3 Events [1435] chapter for a list (with examples and description) of the possible events raised by this device type.

Using the Device in the Node System

After creating a Phidgets Spatial 0/0/3 device, you can also use it in the node system as an input node to retrieve acceleration information.

Please see the chapter "Phidgets Spatial 0/0/3 Input [1021]" for more information about the node itself or the chapter "Tutorial: Nodes" [939] for information about the node system.

7.6.14.11.1 Phidgets - Spatial 0/0/3 Events

This chapter gives an overview of the events that are raised by a Phidgets Spatial 0/0/3 device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Spatial 0/0/3" [1434] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Spatial 0/0/3 Members" [1436].

Acceleration

Example:
BarGraph4.Value = y

This event is raised whenever the registered acceleration of the Phidgets_Spatial_0_0_3_1 device changes. The event returns three parameters to WD which are named "x", "y" and "z", and represent the measured acceleration value in each of the three directions, the unit is g.
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer

BarGraph [852] with ID 4 will be assigned with the same value as the acceleration value in y-direction from the Phidgets device.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

OutOfRange

Example:
Label1.StartFlash

This event is raised whenever one or more of the most recent values the device has measured are outside the sensor's valid range.
The event does not return any parameters for the associated script.
If you select this event and copy the example into the scripting field of the Event Listener, Label [888] with ID 1, will start flashing to draw attention on the detected movement being too fast.

7.6.14.11.2 Phidgets - Spatial 0/0/3 Members

This page is still under construction.

Please check out our Online Manual for updates!

7.6.14.12 Phidgets - Spatial 3/3/3

PhidgetSpatial Precision 0/0/3 - P/N 1044_1B
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

The Spatial 3/3/3 is a sensor family providing an accelerometer, a gyroscope and a compass all in one device.

WD supports the Basic and the Precision model of the USB connected PhidgetSpatial 3/3/3, as well as the VINT connected Spatial Phidget.

Additionally, any Spatial 3/3/3 Phidget can be set up as a Spatial 0/0/3 device [1434], if only the acceleration values are needed.

Adding a New Phidgets Spatial 3/3/3

To add a Phidgets Spatial 3/3/3 device, open the Devices menu and select Phidgets Device > Spatial 3/3/3 Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

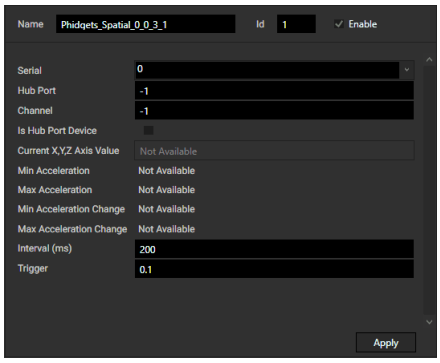
The **name** is the unique identifier for this Phidgets Spatial 3/3/3 object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).

If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Spatial 3/3/3 is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Phidgets Spatial 3/3/3 Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Spatial 3/3/3 Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub, use its serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.
Select an update **Interval** time according to your needs.

When you are using the compass function of this Phidget, you can perform a magnetic error correction beforehand. Follow the instructions on the product website for calibrating the device and copy/paste the parameters displayed in the text box to the **Compass Correction** in the WD configuration.

Press **Apply** to confirm your settings.

After the connection is established, the **Value** fields will update as soon as a change is registered.

Using the Device in Regular Scripting

After adding a Phidgets Spatial 3/3/3 device, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_Spatial_3_3_3_1"), Script Assist will offer you a list of all available members.

You can for example enable the device using scripting:
```
Phidgets__Spatial_3_3_3_1.Enable
```

You can also read out the minimum acceleration value (in g) which triggers a response of the device:
```
vdouble = Phidgets_Spatial_3_3_3_1.MinAccelerationChangeTrigger
```

To compensate for the drift that is inherent to all gyroscopes, execute the ZeroGyroscope method:
```
Phidgets_Spatial_3_3_3_1.ZeroGyroscope
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Label5.Text = "Current X acceleration: " +
Phidgets_Spatial_3_3_3_1.AccelerationAxisX + "g"
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Label5.Text = "Current X acceleration: " +
Phidgets_Spatial_3_3_3_1.AccelerationAxisX + "g"
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. the measured acceleration values for the three axis, and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Spatial 3/3/3 Members [1439].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Spatial 3/3/3 Events [1438] chapter for a list (with examples and description) of the possible events raised by this device type.

Using the Device in the Node System

After creating a Phidgets Spatial 3/3/3 device, you can also use it in the node system as an input node to retrieve acceleration, compass and gyroscope information.

Please see the chapter "Phidgets Spatial 3/3/3 Input [1022]" for more information about the node itself or the chapter "Tutorial: Nodes" [939] for information about the node system.

7.6.14.12.1 Phidgets - Spatial 3/3/3 Events

This chapter gives an overview of the events that are raised by a Phidgets Spatial 3/3/3 device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Spatial 3/3/3" [1436] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Spatial 3/3/3 Members" [1439].

Acceleration

Example:
BarGraph4.Value = y

This event is raised whenever the registered acceleration of the Phidgets_Spatial_3_3_3_1 device changes. The event returns three parameters to WD which are named "x", "y" and "z", and represent the measured acceleration value in each of the three directions, the unit is g.
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer BarGraph [852] with ID 4 will be assigned with the same value as the acceleration value in y-direction from the Phidgets device.

CompassBearing

Example:
Gauge4.Value = bearing

This event is raised whenever the Phidgets_Spatial_3_3_3_1 device registers a change in the surrounding magnetic field.
The event returns six parameters to WD. The first three are named "x", "y" and "z", and hold double values representing the strengths of the magnetic field in each of the three directions, the unit is Gauss (G).
The following three parameters, "pitch", "roll" and "bearing" hold the rotational angles of the device within the surrounding magnetic field as double values in degrees (°).
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer Gauge [856] with ID 4 will be assigned with the same value as the bearing value from the Phidgets device and indicates its current orientation in the x/y-plane.

GyroscopeHeading

Example:
BarGraph4.Value = y

This event is raised whenever the gyroscope of the Phidgets_Spatial_3_3_3_1 device detects a changes. The event returns three parameters to WD which are named "x", "y" and "z", and represent the measured angular velocity value in each of the three planes, the unit is degrees per second (°/s).
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer BarGraph [852] with ID 4 will be assigned with the same value as the velocity value in y-plane from the Phidgets device.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

OutOfRange

Example:
Label1.StartFlash

This event is raised whenever one or more of the most recent values the device has measured are outside the sensor's valid range.
The event does not return any parameters for the associated script.
If you select this event and copy the example into the scripting field of the Event Listener, Label [888] with ID 1, will start flashing to draw attention on the detected movement being too fast.

7.6.14.12.2 Phidgets - Spatial 3/3/3 Members

This page is still under construction.

Please check out our Online Manual for updates!

7.6.14.13 Phidgets - Stepper

A Phidgets Stepper device is a controllable motor and can be connected either directly or via a Hub.

Adding a New Phidgets Stepper

To add a Phidgets Stepper device, open the Devices menu and select Phidgets Device > Stepper Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

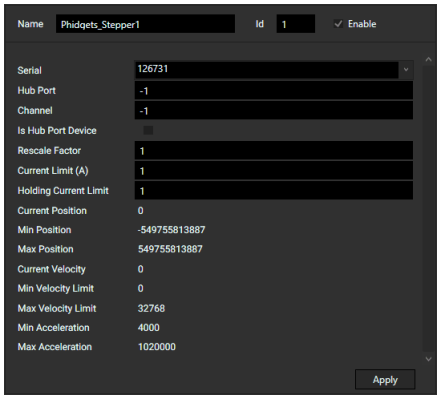
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Stepper object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Stepper is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Stepper Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Stepper Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub, use its serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.
If needed, you can manually adjust the **Rescale Factor**, the **Current Limit** (unit in amp) and the **Holding Current Limit**.

Press **Apply** to confirm your settings.

After the connection is established, the **parameter** fields will be displayed and updated as soon as a change is registered.

Important Note:
Before a Stepper motor can be used, it must be set to "Engaged", using the respective Script described below!
Please also make sure to not exceed the connected motor's capabilities.

Using the Device in Regular Scripting

After adding a Phidgets Stepper, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_Stepper1"), Script Assist will offer you a list of all available members.

Before data transmission can be started, the device needs to be engaged using the following command:
```
Phidgets_Stepper1.Engage
```

After the device is engaged, you can for example set the Stepper to move continuously:
```
Phidgets_Stepper1.MoveContinous(4000,6000)
```

You can also retrieve specific values via scripting in the same way as from other widgets:
```
vbool = Phidgets_Stepper1.isMoving
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Phidgets_Stepper1.RescaleFactor = 2
Label5.Text = "Rescale: " + Phidgets_Stepper1.RescaleFactor
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Phidgets_Stepper1.SetRescaleFactor(2)
Label5.Text = "Rescale: " + Phidgets_Stepper1.GetRescaleFactor
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. "Velocity" or "TargetPosition", and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Stepper Members [1441].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Stepper Events [1441] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.13.1 Phidgets - Stepper Events

This chapter gives an overview of the events that are raised by a Phidgets Stepper device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Stepper" [1439] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Stepper Members" [1441].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

Position

Example:
Fader4.Value = position

This event is raised whenever the position of the Phidgets_Stepper1 device changes.
The event returns one parameter to WD which is a double value with the name "position".
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value as the position value from the Phidgets device.

Velocity

Example:
Fader4.Value = velocity

This event is raised whenever the velocity of the Phidgets_Stepper1 device changes.
The event returns one parameter to WD which is a double value with the name "velocity".
If you select this event and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value as the velocity value from the Phidgets device.

7.6.14.13.2 Phidgets - Stepper Members

This chapter gives an overview of the members available for the Phidgets Stepper device.
Please read the chapter "Phidgets - Stepper" [1439] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Channel

Phidgets_StepperID.Channel

Example:
vint = Phidgets_Stepper1.Channel

This member sets and returns the channel which is displayed next to "Channel" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_Stepper1.Channel`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_Stepper1.Channel)`
Example2:
Phidgets_Stepper1.Channel = 2

The second example shows, how to set the channel to a value of 2.

DeviceSerial

Phidgets_StepperID.DeviceSerial

Example:
vint = Phidgets_Stepper1.DeviceSerial

This member sets and returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_Stepper1.DeviceSerial`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_Stepper1.DeviceSerial)`
Example2:
Phidgets_Stepper1.DeviceSerial = 530161

The second example shows, how to set the channel of the digital output 1 to a value of 530161.

Disable

Phidgets_StepperID.Disable

Example:
Phidgets_Stepper1.Disable

This disables the Phidgets_Stepper1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1442] member to reverse this command.

Enable

Phidgets_StepperID.Enable

Example:
Phidgets_Stepper1.Enable

This enables the Phidgets_Stepper1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1442] member to reverse this command.

HubPort

Phidgets_StepperID.HubPort

Example:
vint = Phidgets_Stepper1.HubPort

This member sets and returns the hub port number which is displayed next to "Hub Port" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_Stepper1.HubPort`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_Stepper1.HubPort)`
Example2:
Phidgets_Stepper1.HubPort = 2

The second example shows, how to set the hub port to a value of 2.

IsConnected

Phidgets_StepperID.IsConnected

Example:
vbool = Phidgets_Stepper1.IsConnected

This member returns the connection status for the Phidgets_Stepper1 device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_Stepper1.IsConnected`
or simply display it in the Debug Logger: `DebugMessage(Phidgets_Stepper1.IsConnected)`

IsEnabled

Phidgets_StepperID.IsEnabled

Example:
vbool = Phidgets_Stepper1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_Stepper1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_Stepper1.IsEnabled)`

IsHubPortDevice

Phidgets_StepperID.IsHubPortDevice

Example:
vbool = Phidgets_Stepper1.IsHubPortDevice

This member returns the status of the check box "Is Hub Port Device" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is a hub port device and "False" if it is not.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_Stepper1.IsHubPortDevice`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_Stepper1.IsHubPortDevice)`

RescaleFactor

Phidgets_StepperID.RescaleFactor

Example:
vdouble = Phidgets_Stepper1.RescaleFactor

Sets or returns "Rescale Factor", also displayed in Configuration dialog

Example2:
Phidgets_Stepper1.RescaleFactor = 1.5

SetRescaleFactor

Phidgets_StepperID.SetRescaleFactor(value)

Example:
Phidgets_Stepper1.SetRescaleFactor(1.5)

Sets "Rescale Factor", also displayed in Configuration dialog

GetRescaleFactor

Phidgets_StepperID.GetRescaleFactor

Example:
vdouble = Phidgets_Stepper1.GetRescaleFactor

Returns "Rescale Factor", also displayed in Configuration dialog

CurrentLimit

Phidgets_StepperID.CurrentLimit

Example:
vdouble = Phidgets_Stepper1.CurrentLimit

Sets or returns "Current Limit A", also displayed in Configuration dialog

Example2:
Phidgets_Stepper1.CurrentLimit = 5

SetCurrentLimit

Phidgets_StepperID.SetCurrentLimit(value)

Example:
Phidgets_Stepper1.SetCurrentLimit(5)

Sets "Current Limit A", also displayed in Configuration dialog

GetCurrentLimit

Phidgets_StepperID.GetCurrentLimit

Example:
vdouble = Phidgets_Stepper1.GetCurrentLimit

Returns "Current Limit A", also displayed in Configuration dialog

HoldingCurrentLimit

Phidgets_StepperID.HoldingCurrentLimit

Example:
vdouble = Phidgets_Stepper1.HoldingCurrentLimit

Sets or returns "Holding Current Limit", also displayed in Configuration dialog

Example2:
Phidgets_Stepper1.HoldingCurrentLimit = 5

SetHoldingCurrentLimit

Phidgets_StepperID.SetHoldingCurrentLimit(value)

Example:
Phidgets_Stepper1.SetHoldingCurrentLimit(5)

Sets "Holding Current Limit", also displayed in Configuration dialog

GetHoldingCurrentLimit

Phidgets_StepperID.GetHoldingCurrentLimit

Example:
vdouble = Phidgets_Stepper1.GetHoldingCurrentLimit

Returns "Holding Current Limit", also displayed in Configuration dialog

MinAcceleration

Phidgets_StepperID.MinAcceleration

Example:
vdouble = Phidgets_Stepper1.MinAcceleration

Reads out "Min Acceleration", also displayed in Configuration dialog

GetMinAcceleration

Phidgets_StepperID.GetMinAcceleration

Example:
vdouble = Phidgets_Stepper1.GetMinAcceleration

Reads out "Min Acceleration", also displayed in Configuration dialog

MaxAcceleration

Phidgets_StepperID.MaxAcceleration

Example:
vdouble = Phidgets_Stepper1.MaxAcceleration

Reads out "Max Acceleration", also displayed in Configuration dialog

GetMaxAcceleration

Phidgets_StepperID.GetMaxAcceleration

Example:
vdouble = Phidgets_Stepper1.GetMaxAcceleration

Reads out "Max Acceleration", also displayed in Configuration dialog

MinVelocityLimit

Phidgets_StepperID.MinVelocityLimit

Example:
vdouble = Phidgets_Stepper1.MinVelocityLimit

Reads out "Min Velocity Limit", also displayed in Configuration dialog

GetMinVelocityLimit

Phidgets_StepperID.GetMinVelocityLimit

Example:
vdouble = Phidgets_Stepper1.GetMinVelocityLimit

Reads out "Min Velocity Limit", also displayed in Configuration dialog

MaxVelocityLimit

Phidgets_StepperID.MaxVelocityLimit

Example:
vdouble = Phidgets_Stepper1.MaxVelocityLimit

Reads out "Max Velocity Limit", also displayed in Configuration dialog

GetMaxVelocityLimit

Phidgets_StepperID.GetMaxVelocityLimit

Example:
vdouble = Phidgets_Stepper1.GetMaxVelocityLimit

Reads out "Max Velocity Limit", also displayed in Configuration dialog

MinPosition

Phidgets_StepperID.MinPosition

Example:
vdouble = Phidgets_Stepper1.MinPosition

Reads out "Min Position", also displayed in Configuration dialog

GetMinPosition

Phidgets_StepperID.GetMinPosition

Example:
vdouble = Phidgets_Stepper1.GetMinPosition

Reads out "Min Position", also displayed in Configuration dialog

MaxPosition

Phidgets_StepperID.MaxPosition

Example:
vdouble = Phidgets_Stepper1.MaxPosition

Reads out "Max Position", also displayed in Configuration dialog

GetMaxPosition

Phidgets_StepperID.GetMaxPosition

Example:
vdouble = Phidgets_Stepper1.GetMaxPosition

Reads out "Max Position", also displayed in Configuration dialog

Engage

Phidgets_StepperID.Engage

Example:
Phidgets_Stepper1.Engage

Engages the Stepper

Disengage

Phidgets_StepperID.Disengage

Example:
Phidgets_Stepper1.Disengage

Disengages the Stepper

StepToPosition

Phidgets_StepperID.StepToPosition(target,accel,velo)

Example:
Phidgets_Stepper1.StepToPosition(50,2,10)

Sets a target position with acceleration and velocity

MoveContinous

Phidgets_StepperID.MoveContinous(Accel, Velo)

Example:
Phidgets_Stepper1.MoveContinous(2,10)

Sets a continuous movement with acceleration and velocity

SetZeroPosition

Phidgets_StepperID.SetZeroPosition

Example:
Phidgets_Stepper1.SetZeroPosition

Sets the current Position as 0

Velocity

Phidgets_StepperID.Velocity

Example:
vdouble = Phidgets_Stepper1.Velocity

Reads out "Current Velocity", also displayed in Configuration dialog

GetVelocity

Phidgets_StepperID.GetVelocity

Example:
vdouble = Phidgets_Stepper1.GetVelocity

Reads out "Current Velocity", also displayed in Configuration dialog

Position

Phidgets_StepperID.Position

Example:
vint = Phidgets_Stepper1.Position

Reads out "Current (Motor) Postion", also displayed in Configuration dialog

GetPosition

Phidgets_StepperID.GetPosition

Example:
vint = Phidgets_Stepper1.GetPosition

Reads out "Current (Motor) Postion", also displayed in Configuration dialog

TargetPosition

Phidgets_StepperID.TargetPosition

Example:
vint = Phidgets_Stepper1.TargetPosition

reads out TargetPosition

GetTargetPosition

Phidgets_StepperID.GetTargetPosition

Example:
vint = Phidgets_Stepper1.GetTargetPosition

reads out TargetPosition

isEngaged

Phidgets_StepperID.isEngaged

Example:
vbool = Phidgets_Stepper1.isEngaged

returns if Motor is engaged

isMoving

Phidgets_StepperID.isMoving

Example:
vbool = Phidgets_Stepper1.isMoving

returns if Motor is moving

7.6.14.14 Phidgets - Temperature Phidget

Temperature Phidget - P/N TMP1000_0
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

The Phidgets Temperature Phidget (TMP1000_0) is a sensor that can measure the ambient temperature of the surrounding air from -40°C to 85°C.

If you are using a different sensor, for example the Precision Temperature Sensor (PN_1124), please add it as a Voltage Ratio Input [1460] device (or respective other type).

Adding a New Temperature Phidget

To add a Phidgets Temperature Phidget device, open the Devices menu and select Phidgets Device > Temperature Phidget Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

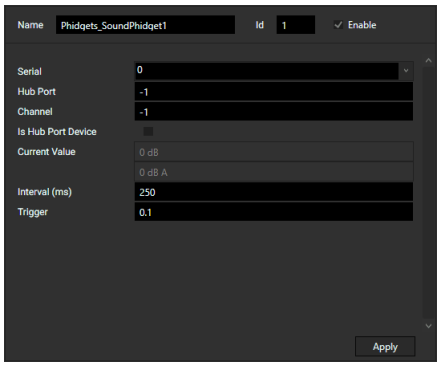
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Temperature Phidget object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter). If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Temperature Phidget is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Temperature Phidget Device and can be opened from here or with Devices menu > Configuration.

The Temperature Phidget Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub, use its serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog.
Select an update **Interval** time and a **Trigger** value (for sensitivity) according to your needs.

Press **Apply** to confirm your settings.

After the connection is established, the **Current Value** field as well as **Min and Max Temperature** will be updated as soon as a change is registered.

Using the Device in Regular Scripting

After adding a Temperature Phidget, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_TemperaturePhidget1"), Script Assist will offer you a list of all available members.

You can for example enable the device using scripting:
```
Phidgets_TemperaturePhidget1.Enable
```

You can also retrieve the current measurement temperature in °C:
```
vdouble_C = Phidgets_TemperaturePhidget1.Value
```

For calculating the temperature in °F, you can use this unnecessarily complicated formula:
```
vdouble_F = Phidgets_TemperaturePhidget1.Value * 1.8 + 32
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Phidgets_TemperaturePhidget1.Interval = 100
Label5.Text = "Current Update Interval: " + Phidgets_TemperaturePhidget1.Interval
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Phidgets_TemperaturePhidget1.SetInterval(100)
Label5.Text = "Current Update Interval: " + Phidgets_TemperaturePhidget1.GetInterval
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. "Value", and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Temperature Phidget Members [1452].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Temperature Phidget Events [1451] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.14.1 Phidgets - Temperature Phidget Events

This chapter gives an overview of the events that are raised by a Phidgets Voltage Input device and that can be used in the Event Listener [1353].
Please read the chapters Phidgets - Temperature Phidget [1450] and Event Listener [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter Phidgets - Temperature Phidget Members [1452].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

Value

Example:
Fader4.Value = value

This event is raised whenever the Temperature Phidget device changes its value.
The event returns one parameter to WD which is a double with the name "value" and holds the current value in degrees Celsius (°C).
If you select the event and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value from the Temperature Phidget device whenever it changes.

7.6.14.14.2 Phidgets - Temperature Phidget Members

This page is still under construction.

Please check out our Online Manual for updates!

7.6.14.15 Phidgets - Voltage Input

Example: Light Sensor 1000 lux - P/N 1142_0
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

A Phidgets Voltage Input connects to an Analog Input (e.g. of an InterfaceKit) [1399] or VINT Hub port

The depicted Light Sensor for example can be connected to both.

Please make sure that the device is a real "Voltage Input" and not a "Voltage Ratio Input [1460]" device.

Adding a New Phidgets Voltage Input

To add a Phidgets Voltage Input device, open the Devices menu and select Phidgets Device > Voltage Input Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

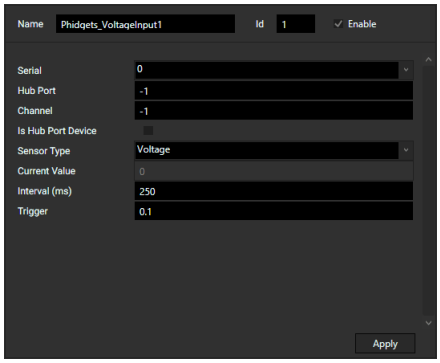
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Voltage Input object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter). If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Voltage Input is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Voltage Input Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Voltage Input Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub or InterfaceKit, use their serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub or InterfaceKit) is shown in the Phidget Control Panel [1385].
To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog and select the product number from the **Sensor Type** drop-down.
Select an update **Interval** time and a **Trigger** value (for sensitivity) according to your needs.

Press **Apply** to confirm your settings.

After the connection is established, the **Current Value** field will update as soon as a change is registered and also display the correct value units.

Using the Device in Regular Scripting

After adding a Phidgets Voltage Input, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_VoltageInput1"), Script Assist will offer you a list of all available members.

You can for example set the sensor type to an Indoor Light Sensor:
```
Phidgets_VoltageInput1.SetSensorType("PN_1142")
```

You can also retrieve specific values via scripting in the same way as from other widgets:
```
vint = Phidgets_VoltageInput1.GetValue
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:
```
Phidgets_VoltageInput1.Interval = 100
Label5.Text = "Current Update Interval: " + Phidgets_VoltageInput1.Interval
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:
```
Phidgets_VoltageInput1.SetInterval(100)
Label5.Text = "Current Update Interval: " + Phidgets_VoltageInput1.GetInterval
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. "Unit", and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Voltage Input Members [1454].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Voltage Input Events [1454] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.15.1 Phidgets - Voltage Input Events

This chapter gives an overview of the events that are raised by a Phidgets Voltage Input device and that can be used in the Event Listener [1353].
Please read the chapters Phidgets - Voltage Input [1452] and Event Listener [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter Phidgets - Voltage Input Members [1454].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

Value

Example:
Fader4.Value = value

This event is raised whenever the Phidgets_VoltageInput device changes its value.
The event returns one parameter to WD which is a double with the name "value" and holds the current value.
If you select the event and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value from the Phidgets_VoltageInput device whenever it changes.

7.6.14.15.2 Phidgets - Voltage Input Members

This chapter gives an overview of the members available for the Phidgets Voltage Input device.
Please read the chapter "Phidgets - Voltage Input" [1452] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Channel

Phidgets_VoltageInputID.Channel

Example:
vint = Phidgets_VoltageInput1.Channel

This member sets and returns the channel which is displayed next to "Channel" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.Channel`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.Channel)`
Example2:
Phidgets_VoltageInput1.Channel = 2

The second example shows, how to set the channel to a value of 2.

DeviceSerial

Phidgets_VoltageInputID.DeviceSerial

Example:
vint = Phidgets_VoltageInput1.DeviceSerial

This member sets and returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.DeviceSerial`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.DeviceSerial)`
Example2:
Phidgets_VoltageInput1.DeviceSerial = 530161

The second example shows, how to set the channel of the digital output 1 to a value of 530161.

Disable

Phidgets_VoltageInputID.Disable

Example:
Phidgets_VoltageInput1.Disable

This disables the Phidgets_VoltageInput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1455] member to reverse this command.

Enable

Phidgets_VoltageInputID.Enable

Example:
Phidgets_VoltageInput1.Enable

This enables the Phidgets_VoltageInput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1455] member to reverse this command.

GetInterval

Phidgets_VoltageInputID.GetInterval

Example:
vint = Phidgets_VoltageInput1.GetInterval

This member returns the interval time which is displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.GetInterval`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.GetInterval)`
An alternative command is: `Phidgets_VoltageInput1.Interval` [1457]

GetSensorType

Phidgets_VoltageInputID.GetSensorType

Example:
vstring = Phidgets_VoltageInput1.GetSensorType

This member returns the sensor type which is displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.GetSensorType`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageInput1.GetSensorType)`
An alternative command is: `Phidgets_VoltageInput1.SensorType` [1458]

GetTrigger

Phidgets_VoltageInputID.GetTrigger

Example:
vint = Phidgets_VoltageInput1.GetTrigger

This member returns the trigger value which is displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.GetTrigger`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.GetTrigger)`
An alternative command is: `Phidgets_VoltageInput1.Trigger` [1459]

GetUnit

Phidgets_VoltageInputID.GetUnit

Example:
vstring = Phidgets_VoltageInput1.GetUnit

This member returns the unit, e.g. "v" for "Volts"..
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.GetUnit`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.GetUnit)`
An alternative command is: `Phidgets_VoltageInput1.Unit` [1459]

GetValue

Phidgets_VoltageInputID.GetValue

Example:
vdouble = Phidgets_VoltageInput1.GetValue

This member returns the value which is displayed next to "Current Value" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.GetValue`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.GetValue)`
An alternative command is: `Phidgets_VoltageInput1.Value` [1460]

HubPort

Phidgets_VoltageInputID.HubPort

Example:
vint = Phidgets_VoltageInput1.HubPort

This member sets and returns the hub port number which is displayed next to "Hub Port" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.HubPort`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.HubPort)`
Example2:
Phidgets_VoltageInput1.HubPort = 2

The second example shows, how to set the hub port to a value of 2.

Interval

Phidgets_VoltageInputID.Interval(test)

Example:
vint = Phidgets_VoltageInput1.Interval

This member sets and returns the interval time which is displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.Interval`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.Interval)`
An alternative command is: `Phidgets_VoltageInput1.GetInterval` [1455]
Example2:
Phidgets_VoltageInput1.Interval = 500

The second example shows, how to set the interval time to a value of 500ms.

An alternative command is: `Phidgets_VoltageInput1.SetInterval` [1458]

IsConnected

Phidgets_VoltageInputID.IsConnected

Example:
vbool = Phidgets_VoltageInput1.IsConnected

This member returns the connection status for the Phidgets_VoltageInput1 device as a Boolean value. The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_VoltageInput1.IsConnected`
or simply display it in the Debug Logger: `DebugMessage(Phidgets_VoltageInput1.IsConnected)`

IsEnabled

Phidgets_VoltageInputID.IsEnabled

Example:
vbool = Phidgets_VoltageInput1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.IsEnabled)`

IsHubPortDevice

Phidgets_VoltageInputID.IsHubPortDevice

Example:
vbool = Phidgets_VoltageInput1.IsHubPortDevice

This member returns the status of the check box "Is Hub Port Device" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is a hub port device and "False" if it is not.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.IsHubPortDevice`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageInput1.IsHubPortDevice)`

SensorType

Phidgets_VoltageInputID.SensorType

Example:
vstring = Phidgets_VoltageInput1.SensorType

This member sets and returns the sensor type which is displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.SensorType`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.SensorType)`
An alternative command is: `Phidgets_VoltageInput1.GetSensorType` [1455]

Example2:
Phidgets_VoltageInput1.SensorType = "PN_1102"

The second example shows, how to set the sensor type to one of the available types (as seen in the above mentioned drop-down list).

An alternative command is: `Phidgets_VoltageInput1.SetSensorType` [1459]

SetInterval

Phidgets_VoltageInputID.SetInterval(value)

Example:
Phidgets_VoltageInput1.SetInterval(500)

This member sets the interval time which is displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The example shows how to set the interval time to a value of 500ms.

An alternative command is: Phidgets VoltageInput1.Interval [1457]

SetSensorType

Phidgets_VoltageInputID.SetSensorType(value)

Example:
Phidgets_VoltageInput1.SetSensorType("PN_1102")

This member sets the sensor type which is displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The example shows, how to set the sensor type to one of the available types.

An alternative command is: Phidgets VoltageInput1.SensorType [1458]

SetTrigger

Phidgets_VoltageInputID.SetTrigger(value)

Example:
Phidgets_VoltageInput1.SetTrigger(0.2)

This member sets the trigger value which is displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The second example shows, how to set the trigger value to 0.2.

An alternative command is: Phidgets VoltageInput1.Trigger [1459]

Trigger

Phidgets_VoltageInputID.Trigger

Example:
vdouble = Phidgets_VoltageInput1.Trigger

This member sets and returns the trigger value which is displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.Trigger`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.Trigger)`
An alternative command is: Phidgets VoltageInput1.GetTrigger [1456]

Example2:
Phidgets_VoltageInput1.Trigger = 0.2

The second example shows, how to set the trigger value to 0.2.

An alternative command is: Phidgets VoltageInput1.SetTrigger [1459]

Unit

Phidgets_VoltageInputID.Unit

Example:
vstring = Phidgets_VoltageInput1.Unit

This member returns the unit, e.g. "v" for "Volts"..
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.Unit`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.Unit)`
An alternative command is: `Phidgets_VoltageInput1.GetUnit` [1456]

Value

Phidgets_VoltageInputID.Value

Example:
vdouble = Phidgets_VoltageInput1.Value

This member returns the value which is displayed next to "Current Value" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageInput1.Value`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageInput1.Value)`
An alternative command is: `Phidgets_VoltageInput1.GetValue` [1456]

7.6.14.16 Phidgets - Voltage Ratio Input

Example: Slider 60 - P/N 1112_1
(Image licensed by Phidgets under CC BY-NC-ND 3.0)

A Phidgets Voltage Input connects to an Analog Input (e.g. of an InterfaceKit) [1399] or VINT Hub port
The depicted Slider for example can be connected to both.

Please make sure that the device is a real "Voltage Ratio Input" and not a "Voltage Input [1452]" device.

Adding a New Phidgets Voltage Input

To add a Phidget Voltage Ratio Input device, open the Devices menu and select Phidgets Device > Voltage Ratio Input Device. This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

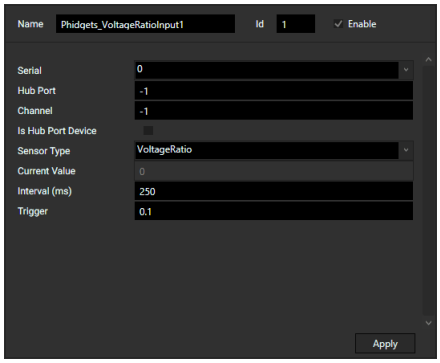
On the right side, you can name, dis-/enable the device or configure it as explained in the next paragraph.

The **name** is the unique identifier for this Phidgets Voltage Ratio Input object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The "**Enable**" check box is ticked per default. On the left side, you should see that the icon in front of your Phidgets Voltage Ratio Input is a filled blue circle. An empty blue circle indicates an enabled device which is not connected. A filled gray circle indicates a disabled device.

You can close the dialog at any time. The newly created device will also be added to the Devices menu > Phidgets Device > Voltage Ratio Input Device and can be opened from here or with Devices menu > Configuration.

The Phidgets Voltage Ratio Input Settings



Select the **Serial** number of the connected device from the drop-down. If your device is connected to a Hub or InterfaceKit, use their serial number respectively.
If the correct serial does not show in the drop-down, make sure that the device (or Hub or InterfaceKit) is shown in the Phidget Control Panel [1385]. To access devices connected via the Network Server feature, you need to add the respective Phidgets Server [1425] device first.

Fill out **Hub Port**, **Channel** and **Is Hub Port Device** according to the Phidget Addressing Information [1386] dialog and select the product number from the **Sensor Type** drop-down.
Select an update **Interval** time and a **Trigger** value (for sensitivity) according to your needs.

Press **Apply** to confirm your settings.

After the connection is established, the **Current Value** field will update as soon as a change is registered and also display the correct value units.

Using the Device in Regular Scripting

After adding a Phidgets Voltage Ratio Input, you can also use it via scripting which allows to perform actions on the device as well as retrieve information from it.

To send commands to the device, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is e.g. "Phidgets_VoltageRatioInput1"), Script Assist will offer you a list of all available members.

You can for example set the sensor type to a Slider Sensor:

```
Phidgets_VoltageRatioInput1.SetSensorType("PN_1112")
```

You can also retrieve specific values via scripting in the same way as from other widgets:

```
vint = Phidgets_VoltageRatioInput1.GetValue
```

The scripting structure for Phidgets provides two different approaches for setting and retrieving values.
If you are already familiar with the object-based notation [1904] and using properties, all device properties can be used like any other object property you have already encountered:

```
Phidgets_VoltageRatioInput1.Interval = 100
Label5.Text = "Current Update Interval: " + Phidgets_VoltageRatioInput1.Interval
```

If you prefer a command structure that distinctly implies the action to be performed, you can use the "Set..." and "Get..." members:

```
Phidgets_VoltageInput1.SetInterval(100)
Label5.Text = "Current Update Interval: " + Phidgets_VoltageRatioInput1.GetInterval
```

Both approaches can be used interchangeably.

Please keep in mind that some properties are read-only properties, e.g. "Unit", and do not provide a "Set..." member.
Furthermore, the higher-level properties of the device itself, like the Serial number or the Enabled state, need to be scripted as properties and do not provide the "Set..." and "Get..." members.

For a list with all available members and commands with examples and descriptions, please refer to the chapter Phidgets Voltage Ratio Input Members [1462].

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and the Phidgets Voltage Ratio Input Events [1462] chapter for a list (with examples and description) of the possible events raised by this device type.

7.6.14.16.1 Phidgets - Voltage Ratio Input Events

This chapter gives an overview of the events that are raised by a Phidgets Voltage Ratio Input device and that can be used in the Event Listener [1353].
Please read the chapters "Phidgets - Voltage Ratio Input" [1460] and "Event Listener" [1353] if you like to know how to add and use both. The available members that can be used in our scripting language are explained in the chapter "Phidgets - Voltage Ratio Input Members" [1462].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "True" or "False".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "True" or "False" into the Label [888] with ID 1 whenever the device dis-/connects.

Value

Example:
Fader4.Value = value

This event is raised whenever the Phidgets_VoltageRatioInput device changes its value.
The event returns one parameter to WD which is a double with the name "value" and holds the current value.
If you select the event and copy the example into the scripting field of the Event Listener, the Widget Designer Fader [874] with ID 4 will be assigned with the same value from the Phidgets_VoltageRatioInput device whenever it changes.

7.6.14.16.2 Phidgets - Voltage Ratio Input Members

This chapter gives an overview of the members available for the Phidgets Voltage Ratio Input device.
Please read the chapter "Phidgets - Voltage Ratio Input" [1460] if you like to know how to add and use this device.
The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Channel

Phidgets_VoltageRatioInputID.Channel

Example:
vint = Phidgets_VoltageRatioInput1.Channel

This member sets and returns the channel which is displayed next to "Channel" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.Channel`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageRatioInput1.Channel)`

Example2:
Phidgets_VoltageRatioInput1.Channel = 2

The second example shows, how to set the channel to a value of 2.

DeviceSerial

Phidgets_VoltageRatioInputID.DeviceSerial

Example:
vint = Phidgets_VoltageRatioInput1.DeviceSerial

This member sets and returns the serial number which is displayed next to "Serial" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.DeviceSerial`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.DeviceSerial)`

Example2:
Phidgets_VoltageRatioInput1.DeviceSerial = 530161

The second example shows, how to set the serial number of the device in the Configuration menu to 530161.

Disable

Phidgets_VoltageRatioInputID.Disable

Example:
Phidgets_VoltageRatioInput1.Disable

This disables the Phidgets_VoltageRatioInput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1463] member to reverse this command.

Enable

Phidgets_VoltageRatioInputID.Enable

Example:
Phidgets_VoltageRatioInput1.Enable

This enables the Phidgets_VoltageRatioInput1 device in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1463] member to reverse this command.

GetInterval

Phidgets_VoltageRatioInputID.GetInterval

Example:
vint = Phidgets_VoltageRatioInput1.GetInterval

This member returns the interval time which is displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.GetInterval`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.GetInterval)`
An alternative command is: `Phidgets_VoltageRatioInput1.Interval` [1465]

GetSensorType

Phidgets_VoltageRatioInputID.GetSensorType

Example:
vstring = Phidgets_VoltageRatioInput1.GetSensorType

This member returns the sensor type which is displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.GetSensorType`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.GetSensorType)`
An alternative command is: `Phidgets_VoltageRatioInput1.SensorType` [1466]

GetTrigger

Phidgets_VoltageRatioInputID.GetTrigger

Example:
vint = Phidgets_VoltageRatioInput1.GetTrigger

This member returns the trigger value which is displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.GetTrigger`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.GetTrigger)`
An alternative command is: `Phidgets_VoltageRatioInput1.Trigger` [1467]

GetUnit

Phidgets_VoltageRatioInputID.GetUnit

Example:
vstring = Phidgets_VoltageRatioInput1.GetUnit

This member returns the unit, e.g. "v" for "Volts"..
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.GetUnit`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageRatioInput1.GetUnit)`
An alternative command is: `Phidgets_VoltageRatioInput1.Unit` [1468]

GetValue

Phidgets_VoltageRatioInputID.GetValue

Example:
vdouble = Phidgets_VoltageRatioInput1.GetValue

This member returns the value which is displayed next to "Current Value" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.GetValue`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.GetValue)`
An alternative command is: `Phidgets_VoltageRatioInput1.Value` [1468]

HubPort

Phidgets_VoltageRatioInputID.HubPort

Example:
vint = Phidgets_VoltageRatioInput1.HubPort

This member sets and returns the hub port number which is displayed next to "Hub Port" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.HubPort`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageRatioInput1.HubPort)`
Example2:
Phidgets_VoltageRatioInput1.HubPort = 2

The second example shows, how to set the hub port to a value of 2.

Interval

Phidgets_VoltageRatioInputID.Interval(test)

Example:
vint = Phidgets_VoltageRatioInput1.Interval

This member sets and returns the interval time which is displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.Interval`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.Interval)`
An alternative command is: `Phidgets_VoltageRatioInput1.GetInterval` [1463]
Example2:
Phidgets_VoltageRatioInput1.Interval = 500

The second example shows, how to set the interval time to a value of 500ms.

An alternative command is: `Phidgets_VoltageRatioInput1.SetInterval` [1467]

IsConnected

Phidgets_VoltageRatioInputID.IsConnected

Example:
vbool = Phidgets_VoltageRatioInput1.IsConnected

This member returns the connection status for the Phidgets_VoltageRatioInput1 device as a Boolean value.
The result is "True" if the device is connected and "False" if it is currently disconnected.
The first example shows how to use it with an existing (global) Boolean variable named "vbool".

You could also write it into a Widget, e.g. a Label via the script: `Label1.Text = Phidgets_VoltageRatioInput1.IsConnected`
or simply display it in the Debug Logger:
`DebugMessage(Phidgets_VoltageRatioInput1.IsConnected)`

IsEnabled

Phidgets_VoltageRatioInputID.IsEnabled

Example:
vbool = Phidgets_VoltageRatioInput1.IsEnabled

This member returns the status of the check box "Enable" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.IsEnabled`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.IsEnabled)`

IsHubPortDevice

Phidgets_VoltageRatioInputID.IsHubPortDevice

Example:
vbool = Phidgets_VoltageRatioInput1.IsHubPortDevice

This member returns the status of the check box "Is Hub Port Device" of the configuration dialog when selecting the according Phidget device. The result, a Boolean value (or string), is "True" if the device is a hub port device and "False" if it is not.
The first example shows how to use it with an existing (global) Boolean variable [1900] named "vbool".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.IsHubPortDevice`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.IsHubPortDevice)`

SensorType

Phidgets_VoltageRatioInputID.SensorType

Example:
vstring = Phidgets_VoltageRatioInput1.SensorType

This member sets and returns the sensor type which is displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.SensorType`
or simply display it in the Debug Logger [812]:
`DebugMessage(Phidgets_VoltageRatioInput1.SensorType)`
An alternative command is: `Phidgets_VoltageRatioInput1.GetSensorType` [1464]
Example2:
Phidgets_VoltageRatioInput1.SensorType = "PN_1102"

The second example shows, how to set the sensor type to one of the available types (as seen in the above mentioned drop-down list).

An alternative command is: Phidgets_VoltageRatioInput1.SetSensorType [1467]

SetInterval

Phidgets_VoltageRatioInputID.SetInterval(value)

Example:
Phidgets_VoltageRatioInput1.SetInterval(500)

This member sets the interval time which is displayed next to "Interval (ms)" in the Configuration menu when selecting the according Phidget device.
The example shows how to set the interval time to a value of 500ms.

An alternative command is: Phidgets_VoltageRatioInput1.Interval [1465]

SetSensorType

Phidgets_VoltageRatioInputID.SetSensorType(value)

Example:
Phidgets_VoltageRatioInput1.SetSensorType("PN_1102")

This member sets the sensor type which is displayed in the drop-down list next to "Sensor Type" in the Configuration menu when selecting the according Phidget device.
The example shows, how to set the sensor type to one of the available types.

An alternative command is: Phidgets_VoltageRatioInput1.SensorType [1466]

SetTrigger

Phidgets_VoltageRatioInputID.SetTrigger(value)

Example:
Phidgets_VoltageRatioInput1.SetTrigger(0.2)

This member sets the trigger value which is displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The second example shows, how to set the trigger value to 0.2.

An alternative command is: Phidgets_VoltageRatioInput1.Trigger [1467]

Trigger

Phidgets_VoltageRatioInputID.Trigger

Example:
vdouble = Phidgets_VoltageRatioInput1.Trigger

This member sets and returns the trigger value which is displayed next to "Trigger" in the Configuration menu when selecting the according Phidget device.
The first example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.Trigger`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageRatioInput1.Trigger)`
An alternative command is: Phidgets_VoltageRatioInput1.GetTrigger [1464]

Example2:
Phidgets_VoltageRatioInput1.Trigger = 0.2

The second example shows, how to set the trigger value to 0.2.

An alternative command is: Phidgets_VoltageRatioInput1.SetTrigger [1467]

Unit

Phidgets_VoltageRatioInputID.Unit

Example:
vstring = Phidgets_VoltageRatioInput1.Unit

This member returns the unit, e.g. "v" for "Volts"..
The example shows how to return the value to an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.Unit`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageRatioInput1.Unit)`
An alternative command is: Phidgets_VoltageRatioInput1.GetUnit [1464]

Value

Phidgets_VoltageRatioInputID.Value

Example:
vdouble = Phidgets_VoltageRatioInput1.Value

This member returns the value which is displayed next to "Current Value" in the Configuration menu when selecting the according Phidget device.
The example shows how to return the value to an existing (global) double variable [1900] named "vdouble".

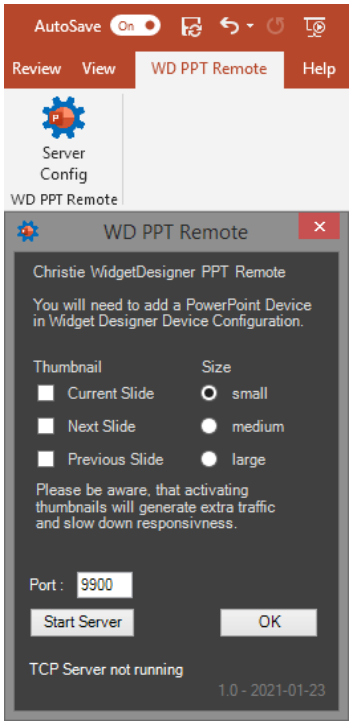
You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Phidgets_VoltageRatioInput1.Value`
or simply display it in the Debug Logger [812]: `DebugMessage(Phidgets_VoltageRatioInput1.Value)`
An alternative command is: Phidgets_VoltageRatioInput1.GetValue [1464]

7.6.15 PowerPoint

The PowerPoint connection in the Configuration dialog allows to easily remote control a PowerPoint presentation and receives feedback information that can be displayed and used via scripting. This involves the current and total slide number, speaker notes and even thumbnails. The PowerPoint feature consists out of two elements which communicate with each other via a TCP connection. Besides the "PowerPoint device" in Widget Designer, you need our PowerPoint plugin (i.e. add-in).

Note that the PowerPoint device is only available in the licensed Widget Designer edition, not the Free version.

Configuring the PowerPoint Add-In



The plugin can be downloaded from our Download-Center. Install it on the PC that runs the PowerPoint presentation. Make sure it is a Windows system with Win8.1 or 10. The PowerPoint application should be from Office 365, Office 2010 or newer.

After the installation, simply start PowerPoint and look for the tab "**WD PPT Remote**". There, click the icon "**Server Config**" and "**Start Server**" to open the TCP connection.
Further, you can change the **port** number, for example if you have more than one PowerPoint PC transmitting information to the same Widget Designer.
With the **Thumbnail** options you can choose whether Widget Designer receives thumbnails from the previous, current, and/or next slide. You can also choose from three different **sizes**: Small-192x108, Medium-320x180, Large-640x360 pixels.
Each time, when a slide is changed, PowerPoint generates new thumbnails and transmits them. Please be aware that this does not only increase network traffic but also slows down the responsiveness. In the moment when PowerPoint creates thumbnails, no commands can be processed.
You can close the dialog and then start the slide show.

There is no restriction in the number of open PowerPoint presentations. However, you will note, that there is only one TCP Server for all open instances. The presentation that is in "Slide Show" mode transmits information via the add-in. In other words, if you have multiple presentations, go ahead and load them as usual but make sure that only one of them is presenting and the others are in normal view mode. As soon as another presentation is toggles in Slide Show mode, the transmitted information adopts.

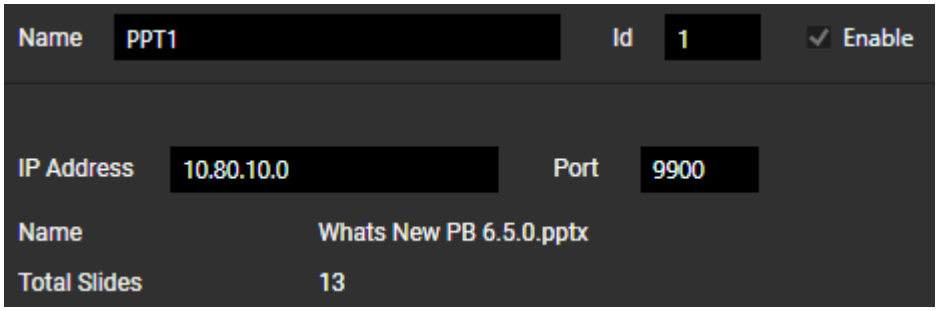
As mentioned, you can connect multiple PowerPoint systems to one Widget Designer. You just need to choose another port and add more "PowerPoint devices" in WD.

Further, you can connect multiple Widget Designer systems to one PowerPoint add-in. Technically said, the add-in starts a TCP Server that can connect to multiple TCP Clients which is the PowerPoint device in the Configuration dialog. Information (i.e. feedback) is sent to all Clients at the same time and commands can be received from either one of them. That allows remote controlling the same presentation from various points.

Adding a New PowerPoint Device

To add a PowerPoint device in Widget Designer, open the Devices menu and select "PowerPoint > Create PowerPoint". This opens the Configuration dialog [1305]. Alternatively, you can add a new device in the Configuration dialog with the "+" button.

On the right side you see several options:



The **Name** is the unique identifier for this PowerPoint object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your PowerPoint device is a filled blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A gray circle indicates a disabled device.

Now, please enter the **IP address** of the PowerPoint system. If needed, adjust the **Port**. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

If the connection is successful, the **Name** displays the file name and **Total Slides** shows the number of all saved slides. Other information can be received via scripting or an Event Listener, both described below.

You can close the dialog now. The newly created device will also be added to the Devices menu > PowerPoint > PPT and can be opened from here or with Devices menu > Configuration.

Using the Device in Regular Scripting

After creating a device, you can also use it via scripting which allows to perform actions in the PowerPoint application as well as retrieve information from it.

To send commands to the PowerPoint application, you can use a CustomScript button [822] or the Macro editor [1897]. For calling the previous or next slide, the PowerPoint Display [860] is most likely the best choice.

Whichever scripting field you choose, as soon as you enter the device's identifier name (per default that is "PPT1"), Script Assist will offer you a list of all PowerPoint Members [1471].

You can for example trigger the next action (which is your specified animation or text):

```
PPT1.Next
```

... or jump to a specific slide number:

```
PPT1.GoTo(3)
```

You can also retrieve information via scripting, e.g. the current slide number:

```
vint = PPT1.GetCurrentSlide
```

Using the Device with Event Listeners

Event Listeners can be added to the Configuration dialog and provide a simple way of listening to triggers from connected devices.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and PowerPoint Events [1470] for an overview (with examples and description) over the events raised by the add-in.

Using the Device as a Widget

There is also a PowerPoint Display [860] widget available which shows the thumbnails sent from PowerPoint and allows to execute a script if you click on the image, hence it is a perfect control to see the current slide and call the next one for example.

7.6.15.1 PowerPoint Events

This chapter gives an overview of the events that are raised by a PowerPoint connection and that can be used in the Event Listener [1353].
Please read the chapters " PowerPoint [1468]" and "Event Listener" [1353] if you like to know how to add and use both. The available PowerPoint members that can be used in our scripting language are explained in the chapter " PowerPoint Members" [1471].

CurrentSlide

Example:
vint = currentSlide

This event is raised whenever the slide number changes.
The event returns one parameter to WD which is an integer value with the name "currentSlide" and holds the slide number, e.g. 4.

If you select this event and copy the example into the scripting field of the Event Listener, it will assign the number of the current slide to the integer variable [1900] with the name "vint" whenever the slide number changes.

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the TCP client dis-/connects.

LastCommandReceived

Example:
Label1.Text = LastCommandReceived

This event is raised whenever the connected PowerPoint add-in receives a command.
The event returns one parameter to WD which is a string value with the name "LastCommandReceived" and holds for example "PPT Internal" if the last command was raised by the PowerPoint application itself. If WD triggers an action, it would appear here instead, e.g. "NEXT" or "GOTO 3".
If you select this event and copy the example into the scripting field of the Event Listener, it will write the last command into the Label [888] with ID 1 whenever an action was triggered.

7.6.15.2 PowerPoint Members

This page is still under construction.

Please check out our Online Manual for updates!

7.6.16 TCP Client

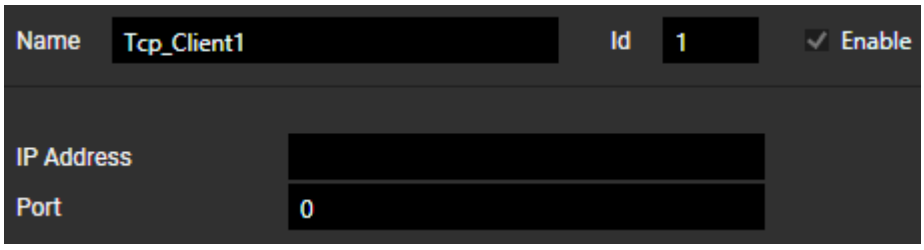
The TCP Client in the Configuration dialog allows to easily connect to another TCP server. The IP and port are scriptable which allows you to use the TCP Client as an interface that connects to various devices on the fly. Incoming messages, e.g. feedback or status information can be received via Event Listeners. Messages can be sent out via scripting. Note that the TCP Client is only available in the licensed Widget Designer edition, not the Free version.

The Connection Manager [1258] offers another possibility to add TCP Clients. Those connections can also be used by the node system which allows a more advanced message handling or is more useful for streaming information.

Adding a New TCP Client

To add a TCP Client, open the Devices menu and select "TCP Client > Create TCP Client". This opens the Configuration dialog [1305]. Alternatively, you can add a new TCP Client in the Configuration dialog with the "+" button.

On the right side you see several options:



The **Name** is the unique identifier for this object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your TCP Client is a filled blue circle. An empty blue circle indicates an enabled device where the IP address is not available. A filled gray circle indicates a disabled device.

Now, please enter the **IP Address** and **Port** of the remote TCP server. Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

You can close the dialog now. The newly created connection will also be added to the Devices menu > TCP Client and can be opened from here or with Devices menu > Configuration.

Using the Connection in Regular Scripting

After creating the connection, you can use it to send or receive messages via scripting.

To send messages to the TCP server, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Tcp_Client1"), Script Assist will offer you a list of all TCP Client Members [1473].

To send a message, choose the "Send" member. Literal text should always be enclosed in either single or double quotation marks otherwise the text is handled like a variable [1900]. You can also combine both like in the example. The global variable "Now" sends the current date and time:

```
Tcp_Client1.Send("Test")
Tcp_Client1.Send("This is my time: " + Now)
```

Displaying incoming messages is also easy:

```
vstring = Tcp_Client1.LastMessageReceived
```

Note that this assigns the last message any time the command is executed. If you like to assign it automatically whenever the message changes, use events instead.

Using the Connection with Event Listeners

Regarding any TCP or UDP connection, the Event Listener is the easiest way to handle incoming messages. Event Listeners can also be added to the Configuration dialog.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and TCP Client Events [1472] for an overview (with examples and description) of the available events.

7.6.16.1 TCP Client Events

This chapter gives an overview of the events that are raised by a TCP Client and that can be used in the Event Listener [1353].
Please read the chapters "TCP Client [1471]" and "Event Listener" [1353] if you like to know how to add and use both.
The available TCP Client members that can be used in our scripting language are explained in the chapter "TCP Client Members" [1473].

IsConnected

Example:
Label1.Text = isConnected

This event is raised whenever the connection state changes from disconnected to connected or vice versa. The event returns one parameter to WD which is a Boolean value with the name "isConnected" and holds either "true" or "false".
If you select this event and copy the example into the scripting field of the Event Listener, it will write "true" or "false" into the Label [888] with ID 1 whenever the TCP server dis-/connects.

MessageReceived

Example:
Label1.Text = MessageReceived

This event is raised whenever the connected TCP server sends a message.
The event returns one parameter to WD which is a string value with the name "MessageReceived" and holds the message.
If you select this event and copy the example into the scripting field of the Event Listener, it will write the message into the Label [888] with ID 1 whenever the server sends one.

7.6.16.2 TCP Client Members

This chapter gives an overview of the members available for the TCP Client in the Configuration dialog. Please read the chapter "TCP Client" [1471] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Disable

Tcp_ClientID.Disable

Example:
Tcp_Client1.Disable

This disables the TCP Client connection in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1473] member to reverse this command.

Enable

Tcp_ClientID.Enable

Example:
Tcp_Client1.Enable

This enables the TCP Client connection in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1473] member to reverse this command.

IpAddress

Tcp_ClientID.IpAddress

Example:
vstring = Tcp_Client1.IpAddress

This returns the IP address of the TCP Client connection as a string. The result could look as follows:
10.169.80.10
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Tcp_Client1.IpAddress`
or simply display it in the Debug Logger [812]: `DebugMessage(Tcp_Client1.IpAddress)`

In return, you can also assign an IP address to the TCP Client connection. WD automatically connects to it afterwards.

```
Tcp_Client1.IpAddress = "10.169.80.10"
```

IsConnected

Tcp_ClientID.IsConnected

Example:
vstring = Tcp_Client1.IsConnected

This member returns the connection status for the TCP Client connection as a Boolean value. The result is "True" if the connection is established and "False" if it is not.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Tcp_Client1.IsConnected`
or simply display it in the Debug Logger [812]: `DebugMessage(Tcp_Client1.IsConnected)`

IsEnabled

Tcp_ClientID.IsEnabled

Example:
vstring = Tcp_Client1.IsEnabled

This member returns the status of the check box "Enable" for the TCP Client connection in the Configuration dialog [1305] as a string or Boolean value. The result is "True" if the connection is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Tcp_Client1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Tcp_Client1.IsEnabled)`

LastMessageReceived

Tcp_ClientID.LastMessageReceived

Example:
vstring = Tcp_Client1.LastMessageReceived

This returns the last message that was received from the TCP Client connection as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Tcp_Client1.LastMessageReceived`
or simply display it in the Debug Logger [812]: `DebugMessage(Tcp_Client1.LastMessageReceived)`

Port

Tcp_ClientID.Port

Example:
vint = Tcp_Client1.Port

This returns the port number of the TCP Client connection in the Configuration dialog [1305] as an integer value.
The result could look as follows: 10001
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Tcp_Client1.Port`
or simply display it in the Debug Logger [812]: `DebugMessage(Tcp_Client1.Port)`

In return, you can also assign another port to the TCP Client connection. WD automatically connects to it afterwards.

```
Tcp_Client1.Port = 10002
```

Send

Tcp_ClientID.Send(data)

Example:
Tcp_Client1.Send("This is my time: " + Now)

This allows to send a custom message to the connected TCP server.

Literal text should always be enclosed in either single or double quotation marks otherwise the text is handled like a variable [1900]. You can also combine both like in the example. The global variable "Now" sends the current date and time.

7.6.17 UDP Client

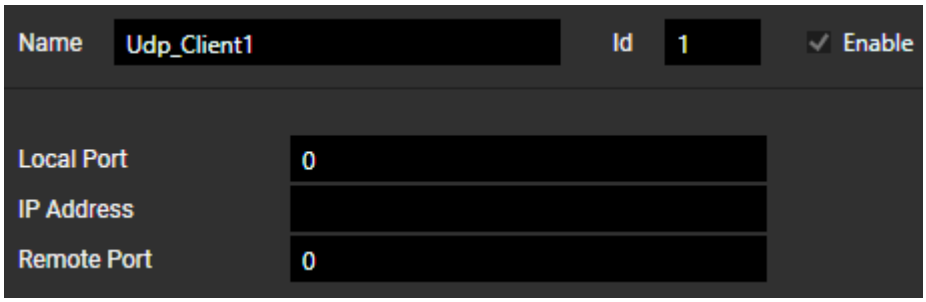
The UDP Client in the Configuration dialog allows to easily create a UDP connection. The UDP Client is perfect for scenarios where you like send UDP messages to a fixed address and optionally also receive messages. The UDP Server&Client connection however is more useful if you like to listen on a certain port and send messages to various addresses. The IP and ports are scriptable which allows you to use the UDP Client as an interface that connects to various devices on the fly if needed. Incoming messages can be received via Event Listeners. Messages can be sent out via scripting. Note that the UDP Client is only available in the licensed Widget Designer edition, not the Free version.

The Connection Manager [1258] offers other possibilities to add UDP connections. Those connections can also be used by the node system which allows a more advanced message handling or is more useful for streaming information.

Adding a New UDP Client

To add a UDP Client, open the Devices menu and select "UDP Client > Create UDP Client". This opens the Configuration dialog [1305]. Alternatively, you can add a new UDP Client in the Configuration dialog with the "+" button.

On the right side you see several options:



The **Name** is the unique identifier for this object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your UDP Client is a filled blue circle. An empty blue circle indicates an active connection but the used port is not available or blocked. A filled gray circle indicates a disabled device. Note that UDP is a stateless protocol and therefore has no connection status as known from the TCP protocol.

Now, please enter the **Local Port**, which is the source port the UDP Client listens on. In addition, this defines the sending port in case you are sending messages. If you not enter a specific port and leave "0", WD picks any free port. This is of interest if you are just sending messages, as the sending port is usually not important.

Next, enter the **IP Address** and **Remote Port** of the remote UDP server. This defines the destination of messages you are sending.

Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

You can close the dialog now. The newly created connection will also be added to the Devices menu > UDP Client and can be opened from here or with Devices menu > Configuration.

Using the Connection in Regular Scripting

After creating the connection, you can use it to send or receive messages via scripting.

To send messages to the UDP server, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Udp_Client1"), Script Assist will offer you a list of all UDP Client Members [1477].

To send a message, choose the "Send" member. Literal text should always be enclosed in either single or double quotation marks otherwise the text is handled like a variable [1900]. You can also combine both like in the example. The global variable "Now" sends the current date and time:

```
Udp_Client1.Send("Test")
Udp_Client1.Send("This is my time: " + Now)
```

Displaying incoming messages is also easy:

```
vstring = Udp_Client1.LastMessageReceived
```

Note that this assigns the last message any time the command is executed. If you like to assign it automatically whenever the message changes, use events instead.

Using the Connection with Event Listeners

Regarding any TCP or UDP connection, the Event Listener is the easiest way to handle incoming messages. Event Listeners can also be added to the Configuration dialog.

Please see the chapter Event Listener [1353] for a detailed description of its functionality and UDP Client Events [1476] for an overview (with examples and description) of the available events.

7.6.17.1 UDP Client Events

This chapter gives an overview of the events that are raised by a TCP Client and that can be used in the Event Listener [1353].
Please read the chapters "UDP Client [1475]" and "Event Listener" [1353] if you like to know how to add and use both.
The available UDP Client members that can be used in our scripting language are explained in the chapter "UDP Client Members" [1477].

MessageReceived

Example:
Label1.Text = MessageReceived

This event is raised whenever the connected UDP server sends a message.
The event returns one parameter to WD which is a string value with the name "MessageReceived" and holds the message.
If you select this event and copy the example into the scripting field of the Event Listener, it will write the message into the Label [888] with ID 1 whenever the server sends one.

7.6.17.2 UDP Client Members

This chapter gives an overview of the members available for the UDP Client in the Configuration dialog. Please read the chapter "UDP Client" [1475] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Disable

Udp_ClientID.Disable

Example:
Udp_Client1.Disable

This disables the TCP Client connection in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1477] member to reverse this command.

Enable

Udp_ClientID.Enable

Example:
Udp_Client1.Enable

This enables the TCP Client connection in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1477] member to reverse this command.

IpAddress

Udp_ClientID.IpAddress

Example:
vstring = Udp_Client1.IpAddress

This returns the IP address of the TCP Client connection as a string. The result could look as follows: 10.169.80.10
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Client1.IpAddress`
or simply display it in the Debug Logger [812]: `DebugMessage(Udp_Client1.IpAddress)`

In return, you can also assign an IP address to the TCP Client connection. WD automatically connects to it afterwards.

```
Udp_Client1.IpAddress = "10.169.80.10"
```

IsEnabled

Udp_ClientID.IsEnabled

Example:
vstring = Udp_Client1.IsEnabled

This member returns the status of the check box "Enable" for the TCP Client connection in the Configuration dialog [1305] as a string or Boolean value. The result is "True" if the connection is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Client1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Udp_Client1.IsEnabled)`

LastMessageReceived

Udp_ClientID.LastMessageReceived

Example:
vstring = Udp_Client1.LastMessageReceived

This returns the last message that was received from the TCP Client connection as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Client1.LastMessageReceived`
or simply display it in the Debug Logger [812]: `DebugMessage(Udp_Client1.LastMessageReceived)`

LocalPort

Udp_ClientID.LocalPort

Example:
vint = Udp_Client1.LocalPort

This returns the local port number of the TCP Client connection in the Configuration dialog [1305] as an integer value. The result could look as follows: 10001
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Client1.LocalPort`
or simply display it in the Debug Logger [812]: `DebugMessage(Udp_Client1.LocalPort)`

In return, you can also assign another local port to the TCP Client connection. WD automatically connects to it afterwards.
`Udp_Client1.LocalPort = 10002`

RemotePort

Udp_ClientID.RemotePort

Example:
vint = Udp_Client1.RemotePort

This returns the remote port number of the TCP Client connection in the Configuration dialog [1305] as an integer value. The result could look as follows: 10001
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Client1.RemotePort`
or simply display it in the Debug Logger [812]: `DebugMessage(Udp_Client1.RemotePort)`

In return, you can also assign another remote port to the TCP Client connection. WD automatically connects to it afterwards.
`Udp_Client1.RemotePort = 10002`

Send

Udp_ClientID.Send(data)

Example:
Udp_Client1.Send("This is my time: " + Now)

This allows to send a custom message to the connected TCP server.

Literal text should always be enclosed in either single or double quotation marks otherwise the text is handled like a variable [1900]. You can also combine both like in the example. The global variable "Now" sends the current date and time.

7.6.18 UDP Server&Client

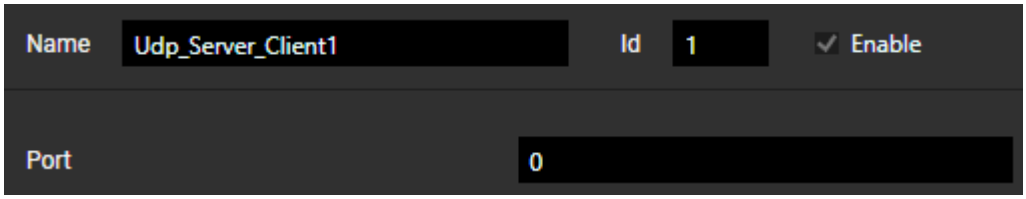
The UDP Server&Client in the Configuration dialog allows to easily create a UDP connection. The UDP Server&Client connection is perfect if you like to listen on a certain port and optionally send messages to various addresses. The UDP Client however is more useful for scenarios where you like send UDP messages to a fixed address and optionally also receive messages. The port is scriptable which allows you to use the UDP Server&Client as an interface that connects to various devices on the fly if needed. Incoming messages can be received via Event Listeners. Messages can be sent out via scripting. Note that the UDP Server&Client is only available in the licensed Widget Designer edition, not the Free version.

The Connection Manager [1258] offers other possibilities to add UDP connections. Those connections can also be used by the node system which allows a more advanced message handling or is more useful for streaming information.

Adding a New UDP Client

To add a UDP Server&Client, open the Devices menu and select "UDP Server/Client > Create UDP Server/Client". This opens the Configuration dialog [1305]. Alternatively, you can add a new UDP Server/Client in the Configuration dialog with the "+" button.

On the right side you see several options:



The **Name** is the unique identifier for this object in WD and the general rules for naming objects apply here as well (only letters, numbers and underscore are allowed, the first symbol must be a letter).
If you like, you can change the default name to a more descriptive one. When scripting, enter the name to access available members as described further down.

The **ID** is currently used internally only. In upcoming versions it will be possible to use it for addressing the device.

The "**Enable**" check box is ticked per default. On the left side, you should later see that the icon in front of your UDP Server/Client is a filled blue circle. An empty blue circle indicates an active connection but the used port is not available or blocked. A filled gray circle indicates a disabled device. Note that UDP is a stateless protocol and therefore has no connection status as known from the TCP protocol.

Now, please enter the **Port**, which is the source port the UDP Client listens on. In addition, this defines the sending port in case you are sending messages. If you not enter a specific port and leave "0", WD picks any free port. This is of interest if you are just sending messages, as the sending port is usually not important.

Click the "**Apply**" button to save any changes done here. You can also use the shortcut [Ctrl + Enter] to do so.

You can close the dialog now. The newly created connection will also be added to the Devices menu > UDP Client and can be opened from here or with Devices menu > Configuration.

Using the Connection in Regular Scripting

After creating the connection, you can use it to send or receive messages via scripting.

To send messages to the UDP server, first create a CustomScript button [822] or use the Macro editor [1897] or anything else with a scripting field. If you enter the device's identifier name into the script field (per default that is "Udp_Client1"), Script Assist will offer you a list of all UDP Client Members [1477].

To send a message, choose the "Send" member. Literal text should always be enclosed in either single or double quotation marks otherwise the text is handled like a variable [1900]. You can also combine both like in the example. The global variable "Now" sends the current date and time:

```
Udp_Server_Client1.Send("2.255.255.255",55550,"Test")
Udp_Server_Client1.Send("2.255.255.255",55550,"This is my time: " + Now")
```

Displaying incoming messages is also easy:

```
vstring = Udp_Server_Client1.LastMessageReceived
```

Note that this assigns the last message any time the command is executed. If you like to assign it automatically whenever the message changes, use events instead.

Using the Connection with Event Listeners

Regarding any TCP or UDP connection, the Event Listener is the easiest way to handle incoming messages. Event Listeners can also be added to the Configuration dialog. please see the chapter Event Listener [1353] for a detailed description of its functionality and UDP Client Events [1476] for an overview (with examples and description) of the available events.

7.6.18.1 UDP Server&Client Events

This chapter gives an overview of the events that are raised by a TCP Client and that can be used in the Event Listener [1353].
Please read the chapters "UDP Server&Client [1479]" and "Event Listener" [1353] if you like to know how to add and use both. The available UDP Client members that can be used in our scripting language are explained in the chapter "UDP Server&Client Members" [1480].

MessageReceived

Example:
Label1.Text = Label1.Text = Data + " from IP: " + Address + " port " + Port

This event is raised whenever the connected UDP server sends a message.
The event returns three parameters to WD. The first is a string value with the name "Data" and holds the message. The second is also a string value with the name "Address" and holds the remote IP address. Lastly, "Port" is an integer value and holds the remote port which is of interest when replying to this message.
If you select this event and copy the example into the scripting field of the Event Listener, it will write a message like "SomeCustomText from IP: 2.0.0.2 port 55551" into the Label [888] with ID 1 whenever the server sends one.

7.6.18.2 UDP Server&Client Members

This chapter gives an overview of the members available for the UDP Client in the Configuration dialog.
Please read the chapter "UDP Server&Client" [1479] if you like to know how to add and use this device. The chapter Object and Member Notation [1904] in Widget Designer explains how to use members in general.

Disable

Udp_Server_ClientID.Disable

Example:
Udp_Server_Client1.Disable

This disables the TCP Client connection in the Configuration dialog [1305] as if the "Enable" check box in the window was unchecked. Use the `.Enable` [1481] member to reverse this command.

Enable

Udp_Server_ClientID.Enable

Example:
Udp_Server_Client1.Enable

This enables the TCP Client connection in the Configuration dialog [1305] as if the "Enable" check box in the window was checked. Use the `.Disable` [1480] member to reverse this command.

IsEnabled

Udp_Server_ClientID.IsEnabled

Example:
vstring = Udp_Server_Client1.IsEnabled

This member returns the status of the check box "Enable" for the TCP Client connection in the Configuration dialog [1305] as a string or Boolean value. The result is "True" if the connection is enabled and "False" if it is currently not enabled.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Server_Client1.IsEnabled`
or simply display it in the Debug Logger [812]: `DebugMessage(Udp_Server_Client1.IsEnabled)`

LastMessageReceived

Udp_Server_ClientID.LastMessageReceived

Example:
vstring = Udp_Server_Client1.LastMessageReceived

This returns the last message that was received from the TCP Client connection as a string value.
The first example shows how to use it with an existing (global) string variable [1900] named "vstring".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Server_Client1.LastMessageReceived`
or simply display it in the Debug Logger [812]:
`DebugMessage(Udp_Server_Client1.LastMessageReceived)`

Port

Udp_Server_ClientID.Port

Example:
vint = Udp_Server_Client1.Port

This returns the port number of the TCP Client connection in the Configuration dialog [1305] as an integer value.
The result could look as follows: 10001
The first example shows how to use it with an existing (global) integer variable [1900] named "vint".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = Udp_Server_Client1.Port`
or simply display it in the Debug Logger [812]: `DebugMessage(Udp_Server_Client1.Port)`

In return, you can also assign another port to the TCP Client connection. WD automatically connects to it afterwards.
`Udp_Server_Client1.Port = 10002`

Send

Udp_Server_ClientID.Send(address,port,data)

Example:
Udp_Server_Client1.Send("2.255.255.255",55550,"This is my time: " + Now")

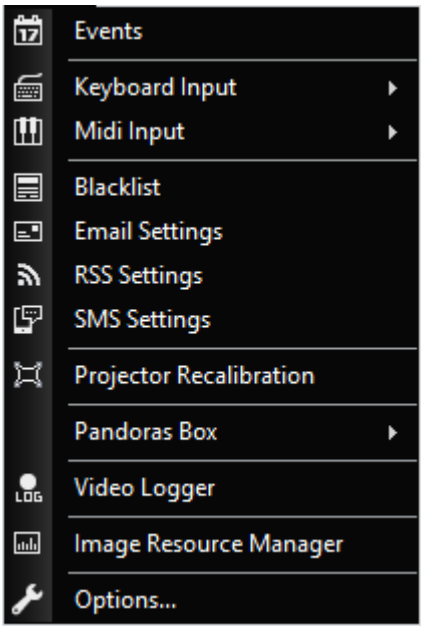
This allows to send a custom message to the address 2.255.255.255 and port 55550.

Literal text should always be enclosed in either single or double quotation marks otherwise the text is handled like a variable[1900]. You can also combine both like in the example. The global variable "Now" sends the current date and time.

7.7 Tools

The Tools menu lists all tools available in Widget Designer. In contrast to a physical hardware device [1277], a tool is rather a software tool. In addition, it is also possible to set up a connection [1255], e.g. a TCP connection.

Once you have chosen an entry a dialog opens with more options.



Events [1483]
Keyboard Input [1483]
Midi Input [1485]
Blacklist [1490] (not included in the WD Free Edition)
Email Settings [1490] (not included in the WD Free Edition)
RSS Settings [1493] (not included in the WD Free Edition)
SMS Settings [1494] (not included in the WD Free Edition)
Projector Recalibration [1496]
Pandoras Box [1503]
Video Logger [1507]
Image Resource Manager [1509]
Options… [1510]

7.7.1 Events

The Event Editor is now available in the EventViewer widget [871] itself.

7.7.2 Keyboard Input

Widget Designer supports multiple hardware Keyboard interfaces. You may setup any keyboard interface with scripts to define the custom button actions.

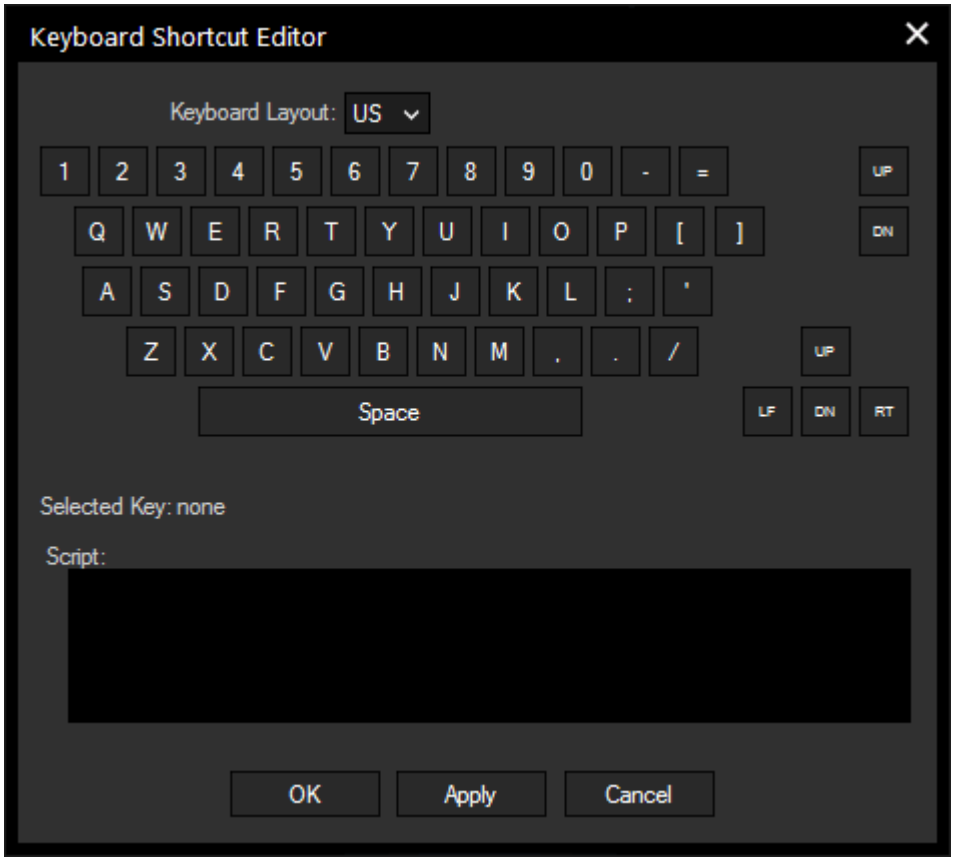
To setup any Keyboard input click on the desired button that you want to setup and enter the script into the text field. To store the script for the button press apply or close to apply and exit the editor.

See here the list of optional keypads:

The Shortcut Editor [1484] allows you to customize keyboard shortcuts for direct script control.

7.7.2.1 Shortcut Editor

Widget Designer supports customizable keyboard shortcuts for direct script control.



Please choose the Keyboard Layout (US or DE) from the list.

Now select the key that should be assign with a command either by clicking on it in the Keyboard Shortcut Editor or by pressing the key on your keyboard.

The key will be displayed above the text field, e.g.: Selected Key: W.

In the text box you may enter commands to be executed. The topic Script Language [1511] explains this in more detail.
See here a list of all commands [1511]. If you want to disable / enable the keyboard shortcuts you set up, please use the commands `WDKeyboardShortcutsDisable` [1748] **and** `WDKeyboardShortcutsEnable` [1748]**.**

7.7.2.2 Logitech Presenter

With the Editor for the Logitech Presenter R400 you can setup which commands should be executed upon a certain button press.

First, check the "Enabled" check box. To enable / disable the Logitech Presenter e.g. from a Custom Script Button, these commands [1552] are available.

Now, click one of the buttons [1] to [5] to assign a custom WD command to it. The selected button will be displayed above the scripting field, e.g.: "Button 1 Press Script".

In the text box you may enter commands to be executed. The topic Script Language [1511] explains this in more detail.

7.7.3 Midi Input

Choose between these Midi Inputs:

Midi Notes [1486]
You may choose to setup Midi Notes On/Off commands by using the Midi Note Editor.

Midi Messages [1487]
If you require intercepting any Midi Message you may use the Midi Message Editor.

MIDI DEVICE SUPPORT
If you wish to use an additional Midi Interface to control Widget Designer you may choose one of these supported midi devices to map its buttons and fader to scripts, parameters and value ranges.

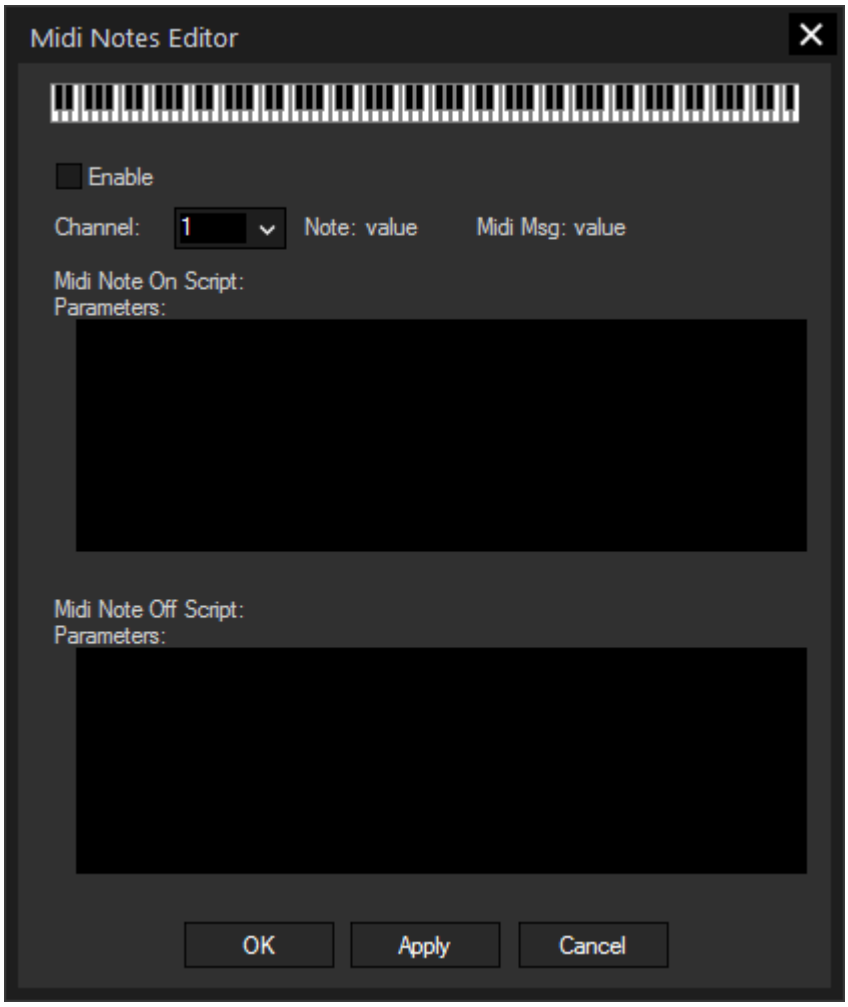
AKAI APC40 [1488]

BCD 2000 (has been removed as it runs on Windows XP 32bit only)
BCF 2000 [1489]

7.7.3.1 Midi Notes Editor

Widget Designer supports standard Windows Midi Devices.

You may choose to setup Midi Notes On/Off commands by using the Midi Note Editor.



In order to receive any Midi Note Message please first setup the Midi Input device in the Connection Manager [1258].

Tick the check box [Enable] to activate the Midi Notes Editor and remove the check if you want to disable incoming Midi Notes again.

Once this is up and running you will be able to see the incoming Midi Note messages in the Midi Note editor.

Scripts

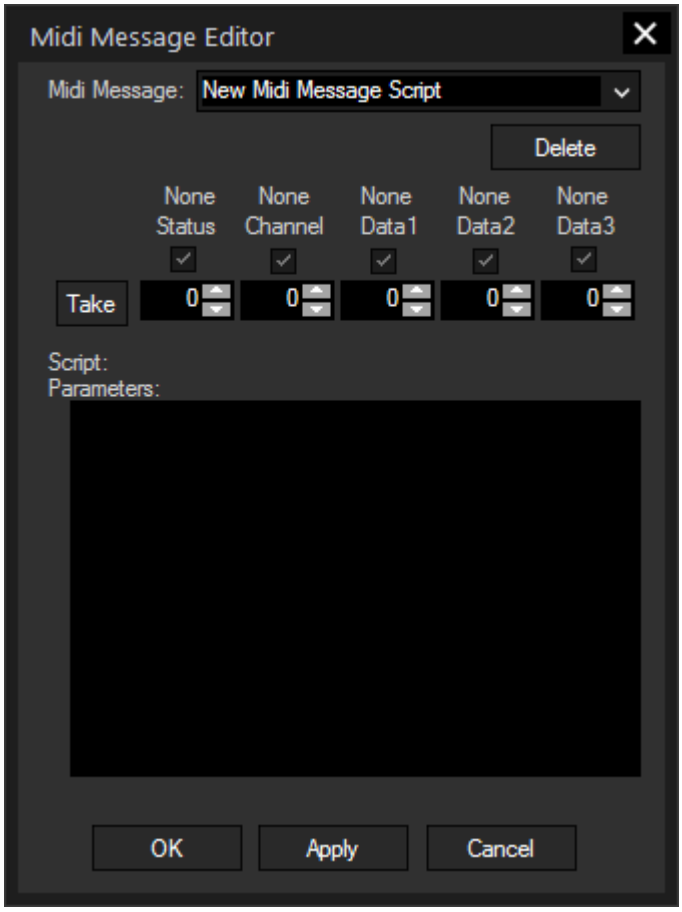
In the Script section you may enter commands to be executed as Midi Note On and/or Midi Note Off script. You may use the drop-down list and the "Add" button or type directly in the text field. The topic Script Language [1511] explains this in more detail.
See here a list of all commands [1520].
To delete a script text box entirely, click [Clear All] above each script.

If you would like to setup the Midi Note On/Off scripts offline, just click on the graphical piano to choose the desired note and enter the desired Note On and/or Note Off script.

Press Apply to store the script for this note.

7.7.3.2 Midi Message Editor

If you require intercepting any Midi Message you may use the Midi Message Editor.



In order to receive any Midi Note Message, please first setup the Midi Input device in the Connection Manager [1258].

With the Midi Message Editor you can filter out specific Midi Data and store commands like a Macro into a List of Midi Message Scripts. This list you can store for your Widget Designers project.

Use the check boxes to define if you wish the script to ignore or filter the related Midi Data.

Use the [Take] button to grab the incoming Midi Data values and store them with the script [1511] as Midi Message Script.

To Store a new Midi Message Script enter a new Name in the top drop-down list and press [Apply] or [OK].

If you wish to delete an existing Midi Message Script, pick the Midi Message Script to delete from the top drop-down list and press the Delete button.

7.7.3.3 AKAI APC40

If you wish to use an additional Midi Interface to control Widget Designer you may choose a AKAI APC40 to map its buttons and fader to commands [1511], parameters and value ranges.

The AKAI APC40 allows using a separate Midi input device which is set up in the top section of the AKAI APC40 Editor:

To enable the Midi Input for the AKAI APC 40 board please choose your Midi Receive Device from the list and press [Apply].

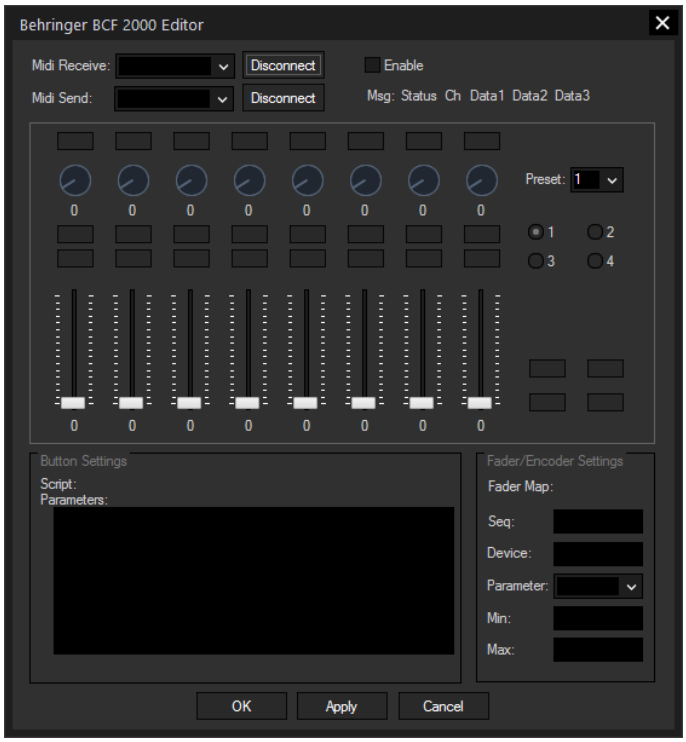
Tick the check box [Enable] to enable the AKAI APC 40 board and remove the check if you want to disable the device again.

Once connected to the AKAI APC40 you may press a button or move a control to let Widget Designer intercept and give you access to the Controls scripts or Fader Encoder Settings.

To change any setting press on the control, enter your values and press [Apply].

7.7.3.4 Behringer BCF 2000

If you wish to use an additional Midi Interface to control Widget Designer you may choose a Behringer BCF 2000 to map its buttons and fader to commands [1511], parameters and value ranges.



The BCF 2000 allows using a separate Midi input device which is setup in the top section of the BCF 2000 Editor.

Since the BCF 2000 is a freely programmable Midi Interface you will need to use the Learn function of the BCF 2000 to store the Midi value from WD into the unit.

To do this first set up the Midi Send Device in Order to receive the values in the BCF2000:

To enable the Midi Input for the BCF 2000 board please choose your Midi Send Device from the list and press [Apply].

Tick the check box [Enable] to enable the BCF 2000 board and remove the check if you want to disable the device again.

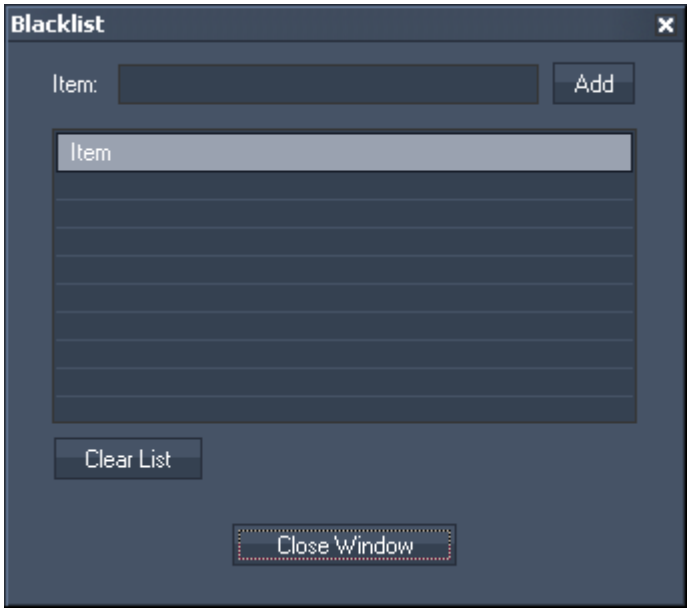
To store the faders, encoders and buttons you will need to select the desired page in both the BCF 200 Editor and on the BCF 2000 Control itself.

Once this is done, hold the learn button on the BCF and press or move any control item to go to the learn mode. Now click or click and drag any fader in the BCF 2000 Editor with the mouse until the BCF 2000 LCD Menu reads "Good".

To store commands or a value range for the individual control, click on the controls and enter your values and confirm with [Apply] or [OK] to store the value in Widget Designer.

7.7.4 Blacklist

Use the blacklist for filtering incoming Emails [1490], SMS [1494] and RSS feeds [1493] for specific words.

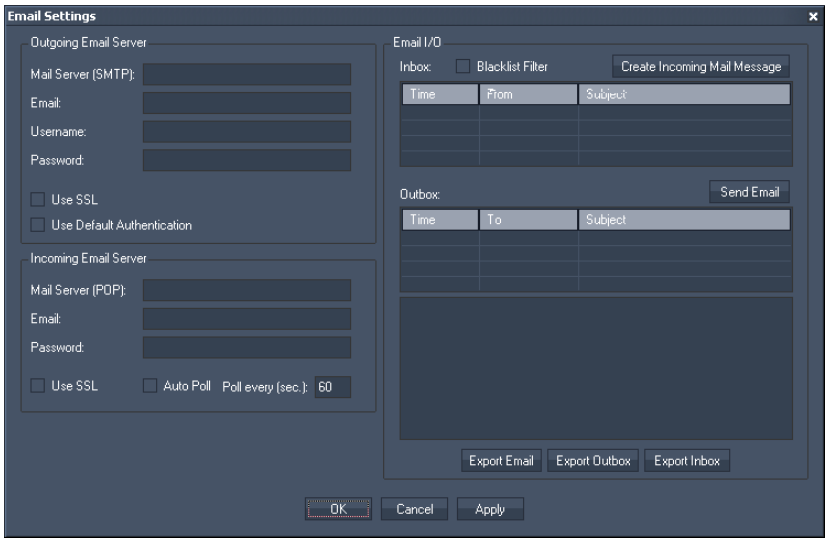


Enter the keywords one after another in the Item text field. Apply each keyword by pressing [Add]

To clear a single keyword from the list, right-click on this entry and choose "Delete".
To clear the whole list, use the button [Clear List].

7.7.5 Email Settings

The Email Settings is the tool to setup all incoming and outgoing Email communication.
To process the incoming emails inside Widget Designer, also use the Email commands "EmailSend,'To','Subject','Message'" and "EmailSendAtt,'To','Att','Subject','Message'", see script language [1511].
Email Input Node [1068].
To process any data inside Widget Designer to send via Email, also use the dedicated .



Outgoing Email Server

Please enter all relevant information for your SMTP Mail Server settings, such as
- Mail Server (SMTP),
- Email Address,
- Username,
- Password.

Tick the check box "Use SSL" to use SSL (Secure Socket Layer).
Tick the check box "Use Default Authentication" to use default authentication.

Incoming Email Server

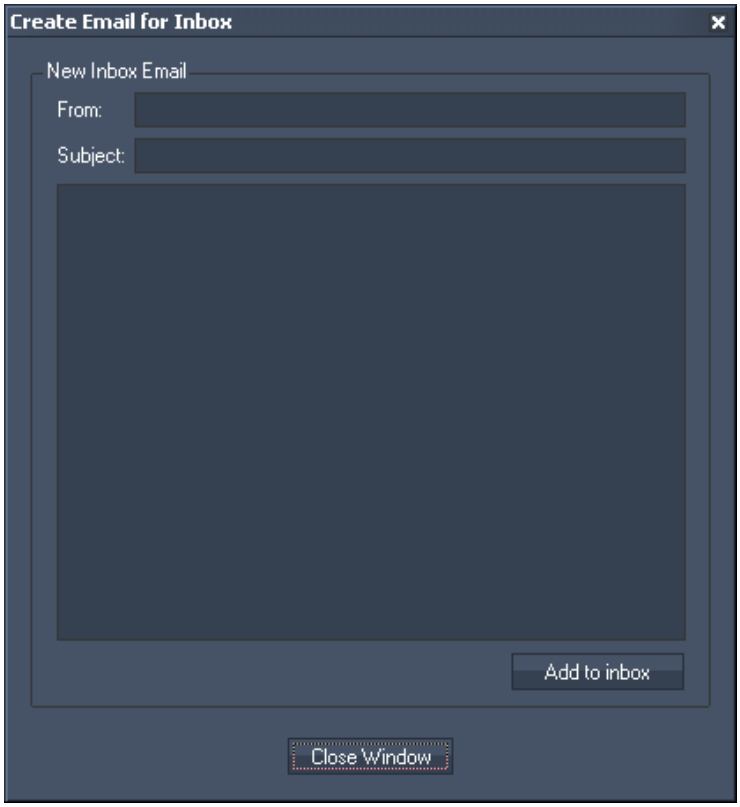
Please enter all relevant information for your POP Mail Server settings, such as
- Mail Server (POP),
- Email Address,
- Password.

Tick the check box "Use SSL" to use SSL (Secure Socket Layer).
Tick the check box "Auto Poll" to automatically poll your email account. Enter the interval time in the text field aside.
If Auto Poll is not checked, your Email account will only be polled once when pressing Apply.

Email I/O

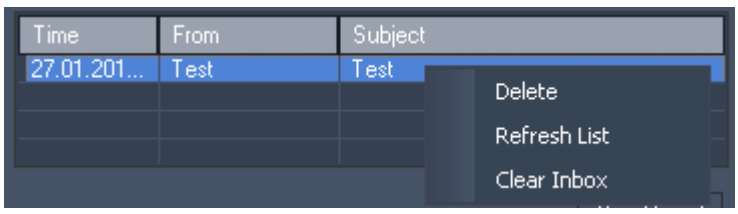
Inbox:
To use the Blacklist [1490] filter, please check this option.

Click on "Create Incoming Mail Message" to get an email message without polling it from the mail server.
The following pop-up window opens:



Click "Add to inbox" to receive this email in your inbox, click on Close Window if you do not want to send the email.

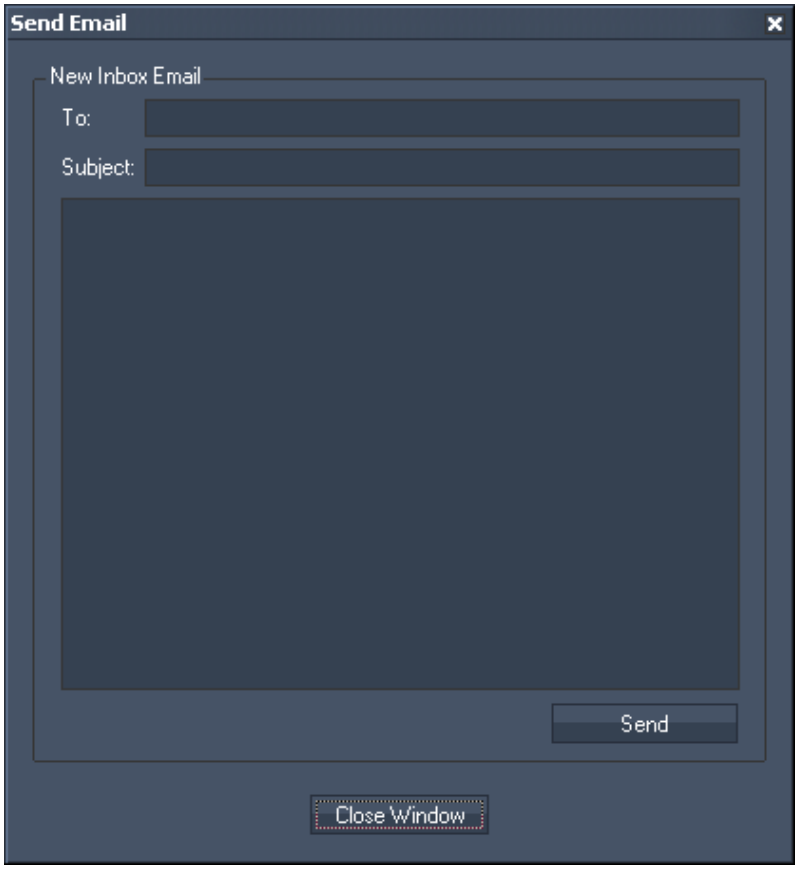
If you right-click on an email in your inbox, you have the opportunities to:



- Delete this mail,
- Refresh the Inbox List,
- Clear the Inbox.

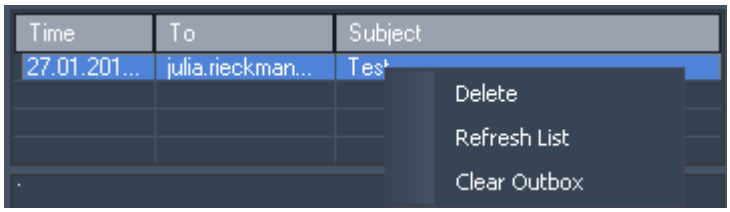
Outbox:
In the Outbox all emails sent will be listed.

Click on "Send Email" to create an email directly in the outbox.
The following pop-up window opens:



Enter the receiver, subject and message and press "Send" to send this email or directly "Close Window" to leave this dialogue.

If you right-click on an email in your outbox, you have the opportunities to:



- Delete this mail,
- Refresh the Outbox List,
- Clear the Outbox.

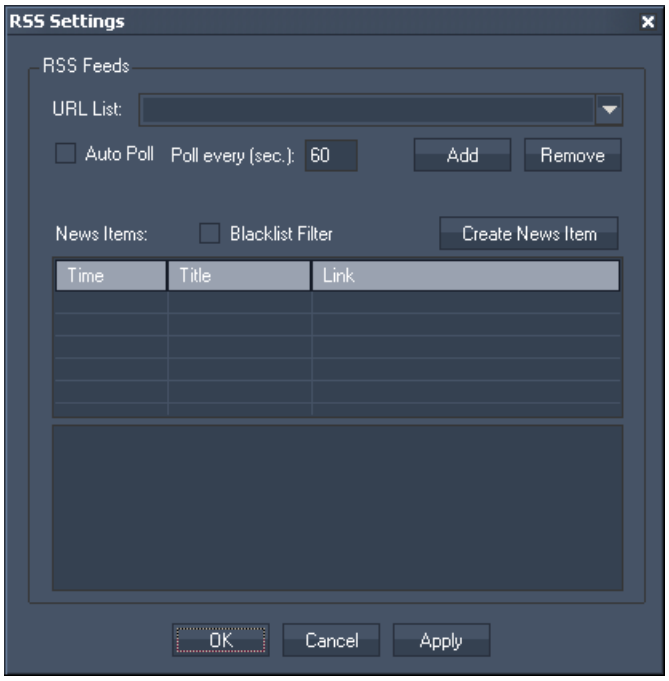
Export Email:
Exports all emails addresses listed in the inbox into a text file.

Export Outbox:
Exports all emails listed in the outbox into a text file.

Export Inbox:
Exports all emails listed in the inbox into a text file.

7.7.6 RSS Settings

The RSS Settings is the tool to setup and create all RSS feeds.
To process the incoming RSS feed inside Widget Designer, also use the RSS Input Node [1072].



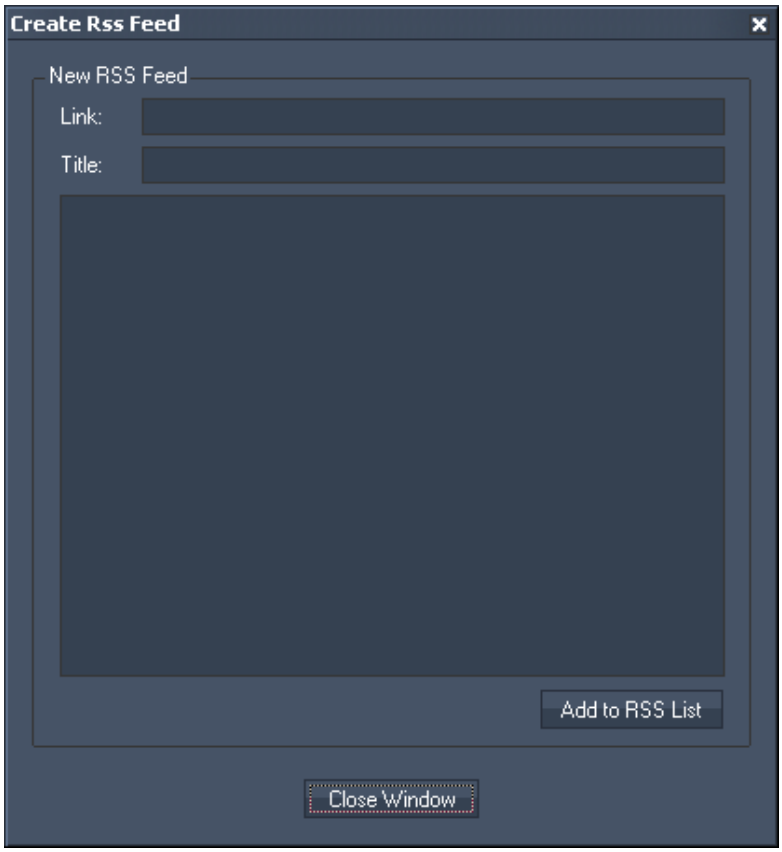
URL List:
Enter a new URL manually in the text field or choose one from the list.
To add an URL to the list, press [Add] after you entered it into the text field. Click [Remove] to remove this entry from the list.

Check "Auto Poll" and enter the Poll interval in seconds to automatically poll the RSS Feed. If this option is unchecked the RSS Feed will only be polled once when pressing [Apply] or [OK].

To use a prepared Blacklist [1490] for filtering the RSS Feeds for specific words, please check the option Blacklist Filter.

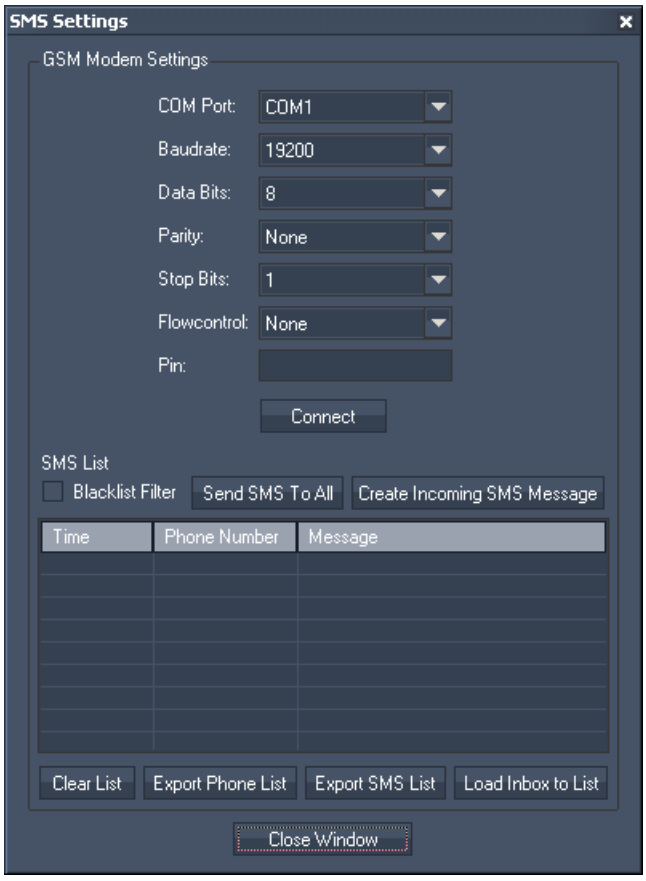
All received RSS Feeds will be displayed in the incoming RSS list.

Create new Items:
Click on this button create a RSS feed manually.
In the upcoming dialog enter the link that should be displayed, a title and the RSS message text.
Press [Add to RSS List] to transfer it to the incoming RSS list.



7.7.7 SMS Settings

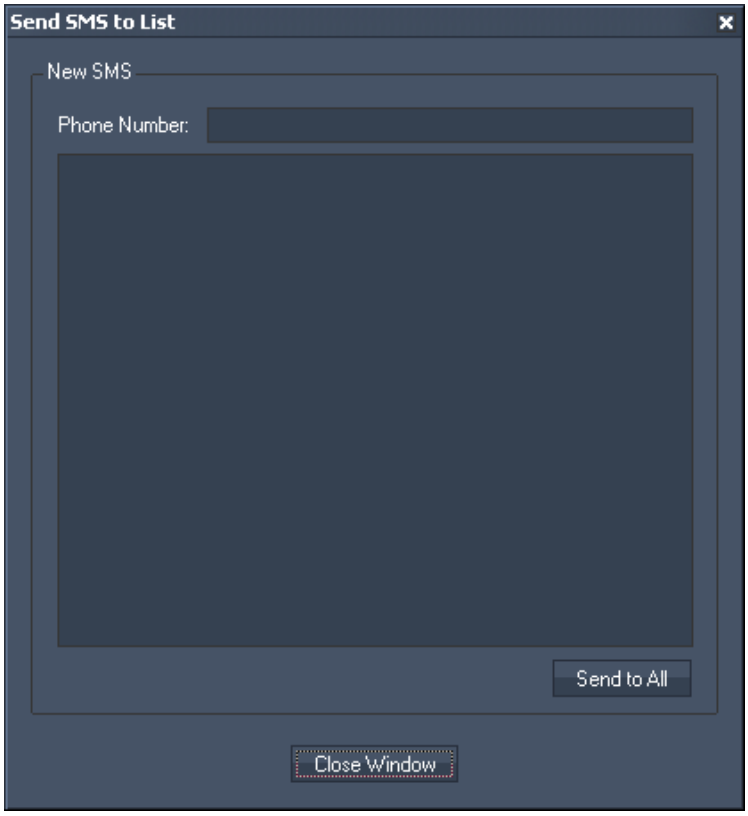
Setup your GSM modem in this tool to be able to receive and send SMS via the Widget Designer.



GSM Modem Settings:
Choose the COM Port, Baud rate, Data Bits, Parity, Stop Bits and the Flow control. Enter the Pin for the GSM Modem and press [Connect].

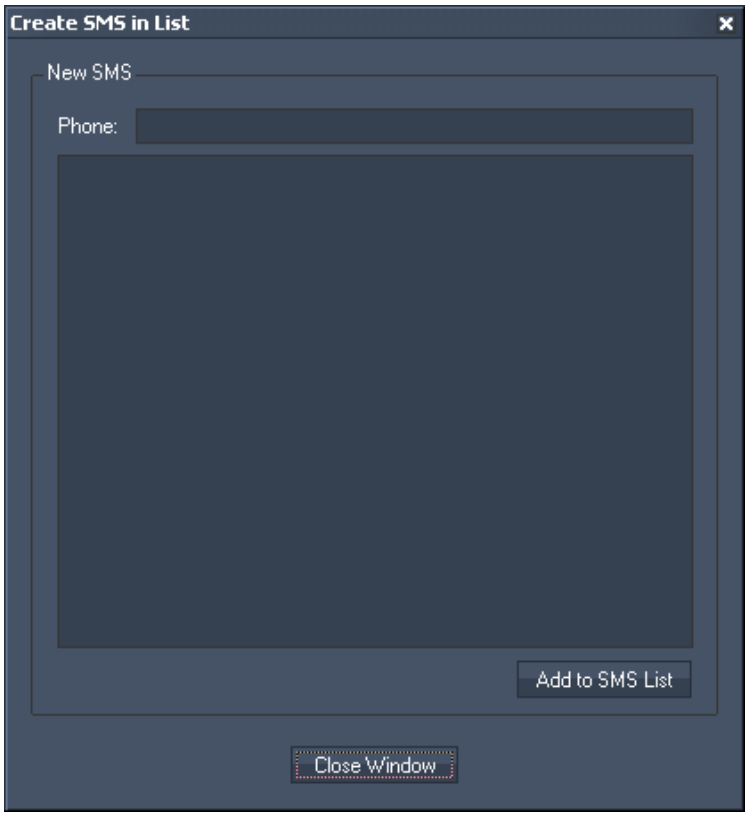
Blacklist Filter
To use a prepared Blacklist 1490 for filtering the incoming SMS for specific words, please check the option Blacklist Filter.

Send SMS To All:
Click on this Button to send a SMS message to all Phone numbers in the SMS List. The following pop-up window opens:



Additionally to all Phone numbers in the SMS list, the new SMS can be sent to any number entered in the text field next to Phone Number.
Enter the SMS text in the text box.

Create Incoming SMS Message:
Click on "Create Incoming SMS Message" to create a SMS message without polling it via the GSM Modem.
The following pop-up window opens:



Enter the phone number that should appear as faked SMS sender in the text field.

Use the text box to enter the SMS text.

Push [Add to SMS List] to add this entry to the SMS Inbox list.
If you want to discard this entry, just press [Close Window] without adding it.

Clear List:
Click this button to clear the SMS List.

Export Phone List:
Click this button to export all SMS sender's phone numbers to a text file.

Export SMS List:
Click this button to export all SMS (phone number and message text) to a text file.

Load Inbox to List:
Load all SMS from your SIM Card into the Inbox.

7.7.8 Projector Recalibration

The Projector Recalibration is a Tool needed to set up the connection between a Calibration Link [2000] and other Pandoras Box software components for achieving an automated recalibration of a warped projector output.

This chapter covers the concept of Recalibration realized in WD. The next chapters Hardware Setup [1497] and Software Setup [1498] offer for more detailed information about how to use this feature.

The Principle Behind "Recalibration"

Imagine the following scenario:
You have a fixed installation with a projector (or several ones) and a screen. You set everything up and it's perfectly warped and blended. But it lies in the nature of such installations that they do not remain perfectly warped and blended. May it be because the projector mount is a bit slaggy and sinks a couple of millimeters every few months, or someone moved the projector or screen accidentally. At least every time the device needs a new lamp or filter, it will have to be moved away from its perfectly warped position.

Or an even more extreme example: A stage setup in a theater, on a cruise ship or a touring concert. It happens often that projectors and / or screens need to be dismounted for certain events on a regular basis.

Warping and adjusting anew from scratch is not only time consuming and pricey, it also becomes quite annoying after a while.
This is the situation where the Recalibration jumps in: an easy to use tool that re-calibrates and re-adjusts the whole warping completely automatically, simply by pressing a button.

Our recalibration method does not require cameras, only a device called Calibration Link [2000] and fiber cables of which the ends are mounted into the screen as described in the next chapter [1497].
Furthermore, the normal setup procedure for the projection needs to be done, such as adjusting the focus and position. Then, you warp and soft-edge in PB once and store this setup with the Recalibration Tool as the so called "home location". From now on, this is your reference. If the projection is not perfect anymore and the recalibration process is started, WD calculates the offset and restores the home location. In other words, the projection is adjusted once in the beginning by video technicians. Afterwards, the recalibration process can be started with one button click and no further manual settings are necessary. This is useful when the installation is serviced by staff who does not know how to reset the warp and blend properly but also when time is short and the video technicians have many other jobs to do.

Of course the projected image needs to cover all calibration points (i.e. the fiber tips within the screen) at all times. So if the projection moved a lot, it first needs to be re-positioned and re-focused manually. The automated recalibration then takes care of the exact warping and blending to match the stored home location.

Currently only projections onto 2D-planes can be recalibrated.
The pixel accuracy that can be achieved in a best case scenario is 1mm / +- 1pixel accuracy.

The next chapters cover the Hardware Setup [1497] and Software Setup [1498].

7.7.8.1 Recalibration Hardware Setup

This chapter explains the physical setup of a WD Recalibration scenario, what devices you need and what to pay attention to.

Mounting the Fiber Cables

The Calibration Link is, in principle, a NET Link [2000] equipped with two Calibration Fiber input boards [2004]. Those boards allow measuring the light intensity levels of a 1mm fiber cable. The Recalibration process is based on these measurements. Hence, the quality of light transmission effects the quality of the process significantly.

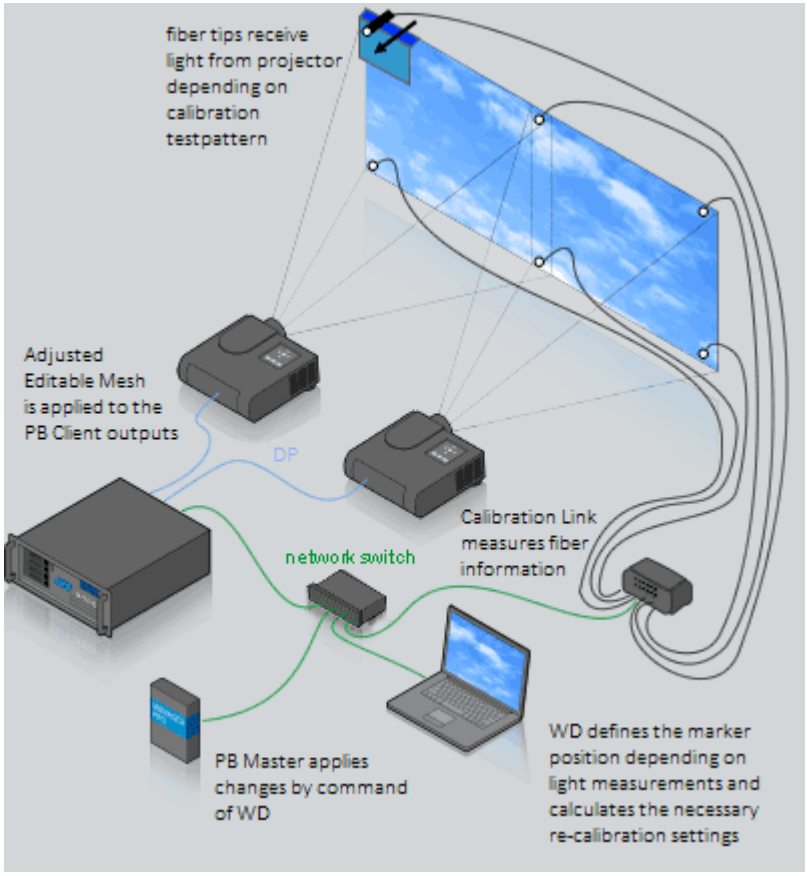
The cable ends are mounted into the screen in a way that the fibers "look" straight at the projector to catch as much light as possible. The more offset the cable ends have, the more precise the calibration can be. At best, they are mounted into the corners.
A minimum number of four cables is required per projector. However, one can be shared by several projectors. So if you have e.g. a soft edge blend with two projectors, you can use six fiber cables (two in the overlap, and the other four near the corner of the screen).

Direct sunlight needs to be avoided.
By the very nature of the cables, they should be bent as little as possible.
Also, the fiber cables that form a group for one projector should be equally long.
The maximum length depends on the brightness of the projector. The longer the fiber cable is, the more light needs to hit the tip of the cable. As a rule of thumb, a 10m fiber cable require at least 100lx. If you have 20m cables then you need to have a minimum brightness of 200lx and so on.

The cable ends must be clean and cut as straight as possible. It is highly recommended to use a professional fiber cutter to do so, as other tools squeeze the fiber and reduce the transmission quality. Neither end of the actual glass fiber itself must stick out from the cable sheath!
Simply insert the cut off end into the Calibration Link's fiber input and move it slightly forward until it withstands, then tighten the screw-able input.

It is recommended that the projector pixels are in a best focus position for all fiber points.

Hardware Setup



The graphic to the left depicts the communication between Pandoras Box (PB) components, Widget Designer (WD), Calibration Link and fiber cables.

First of all, a PB Master runs in stand-alone mode or is connected to other Clients. Widget Designer may run on the same or a separate hardware and is connected via network to the PB Master and the Calibration Link. Fiber cables run from the Calibration Link to the screen and measures light from a projector that is connected to the PB Client.

1. The user starts the recalibration process by clicking a button in WD.
2. WD tells the PB Master to load an Editable Canvas [256] into the designated Client's output.
3. WD sends a sequence of black and white calibration patterns to the Canvas which are displayed on the Client's output (i.e. the projector).
4. According to their position, the fiber cables receive a certain light amount from each calibration pattern.
5. The Calibration Link measures the light intensities and transmits this information via network to WD.
6. WD calculates the position from the fiber tips (markers) relative to the projected image.
7. WD calculates the necessary recalibration settings and sends them to PB.
8. PB applies the adjustments to an Editable Mesh [257] and loads it into the Client's output.

The whole process is completed within a few minutes.

The next chapter [1498] covers the software part and a step-by-step instruction how set this feature up in WD and PB.

7.7.8.2 Recalibration Software Setup

This chapter explains the software setup of WD and PB for a Recalibration scenario. If you want to know more about the physical setup, please refer to the previous chapter [1497].

If you are already experienced with using the Recalibration, you can find a shortened step-by-step list [1502] at the bottom of this page for an overview of the necessary steps.

Please make sure that you are using **WD version 6.1.0** and **PB version 6.1.3** (or higher) for this feature and have **.NET version 4.8** or higher installed.

Preparations in Pandoras Box

Before you start the Recalibration setup, please have your Master and Client setup up and running. Additionally, please add an Editable Mesh for each output that is supposed to be included in the Recalibration, and warp it according to your screen. The Recalibration tool relies on this warp file to determine the adjusted warp parameters.

The more exact your initial warp is, the more exact the Recalibration is going to be afterwards.

Please note that the original Mesh file that you created yourself will at no point be modified by WD.

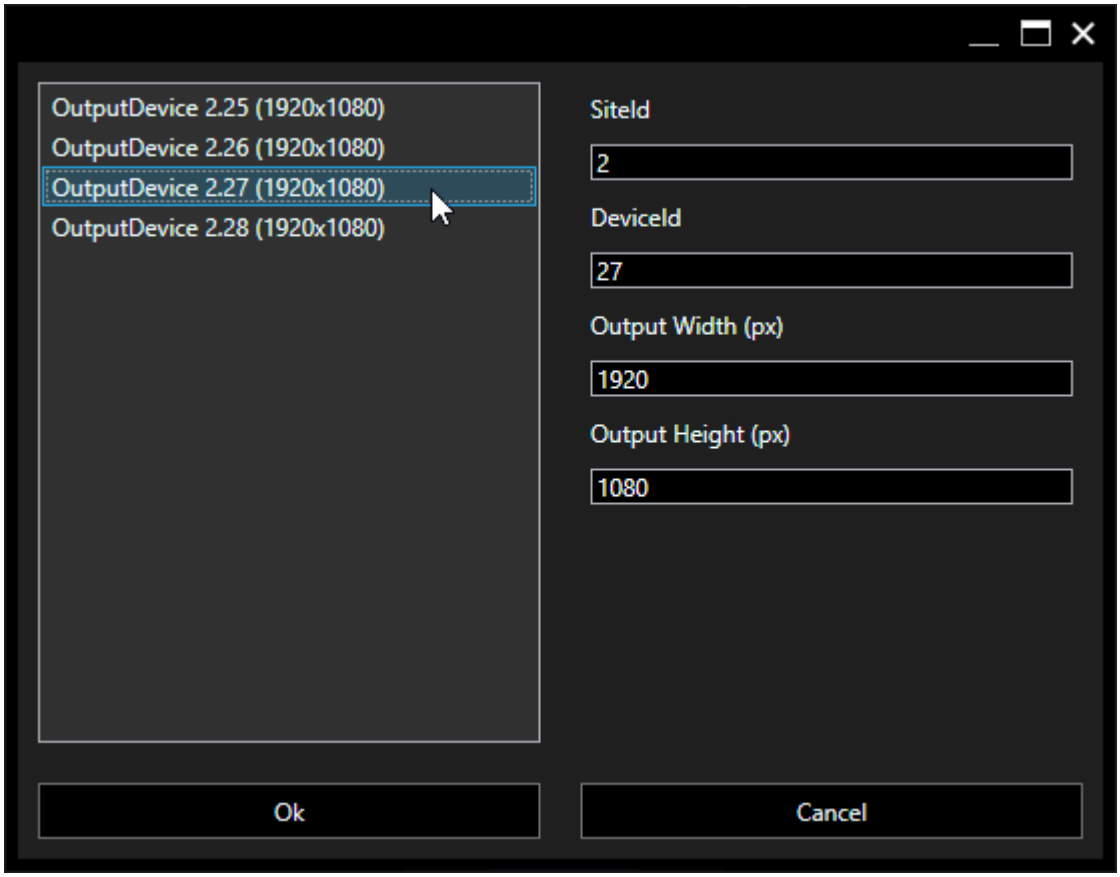
Setup in Widget Designer

First of all, you need to retrieve the measured light intensities. For this, please create a Calibration Link input node [981] and connect it to your Calibration Link hardware.
This node does not require to be connected to an output node.

When it is successfully connected, you should already see incoming values from your fibers. If you want to make sure that the physical placement of the fiber tips in the screen in relation to the projector is good, display a white image and check the received values in the Calibration Link dialog. The maximum intensity level is 1023. If you are not close to this value with a white pattern, please try optimizing your physical setup (i.e. for example the orientation of the fiber tips, see more at the bottom of this page).
Recalibration does also work with lower values, but you should aim for the best conditions possible.

Now make sure to establish a connection [1256] between WD and the PB master and then open the Projector Recalibration dialog from the Tools Menu [813].

When setting up the Recalibration for the first time, you need to select the desired PB outputs. Click on "**Add New**" to add one output to your setup.



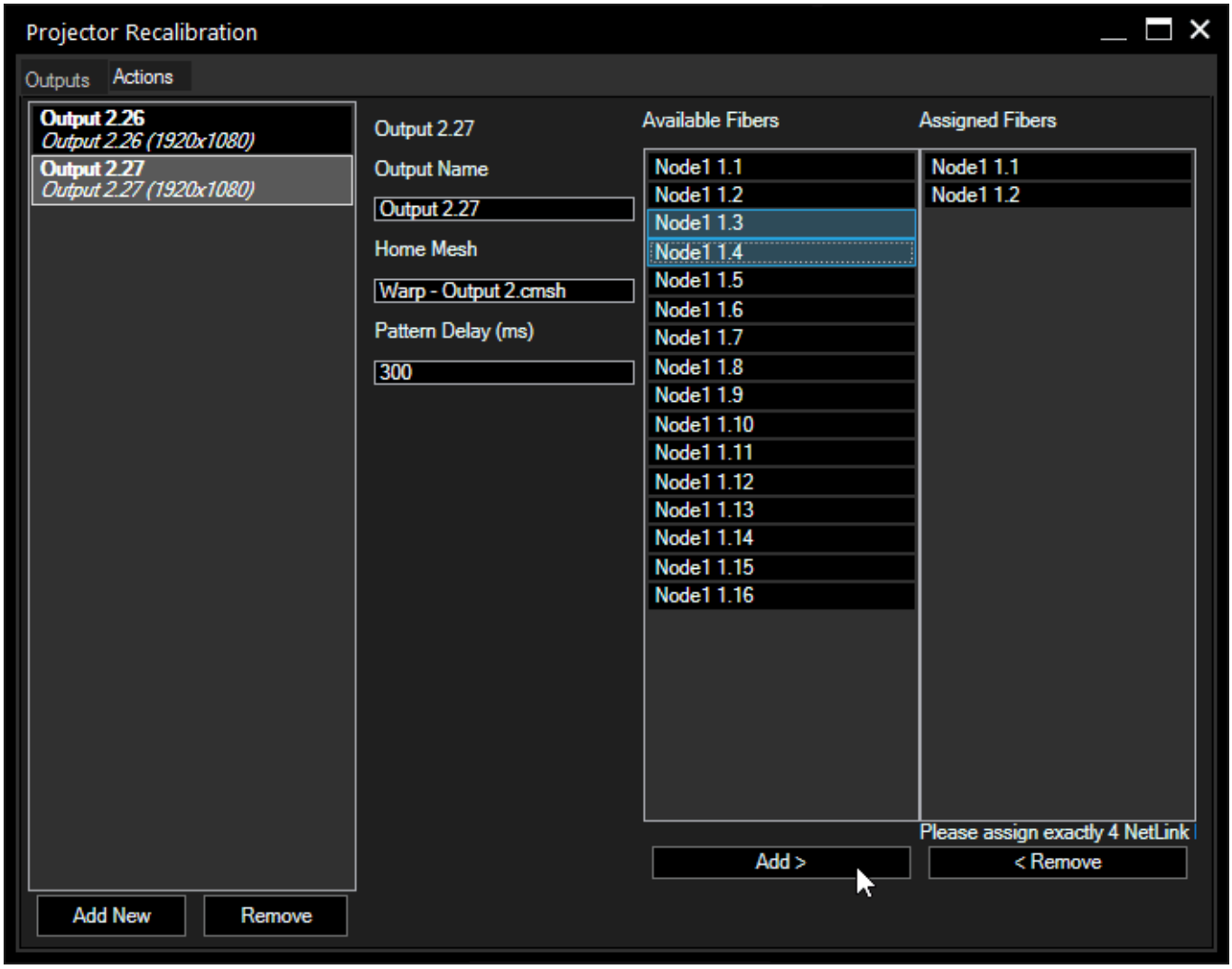
The Client outputs provided by the Pandoras Box Master are displayed in the list. Simply select one and press "**Ok**".

You can also add outputs manually by entering the Site and Device ID, as well as output resolution in the text fields.

Even so, it is recommended to use the settings offered by the list to prevent accidental errors.

The next step is to assign the available fiber inputs to each PB output.

Select the output you want to edit to display its settings:



Output Name:
Here you can enter a different name to be displayed for your output than the automatically assigned one. It can be helpful to keep a better overview when several projectors are supposed to be recalibrated in one project.
This is also the name with which this output can be accessed via script commands [1805] later.

Home Mesh:
Enter here the exact name and folder path as it is displayed in the PB Project Tree of the already warped Editable Mesh, corresponding to the selected output, e.g. "Warps/Warp - Output 2.cmsh"

Pattern Delay:
Enter here the time for the interval in which the calibration pattern changes. In certain cases it can be necessary to increase the interval, especially if the processing capacities of the WD or PB machine is limited or the connection between both is slow.
It is not recommended to enter a much smaller number than the default 300ms.

Add / Remove Fibers:
The left list contains all available input fibers from all Calibration Link nodes set up in the project. Select the **four** inputs corresponding to the output you are currently editing (multi-selection possible) and press "**Add >**" to assign them.
If you want to remove fibers from the assignment, select them and press "**< Remove**".
One fiber input can be assigned to multiple outputs, e.g. when you have an edge blend.

Apply Calibration and Recalibration

Principle:
When you start the calibration, WD creates an Editable Canvas [256] within Pandoras Box and uses this to display a sequence of specific calibration patterns to locate the fiber tips in the screen and calculate their position within the

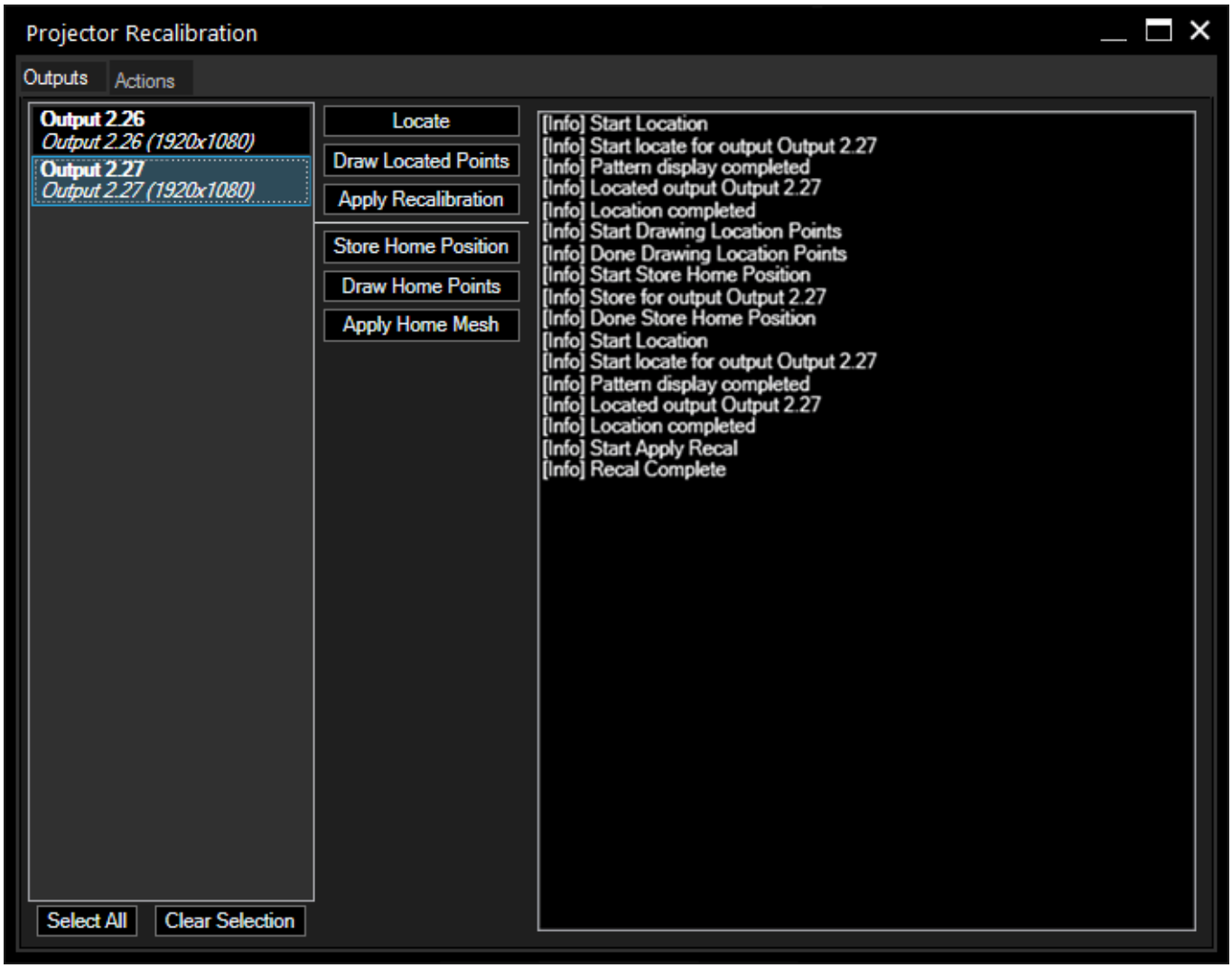
warped output. Those initial points are stored as the home location.
If the projector or screen is now moved slightly and the warping is off, the process of point location is repeated, and based on these light measurements the exact new pixel location can be determined. From the resulting position changes compared to the original configuration, WD calculates the necessary transformation that needs to be made for a new warp.
A second mesh file is created by WD (the original initial warp file is never altered), and the calculated changes are applied.

Realization:
After you have your outputs set up, mesh files and fiber inputs assigned, you can start with the calibration process. Select the **Actions** tab in the top-left corner of the dialog.

First of all, the home points need to be calibrated. Press "**Locate**" to initiate the calibration pattern. Then, press "**Draw Located Points**" to check the fiber positions and afterwards "**Store Home Points**" to save this as the home position.
This creates an Editable Canvas as well as an Editable Mesh for each output.

If your physical setup changes, e.g. the projector is being moved, you need to **locate** the new fiber positions first and then press "**Apply Recalibration**" for calculating and applying the new position adjustments.

Important: All actions reset the respective PB output, active values will be lost.



Locate:
Press this button to start the fiber localization. If this is executed the first time for this output, it will create an Editable Canvas (e.g. "output_2_27") within the PB project.
The Canvas is loaded to the selected output and the calibration pattern sequence starts.

This action only stores the positions of the fiber tips internally, the actual recalibration has to be started manually afterwards.

Draw Located Points:
If you want to check whether the fiber positions were recognized correctly, press this button to display crosshairs around the localized points in the Client output.

Apply Recalibration:
This button now takes the offset between the stored home points and the newly located points, and calculates the transformation of the mesh file needed to meet the new locations. This action is the actual recalibration.
By pressing this button, a copy of the original home mesh is created and modified. The original warp mesh remains as it is.

Store Home Position:
Press this button when you have located your home points for the very first time and before the setup changes. The positions of the fibers are stored internally.
It is crucial that the home positions are as exact as possible. To avoid error propagation between each calibration process, the recalibration always refers to the original home mesh and home positions.
If you have modified your original home mesh, please also store the home positions again.

Draw Home Points:
This displays the positions of the stored home points and visualizes how much the setup has been altered compared to the initial state.

Apply Home Mesh:
Press this button to copy the original Editable Mesh you have warped yourself in the beginning, and apply it to the output.

Log:
The log window displays valuable information [info] about the performed action. If a localization or any other action failed, you can see here why it did not work in an [error] or [warn] message.
It also displays when an action started and when it is finished.
In the above example you can see the messages relating to the following actions:
- Location
- Show Located Points
- Store Home Position
- Location (after setup was altered)
- Apply Recalibration

Recalibration Via Script

There are several scripting commands [1805] available for automating the recalibration process, so that you don't need to interact with the recalibration dialog. You can program a simple CustomScript [822] button that needs to be pressed when the warp is off, or even establish a scheduled [871] recalibration routine.

When you are working with scripts, please note:

- A command does **not** wait until a former action is completed.
- Calling multiple commands might **cancel** any action that is currently in progress (this depends on the command)
- Calling multiple commands will not queue them up, they will be performed at once. If you want to queue commands, please make use of WDWait [1870] with an appropriate wait time.
- All commands reset the Pandoras Box Output, i.e. as if you had initiated a Device Reset. **Active values will be lost!**

Short Step-by-Step List

Find here a short step-by-step list for setting up a recalibration scenario:

1. Setup your PB environment with Master and Clients, create and warp the respective Editable Mesh files in PB for each output you want to recalibrate.
2. In WD, create a Calibration Link input node and connect it with the respective device
3. Connect WD to the PB Master
4. Open Tools > Projector Recalibration
5. Add Client outputs

6. Enter the warp mesh name
7. Assign fiber inputs to output
8. Go to the Actions Tab, select an output and localize the fiber tips
9. Check accuracy with "Draw Located Points" (optional)
10. "Store Home Position"
11. Alter physical setup, localize new fiber tip positions
12. Check accuracy with Draw Located Points (optional)
13. Apply Recalibration
14. Rejoice

Possible error messages

To localize the calibration points, a series of black and white horizontal and vertical line patterns is projected. With pure logical combination Widget Designer then determines where each fiber tip is located.
This entire process of light based measurement and calibration requires that "white" and "black" can be clearly distinguished.

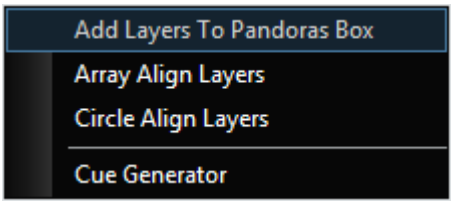
Possible reasons why a black pixel cannot be distinguished from a white one could be that black is too bright or white is too dark or even black equals white:
- too much (side) light on screen that raises the value for black
- broken projector lamp
- poor cable transmission quality
  - broken Calibration Link board
  - broken cables
  - too long cables
  - cables bent too much
  - cable tips dusty, dirty, not cut properly

In the worst case, the projector moved so much that its light does not hit the fiber tips any more. So before starting the process, ensure that the projector covers all of its related fiber points as well as all of its dedicated screen-space and that the focus is alright.

If your value for white exceeds 1023, you may set the Calibration Link to coarse mode. It roughly doubles the value range. If you need to do that, open the Calibration Link and find a dip switch on the circuit board. If you need assistance, please contact Christie Pandoras Box support.

7.7.9 Pandoras Box

In the WD Tools menu [1483], the category "Pandoras Box" offers the following options that add and edit Layers and Cues in a Pandoras Box [67] Master software:

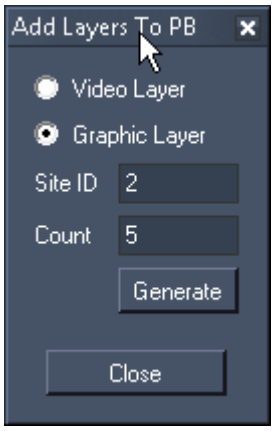


Add Layers to Pandoras Box [1503]
Array Align Layers [1504]
Circle Align Layers [1505]
Cue Generator [1506]

7.7.9.1 Add Layers to Pandoras Box

This tool allows you to easily add one or several new Graphic or Video Layers to the PB V5 Master Device (and its connected Clients) directly from within the Widget Designer.

Please note:
This tool does not work if connected to a PB Master Device below Version 5. If you are working with a Pandoras Box product with limited layers, the maximum layer count cannot be exceeded. Note that PB version 8 comes with unlimited Video Layers and therefore discontinued the Graphic Layer type.



[Video Layer] / [Graphic Layer]:
Choose whether you like to add video or graphic layers.

[Site ID]:
Enter here the ID of the site to which new layers should be added.

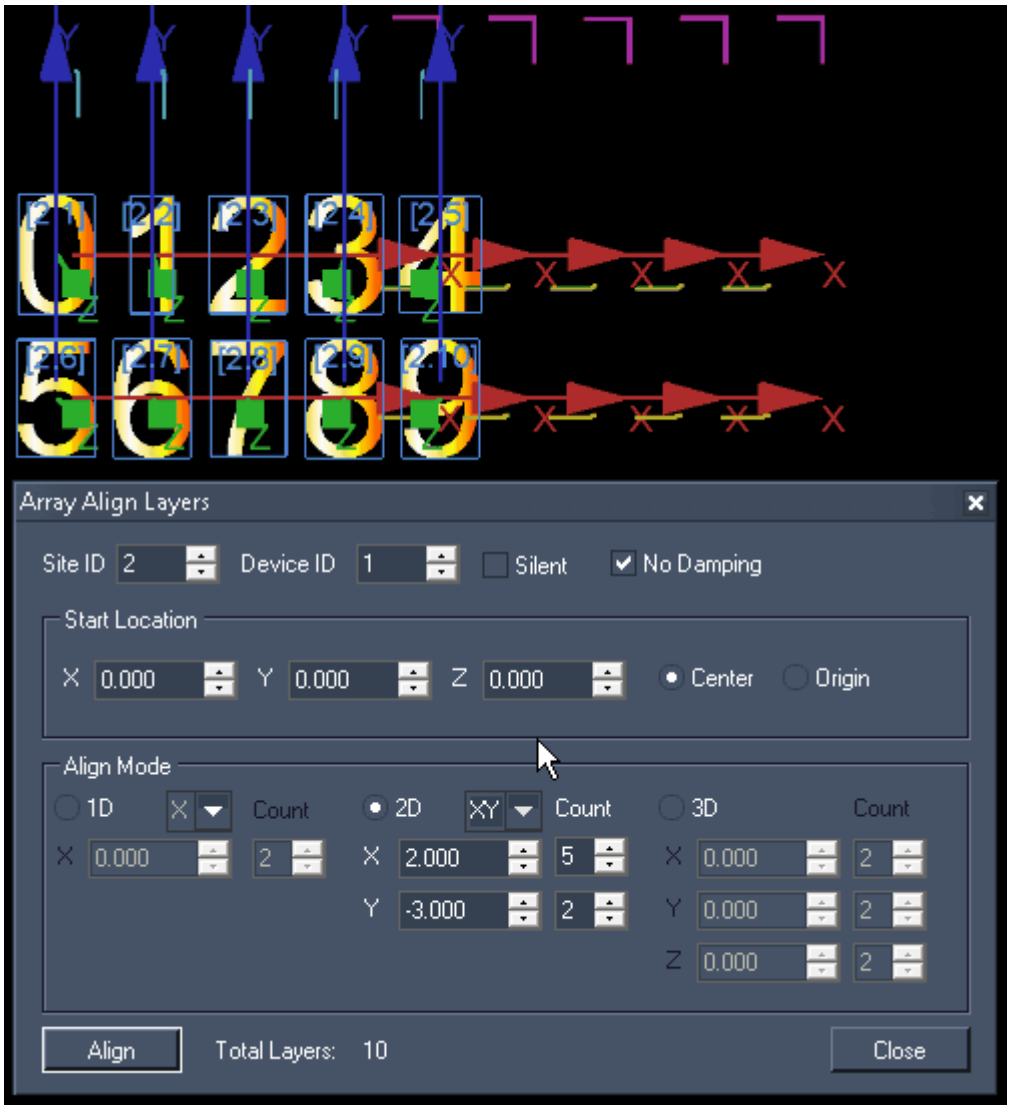
[Count]:
Enter here the amount of layers you want to add to the specific site.

[Generate]:
Press [Generate] in order to generate the specified amount of layers to the site. Depending on the amount, this may take a while and prevent the Master from reacting.

7.7.9.2 Array Align Layers

This tool allows you to arrange several Graphic or Video Layers from your PB Master Device and its connected Clients to form a pattern - a line (1D), a rectangle (2D) or a rectangular box (3D). This kind of arrangement can be done from within the Widget Device faster and easier.
Please read the next topic [1505] if you are interested in a circular alignment.

Please note:
It is recommended to use this tool with a PB Master Device Version 5.



[Site ID] and [Device ID]:
Enter here the ID of the site and the starting device to address the layers which should be arranged.

[Silent]:
Tick the check box if the generated position values should not be entered as active values, but silent values.

[No Damping]:
Tick the check box if the position values should be applied instantly without the Pandoras Box setting for Translation Smoothing. This damping time can be adjusted in the site's Inspector [210].

[Start Location]:
Enter the X, Y, Z position and choose whether the starting point should be centered or not.

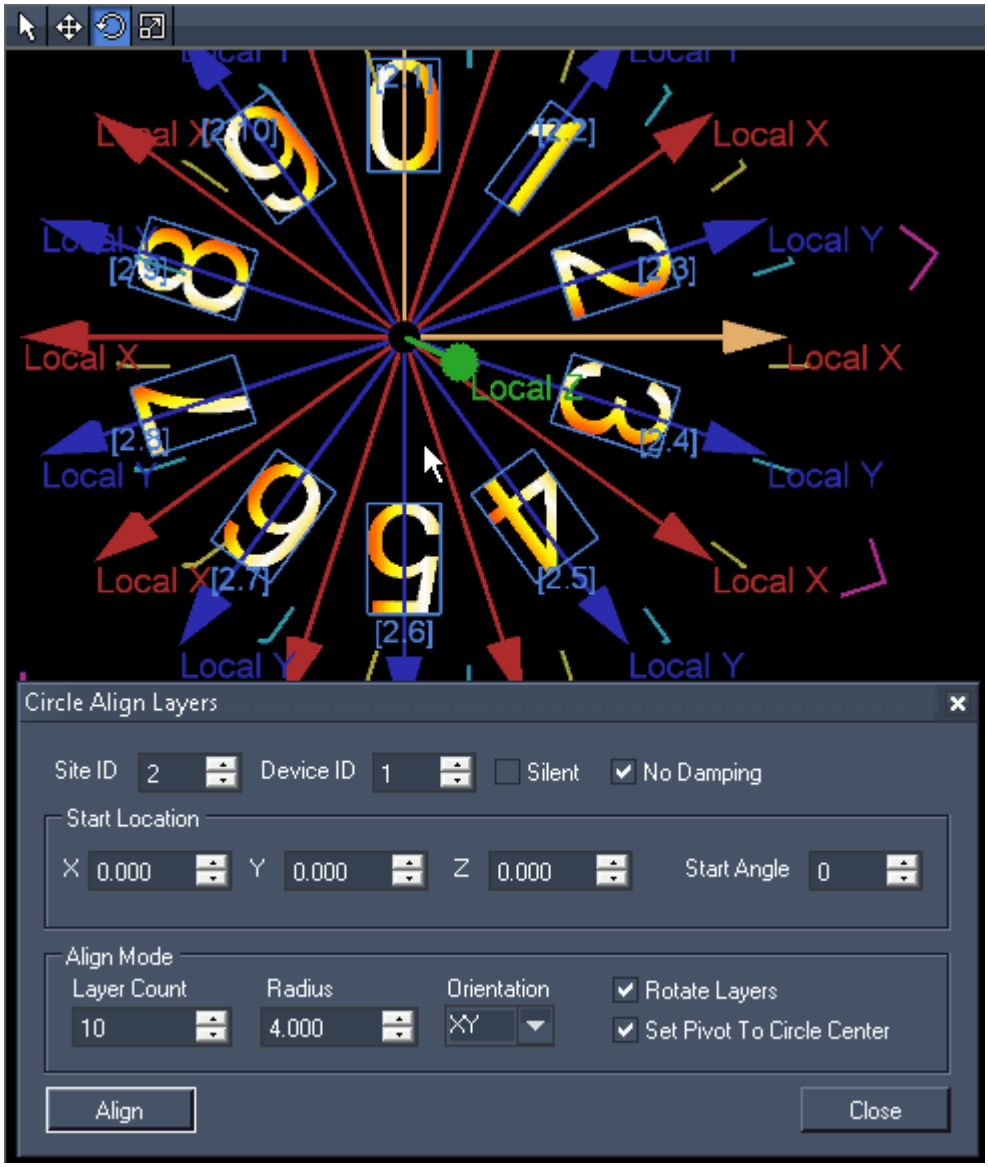
[Align Mode]:
Set the radio buttons to form a line (1D), a rectangle (2D) or a rectangular box (3D). The X, Y, Z values apply for the offset between the layers whereas "Count" defines, how many layers are influenced. In the depicted example the offsets between the 5 horizontal layers are set to 2 and the vertical offset to -3, in total 10 layers (2.1 to 2.10) are

arranged.
Please note that the positive X direction points to the right side, the positive Y direction points to the bottom and Z points into the display.

7.7.9.3 Circle Align Layers

This tool allows you to arrange several Graphic or Video Layers from your PB Master Device and its connected Clients to form a circular pattern. This kind of arrangement can be done from within the Widget Device faster and easier.
Please read the previous topic [1504] if you are interested in an array alignment.

Please note:
It is recommended to use this tool with a PB Master Device Version 5.



[Site ID] and [Device ID]:
Enter here the ID of the site and the starting device to address the layers which should be arranged.

[Silent]:
Tick the check box if the generated position values should not be entered as active values, but silent values.

[No Damping]:
Tick the check box if the position and rotation values should be applied instantly without the Pandoras Box setting for Translation and Rotation Smoothing.
These damping times can be adjusted in the site's Inspector [210].

[Start Location]:
Enter the X, Y, Z position for the starting point. The "Start Angle" defines where the first layer is positioned.

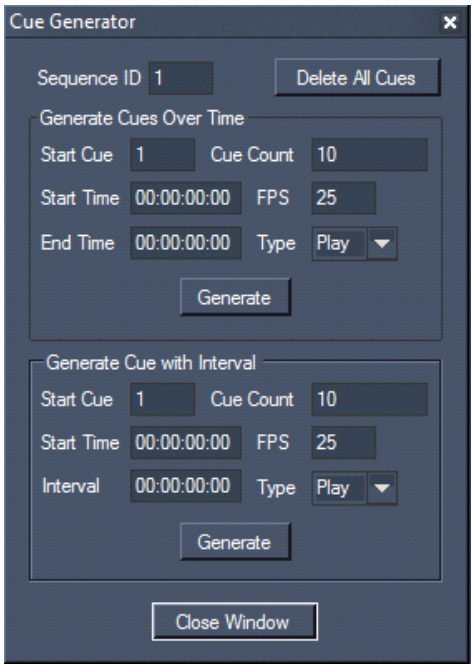
[Align Mode]:
The "Layer Count" influences how many layers are arranged in a circular pattern.
"Radius" enlarges the size of the formed circle.
"Orientation" gives the possibility to span the circle in a XY / YZ / XZ plane.

In the image the option "Rotate Layers" is activated, the layer's lower sides do not face down any more but towards the circle`s center.
"Set Pivot To Circle Center" was ticked as well. The Rotation Pivot Point is positioned in the circle's center. Now, a constant rotation can be applied, for example.

7.7.9.4 Cue Generator

The Cue Generator Tool allows you adding cues into a timeline in Pandoras Box Master Device.

This can be done in two different ways:
- Arranging a specific amount of new cues evenly over a time period.
- Arranging a specific amount of new cues with intervals.



[Sequence ID]
Enter here the ID of the sequence in which you want to add new cues.

Generate Cues Over Time

[Start Cue]
Enter here the ID for the first cue that will be added.

[Cue Count]:
Enter here the amount of cues that will be added.

[Start Time]:
Enter here the time at which the first cue will be placed, in the format (hh:mm:ss:ff).

[End Time]:
Enter here the time at which the last cue will be placed, in the format (hh:mm:ss:ff).

[FPS]:
Enter here the frame rate the sequence is set to in Pandoras Box (FPS = Frames per second), see Sequence Inspector 204.

[Type]:
Enter here the type of the new cues: Play, Pause or Stop.

Generate Cue with Interval

[Start Cue]
Enter here the ID for the first cue that will be added.

[Cue Count]:
Enter here the amount of cues that will be added.

[Start Time]:
Enter here the time at which the first cue will be placed, in the format (hh:mm:ss:ff).

[FPS]:
Enter here the frame rate the sequence is set to in Pandoras Box (FPS = Frames per second), see Sequence Inspector [204].

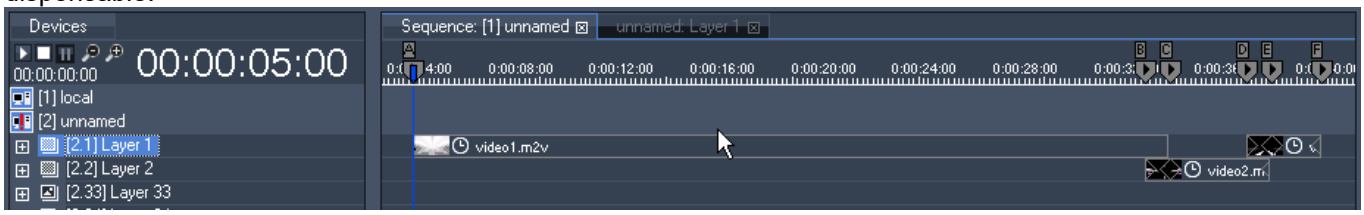
[Interval]:
Enter here the time interval after which the next cue will be added to the timeline, in the format (hh:mm:ss:ff).

[Type]:
Enter here the type of the new cues: Play, Pause or Stop.

7.7.10 Video Logger

The Video Logger allows logging the timeline of Pandoras Box. The feature is useful when clients ask for a playback documentation, e.g. what media was seen for how long.

Please see below a showcase timeline with 3 video containers programmed on two layers. The cues are dispensable.



When importing the log file into a tabulation program like Excel from Microsoft, the result could look like this. For documentation reasons the cues where inserted into the table as well.

	Time	Descript.	Site ID	Device ID	Folder ID	File ID	FilePath	Starttime (hh:mm:ss)	Duration	Total Sec.
A	10:22:59	MediaLog	2	1	-1	-1	None	10:22:36	00:00:22	22.5
B	10:23:28	MediaLog	2	2	-1	-1	None	10:22:36	00:00:51	51.5
C	10:23:29	MediaLog	2	1	10	1	c:\video1.m2v	10:22:59	00:00:29	29.9
D	10:23:32	MediaLog	2	1	-1	-1	None	10:23:29	00:00:02	2.9
E	10:23:33	MediaLog	2	2	10	2	c:\video2.m2v	10:23:28	00:00:04	4.9
F	10:23:35	MediaLog	2	1	10	3	c:\video3.m2v	10:23:32	00:00:02	2.9

At time 0:00 the video logging in Widget Designer was started, 22.5 seconds later the first container on the first layer started. In other words, and this is what the first line in the table tells us, for 22.5 seconds nothing was rendered on that layer.
The next line informs that nothing was seen on the second layer for 51.5 seconds.
Line C states that the file "video1.m2v" saved directly on hard drive C:/ was played back on the first layer (site 2 - device 1) for 29.9 seconds. The duration is given in seconds and in time code language. As well the Folder and File ID (setup in the File Inspector [191]) is logged. When no IDs are assigned "-1" will be saved.

This image shows the Video Logger from Widget Designer. The Video Logger was set up to log Layer 1 and 2 on Site 2. The trigger for the opacity parameter was set to 0.

[Site ID] and [Device ID]
Enter here the ID of the site and the device that should be logged. After setting up the opacity click "Add" to adjoin the layer to the Video Logger's list.

[Min. Opacity]
As soon as a layer exceeds the minimum opacity a log entry is generated.

[Add]
First set up the site and device ID and the opacity trigger value, then click "Add" to adjoin the layer to the Video Logger's list.

[Remove]
If you like to erase a layer with from the Video Logger's list, set up the site and device ID and click "Remove". The opacity value is of no importance.

[Clear devices]
Erases all layers from the Video Logger's list.

[Refresh Devices]
Refreshes the list of layers.

[Start Logging]
Defines a moment 0:00 and starts the logging process with the sequence of the Pandoras Master System that is connected to the Widget Designer.

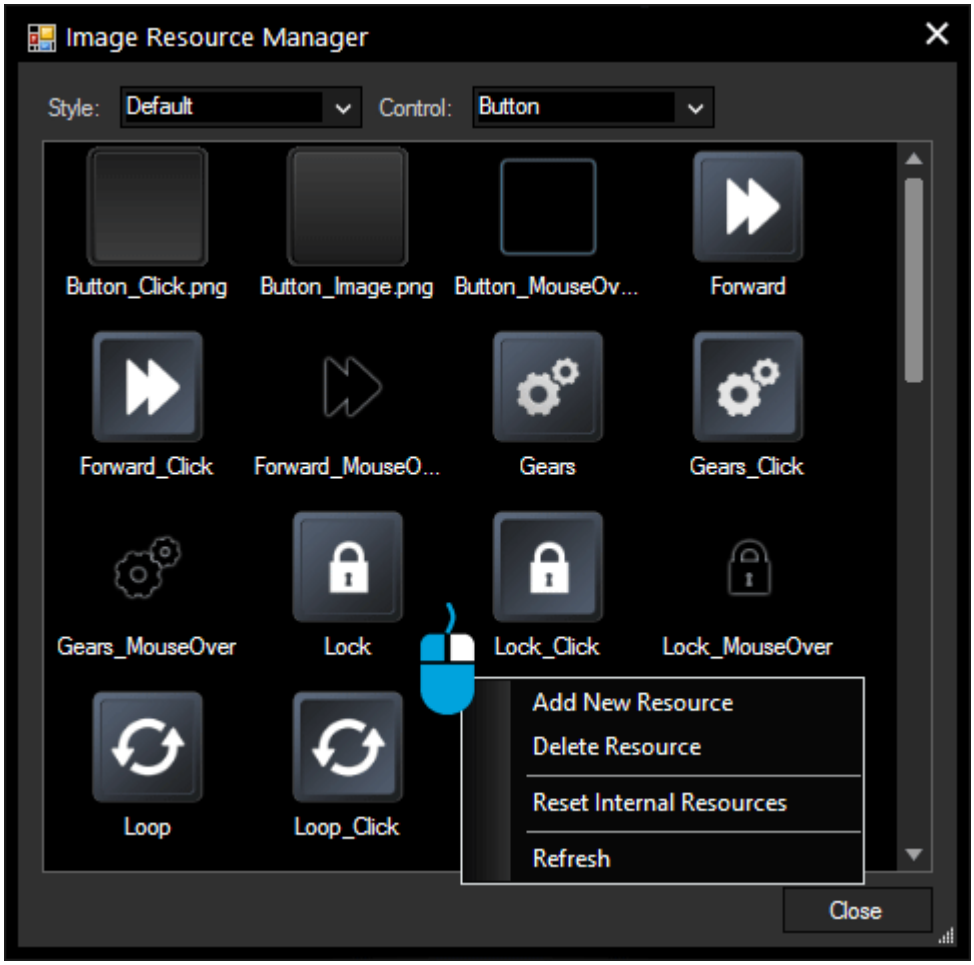
[Clear Log]
Erases all log entries (without pausing the logging process). If you like to erase particular lines only you may as well select them with the mouse or keyboard and press the Delete key.

[Save Log]
Saves the log as a CSV file to a file path of your choice. When importing the CSV into another program you may choose to use the semicolons as a separation symbol.

7.7.11 Image Resource Manager

The Resource Manager gives a fast and easy possibility to manage images within Widget Designer. It can be called from the Tools menu or when clicking a "Res" button in the style settings of a control, e.g. a Fader [874].
Example1: You are using images for controls like buttons and faders.
Example2: You are using the WD Remote App [1275] and send pictures from your Apple mobile device to WD.

The right-click menu lists the commands to add a file from your hard drive and to reset and refresh the resources within the Resource Manager. The "Delete" command works only for added images, not the once that are displayed per default.
If you send images via the WD Remote App [1275], they are displayed automatically in Style:Default, Control:User.



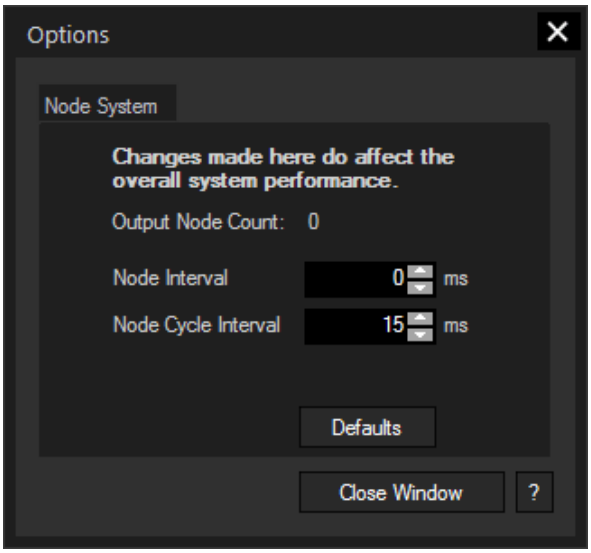
In the item's properties (e.g a fader) you may find a [Res] button which opens the Manager as well. Instead of opening a specific file path for each item separately you may use the Resource Manager for a faster access.

7.7.12 Options

The Options tool allows to set advanced options to influence the performance of the node system

When the user interface contains a lot of controls (labels, faders etc.) that are updated via output nodes, these updates could slow down the whole system performance. In these cases it could make sense to slow down the output node processing to gain overall system performance.

The disadvantage of slowing down the node processing is that all data given out by the nodes, for example Art-Net data, will be slowed down as well.



Node Interval:
By default the Node Interval is set to 0 ms. This means that there is no delay between the processing of values from one node to the following node. If setting the Node Interval to x ms, there will be a pause of x ms after the processing of each(!) node within one node chain.

Node Cycle Interval:
A node cycle interval is the interval, in which all output nodes are processed once, in other words after what time the Output updates / sends out its current value. By default the Node Cycle Interval is set to 15 ms. This means that after all output nodes are processed once there is a wait time of 15 ms, i.e 66 times per second. You may set this wait time to a different value to slow down or speed up the node processing interval.

Defaults:
Press this button to set the Node Interval and Node Cycle Interval back to their default values.

Since the Property pages request the node chain to update 25 times per second to display the current values, it is recommended to have all property dialogs closed during show mode for exact value processing.

7.8 Scripting

The Scripting menu [811] lists all information around using commands in Widget Designer.

The chapter Script Language [1511] explains the general usage of the script language in Widget Designer, e.g. what is a command and how do I write it. Further advanced programming statements are explained, e.g. the if-statement [1886].

In addition it covers helping tools and scripting techniques regarding scripting in Widget Designer.
Functions and Macros [1897]
Variables [1900]
The Object and Member Notation (dot syntax) [1904]

See the chapter Scripting Menu [811] for information about the options you can choose there, e.g. the Debug Logger.

7.8.1 Script Language

The built-in script language allows you to create customized routines of commands. Currently over 1500 commands [1520] are available to control specific features of Pandoras Box, Widget Designer or other 3rd party products and protocols.

Commands can be used as a project's Start-up Script [798] or in some nodes, tools and many widgets. For example, you may assign one or more commands to a button click.

How to enter a command using Script Assistant

In order to use a command, open the widget's Item properties, for example from a Custom Script Button [822].

As soon as you start typing in the dedicated script text field, the feature Script Assistant will automatically search for available commands that contain the typed characters and display the result in a list box. The [Esc] key hides this list. The availability of commands depends also on the fact, whether a widget is already added to the project, i.e. without existing faders, the Script Assistant will not offer fader commands.

You have three possibilities for setting up the script filter of your assistant, just right-click in your script field and choose one of the following filter styles:

- Starts with expression
The Script Assistant will show only entries that start with the entered expression. This is useful if you are already familiar with the command structure and know which group of commands you need.

- Contains expression (this is default)
You do not need to know how exactly the command starts, typing "fader" will suggest "WDFader..." too. Type more characters to shorten the list or use the arrow keys [Up] or [Down] to select a command from it and press [Enter].

- Camel Case
This is the fastest way of getting the correct command if you already know which command you want to use. This filter type requires deep knowledge of the command structure, as you only type in the letters that are written with capitals in the command (this way of writing with capitals inside an expression is called "camel case").
e.g.: type in "wdfd" and the command "WDFaderDown" will appear in the list box.

If you have one preferred filter style, you can set it up in your User Profile [795] as the default value.

Now, the placeholders need to be filled with concrete information, e.g. a numeric ID for a certain element that you would like to control or send values to. In addition you may use variables. Again, the Script Assistant offers a drop-down with all available variables.
The Script Assistant will guide you through the arguments, too, by listing all arguments and their respective data types (like, String, Integer, Bool, ...)

Please note that all literal values, meaning any kind of character string, has to be enclosed in either single or double quotation marks. If one of those is part of your string, just use the other one for declaring the literal. Do not use accents like ´ or `!
WDLabelText(1,'Hello World!')
WDLabelText(2,"Hello World!")

WDLabelText(3,'Hello "World"!')
Some strings, like Pandoras Box device parameters or variable names, already appear at the Script assistant with quotation marks. You can select one of those without adding anything and proceed with the next argument. If you like to use a variable's value, type the variable name without quotation marks.

All commands affecting a widget are available with the object and member notation [1904], too.

Multiple commands (i.e. one script) can be executed by delimiting every command with a carriage return at the end of each command, simply hit the "enter" key. To delete a command, select it and press the "Delete" key.

Command examples

For example, to set the opacity of Layer 1 on Server 1 to 128 you will need the following command:
`DeviceSetParam(SiteID,DeviceID,ParamName,AbsoluteValue)`

Since this is only a template of the command, you will need to enter the numeric values for the SiteID, DeviceID, AbsoluteValue and the ParamName [1514] as plain text.
The correct command for this example would be written like this:
`DeviceSetParam(1,1,"Opacity",128)`

Alternatively you may enter a variable [1900] name for any placeholder. Assuming you have an integer variable defined with the name varNumber and the value of 1, you could write the command as follows:
`DeviceSetParam(varNumber,varNumber,"Opacity",128)`

Advanced script techniques

Normally, you would enter a command and then fill out the placeholders (also called the "arguments" of a command). But there is also a technique that allows more direct programming. Still the syntax must conform, but as it is more open it can be used more flexible. You have direct script access to Variables and certain member values including nodes. This allows direct value assignment and item control. For further information and examples click the according links.

direct command	corresponding common command
Variables [1900]	

`Variable = Value`
`varNumber = 123`
`ListVariable[Index] = Value`
`varList[5] = 123`
	`VValue(VarName,Value)`
	`VValue(varNumber,123)`
	`VSetArrayValue(VarName,Index,Value)`
	`VSetArrayValue("varList",5,123)`

Members [1904]	

`MemberID.MemberValue = Value`
`label1.text = "Hello"`
 	
`fader2.value = 123`
`fader2.value = varNumber`
	`WDLabelText(ID,Text)`
	`WDLabelText(1,"Hello")`
	`WDFaderValue(ID,Value)`
	`WDFaderValue(2,123)`
	`WDFaderValue(2,varNumber)`

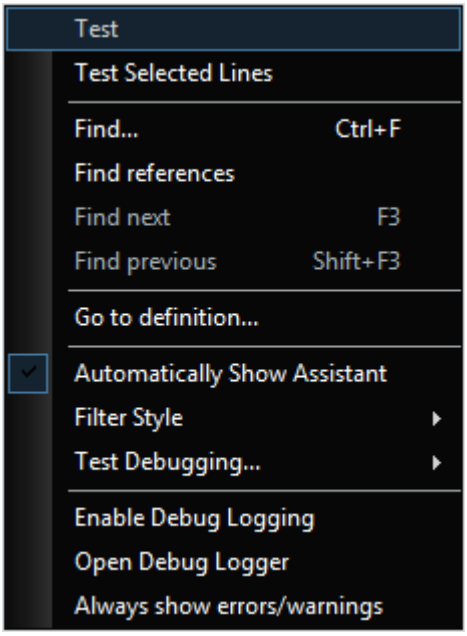
Nodes [1928]	

`NodeID.Nodecommand`
`node1.ResetPlaylist`
`node2.PowerOn`	

The last row mentions Node Commands [1928]. They cannot only access functions of a node but also the parameters. Please see the according chapter for more details.

Of course, mathematical functions can be applied to values and variables, too. Please refer to the chapter Mathematical expressions and conditions [1894] for more information.
Since Version 6, it is even possible to use mathematical expressions as command arguments or programming statements, e.g.:
```
DeviceSetParam(varNumber,varNumber+10,"Opacity",128)
If  varRes >= (varNumber*5) {True Script}
```

Useful Tools from the Context Menu



When you right-click somewhere inside a scripting field, the context menu opens up and offers you some practical items:

Test: Executes the whole script inside this field

Test selected Lines: Executes all highlighted lines or the line in which the cursor is located

Find... : Searches for the highlighted expression (only text search)

Find references: Searches for the highlighted expression (searches for references to the marked object)

Find next / previous: Goes through all found expressions

Go to definition...: Shows the origin of the object, i.e. the line where a local variable is initialized, the variable list if the object is a global variable, or it opens the function / macro editor if the object is one of those.

Automatically show Assistant: Uncheck this if you do not need the Script Assistant

Filter Style: Select the Script Assistant's filter style which is described above

Test Debugging: Enables the Debug Logger [812] for the "Test" function mentioned above and clears it if necessary

Enable Debug Logging: Uncheck this if you do not want the Debug Logger to open

Open Debug Logger: This opens the dialog Debug Logger [812]

Always show errors / warnings: Per default, the Debug Logger does not open automatically to show whether a script includes an error. It opens only if you choose the "Test"command from the right-click menu. If you like to always see errors and warnings, enable this option.

More information

WD commands can be written in upper or lower case style or mixed, with one exception:
Wherever you need to specify text-based parameters such as Opacity, the entry is case sensitive and must not have additional blanks (space characters).
Please see the next page for a list with all parameter names [1514] from Pandoras Box. Then, a Command List [1520] explains all available commands.
Advanced users can merge commands in functions and macros [1897] or combine them with If-queries and for loops repetitions as explained in the topic programming statements [1886].

7.8.1.1 Parameter List

This list helps with commands [1511] in Widget Designer where a so called "ParamName" needs to be defined (e.g. `DeviceSetParam(SiteID,DeviceID,ParamName,AbsoluteValue))`
The overview lists all names from parameters within Pandoras Box and their value range.

Please note:
- the name is case sensitive and spaces are important, e.g. "X Scale", not "x scale" or "XScale"
- all lists show the parameter name for the first axis, edge or color. If there is "X Pos", it would be "Y Pos" and "Z Pos" for the other axes; if there is "Keystone R", ir would be "Keystone L",
- for effects [322] the entire name consists of the FX name and the FX parameter, both combined with a pipe "|" character (vertical bar) but no spaces before or after the pipe, e.g. "Blur|Mix" or "B&W Add|Invert"
- parameters for particle systems [185] are listed under Particle Layer [1517]; if you are using the older workflow with Particle Systems on a Video Layer, see here
- if the value needs to be a decimal value, always use a point "." as separator not a comma "," e.g. "0.5"
- per default, active values are applied to the parameter not instantly or abruptly (0ms) but within 500ms in order to smooth movement. More information can be found in the chapter Device Inspector > "Parameter Value Smoothing [210]".
- in PB, the default setup in the Configuration tab > Unit Management [161] is, that the position values are displayed in pixels but that incoming values (via SDK or Widget Designer, not DMX) are interpreted as generic units. Depending on your needs, you can either display position parameters as generic units or interpret incoming values as pixels. See the linked chapter for more information.

Video Layer (see here [647] for description)	
Media	either file name, or 1 - 255 for Folder ID & 1 - 255 for File ID
Mesh	either file name, or 1 - 255 for Folder ID & 1 - 255 for File ID
Opacity	0 - 255
Playback Transport	0 or "Stop"
	64 (1-127) or "Play"
	128 or "Pause"
	192 (129-255) or "Loop"
Playback Speed	0-127 = slower 128 = 1:1 Speed 129-255 = faster
Inpoint	0 - 65535
Outpoint	0 - 65535
Volume	0 - 2 (= -96.00 - +6 dB) [663]
X Pos (Y..., Z... for other axes)	-999.999 – +999.999
X Angle	-9999.99 – +9999.99
X Rot Mode	0; 1
X Rot Speed	0 - 655535
X Scale	-999.999 – +999.999
Rot Pivot X Pos	-999.999 – +999.999
Scale Pivot X Pos	-999.999 – +999.999
Blend Mode	0 - 20
'FXName'|'FXParameter' (e.g. Blur|Mix) see FX List [327] for more info	
Particle System|'PSParameter' (e.g. Particle System|Wind) and Particle System|Particle Emitter|'PEParameter' (e.g. Particle System|Particle Emitter|Radius) see here [185] for more info about Particles; see below for parameters of Particle Layer [1517]	

Light Layer (see here 666 for description)	
Media	either file name, or 1 - 255 for Folder ID & 1 - 255 for File ID
Light Intensity	0 - 255
Playback Transport	0 or "Stop"
	64 (1-127) or "Play"
	128 or "Pause"
	192 (127-255) or "Loop"
Playback Speed	0 - 255
Inpoint	0 - 65535
Outpoint	0 - 65535
Light Source X Pos (Y..., Z... for other axes)	-999.999 – +999.999
Light Target X Pos	-999.999 – +999.999
Light Color Red (...Green, Blue, Alpha)	0 - 255
Light Angle	0.000° - 180.000°
Light Aspect	0.000 - 20.000
Light Z Roll	-9999.99° – +9999.99°
Near Plane	0 - 65535
Far Plane	0 - 65535
Light Tolerance	0.000 - 1.000
Shadow Softness	0 - 1000
Light Proj. Mode	0; 1

Particle Layer (see here [185] for description)	
Opacity	0 - 255
Playback Transport	0 or "Stop"
	64 (1-127) or "Play"
	128 or "Pause"
	192 (129-255) or "Loop"
X Pos (Y..., Z... for other axes)	
Gravity	0.000 - 1000.000
Wind	0.000 - 1000.000
Wind Rot X	0 - 360
Blend Mode	0 - 20
Particle Emitter|Media	either file name, or 1 - 255 for Folder ID & 1 - 255 for File ID
Particle Emitter|Mesh	either file name, or 1 - 255 for Folder ID & 1 - 255 for File ID
Particle Emitter|Emitter Type	0 - 6
Particle Emitter|Particle Alignment	0 - 2
Particle Emitter|Radius	0 - 100
Particle Emitter|Length	0 - 100
Particle Emitter|Angle	0 - 360
Particle Emitter|Range	0 - 100
Particle Emitter|X Pos	-999.999 - +999.999
Particle Emitter|X Angle	-1000° - +1000°
Particle Emitter|Rot Pivot X Pos	-999.999 - +999.999
Particle Emitter| Count/Sec	0.000 - 1000.000
Particle Emitter|Time to Live (Sec)	0.000 - 1000.000
Particle Emitter|Mass	1.00 - 100.00
Particle Emitter|Drag	0.000 - 1000.000
Speed, XYZ Scale, XYZ Rot, Color and Particle Opacity	can not be remote controlled via WD

Camera Layer (see here [675] for description)	
Proj. Mode	0; 1
Viewpoint X Pos (Y..., Z... for other axes)	-999.999 – +999.999
Target X Pos	-999.999 – +999.999
FOV	-180.000 – +180.000
Near Plane	0 - 65535
Far Plane	0 - 65535
Aspect	0.000 - 6.000
X Offset	-999.999 – +999.999
Z Roll	-9999.99 – +9999.99
RtClearColor Red (...Green, Blue, Alpha)	0 - 255

Output Layer (see here 682 for description)	
Camera	via command: ShareLayerTexture 1612
Mesh	either file name, or 1 - 255 for Folder ID & 1 - 255 for File ID
Opacity	0 - 255
X Pos (Y..., Z... for other axes)	-999.999 – +999.999
X Angle	-9999.99 – +9999.99
X Rot Mode	0; 1
X Rot Speed	0 - 65535
X Scale	-999.999 – +999.999
Rot Pivot X Pos	-999.999 – +999.999
Scale Pivot X Pos	-999.999 – +999.999
Viewpoint X Pos	-999.999 – +999.999
Target X Pos	-999.999 – +999.999
FOV	-180.000 – +180.000
Near Plane	0 - 65535
Far Plane	0 - 65535
Aspect	0.000 - 6.000
X Offset	-999.999 – +999.999
Z Roll	-9999.99 – +9999.99
RtClearColor Red (...Green, Blue, Alpha)	0 - 255
KS L (...R, ...B, ...T)	0 - 65535
KS LR	0 - 65535
Lin X	0 - 65535
SE Media	either file name, or 1 - 255 for Folder ID & 1 - 255 for File ID
SE L	0 - 65535
SE LC	0 - 65535
SE L Marker	0 = no marker 1 - 255 = black, gray, white marker
SE L Width	0 - 255
State	0; 1

7.8.1.2 Command List

This topic lists all commands that are available in Widget Designer [786]. How they are used is explained in the introducing topic [1511].

A [1520] B [1525] C [1525] D [1530] E [1540] F [1544] G [1544] H [1544] I [1545] J [1546] K [1547] L [1551] M [1552] N [1561] O [1561] P [1564] Q [1576]
R [1576] S [1600] T [1616] U [1618] V [1620] W [1647] X [1885] Y [1885] Z [1885]

A

ActivateAll

ActivateAll

Example:
ActivateAll

Activates all parameters of all layers of all Servers within the Pandoras Box Project.

ActivateDevice

ActivateDevice(SiteID,DeviceID)

Example:
ActivateDevice(1,3)

Activates all parameters of layer 3 of site 1 within the Pandoras Box Project.

ActivateParam

ActivateParam(SiteID,DeviceID,ParamName)

Example:
ActivateParam(1,3,"X Scale")

Activates the parameter [1514] X Scale of layer 3 of site 1 within the Pandoras Box Project.

ActivateSite

ActivateSite(SiteID)

Example:
ActivateSite(1)

Activates all parameters of all layers of site 1 within the Pandoras Box Project.

AddEncryptionKey

AddEncryptionKey(Key)

Example:
AddEncryptionKey("Key1|
uYxJLovsAK+ZAJplLQgpSf1u6wwaA5e0UurBJq2+MTsugQpLiXSOHRCAHdcMBVf2GSfZLRn5UoURjIfnmpJO
F78d33pFdTfdKdj6YjnJEr0=")

This applies to the Media Encryption feature [224] in Pandoras Box. Using this command you can import a key that was exported earlier with another PB project.

AddEncryptionPolicy

AddEncryptionPolicy(Policy)

Example:
AddEncryptionPolicy("Pol1|
Y3dm4ld4S7XB0NzYLinuseF7jXzUexhTBWSysoPlXZKUmgnbuvWfG/t39j+qxUE5FdDSaoZpOzbQUuje0E13F
eKfMHKbfFpvS5FATyI8LvU=")

This applies to the Media Encryption feature [224] in Pandoras Box. Using this command you can import a policy that was exported earlier with another PB project.

AddFileToPBPlaylistByID

AddFileToPBPlaylistByID(FileName,ProjectFolder,TempFolderID,TempFileID,PbPlaylistFolderID,PbPlaylistFileI
D)

Example:
AddFileToPBPlaylistByID("C:\Christie\content\playlist\image5.png","playlist",2,5,3,6)

First(the file)"image5.png" from the path "C:\Christie\content\playlist" is loaded and added to the subfolder "playlist" within your Pandoras Box project. Secondly, the file is assigned with the Folder and File ID 2,5. Last, this file is added to the Playlist [239] with Folder/File ID 3,6.
If you do not need a Folder/File ID in the second step, simply use "0,0" instead of "2,5".

AddGraphicLayer

AddGraphicLayer(SiteID,Count)

Example:
AddVideoLayer(1,3)

Adds three new Graphic Layers to Site 1 (e.g. a Server). Note that graphic layers are discontinued since Pandoras Box version 8.0.0

AddVideoLayer

AddVideoLayer(SiteID,Count)

Example:
AddVideoLayer(1,3)

Adds three new Video Layers [647] to Site 1 (e.g. a Server). Note that, before version 8, some Pandoras Box products had a limited amount of Video Layers.

ApplicationClose

ApplicationClose(Processname)

Example:
ApplicationClose("notepad")

Closes all running notepad applications. Please have a look into the Windows Task Manager to get the correct process name of the running application you want to close, enter this name without ".exe".

ApplicationKill

ApplicationKill(Processname)

Example:
ApplicationKill("PB_Widget_Designer")

This ends the process of Widget Designer of the local WD computer without asking to save any projects. Please have a look into the Windows Task Manager to get the correct process name of the running application you want to close, enter this name without ".exe".

Other examples:
ApplicationKill("mspaint")
ApplicationKill("vlc")

These commands end the process of VLC Player and Microsoft Paint.

ApplicationKillAllOtherWDInstances

ApplicationKillAllOtherWDInstances

Example:
ApplicationKillAllOtherWDInstances

This ends all background processes of Widget Designer that do not belong to the one currently running and have added up by mistake. This command is of interest for permanent installations without technical staff.

ApplicationStart

ApplicationStart(Filepath,Commandline Arguments(optional))

Example:
ApplicationStart("C:\Program Files\Christie\Widget Designer 6.1.0\PB_Widget_Designer.exe")

The first examples use only the first parameter "Filepath". It starts Widget Designer on the local WD computer. The following examples start the VLC Player and Microsoft Paint.

ApplicationStart("C:\program files\VideoLAN\VLC\vlc.exe")
ApplicationStart("mspaint")

Examples with optional parameter for commandline arguments:
ApplicationStart("C:\program files\VideoLAN\VLC\vlc.exe","C:\Music\Sound.mp3")
ApplicationStart("C:\Program Files\Christie\Widget Designer 6.1.0\PB_Widget_Designer.exe" STARTWDF)

If you like to use the optional second parameter "Commandline Arguments", please find out first whether your application supports this and how the commandline should be written.
The first example from above starts the VLC Player and plays back the file "Sound.mp3". The second example starts the Widget Designer free edition.

Widget Designer starts with the following commandline arguments:
- STARTWDF: in the free edition
- STARTWD (or nothing): for the normal edition
- STARTWDU: for the edition with unlimited web clients

ApplyView

ApplyView(ViewID)

Example:
ApplyView(2)

This applies the View [287] with ID 2 to the Pandoras Box user interface. Views can be saved in the Project tab or View tab [310].

ArtNetDisableAllUniverses

ArtNetDisableAllUniverses

Example:
ArtNetDisableAllUniverses

Disables the Art-Net Output on all Universes. This command clears the Universe List in the Connection Manager [1258].

ArtNetDisableUniverse

ArtNetDisableUniverse(Subnet 0-15,Universe 0-15)

Example:
ArtNetDisableUniverse(0,1)

Disables the Art-Net Output on Art-Net Subnet 0, Universe 1. The specified Art-Net Universe will be removed from the Universe List in the Connection Manager [1258].

ArtNetInputDisabled

ArtNetInputDisabled

Example:
ArtNetInputDisabled

Disables the Art-Net Input in the Connection Manager [1258].

ArtNetInputEnabled

ArtNetInputEnabled

Example:
ArtNetInputEnabled

Enables the Art-Net Input in the Connection Manager [1258].

ArtnetInputSetAdapter

ArtnetInputSetAdapter(NicAdapterName)

Example:
ArtnetInputSetAdapter("Lan1")

Enables the Art-Net Input in the Connection Manager [1258] and sets the network adapter to the entry "Lan1". Make sure, that the connection is disabled before setting the network adapter, e.g. via the command ArtNetInputDisabled [1523].

ArtNetOutputDisabled

ArtNetOutputDisabled

Example:
ArtNetOutputDisabled

Disables the Art-Net Output in the Connection Manager [1258].

ArtNetOutputEnabled

ArtNetOutputEnabled

Example:
ArtNetOutputEnabled

Enables the Art-Net Output in the Connection Manager [1258].

ArtnetOutputSetAdapter

ArtnetOutputSetAdapter(IP address,NicAdapterName)

Example:
ArtnetOutputSetAdapter(,"2.0.0.1","Lan1)

Enables the Art-Net Output in the Connection Manager [1258], sets the IP to the address 2.0.0.1 and the network adapter to the entry "Lan1". Make sure, that the connection is disabled before setting the network adapter, e.g. via the command ArtNetOutputDisabled [1524].

ArtNetSetVal16bit

ArtNetSetVal16bit(Subnet 0-15,Universe 0-15,Channel,Value 0-65535)

Example:
ArtNetSetVal16bit(0,1,41,32768)

Sets the 16bit parameter with the following starting address to a value of 32768: Channels 41 on Art-Net Subnet 0, Universe 1.

Please note, that the next channel you can control is now Channel 43 as the 16 bit value occupied Channel 41 and 42.

Art-Net Output needs to be enabled in the Connection Manager [1258].

ArtNetSetVal24bit

ArtNetSetVal24bit(Subnet 0-15,Universe 0-15,Channel,Value 0-16.777.215)

Example:
ArtNetSetVal24bit(0,1,44,2.255.248)

Sets the 24bit parameter with the following starting address to a value of 2.255.248: Channels 44 on Art-Net Subnet 0, Universe 1.

Please note, that the next channel you can control is now Channel 47 as the 24 bit value occupied Channel 44, 45 and 46.

Art-Net Output needs to be enabled in the Connection Manager [1258].

ArtNetSetVal8bit

ArtNetSetVal8bit(Subnet 0-15,Universe 0-15,Channel,Value 0-255)

Example:
ArtNetSetVal8bit(0,1,38,255)

Sets the Channel 38 on Art-Net Subnet 0, Universe 1 to the value 255.

Art-Net Output needs to be enabled in the Connection Manager [1258].

B B

C C

ClearActiveDevice

ClearActiveDevice(SiteID,DeviceID)

Example:
ClearActiveDevice(1,3)

Clears all active parameters of layer 3 of site 1 within the Pandoras Box Project.

ClearActiveParam

ClearActiveParam(SiteID,DeviceID,ParamName)

Example:
ClearActiveParam(1,3,"X Scale")

Sets a clear active to the parameter [1514] X Scale of layer 3 of site 1 within the Pandoras Box Project.

ClearActiveSite

ClearActiveSite(SiteID)

Example:
ClearActiveSite(1)

Clears all active parameters of all layers of site 1 within the Pandoras Box Project.

ClearAllActive

ClearAllActive

Example:
ClearAllActive

Clears all active parameters of all layers of all Servers within the Pandoras Box Project.

ClearSelection

ClearSelection

Example:
ClearSelection

Clears the device selection within the Pandoras Box project.

CloseAirScanProperties

CloseAirScanProperties

Example:
CloseAirScanProperties

Closes the dialog for the AirScan tool [1277].

CloseAllDialogs

CloseAllDialogs

Example:
CloseAllDialogs

This closes all dialogs from Widget Designer.

CloseArtNetMonitor

CloseArtNetMonitor

Example:
CloseArtNetMonitor

Closes the Art-NetMonitor [2051] which can also be accessed through the Connection Manager [1258].

CloseArtNetUniverselist

CloseArtNetUniverselist

Example:
CloseArtNetUniverselist

Closes the Art-Net Universe List which can also be accessed through the Connection Manager [1258].

CloseCameraTrackerDialog

CloseCameraTrackerDialog

Example:
CloseCameraTrackerDialog

Closes the Camera Tracker [1291] dialog.

CloseCitpBrowser

CloseCitpBrowser

Example:
CloseCitpBrowser

Closes the Thumbnail Browser which can also be accessed through the dialog PB Network Configuration [1256].

CloseComConnections

CloseComConnections

Example:
CloseComConnections

Closes the COM Connection [1269] dialog which gives an better overview than the Connection Manager [1258].

CloseConnectionManager

CloseConnectionManager

Example:
CloseConnectionManager

Closes the Connection Manager [1258].

CloseEmailSettings

CloseEmailSettings

Example:
CloseEmailSettings

Closes the Email Settings Tool [1490].

CloseIpConfiguration

CloseIpConfiguration

Example:
CloseIpConfiguration

Opens the PB Network Configuration [1256] dialog.

CloseKeyboardShortcuts

CloseKeyboardShortcuts

Example:
CloseKeyboardShortcuts

Closes the Keyboard Shortcut Editor [1484].

CloseKinectDialog

CloseKinectDialog

Example:
CloseKinectDialog

Closes the Kinect Tool [1283].

CloseMacroEditor

CloseMacroEditor(MacroName)

Example:
CloseMacroEditor("MyMacro")

Closes the Macro Editor with the macro "MyMacro".

CloseMidiNoteEditor

CloseMidiNoteEditor

Example:
CloseMidiNoteEditor

Closes the Midi Note Editor [1486].

ClosePageBrowser

ClosePageBrowser

Example:
ClosePageBrowser

Closes the Page Browser [799].

CloseRemoteInput

CloseRemoteInput

Example:
CloseRemoteInput

Closes the Remote Input Tool [1273].

CloseSmsSettings

CloseSmsSettings

Example:
CloseSmsSettings

Closes the SMS Settings Tool [1494].

CloseTcpConnections

CloseTcpConnections

Example:
CloseTcpConnections

Closes the TCP Connection [1265] dialog which gives an better overview than the Connection Manager [1258].

CloseUdpConnections

CloseUdpConnections

Example:
CloseUdpConnections

Closes the UDP Connection [1267] dialog which gives an better overview than the Connection Manager [1258].

CloseVariableList

CloseVariableList

Example:
CloseVariableList

Closes the dialog Variable List [1901].

COMInject

COMInject(ID,Message)

Example:
COMInject(1,"Play")

Injects the message "Play" directly into the stream of the COM Connection with ID 1, without waiting for other messages or packages to be finished.

The UDP Connection needs to be defined first in the Connection Manager [1258].

Incoming or outgoing values in TCP-, UDP- and COM Port nodes can be entered as string, decimal or hexadecimal values as explained in the chapter Syntax TCP-/UDP-/Serial Messages [944].

Example:
Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

For two commands - e.g. a carriage return and a line feed - you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

COMSend

COMSend(ID,Message)

Example:
COMSend(1,"Play")

Sends the message "Play" via the COM Port Connection with the ID 1 in the Connection Manager [1258].

COMSendDec

COMSendDec(ID,Message)

Example:
COMSendDec(1,"72 105 33")

Sends the message "Hi!" in decimal values via the TCP Connection with the ID 1 in the Connection Manager [1258]. For more information see the chapter "Syntax TCP-/UDP-/Serial Messages" [944].

COMSendHex

COMSendHex(ID,Message)

Example:
COMSendHex(1,"48 69 21")

Sends the message "Hi!" in hexadecimal values via the COM Port Connection with the ID 1 in the Connection Manager [1258]. For more information see the chapter "Syntax TCP-/UDP-/Serial Messages" [944].

COMStart

COMStart(ID)

Example:
COMStart(1)

Starts the COM Port Connection with the ID 1 in the Connection Manager [1258].

COMStop

COMStop(ID)

Example:
COMStop(1)

Stops the COM Port Connection with the ID 1 in the Connection Manager [1258].

D D

DebugClear

DebugClear

Example:
DebugClear

This clears the log entries of the Debug Logger [812] dialog.
Please note: This feature is currently not available for the WD Free Version.

DebugDisable

DebugDisable

Example:
DebugDisable

Disables the Debug Logging so that error messages are not added to the Debug Logger [812] dialog.
Please note: This feature is currently not available for the WD Free Version.

DebugEnable

DebugEnable

Example:
DebugEnable

Enables the Debug Logging so that error messages are added to the Debug Logger [812] dialog.
Please note: This feature is currently not available for the WD Free Version.

DebugJson

DebugJson(JsonVarName)

Example:
DebugJson(varJson)

This displays the value of the variable named varJson (with the JSON type [1926]) in the Debug Logger [812] dialog.
Please note: This feature is currently not available for the WD Free Version.

DebugMessage

DebugMessage(Message,Arbitrary Value)

Example:
DebugMessage("Home Button clicked")

This writes the message "Home Button clicked" into the Debug Logger [812] and stores it there for debugging or logging purposes. Text must be enclosed in double or single quotation marks and can be combined e.g. with variables using a plus sign as seen in the examples below. Note: Debug Logging has to be enabled via the command "`DebugEnable`" or in the Scripting menu. Here you can also choose to open the Debug Logger manually..

Example2:
DebugMessage("Macro 'Restart' executed at "+Now)

The second example uses the global variable [1900] "Now" which expresses the current date and time. The Debug Logger displays:

```
Macro 'Restart' executed at 2017-03-09 09:21:05.472
```

Example3:
DebugMessage("info1","info2",varstring,varint)

If you like to display various messages in one line, you can separate them using a comma as done in the third example. The Debug Logger displays:

```
"info1","info2","text stored in string variable",99
```

DebugShow

DebugShow

Example:
DebugShow

This opens Debug Logger [812] dialog.
Please note: This feature is currently not available for the WD Free Version.

DeviceAcceptDmxById

DeviceAcceptDmxById(SiteID,DeviceID,"On"/"Off")

Example:
DeviceAcceptDmxById(2,1,"On")

Patches Layer 1 of Site 2 in the Patch tab [228] so that it can be controlled via an attached DMX / Art-Net device.

Example 2:
DeviceAcceptDmxById(2,1,"Off")

Unpatches Layer 1 of Site 2 in the Patch tab [228] so that it cannot be controlled via an attached DMX / Art-Net device anymore.

Please note:
Art-Net Input needs to be enabled in Configuration Tab [139] in order to remote control the Layer via Art-Net.

To change the Devices DMX / Art-Net start address (Channel, Art-Net Subnet and Universe) use either the Patch Tab in PB or the WD command DeviceSetDmxAddress [1534].

DeviceAddToSelection

DeviceAddToSelection(SiteID,DeviceID)

Example:
DeviceAddToSelection(1,3)

Selects layer 3 of Site 1 (e.g. a Server) and adds it to the current selection within the Pandoras Box project: If other layers were selected before, they are still selected. If you like to select solely layer 3, use the command `DeviceSelect(SiteID,DeviceID)`

DeviceBringToFront

DeviceBringToFront(SiteID,DeviceID)

Example:
DeviceBringToFront(1,3)

Changes the layer order in the Pandoras Box Device Tree [176] for Site 1 (e.g. a Server). Per default, the layer 3 is rendered after layer 1 and 2 and before layer 4,5,6 etc. With this command layer 3 is moved on top of all other layers. Please note that you might need to toggle the layers to be visible.

DeviceFadeParam

DeviceFadeParam(SiteID,DeviceID,Param,StartValue,EndValue,Time)

Example:
DeviceFadeParam(2,3,"Opacity",45,108,11)

The parameter "Opacity" of layer 3 of site 2 starts at value 45 and increases gradually to 108 in 11 seconds. In addition the parameter will be activated and highlighted in red.
Note that the parameter name is case-sensitive, see the topic "Parameter List" [1514]..

DeviceFadeToParam

DeviceFadeToParam(SiteID,DeviceID,Param,Value,Time)

Example:
DeviceFadeToParam(2,4,"Opacity",103,8)

The parameter "Opacity" of layer 4 of site 2 will gradually change in 8 seconds to value 103. In addition the parameter will be activated and highlighted in red.
Note that the parameter name is case-sensitive, see the topic "Parameter List" 1514.

DeviceMoveToBack

DeviceMoveToBack(SiteID,DeviceID)

Example:
DeviceMoveToBack(1,3)

Changes the layer order in the Pandoras Box Device Tree 173 for Site 1 (e.g. a Server). Per default, the layer 3 is rendered after layer 1and 2 and before layer 4,5,6 etc. With this command layer 3 is moved one step behind layer 2 but stays in front of layer 1. To move it behind all other layers, use the command `DeviceSendToBack(SiteID,DeviceID)`. Please note that you might need to toggle the layers to be visible.

DeviceMoveToFront

DeviceMoveToFront(SiteID,DeviceID)

Example:
DeviceMoveToFront(1,3)

Changes the layer order in the Pandoras Box Device Tree 173 for Site 1 (e.g. a Server). Per default, the layer 3 is rendered after layer 1and 2 and before layer 4,5,6 etc. With this command layer 3 is moved one step in front of layer 4 but stays behind layers 5,6 etc. To move it in front all other layers, use the command `DeviceBringToFront(SiteID,DeviceID)`. Please note that you might need to toggle the layers to be visible.

DeviceSelect

DeviceSelect(SiteID,DeviceID)

Example:
DeviceSelect(1,3)

Selects layer 3 of Site 1 (e.g. a Server) within the Pandoras Box project: If other layers were selected before, they are now deselected. If you like to add layer 3 to the current selection, use the command `DeviceAddToSelection(SiteID,DeviceID)`

DeviceSendToBack

DeviceSendToBack(SiteID,DeviceID)

Example:
DeviceSendToBack(1,3)

Changes the layer order in the Pandoras Box Device Tree 173 for Site 1 (e.g. a Server). Per default, the layer 3 is rendered after layer 1and 2 and before layer 4,5,6 etc. With this command layer 3 is moved behind all other layers. Please note that you might need to toggle the layers to be visible.

DeviceSetDmxAddress

DeviceSetDmxAddress(SiteID,DeviceID,DMXStart,Universe,Subnet)

Example:
DeviceSetDmxAddress(2,1,0,5,4)

Patches Layer [2,1] in Pandoras Box to the DMX start address 1, Art-Net Universe 5 and Subnet ID 4.

DeviceSetIp

DeviceSetIp(SiteID,IP Address)

Example:
DeviceSetIp(3,"2.0.0.103")

Sets the IP of Site 3 inside your PB project to 2.0.0.103.

Please note:
This command is only working from PB version 5237 on.

DeviceSetMedia

DeviceSetMedia(SiteID,DeviceID,FolderID,FileID)

Example:
DeviceSetMedia(1,3,2,4)

Assigns the media file with the File and Folder ID 2,4 to layer 3 of Site 1 (e.g. a Server).

DeviceSetMediaByName

DeviceSetMediaByName(SiteID,DeviceID,MediaName)

Example:
DeviceSetMediaByName(1,3,"Testpattern/Calibrate.png")

Assigns the media file with the name "Calibrate.png" from the subfolder "Testpattern" within the project folder to layer 3 of the Site 1 (e.g. a Server). If there is more than one file in the subfolder that holds this name, the first file is taken. If the file is not in a special subfolder, simply use "Calibrate.png". Please note that "MediaName" is case-sensitive.

DeviceSetMediaInSelection

DeviceSetMediaInSelection(FolderID,FileID)

Example:
DeviceSetMediaInSelection(2,4)

Assigns the media file with the File and Folder ID 2,4 to all layers that are currently selected.

DeviceSetMediaParamByID

DeviceSetMediaParamByID(SiteID,DeviceID,ParamName,FolderID,FileID)

Example:
DeviceSetMediaParamByID(1,3,"Quad Media Overlay|Media1",2,4)

Assigns the media file with the File and Folder ID 2,4 to layer 3 of the Site 1 (e.g. a Server) but not as the main media but the first media for the effect named "Quad Media Overlay".

DeviceSetMediaParamByName

DeviceSetMediaParamByName(SiteID,DeviceID,ParamName,MediaName)

Example:
DeviceSetMediaParamByName(1,3,"Quad Media Overlay|Media1","Testpattern/Calibrate.png")

Assigns the media file with the name "Calibrate.png" from the subfolder "Testpattern" within the project folder to layer 3 of the Site 1 (e.g. a Server) but not as the main media but the first media for the effect named "Quad Media Overlay".
If there is more than one file in the subfolder that holds this name, the first file is taken. If the file is not in a special subfolder, simply use "Calibrate.png". Please note that "MediaName" and "ParamName" is case-sensitive. "ParamName" consists of the name of the effect [327] followed by the character pipe (vertical bar) and the name of the media field (in most cases simply "Media").

DeviceSetMesh

DeviceSetMesh(SiteID,DeviceID,FolderID,FileID)

Example:
DeviceSetMesh(1,3,2,4)

Assigns the mesh file with the File and Folder ID 2,4 to layer 3 of Site 1 (e.g. a Server).

DeviceSetMeshByName

DeviceSetMeshByName(SiteID,DeviceID,MediaName)

Example:
DeviceSetMeshByName(1,3,"Objects/Cone.x")

Assigns the mesh file with the name "Cone.x" from the subfolder "Objects" within the project folder to layer 3 of the Site 1 (e.g. a Server). If there is more than one file in the subfolder that holds this name, the first file is taken. If the file is not in a special subfolder, simply use "Cone.x". Please note that "MediaName" is case-sensitive.

DeviceSetMeshInSelection

DeviceSetMeshInSelection(FolderID,FileID)

Example:
DeviceSetMeshInSelection(2,4)

Assigns the mesh file with the File and Folder ID 2,4 to all layers that are currently selected.

DeviceSetMeshParamByID

DeviceSetMeshParamByID(SiteID,DeviceID,ParamName,FolderID,FileID)

Example:
DeviceSetMeshParamByID(1,3,"FXName|Mesh1",2,4)

Assigns the mesh file with the File and Folder ID 2,4 to layer 3 of the Site 1 (e.g. a Server) but not as the main mesh but the first mesh of the effect named "FXName".

DeviceSetMeshParamByName

DeviceSetMeshParamByName(SiteID,DeviceID,ParamName,MediaName)

Example:
DeviceSetMeshParamByName(1,3,"FXName|Mesh1","Objects/Car.x")

Assigns the mesh file with the name "Car.x" from the subfolder "Objects" within the project folder to layer 3 of the Site 1 (e.g. a Server) but not as the main mesh but the first mesh of the effect named "FXName".

If there is more than one file in the subfolder that holds this name, the first file is taken. If the file is not in a special subfolder, simply use "Car.x". Please note that "MediaName" and "ParamName" is case-sensitive. "ParamName" consists of the name of the effect followed by the character pipe (vertical bar) and the name of the mesh field.

DeviceSetParam

DeviceSetParam(SiteID,DeviceID,ParamName,AbsoluteValue)

Example:
DeviceSetParam(1,2,"X Pos",3)

Sets the X Position of Server 1, Layer 2 to the value 3.
If you like to send string values, use the command DeviceSetTextParam 1537.

In the manual, there is a topic with all parameter names 1514.

DeviceSetParamDirect

DeviceSetParamDirect(SiteID,DeviceID,ParamName,AbsoluteValue)

Example:
DeviceSetParamDirect(1,2,"X Pos",3)

In Pandoras Box, the X Position of Server 1, Layer 2 is set to the value 3 WITHOUT applying translation smoothing in the Device Inspector 210. Since PB 9088 all parameters can be smoothend. Before, this applied only to position, scaling and rotation.
Note that the parameter name is case-sensitive, see the topic "Parameter List" 1514.

DeviceSetParamDirectSilent

DeviceSetParamDirectSilent(SiteID,DeviceID,ParamName,AbsoluteValue)

Example:
DeviceSetParamDirectSilent(1,2,"X Pos",3)

This command combines the commands DeviceSetParamDirect and DeviceSetParamSilent:
In Pandoras Box it sets the X Position of Server 1, Layer 2 to the value 3 WITHOUT applying translation smoothing in the Device Inspector 210 and WITHOUT setting the parameter active in PB if it was not active before.
Note that the parameter name is case-sensitive, see the topic "Parameter Llist" 1514.

DeviceSetParamInSelection

DeviceSetParamInSelection(ParamName,AbsoluteValue)

Example:
DeviceSetParamInSelection("Opacity",255)

Sets the Opacity of all selected Devices in PB to the value 255.
Note that the parameter name is case-sensitive, see the topic "Parameter List" 1514.

DeviceSetParamRelative

DeviceSetParamRelative(SiteID,DeviceID,ParamName,RelativeValue)

Example:
DeviceSetParamRelative(1,2,"Opacity",100)

Adds the value 100 to the parameter 1514 Opacity of Server 1, Layer 2.

DeviceSetParamRelativeInSelection

DeviceSetParamRelativeInSelection(ParamName,RelativeValue)

Example:
DeviceSetParamRelativeInSelection("Opacity",20)

Adds the value 20 to the Opacity parameter of all selected Devices in PB.
Note that the parameter name is case-sensitive, see the topic "Parameter List" 1514.

DeviceSetParamSilent

DeviceSetParamSilent(SiteID,DeviceID,ParamName,AbsoluteValue)

Example:
DeviceSetParamSilent(1,2,"X Pos",3)

Sets the X Position of Server 1, Layer 2 to the value 3 WITHOUT setting the parameter active in PB if it was not active before.
Note that the parameter name is case-sensitive, see the topic "Parameter List" 1514.

DeviceSetPreset

DeviceSetPreset(SiteID,DeviceID,PresetID1,PresetID2)

Example:
DeviceSetPreset(1,3,2,4)

Applies the Preset 283 with the ID 2.4 to layer 3 of Site 1 (e.g. a Server). Please note that the command will not paste keys into the timeline but apply the value of the first key.

DeviceSetTextParam

DeviceSetTextParam(SiteID,DeviceID,ParamName,String)

Example:
DeviceSetTextParam(1,3,"Notch|Text|Input Name","New")

This applies the string "New" to the text parameter "Input Name" from the Notch Layer with ID 3 on Site ID 1. The parameter can be found in the group "Notch > Text" which is expressed using the pipe "|"character.

DeviceUnselect

DeviceUnselect(SiteID,DeviceID)

Example:
DeviceUnselect(1,3)

Unselects layer 3 of Site 1 (e.g. a Server) within the Pandoras Box project. If other layers were selected before, they are still selected.

DirCopy

DirCopy(SourcePath,TargetPath)

Example:
DirCopy("C:\Christie\WD_test\Dir_1","C:\Christie\WD_test\Dir_2")

Copies the content of directory Dir_1 into directory Dir_2. If there are files in Dir_2 with the same name as in Dir_1, they are not being replaced.

DirCopyBackup

DirCopyBackup(SourcePath)

Example:
DirCopyBackup("C:\Christie\WD_test\Dir_1")

Generates a backup of the selected folder Dir_1 and saves it with date and time at the same directory as the source folder "C:\Christie\WD_test\Dir_1(2015-09-08_10-12-09)".

DirCopyChanges

DirCopyChanges(SourcePath,TargetPath)

Example:
DirCopyChanges("C:\Christie\WD_test\Dir_1","C:\Christie\WD_test\Dir_2")

Copies the content of directory Dir_1 into directory Dir_2, files in Dir_2 with the same name as files in Dir_1 are being overwritten.

DirCopyOverwrite

DirCopyOverwrite(SourcePath,TargetPath)

Example:
DirCopyOverwrite("C:\Christie\WD_test\Dir_1","C:\Christie\WD_test\Dir_2")

Copies the content of directory Dir_1 into directory Dir_2, files in Dir_2 with the same name as files in Dir_1 are being overwritten.

DirCopyUI

DirCopyUI(SourcePath,TargetPath)

Example:
DirCopyUI("C:\Christie\WD_test\Dir_1","C:\Christie\WD_test\Dir_2")

Copies the content of directory Dir_1 into directory Dir_2. If there are files in Dir_2 with the same name as in Dir_1, a dialog window how to proceed with those files opens at the Widget Designer.

DirDelete

DirDelete(Path)

Example:
DirDelete("C:\Christie\WD_test\Dir_3")

Deletes the whole directory Dir_3 permanently including its content.

DMXLinkInDisable

DMXLinkInDisable

Example:
DMXLinkInDisable

Disables the Widget Designer to receive values via the Christie DMX Link [1999]. This can also be done in the Connection Manager [1258].

DMXLinkInEnable

DMXLinkInEnable

Example:
DMXLinkInEnable

Enables the Widget Designer to receive values via the Christie DMX Link [1999]. This can also be done in the Connection Manager [1258].

DMXLinkInReset

DMXLinkInReset

Example:
DMXLinkInReset

Resets the value of all channels received via the Christie DMX Link [1999] which is enabled in the Connection Manager [1258].

DMXLinkOutDisable

DMXLinkOutDisable

Example:
DMXLinkOutDisable

Disables the Widget Designer to send out values via the Christie DMX Link [1999]. This can also be done in the Connection Manager [1258].

DMXLinkOutEnable

DMXLinkOutEnable

Example:
DMXLinkOutEnable

Enables the Widget Designer to send out values via the Christie DMX Link [1999]. This can also be done in the Connection Manager [1258].

DMXLinkOutReset

DMXLinkOutReset

Example:
DMXLinkOutReset

Resets the value of all channels send out via the Christie DMX Link [1999] which is enabled in the Connection Manager [1258].

DMXLinkOutSetVal16bit

DMXLinkOutSetVal16bit(Channel,Value 0-65535)

Example:
DMXLinkOutSetVal16bit(255,32768)

Sends out the value 32768 on channel 255 via the Christie DMX Link [1999] which is enabled in the Connection Manager [1258].

DMXLinkOutSetVal8bit

DMXLinkOutSetVal8bit(Channel,Value 0-255)

Example:
DMXLinkOutSetVal8bit(255,20)

Sends out the value 20 on channel 255 via the Christie DMX Link [1999] which is enabled in the Connection Manager [1258].

E E

EmailClearInbox

EmailClearInbox

Example:
EmailClearInbox

Clears the Email Inbox from the Email tool [1490].

EmailClearOutbox

EmailClearOutbox

Example:
EmailClearOutbox

Clears the Email Outbox from the Email tool [1490].

EmailExportInbox

EmailExportInbox(FilePath)

Example:
EmailExportInbox("C:\Christie\content\Inbox.txt")

Exports all emails listed in the Inbox from the Email tool [1490] into the text file "Inbox.txt" under the path "C:\Christie\content".

EmailExportOutbox

EmailExportOutbox(FilePath)

Example:
EmailExportOutbox("C:\Christie\content\Outbox.txt")

Exports all emails listed in the Outbox from the Email tool [1490] into the text file "Outbox.txt" under the path "C:\Christie\content".

EmailOutputAddressSet

EmailOutputAddressSet(EmailAddress)

Example:
EmailOutputAddressSet("support.pandorasbox@christiedigital.com")

Sets the Email Address for the outgoing Email Server in the Email Settings Tool [1490] to "support.pandorasbox@christiedigital.com".

EmailOutputPasswordSet

EmailOutputPasswordSet(Password)

Example:
EmailOutputPasswordSet("SECRET")

Sets the Email Password for the outgoing Email Server in the Email Settings Tool [1490] to "SECRET".

EmailOutputSendDisable

EmailOutputSendDisable

Example:
EmailOutputSendDisable

When this command is executed, no email can be sent out. Use the command EmailSendEnable to enable this function again.

Please use the Email Settings Tool [1490] to setup all incoming and outgoing Email communication.

EmailOutputUseDefaultAuthentication

EmailOutputUseDefaultAuthentication(True/False)

Example:
EmailOutputUseDefaultAuthentication("True")

Checks the option "Use Default Authentification" for the outgoing Email Server in the Email Settings Tool [1490].

EmailOutputUsernameSet

EmailOutputUsernameSet(Username)

Example:
EmailOutputUsernameSet("support")

Sets the Email Username for the outgoing Email Server in the Email Settings Tool [1490] to "support".

EmailSend

EmailSend(To,Subject,Message)

Example:
EmailSend("support.pandorasbox@christiedigital.com","Question WD","What are the system requirements for running WD?")

Sends an Email with the subject "Question WD" and the message "What are the system requirements for running WD?" to support.pandorasbox@christiedigital.com.

Please use the Email Settings Tool [1490] first to setup all incoming and outgoing Email communication.

EmailSendAtt

EmailSendAtt(To,Att,Subject,Message)

Example:
EmailSendAtt("support.pandorasbox@christiedigital.com","C:\Christie\commandlist.txt","Question WD","Please explain the commands!")

Sends an Email with the subject "Question WD", the message "Please explain the commands!" and the attachment "commandlist.txt" from the specified directory (C:\Christie\commandlist.txt) to support.pandorasbox@christiedigital.com.

Please use the Email Settings Tool [1490] first to setup all incoming and outgoing Email communication.

EmailSendEnable

EmailSendEnable

Example:
EmailSendEnable

Enables the EmailSend function after it was disabled.

Please use the Email Settings Tool [1490] to setup all incoming and outgoing Email communication.

EmailSendFromTextBox

EmailSendFromTextBox(To,Subject From TextBoxID,Message From TextBoxID)

Example:
EmailSendFromTextBox("support.pandorasbox@christiedigital.com",1,2)

Sends an Email with the text of TextBox 1 as subject and with the text of TextBox 2 as message to support.pandorasbox@christiedigital.com.

Please use the Email Settings Tool [1490] first to setup all incoming and outgoing Email communication.

EmailSendHTML

EmailSendHTML(To,Subject,HTMLcodedMessage)

Example:
EmailSendHTML("support.pandorasbox@christiedigital.com","Format",vCode)

Sends an HTML formatted Email with the subject "Format" to support.pandorasbox@christiedigital.com. The message of the Email is the content of the string variable [1900] "vCode" which contains HTML code (see below).

Please use the Email Settings Tool [1490] first to setup all incoming and outgoing Email communication.

Example for the message which can be copied to the variable for testing:
<!DOCTYPE html> <html> <head> <meta name="viewport" content="width=device-width"> <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"> <style> @media only screen and (max-width: 400px) { body {font-size: 28px; color:red} }</style> </head> <body style="background-color: #000"> Background Text This is HTML formatted text. </body> </html>

EmailServerSet

EmailServerSet(EmailServerSmpt)

Example:
EmailServerSet("smtp.googlemail.com")

Sets the SMTP Mail Server in the Email Settings Tool [1490] to smtp.googlemail.com.

EmailUseSSL

EmailUseSSL(True/False)

Example:
EmailUseSSL(True)

Checks the option "Use SSL" for the outgoing Email Server in the Email Settings Tool [1490].

FileCopy

FileCopy(SourcePath,TargetPath)

Example:
FileCopy("C:\Christie\WD_test\Dir_1\file_1.txt","C:\Christie\WD_test\Dir_2\file_flollop.txt")

Copies the Content of the file "file_1.txt" in directory Dir_1 and pastes it to the newly generated file "file_flollop.txt" in directory Dir_2.

FileDelete

FileDelete(FileName)

Example:
FileDelete("C:\Christie\WD_test\Dir_2\file_4.txt")

Deletes the file "file_4.txt" permanently.

FullScreenByID

FullScreenByID(SiteID)

Example:
FullScreenByID(4)

Switches the site 4 to fullscreen mode in Pandoras Box project whether it is in windowed mode or in fullscreen mode already.
For addressing the machines combined to a Virtual Site 311, you need to use the Site IDs of the single machines as displayed in the Virtual Site Tab.
To toggle the site back into windowed mode , use WindowedModeByID 1884

FullScreenByIP

FullScreenByIP(IP address)

Example:
FullScreenByIP("2.0.0.1")

Switches the Pandoras Box master or client with the IP address 2.0.0.1 to fullscreen mode.
For addressing the machines combined to a Virtual Site 311, you need to use the IPs of the single machines as displayed in the Virtual Site Tab or the command `FullScreenByID` and the individual SiteID.
To toggle the site back into windowed mode, use WindowedModeByIP 1884

HTTPRequestToVar

HTTPRequestToVar(URL,VarName)

Example:
HTTPRequestToVar("https://postman-echo.com/get?foo1=bar1&foo2=bar2","var1")

This allows to send single HTTP commands via scripts directly to a web server and writes the answer to a WD Variable. Please make sure the variable has the correct type. It is not necessary to establish the connection from Widget Designer to the external web server using the Connection Manager but they need to be connected via a switch for example.

The HTTP Request is useful when working with an external web server. The first example refers to the web server from the web site postman-echo.com

Example 2:
HTTPRequestToVar("http://192.168.50.40/cgi-bin/shutter_on","var1")

As well, you may control external devices, e.g. a projector supporting the HTTP protocol. This is an alternative to controlling them via a RS232 / Serial protocol.

Use the command HTTPRequestToVarWithLogin [1545] if the external web server requires a username and / or a passwort and supports a HTTP Basic Authentication (in most browsers that gives you a pop-up window asking for credentials).
The commands WDHTTPRequest [1735] and WDHTTPRequestWithLogin [1735] allow sending a request without writing the answer from the web server to a variable.

HTTPRequestToVarWithLogin

HTTPRequestToVarWithLogin(Username,Password,URL,VarName)

Example:
HTTPRequestToVarWithLogin("postman","password","https://postman-echo.com/basic-auth","var1")

This allows to send single HTTP commands via scripts directly to a web server including required login information and writes the answer to a WD Variable. Please make sure the variable has the correct type. It is not necessary to establish the connection from Widget Designer to the external web server using the Connection Manager but they need to be connected via a switch for example.

The HTTP Request is useful when working with an external web server. The first example refers to the web server from the web site postman-echo.com

Example 2:
HTTPRequestToVar("User1","Password1","http://192.168.50.40/cgi-bin/shutter_on","var1")

As well, you may control external devices, e.g. a projector supporting the HTTP protocol. This is an alternative to controlling them via a RS232 / Serial protocol.

Use the command HTTPRequestToVar [1544] if the external web server does not require a username and / or a passwort nor supports a HTTP Basic Authentication (in most browsers that gives you a pop-up window asking for credentials).
The commands WDHTTPRequest [1735] and WDHTTPRequestWithLogin [1735] allow sending a request without writing the answer from the web server to a variable.

IncludeInSpread

IncludeInSpread(SiteID)

Example:
IncludeInSpread(2)

This deactivates the option "Spare from Spread" for Site ID 2 in Pandoras Box. The option can be found in the Device Inspector [210] .

J J

JsonLoadFileToVar

JsonLoadFileToVar(FilePath,VarName)

Example:
JsonLoadFileToVar("C:\Christie\content\participants.txt","varJson")

This loads the JSON formatted [1926] content of the text file "participants.txt" and writes it in the variable "varJson". The variable can be either of String [1914] or of Json [1922] type.

Exemplary content of the JSON file:

```
{
  "person": [
    {
    "id": "1",
    "name": "Alan",
    "country": "Wales"
    },{
    "id": "2",
    "name": "Louis",
    "country": "France"
    }
  ]
}
```

In case the imported data does not show the same letters etc., check the encoding of your file. Widget Designer imports the following encodings: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8 with a byte order mark (BOM).

JsonLoadXmlFileToVar

JsonLoadXmlFileToVar(FilePath,VarName)

Example:
JsonLoadFileToVar("C:\Christie\content\participants.xml","varJson")

This loads the XML formatted content of the text file "participants.xml" and writes it in the variable "varJson". The variable can be either of String [1914] or of Json [1922] type and the file format can be XML or TXT.

Exemplary content of the XML file:

```
<root>
  <person id='1'>
    <name>Alan</name>
    <country>Wales</country>
  </person>
  <person id='2'>
    <name>Louis</name>
    <country>France</country>
  </person>
</root>
```

Resulting Json Variable:

```
{
  "root": {
```

```
    "person": [
      {
        "@id": "1",
        "name": "Alan",
        "country": "Wales"
      },
      {
        "@id": "2",
        "name": "Louis",
        "country": "France"
      }
    ]
  }
}
```

In case the imported data does not show the same letters etc., check the encoding of your file. Widget Designer imports all common encodings except ANSI. That is: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8 (with or without BOM).

JsonRequestItemToVar

JsonRequestItemToVar(URL,JsonItem,VarName)

Example:
JsonRequestItemToVar("http://time.jsontest.com/","date","varString")

Writes the value of the requested Json item "date" from the URL "http://time.jsontest.com/" into the variable "varString".
Note: Make sure the variable has the correct type: String 1914 or Json 1922.

JsonRequestToVar

JsonRequestToVar(URL,VarName)

Example:
JsonRequestToVar("http://time.jsontest.com/","varJson")

Writes the value of the requested JSON data from the URL "http://time.jsontest.com/" into the variable "varJson".
Note: Make sure the variable has the correct type: String 1914 or Json 1922.

JsonRequestToVarWithLogin

JsonRequestToVarWithLogin(Username,Password,URL,VarName)

Example:
JsonRequestToVarWithLogin("christie",abc","http:\\localhost:8000\timeEntry.json","varJson")

This command logs in to "http:\\localhost:8000\timeEntry.json" with username "christie" and password "abc" and stores the Json data into the variable varJson.
Note: Make sure the variable has the correct type: String 1914 or Json 1922.

K K

KeyboardKeyPress

KeyboardKeyPress(Keycode)

Example:
KeyboardKeyPress("A")

The first example executes a keystroke with the character A.

KeyboardKeyPress("Abc")
This example executes a keystroke with the characters A, then b and c.

KeyboardKeyPress("^c^v")
This example executes the shortcuts to copy and paste.

To combine a key with [Shift], precede the key code with + (plus sign).
To combine a key with [Ctrl], precede the key code with ^ (caret).
To combine a key with [Alt], precede the key code with % (percent sign).

To specify repeating keys, use the form {key number}. You must put a space between key and number.

KeyboardKeyPress("{h 10}")
This example presses the letter h 10 times.

KeyboardKeyPress("{LEFT 42}")
This example presses the [left arrow] key 42 times.

KeyboardKeyPress("{DEL}")
This example presses the [Delete] key.

These keys are available:

{BACKSPACE} or {BS}
{BREAK}
{CAPSLOCK}
{CLEAR}
{DELETE} or {DEL}
Arrow keys: {LEFT} {RIGHT} {DOWN} {UP}
{END}
Enter on the numeric keypad {ENTER}
ENTER ~
ESC {ESCAPE} or {ESC}
{HELP}
{HOME}
{INSERT}
{NUMLOCK}
Page Down and up {PGDN} {PGUP}
{RETURN}
{SCROLLLOCK}
{TAB}
{F1} through {F15}.

If you want to send a keystroke to an application that currently is not activated, you may want to use the command WindowFocus(WindowTitleText) 1884.

KeyboardToggleVolume

KeyboardToggleVolume

Example:
KeyboardToggleVolume

Toggles the Keyboards Sound On / Off Button.

KeyboardVolumeDown

KeyboardVolumeDown

Example:
KeyboardVolumeDown

Executes the Keyboards Volume Down Button.

KeyboardVolumeUp

KeyboardVolumeUp

Example:
KeyboardVolumeUp

Executes the Keyboards Volume Up Button.

KinectDisable

KinectDisable

Example:
KinectDisable

Disables the Kinect [1283] device.

KinectEnable

KinectEnable

Example:
KinectEnable

Enables the Kinect [1283] device.

KinectHideDialog

KinectHideDialog

Example:
KinectHideDialog

This hides the Kinect [1283] Settings dialog.

KinectSetBlur

KinectSetBlur(0-15)

Example:
KinectSetBlur(8)

Changes the blur option in the Kinect Tool [1283] to the value 8.

KinectSetDamping

KinectSetDamping(Value)

Example:
KinectSetDamping(0.1)

Sets the damping in the Kinect tool [1283] to 0,1.

The damping allows to reduce noisy input values. This value can be changed from 0,1 = maximum damping up to 1 = no damping.

KinectSetDepthThreshold

KinectSetDepthThreshold(0-2047)

Example:
KinectSetDepthThreshold(1000)

Sets the Depth Threshold in the Kinect Tool [1283] to the value 1000.

KinectSetInflate

KinectSetInflate(0-15)

Example:
KinectSetInflate(8)

Changes the inflate option in the Kinect Tool [1283] to the value 8.

KinectSetMaxDelta

KinectSetMaxDelta(Value)

Example:
KinectSetMaxDelta(50)

Changes the maximum delta option in the Kinect Tool [1283] to 50 (px).

KinectSetMaxHeight

KinectSetMaxHeight(Value)

Example:
KinectSetMaxHeight(250)

Sets the maximum height of a point in the Kinect Tool [1283] to the value 250 (px).

KinectSetMaxWidth

KinectSetMaxWidth(Value)

Example:
KinectSetMaxWidth(250)

Sets the maximum width of a point in the Kinect Tool [1283] to the value 250 (px).

KinectSetMinHeight

KinectSetMinHeight(Value)

Example:
KinectSetMinHeight(50)

Sets the minimum height of a point in the Kinect Tool [1283] to the value 50 (px).

KinectSetMinWidth

KinectSetMinWidth(Value)

Example:
KinectSetMinWidth(50)

Sets the minimum width of a point in the Kinect Tool [1283] to the value 50 (px).

KinectSetNearThreshold

KinectSetNearThreshold(0-2047)

Example:
KinectSetNearThreshold(800)

Sets the Near Threshold in the Kinect Tool [1283] to the value 900.

KinectSetPoints

KinectSetPoints(1-8)

Example:
KinectSetPoints(4)

Changes the current amount of points in the Kinect Tool [1283] to 4 points.

KinectShowDialog

KinectShowDialog

Example:
KinectShowDialog

This shows the Kinect [1283] Settings dialog.

L L

LogitechPresenterR400Disabled

LogitechPresenterR400Disabled

Example:
LogitechPresenterR400Disabled

Disables the Logitech Presenter R400.

LogitechPresenterR400Enabled

LogitechPresenterR400Enabled

Example:
LogitechPresenterR400Enabled

Enables the Logitech Presenter R400.

M M

MagicQActivatePlayback

MagicQActivatePlayback(ID)

Example:
MagicQActivatePlayback(8)

Activates (starts) the playback 8 within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQChangePage

MagicQChangePage(ID)

Example:
MagicQChangePage(11)

Changes to playback page 11 within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQChannelLevel

MagicQChannelLevel(ID,Level)

Example:
MagicQChannelLevel(8,90)

Sets the Level of channel 8 within MagicQ to 90%.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQFastBackOnPlayback

MagicQFastBackOnPlayback(ID)

Example:
MagicQFastBackOnPlayback(8)

Goes back one step in the Cue Stack of Playback 8 without taking the step and fade times into account.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQFastForwardOnPlayback

MagicQFastForwardOnPlayback(ID)

Example:
MagicQFastForwardOnPlayback(8)

Goes back one step in the Cue Stack of Playback 8 without taking the step and fade times into account.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQGoOnPlayback

MagicQGoOnPlayback(ID)

Example:
MagicQGoOnPlayback(8)

Gives a Go on playback 8 within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQJumpToCue

MagicQJumpToCue(ID,CueID)

Example:
MagicQJumpToCue(8,4)

Jumps to cue 4 at playlist 8 within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQPlaybackLevel

MagicQPlaybackLevel(ID,Level)

Example:
MagicQPlaybackLevel(8,90)

Sets the Level of Playback 8 within MagicQ to 90%.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQReleasePlayback

MagicQReleasePlayback(ID)

Example:
MagicQReleasePlayback(8)

Releases (stops) the playback 8 within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQRemoteTrigger

MagicQRemoteTrigger(State)

Example:
MagicQRemoteTrigger(State)

Sends a remote trigger signal to MagicQ.

Note: The absolute state is 0 or 1, 2 can be used for toggling the current state.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQStopOnPlayback

MagicQStopOnPlayback(ID)

Example:
MagicQStopOnPlayback(8)

Sets playback 8 within MagicQ to pause.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQTestCue

MagicQTestCue(ID)

Example:
MagicQTestCue(4)

Sets cue 4 in test mode within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQTestCueStack

MagicQTestCueStack(ID)

Example:
MagicQTestCueStack(6)

Sets cue stack 6 in test mode within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQTestPlayback

MagicQTestPlayback(ID)

Example:
MagicQTestPlayback(8)

Sets the playback 8 in test mode within MagicQ: activates playback 8 and sets the level to 100%.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQUntestCue

MagicQUntestCue(ID)

Example:
MagicQUntestCue(4)

Takes cue 4 out of test mode within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQUntestCueStack

MagicQUntestCueStack(ID)

Example:
MagicQUntestCueStack(6)

Takes cue stack 6 out of test mode within MagicQ.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MagicQUntestPlayback

MagicQUntestPlayback(ID)

Example:
MagicQUntestPlayback(8)

Takes the playback 8 out of test mode within MagicQ: releases the playback 8 and sets the level to 0%.

This command is a Chamsys MagicQ Remote Playback Command based on the MagicQ default UDP Port 6553.

MA-NetInputDisabled

MANetInputDisabled

Example:
MANetInputDisabled

Disables the MA Net Input in the Connection Manager [1258].

MA-NetInputEnabled

MANetInputEnabled

Example:
MANetInputEnabled

Enables the MA Net Input in the Connection Manager [1258].

MANetSessionID

MANetSessionID(ID)

Example:
MANetSessionID(2)

Sets the MA Net Session ID in the Connection Manager [1258] to 2.

MidiInputDisabled

MidiInputDisabled

Example:
MidiInputDisabled

Disables the Midi Input in the Connection Manager [1258].

MidiInputEnabled

MidiInputEnabled

Example:
MidiInputEnabled

Enables Midi Input in the Connection Manager [1258]. Please choose a Midi Device first.

MidiNotesMuteDisable

MidiNotesMuteDisable

Example:
MidiNotesMuteDisable

Disables incoming Midi Notes in the Midi Notes Editor Tool [1486].

MidiNotesMuteEnable

MidiNotesMuteEnable

Example:
MidiNotesMuteEnable

Activates incoming Midi Notes in the Midi Notes Editor Tool [1486].

MidiOutputDisabled

MidiOutputDisabled

Example:
MidiOutputDisabled

Disables the Midi Output in the Connection Manager [1258].

MidiOutputEnabled

MidiOutputEnabled

Example:
MidiOutputEnabled

Enables Midi Output in the Connection Manager [1258]. Please choose a Midi Device first.

MidiSendControlChange

MidiSendControlChange(Channel 1-16,Control 0-127,Value 0-127)

Example:
MidiSendControlChange(1,35,127)

Sets the Control 35 on Midi Channel 1 to the value 127.

The Midi connection needs to be enabled in the Connection Manager [1258].
If you have more than one Midi device, please use the command "MidiSendControlChangeById [1557]".

MidiSendControlChangeByID

MidiSendControlChangeByID(ConnectionID,Channel 1-16,Control 0-127,Value 0-127)

Example:
MidiSendControlChangeByID(3,1,35,127)

This sends a message to the Midi device set up as "Midi Output 3" in the Connections > Midi Connections dialog [1271]. It sets the Control 35 on Midi Channel 1 to the value 127.

MidiSendNoteOff

MidiSendNoteOff(Channel 1-16,Note C0-G10)

Example:
MidiSendNoteOff(1,"E5")

Sends the Midi Note Off command to Note E5 on channel 1.

The Midi connection needs to be enabled in the Connection Manager [1258].
If you have more than one Midi device, please use the command "MidiSendNoteOffByID [1557]".

MidiSendNoteOffByID

MidiSendNoteOffByID(ConnectionID,Channel 1-16,Note C0-G10)

Example:
MidiSendNoteOffByID(3,1,"E5")

This sends a message to the Midi device set up as "Midi Output 3" in the Connections > Midi Connections dialog [1271]. It sends the Midi Note Off command to Note E5 on channel 1.

MidiSendNoteOn

MidiSendNoteOn(Channel 1-16,Note C0-G10,Velocity 0-127)

Example:
MidiSendNoteOn(1,"E5",80)

Sends the Midi Note On command to Note E5 on channel 1 with velocity 80.

The Midi connection needs to be enabled in the Connection Manager [1258].
If you have more than one Midi device, please use the command "MidiSendNoteOnByID [1558]".

MidiSendNoteOnByID

MidiSendNoteOnByID(ConnectionID,Channel 1-16,Note C0-G10,Velocity 0-127)

Example:
MidiSendNoteOnByID(3,1,"E5",80)

This sends a message to the Midi device set up as "Midi Output 3" in the Connections > Midi Connections dialog [1271]. It sends the Midi Note On command to Note E5 on channel 1 with velocity 80.

MidiSendProgramChange

MidiSendProgramChange(Channel 1-16,Program 0-127)

Example:
MidiSendProgramChange(1,20)

Changes the current program for Midi channel 1 to the program 20.

The Midi connection needs to be enabled in the Connection Manager [1258].
If you have more than one Midi device, please use the command "MidiSendProgramChangeByID [1558]".

MidiSendProgramChangeByID

MidiSendProgramChangeByID(ConnectionID,Channel 1-16,Program 0-127)

Example:
MidiSendProgramChangeByID(3,1,20)

This sends a message to the Midi device set up as "Midi Output 3" in the Connections > Midi Connections dialog [1271]. It changes the current program for Midi channel 1 to the program 20.

MidiSendRawMsg

MidiSendRawMsg(Status,Channel,Data1,Data2,Data3)

Example:
MidiSendRawMsg(10,11,127,0,0)

Sends a Midi Raw Message with the 5 Midi control bytes 10,11,127,0,0.

The Midi connection needs to be enabled in the Connection Manager [1258].

MidiSendRawMsgByID

MidiSendRawMsgByID(Status,Channel,Data1,Data2,Data3)

Example:
MidiSendRawMsgByID(10,11,127,0,0)

Sends a Midi Raw Message with the 5 Midi control bytes 10,11,127,0,0 via the Midi device with ID 3. Please select and activate the according MIDI output device in the Connections > Midi Connections dialog [1271].

MIDISendSysEx

MIDISendSysEx([Data])

Example:
MIDISendSysEx("[hf0 h7f h01 h02 h01 h01 h32 h34 hF7]")

This sends a Midi "System Exclusive Message". Please select and activate a Midi output device in the Connection Manager [1260]. The hexadecimal message needs to be enclosed in square brackets and as any string it is put in quotation marks. The data itself is written in the hxx format.

If you send this example to Pandoras Box (with activated MSC), it will go to cue 24. For more information about Pandoras Box SysEx please go to the chapter "MSC Midi Show Control" [716] in the manual.
If you have more than one Midi device, please use the command MIDISendSysExByID.

MIDISendSysExByID

MIDISendSysExByID(ID,[Data])

Example:
MIDISendSysExByID(3,"[hf0 h7f h01 h02 h01 h01 h32 h34 hF7]")

This sends a Midi "System Exclusive Message" via the Midi device with ID 3. Please select and activate the according MIDI output device in the Connections > Midi Connections dialog [1271]. The hexadecimal message needs to be enclosed in square brackets and as any string it is put in quotation marks. The data itself is written in the hxx format.

If you send this example to Pandoras Box (with activated MSC), it will go to cue 24. For more information about Pandoras Box SysEx please go to the chapter "MSC Midi Show Control" [716] in the manual.

MouseLeftClick

MouseLeftClick

Example:
MouseLeftClick

Executes a Mouse Left Click.

MouseLeftDown

MouseLeftDown

Example:
MouseLeftDown

Simulates the depression of the left mouse button.

MouseLeftUp

MouseLeftUp

Example:
MouseLeftUp

Simulates the release of the left mouse button

MouseMove

MouseMove(xPos,yPos)

Example:
MouseMove(200,100)

Moves the mouse cursor to the X Position 200 px and Y Position 100 px.

Tip: If you want to display the current mouse values, create a Mouse Input Node [1016] and see its Item Properties.

MouseRightClick

MouseRightClick

Example:
MouseRightClick

Executes a Mouse Right Click.

MouseRightDown

MouseRightDown

Example:
MouseRightDown

Simulates the depression of the right mouse button.

MouseRightUp

MouseRightUp

Example:
MouseRightUp

Simulates the release of the right mouse button

OpenAirScanProperties

OpenAirScanProperties

Example:
OpenAirScanProperties

Opens the dialog for the AirScan tool [1277].

OpenArtNetMonitor

OpenArtNetMonitor

Example:
OpenArtNetMonitor

Opens the Art-NetMonitor [2051] which can also be accessed through the Connection Manager [1258].

OpenArt-NetUniverseList

OpenArtNetUniverseList

Example:
OpenArtNetUniverseList

Opens the Art-Net Universe List which can also be accessed through the Connection Manager [1258].

OpenCameraTrackerDialog

OpenCameraTrackerDialog

Example:
OpenCameraTrackerDialog

Opens the Camera Tracker [1291] dialog.

OpenCITPBrowser

OpenCITPBrowser

Example:
OpenCITPBrowser

Opens the Thumbnail Browser which can also be accessed through the dialog PB Network Configuration [1256].

OpenComConnections

OpenComConnections

Example:
OpenComConnections

Opens the COM Connection [1269] dialog which gives an better overview than the Connection Manager [1258].

OpenConnectionManager

OpenConnectionManager

Example:
OpenConnectionManager

Opens the Connection Manager [1258].

OpenEmailSettings

OpenEmailSettings

Example:
OpenEmailSettings

Opens the Email Settings Tool [1490].

OpenHelp

OpenHelp

Example:
OpenHelp

Opens this Help File.

OpenIPConfiguration

OpenIPConfiguration

Example:
OpenIPConfiguration

Opens the PB Network Configuration [1256] dialog.

OpenKeyboardShortcuts

OpenKeyboardShortcuts

Example:
OpenKeyboardShortcuts

Opens the Keyboard Shortcut Editor [1484].

OpenKinectDialog

OpenKinectDialog

Example:
OpenKinectDialog

Opens the Kinect Tool [1283].

OpenLoadFileDialog

OpenLoadFileDialog

Example:
OpenLoadFileDialog

Opens the "Load File" dialog in WD to load an existing project.

OpenMacroEditor

OpenMacroEditor(MacroName)

Example:
OpenMacroEditor("MyMacro")

Opens the Macro Editor dialog in WD and loads the macro "MyMacro".

OpenMidiNoteEditor

OpenMidiNoteEditor

Example:
OpenMidiNoteEditor

Opens the Midi Note Editor [1486].

OpenPageBrowser

OpenPageBrowser

Example:
OpenPageBrowser

Opens the Page Browser [799].

OpenRemoteInput

OpenRemoteInput

Example:
OpenRemoteInput

Opens the Remote Input Tool [1273].

OpenSaveAsFileDialog

OpenSaveAsFileDialog

Example:
OpenSaveAsFileDialog

Opens the "Save as" dialog in WD to save the current project to a different name and directory.

OpenSMSSettings

OpenSMSSettings

Example:
OpenSMSSettings

Opens the SMS Settings Tool [1494].

OpenTcpConnections

OpenTcpConnections

Example:
OpenTcpConnections

Opens the TCP Connection [1265] dialog which gives an better overview than the Connection Manager [1258].

OpenUdpConnections

OpenUdpConnections

Example:
OpenUdpConnections

Opens the UDP Connection [1267] dialog which gives an better overview than the Connection Manager [1258].

OpenVariableList

OpenVariableList

Example:
OpenVariableList

Opens the dialog Variable List [1901].

P P

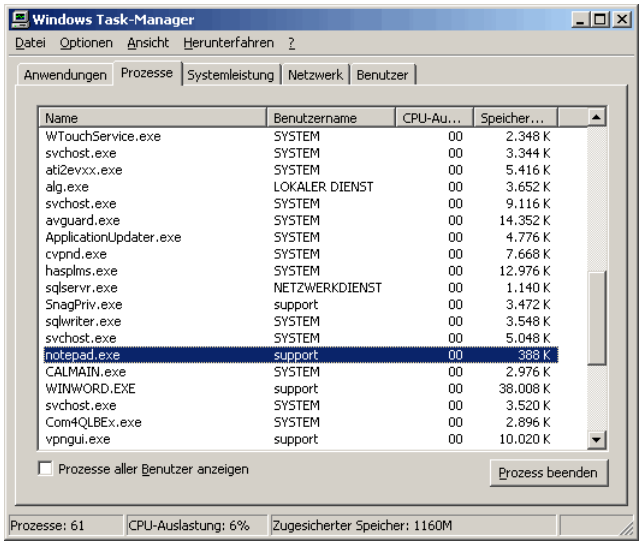
PBApplicationClose

PBApplicationClose(IP address,Processname,NicAdapterName(optional))

Example:
PBApplicationClose("10.169.10.60","notepad","Lan1")

Closes the running notepad application on the PB computer with the IP address 10.169.10.60 in the local network. Please have a look into the Windows Task Manager to get the correct process name of the running application you want to close, enter this name without ".exe".

The command is sent out via NIC adapter "Lan1" of the local PC.



Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

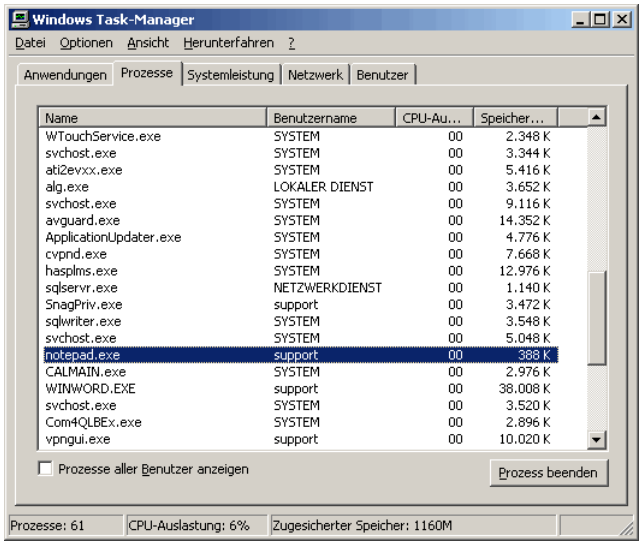
PBApplicationCloseAll

PBApplicationCloseAll(Processname,NicAdapterName(optional))

Example:
PBApplicationCloseAll("notepad","Lan1")

Closes the running notepad application on all PB computers in the local network. Please have a look into the Windows Task Manager to get the correct process name of the running application you want to close, enter this name without ".exe".

The command is sent out via NIC adapter "Lan1" of the local PC.



Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBApplicationStart

PBApplicationStart(IP address,Filepath,NicAdapterName(optional))

Example:
PBApplicationStart("10.169.10.60","notepad.exe","Lan1")

Opens the notepad on the PB computer with the IP address 10.169.10.60 in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Example 2:
PBApplicationStart("10.169.10.60","C:\Program Files\Internet Explorer\iexplore.exe")

Opens the Internet Explorer on the PB computer with the IP address 10.169.10.60 in the local network.
If the application you want to start is not a standard Windows application, please use the whole file path.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBApplicationStartAll

PBApplicationStartAll(Filepath,NicAdapterName(optional))

Example:
PBApplicationStartAll("notepad.exe","Lan1")

Opens the notepad on all PB computers in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Example 2:
PBApplicationStartAll("C:\Program Files\Internet Explorer\iexplore.exe")

Opens the Internet Explorer on all PB computers in the local network.
If the application you want to start is not a standard Windows application, please use the whole file path.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBApplicationStartAllEx

PBApplicationStartAllEx(Filepath,Application arguments,NicAdapterName(optional))

Example:
PBApplicationStartEx("C:\path\the_answer.bat","42","Lan1")

Executes "the_answer.bat" on the PB computers in the local network and passes the argument "42". The command is sent out via NIC adapter "Lan1" of the local PC.
If your application or file requires more than one argument, you can separate them by a white space, e.g. "mattresses flollopping and volluing floopily".

If the application you want to start is not a standard Windows application, please use the whole file path.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and

therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBApplicationStartEx

PBApplicationStartEx(IP address,Filepath,Application arguments,NicAdapterName(optional))

Example:
PBApplicationStartEx("10.169.10.60","C:\path\the_answer.bat","42","Lan1")

Executes "the_answer.bat" on the PB computer with the IP address 10.169.10.60 in the local network and passes the argument "42". The command is sent out via NIC adapter "Lan1" of the local PC.
If your application or file requires more than one argument, you can separate them by a white space, e.g. "mattresses flollopping and volluing floopily".

If the application you want to start is not a standard Windows application, please use the whole file path.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBCanvasUpdatesDisableForSite

PBCanvasUpdatesDisableForSite(FolderID,FileID,SiteID)

Example:
PBCanvasUpdatesDisableForSite(1,3,2)

This applies to the Canvas [280] with Folder ID 1 and File ID 3 and sets the option "Block canvas updates for selected nodes" in the Canvas Inspector [199] so that the Site with ID 2 is blocked. The Site does not receive Canvas updates any more even if it is listed in the File Location Table (at the bottom of the Inspector).

Note that this command works only with a PB version below 6.0.4 as then, the option was removed. Now, a Canvas is spread like other Resource too, so simply restrict the spreading to those Sites that should display the Canvas. The following commands could be helpful: ResourceAttachByID [1583], ResourceDetachByID [1585] or ResourceSpreadToSiteById [1600]

PBCanvasUpdatesEnableForSite

PBCanvasUpdatesEnableForSite(FolderID,FileID,SiteID)

Example:
PBCanvasUpdatesEnableForSite(1,3,2)

This applies to the Canvas [280] with Folder ID 1 and File ID 3 and removes the option "Block canvas updates for selected nodes" in the Canvas Inspector [199] for Site with ID 2 if it was blocked before. The Site does receive Canvas updates again if it is listed in the File Location Table (at the bottom of the Inspector).

Note that this command works only with a PB version below 6.0.4 as then, the option was removed. Now, a Canvas is spread like other Resource too, so simply restrict the spreading to those Sites that should display the Canvas. The following commands could be helpful: ResourceAttachByID [1583], ResourceDetachByID [1585] or ResourceSpreadToSiteById [1600]

PBCanvasUpdatesExclusiveForSite

PBCanvasUpdatesExclusiveForSite(FolderID,FileID,SiteID)

Example:
PBCanvasUpdatesExclusiveForSite(1,3,2)

This applies to the Canvas [280] with Folder ID 1 and File ID 3 and sets the option "Block canvas updates for selected nodes" in the Canvas Inspector [199] so that only the Site with ID 2 is not blocked. The Site does receive Canvas updates if it is listed in the File Location Table (at the bottom of the Inspector).

Note that this command works only with a PB version below 6.0.4 as then, the option was removed. Now, a Canvas is spread like other Resource too, so simply restrict the spreading to those Sites that should display the Canvas. The following commands could be helpful: ResourceAttachByID [1583], ResourceDetachByID [1585] or ResourceSpreadToSiteById [1600]

PBClose

PBClose(IP address,NicAdapterName(optional))

Example:
PBClose("10.169.10.65","Lan1")

Closes the PB software on the computer with the IP address 10.169.10.65 in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBCloseAll

PBCloseAll(NicAdapterName(optional))

Example:
PBCloseAll("Lan1")

Closes the PB software on all computers in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Example2:
PBCloseAll()

Use this, if you do not need to specify the NIC adapter explicitly.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBCloseProject

PBCloseProject(SaveCurrent)

Example:
PBCloseProject(True)

Closes the current project in PB after saving it.

Example 2:
PBCloseProject(False)

Closes the current project in PB without saving it.

PBCueMap

PBCueMap(ButtonID,SeqID,CueID)

Example:
PBCueMap(2,1,10)

Links Button 2 of the PB Jog/Shuttle Board to Cue 10 of sequence 1 in Pandoras Box Master System.

PBDeactivateOutput2

PBDeactivateOutput2(IP address)

Example:
PBDeactivateOutput2("10.169.10.65")

Deactivates output 2 of the PB computer with the IP address 10.169.10.65.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBDeactivateOutput2All

PBDeactivateOutput2All

Example:
PBDeactivateOutput2All

Deactivates output 2 of all PB computers in the local network.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBFaderMap

PBFaderMap(FaderID,SeqID)

Example:
PBFaderMap(3,1)

Links Fader 3 of the PB Fader Board to the sequence 1 in Pandoras Box Master System.

PBLoadProject

PBLoadProject(Path,Name,SaveCurrent)

Example:
PBLoadProject("C:\Christie\content\projects","Rotation.pbb",True)

Saves the current PB project and opens the project "Rotation.pbb" from the specified directory.

Example 2:
PBLoadProject("C:\Christie\content\projects","Rotation.pbb",False)

Opens the PB project "Rotation.pbb" from the specified directory without saving the current project.

PbPlaylistJumpToNextItemByID

PbPlaylistJumpToNextItemByID(SiteID,DeviceID,FolderID,FileID)

Example:
PbPlaylistJumpToNextItemByID(2,1,3,5)

This jumps to the next entry in the PB Playlist with the Folder/File ID [3,5] which is assigned to the Device 1 of Site 2.

Please use Pandoras Box version 8.3 or higher to navigate its Playlists.

PbPlaylistJumpToNextItemByName

PbPlaylistJumpToNextItemByName(SiteID,DeviceID,ProjectPath)

Example:
PbPlaylistJumpToNextItemByName(2,1,"test/Videoloop")

This jumps to the next entry in the PB Playlist named "Videoloop" from the project folder "test" which is assigned to the Device 1 of Site 2.

Please use Pandoras Box version 8.3 or higher to navigate its Playlists.

PbPlaylistJumpToPreviousItemByID

PbPlaylistJumpToPreviousItemByID(SiteID,DeviceID,FolderID,FileID)

PbPlaylistJumpToPreviousItemByID(2,1,3,5)

This jumps to the previous entry in the PB Playlist with the Folder/File ID [3,5] which is assigned to the Device 1 of Site 2.

Please use Pandoras Box version 8.3 or higher to navigate its Playlists.

PbPlaylistJumpToPreviousItemByName

PbPlaylistJumpToPreviousItemByName(SiteID,DeviceID,ProjectPath)

PbPlaylistJumpToPreviousItemByID(2,1,"test/Videoloop")

This jumps to the previous entry in the PB Playlist named "Videoloop" from the project folder "test" which is assigned to the Device 1 of Site 2.

Please use Pandoras Box version 8.3 or higher to navigate its Playlists.

PBProjectRefresh

PBProjectRefresh

Example:
PBProjectRefresh

This reads-out information from a Pandoras Box project and stores it in a temporary cache. The command needs to be called before other commands like `VGetAsset...`

PBReboot

PBReboot(IP address,NicAdapterName(optional))

Example:
PBReboot("10.169.10.65","Lan1")

Reboots the PB computer with the IP address 10.169.10.65 in the local network. The example names the optional NIC adapter for sending the command with "Lan1". If you do not need to specify the NIC adapter explicitly, simply write PBReboot("10.169.10.65")

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBRebootAll

PBRebootAll(NicAdapterName(optional))

Example:
PBRebootAll("Lan1")

Reboots all PB computers in the local network. The example names the optional NIC adapter for sending the command with "Lan1".

Example2:
PBRebootAll()

Use this, if you do not need to specify the NIC adapter explicitly.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSaveProject

PBSaveProject

Example:
PBSaveProject

Saves the current project of the PB Master device that is connected to the Widget Designer.

PBSetIP1

PBSetIP1(IP address,NewIP,NewMask,NicAdapterName(optional))

Example:
PBSetIP1("10.169.10.65","2.0.0.115","255.255.255.0","Lan1")

Changes the first IP address of the PB computer with the IP address 10.169.10.65 to the IP 2.0.0.115 with the subnet mask 255.255.255.0.

The example names the optional NIC adapter for sending the command with "Lan1". If you do not need to specify the NIC adapter explicitly, simply write PBSetIP1("10.169.10.65","2.0.0.115","255.255.255.0")

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSetIP1DHCP

PBSetIP1DHCP(IP address)

Example:
PBSetIP1DHCP("2.0.0.111")

Sets the first network adapter of the PB computer with the IP address 2.0.0.111 to DHCP.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSetIP2

PBSetIP2(IP address,NewIP,NewMaskNicAdapterName(optional))

Example:
PBSetIP2("10.169.10.65","2.0.0.115","255.255.255.0","Lan1")

Changes the second IP address of the PB computer with the IP address 10.169.10.65 to the IP 2.0.0.115 with the subnet mask 255.255.255.0
The example names the optional NIC adapter for sending the command with "Lan1". If you do not need to specify the NIC adapter explicitly, simply write PBSetIP2("10.169.10.65","2.0.0.115","255.255.255.0")

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSetIP2DHCP

PBSetIP2DHCP(IP address)

Example:
PBSetIP2DHCP("2.0.0.111")

Sets the second network adapter of the PB computer with the IP address 2.0.0.111 to DHCP.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSetOutput1Resolution

PBSetOutput1Resolution(IP address,1024 768 60)

Example:
PBSetOutput1Resolution("10.169.10.65",1920 1080 60)

Sets the resolution of output 1 of the PB computer with the IP address 10.169.10.65 to 1920x1080 px and 60 Hz.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSetOutput1ResolutionAll

PBSetOutput1ResolutionAll(1024 768 60)

Example:
PBSetOutput1ResolutionAll(1024 768 60)

Sets the resolution of output 1 of all PB computers in the local network to 1024x786 px @ 60 Hz.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSetOutput2Resolution

PBSetOutput2Resolution(IP address,1024 768 60)

Example:
PBSetOutput2Resolution("10.169.10.65",1920 1080 60)

Sets the resolution of output 2 of the PB computer with the IP address 10.169.10.65 to 1920x1080 px and 60 Hz.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBSetOutput2ResolutionAll

PBSetOutput2ResolutionAll(1024 768 60)

Example:
PBSetOutput2ResolutionAll(1024 768 60)

Sets the resolution of output 2 of all PB computers in the local network to 1024x786 px @ 60 Hz.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBShutDown

PBShutDown(IP address,NicAdapterName(optional))

Example:
PBShutDown("10.169.10.65","Lan1")

Turns off the PB computer with the IP address 10.169.10.65 in the local network.
The example names the optional NIC adapter for sending the command with "Lan1". If you do not need to specify the NIC adapter explicitly, simply write PBShutDown("10.169.10.65")

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBShutDownAll

PBShutDownAll(NicAdapterName(optional))

Example:
PBShutDownAll("Lan1")

Shuts down all PB computers in the local network.
The example names the optional NIC adapter for sending the command with "Lan1".

Example2:
PBShutDownAll()

Use this, if you do not need to specify the NIC adapter explicitly.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBStandby

PBStandby(IP address,NicAdapterName(optional))

Example:
PBStandby("10.169.10.65","Lan1")

Sets the PB computer with the IP address 10.169.10.65 in standby mode.
The example names the optional NIC adapter for sending the command with "Lan1". If you do not need to specify the NIC adapter explicitly, simply write PBStandby("10.169.10.65")

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBStandbyAll

PBStandbyAll(NicAdapterName(optional))

Example:
PBStandbyAll("Lan1")

Sets all PB computer with the same IP range in standby mode
The example names the optional NIC adapter for sending the command with "Lan1".

Example2:
PBStandbyAll()

Use this, if you do not need to specify the NIC adapter explicitly.

Please note:
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBStartClient

PBStartClient(IP address,NicAdapterName(optional))

Example:
PBStartClient("10.169.10.65","Lan1")

Launches the PB Client software on the computer with the IP address 10.169.10.65 in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBStartClientAll

PBStartClientAll(NicAdapterName(optional))

Example:
PBStartClientAll("Lan1")

Launches the PB Client software on all computers in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Example2:
PBStartClientAll()

Use this, if you do not need to specify the NIC adapter explicitly.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBStartMaster

PBStartMaster(IP address,NicAdapterName(optional))

Example:
PBStartMaster("10.169.10.65","Lan1")

Launches the PB Master software on the computer with the IP address 10.169.10.65 in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PBStartMasterAll

PBStartMasterAll(NicAdapterName(optional))

Example:
PBStartMasterAll("Lan1")

Launches the PB Master software on all computers in the local network. The command is sent out via NIC adapter "Lan1" of the local PC.

Example2:
PBStartMasterAll()

Use this, if you do not need to specify the NIC adapter explicitly.

Please note:
If you do not specify the NIC adapter name, it might happen that the command binds to the wrong adapter and therefore does not send it out where it was intended to.
All PB commands will only work when the PB Menu is installed on the computers, the current version can be found in our Download-Center.

PlayClientSound

PlayClientSound(Filename)

Example:
PlayClientSound("C:\Christie\media\instrument.wav")

Plays the wave file "instrument.wav" from the specified directory on the current client (depending on the script's context). Please note, that WD does not play another format than wave files.

PlayClientSoundLoop

PlayClientSoundLoop(Filename)

Example:
PlayClientSoundLoop(C:\Christie\media\instrument.wav)

Plays (and loops) the wave file "instrument.wav" from the specified directory on the current client (depending on the script's context). Please note, that WD does not play another format than wave files.

PlaySound

PlaySound(Filename)

Example:
PlaySound("C:\Christie\media\instrument.wav")

Plays the wave file "instrument.wav" from the specified directory. Please note, that WD does not play another format than wave files.

PlaySoundLoop

PlaySoundLoop(Filename)

Example:
PlaySoundLoop(C:\Christie\media\instrument.wav)

Loops the wave file "instrument.wav" from the specified directory. Please note, that WD does not play another format than WAV files.

Q Q

R R

ReconnectFaderboardDevices

ReconnectFaderboardDevices

Example:
ReconnectFaderboardDevices

This command resets the USB connection to all available fader board controllers (i.e. Jog/Shuttle [1991] and Fader Extension [1993]) of your Pandoras Box Master and Backup connection. The IP address and Domain for the "Master Connection" and "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ReconnectFaderboardDevicesBackup

ReconnectFaderboardDevicesBackup

Example:
ReconnectFaderboardDevicesBackup

This command resets the USB connection to all available fader board controllers (i.e. Jog/Shuttle [1991] and Fader Extension [1993]) of your Pandoras Box Backup connection. The IP address and Domain for the "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ReconnectFaderboardDevicesMaster

ReconnectFaderboardDevicesMaster

Example:
ReconnectFaderboardDevicesMaster

This command resets the USB connection to all available fader board controllers (i.e. Jog/Shuttle [1991] and Fader Extension [1993]) of your Pandoras Box Master connection. The IP address and Domain for the "Master Connection" and "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

RemoveGraphicLayer

RemoveGraphicLayer(SiteID,DeviceID)

Example:
RemoveGraphicLayer(1,3)

Removes layer 3 from the Site 1 (e.g. a Server) if it is a Graphic Layer. This command does not just toggle the layer invisible, it deletes it entirely. Note that graphic layers are discontinued since Pandoras Box version 8.0.0

RemoveVideoLayer

RemoveVideoLayer(SiteID,DeviceID)

Example:
RemoveVideoLayer(1,3)

Removes layer 3 from the Site 1 (e.g. a Server) if it is a Video Layer. This command does not just toggle the layer invisible, it deletes it entirely.

ResetAll

ResetAll

Example:
ResetAll

Resets all parameters of all layers of all Servers within the Pandoras Box Project.

ResetDevice

ResetDevice(SiteID,DeviceID)

Example:
ResetDevice(1,3)

Resets all parameters of layer 3 of site 1 within the Pandoras Box Project.

ResetParam

ResetParam(SiteID,DeviceID,ParamName)

Example:
ResetParam(1,3,"X Scale")

Resets the parameter [1514] X Scale of layer 3 of site 1 within the Pandoras Box Project.

ResetSerialLinkDeviceConnection

ResetSerialLinkDeviceConnection(SiteID,DeviceID)

Example:
ResetSerialLinkDeviceConnection(3,1)

Resets the connection of the Serial Link Device [692] with the Device ID [2.1] of your Pandoras Box Master and Backup connection. In the status bar of PB and the Devices Tree, you will see that the device is dis- and reconnected.
The IP address and Domain for the "Master Connection" and "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ResetSerialLinkDeviceConnectionBackup

ResetSerialLinkDeviceConnectionBackup(SiteID,DeviceID)

Example:
ResetSerialLinkDeviceConnectionBackup(3,1)

Resets the connection of the Serial Link Device [692] with the Device ID [3.1] of your Pandoras Box Backup connection. In the status bar of PB and the Devices Tree, you will see that the device is dis- and reconnected.
The IP address and Domain for the "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ResetSerialLinkDeviceConnectionMaster

ResetSerialLinkDeviceConnectionMaster(SiteID,DeviceID)

Example:
ResetSerialLinkDeviceConnectionMaster(3,1)

Resets the connection of the Serial Link Device [692] with the Device ID [3.1] of your Pandoras Box Master connection. In the status bar of PB and the Devices Tree, you will see that the device is dis- and reconnected.
The IP address and Domain for the "Master Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ResetSite

ResetSite(SiteID)

Example:
ResetSite(1)

Resets all parameters of all layers of site 1 within the Pandoras Box Project.

ResetTcpDeviceConnection

ResetTcpDeviceConnection(SiteID,DeviceID)

Example:
ResetTcpDeviceConnection(3,1)

Resets the connection of the TCP Device [318] with the Device ID [2.1] of your Pandoras Box Master and Backup connection. In the status bar of PB and the Devices Tree, you will see that the device is dis- and reconnected. The IP address and Domain for the "Master Connection" and "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ResetTcpDeviceConnectionBackup

ResetTcpDeviceConnectionBackup(SiteID,DeviceID)

Example:
ResetTcpDeviceConnectionBackup(2,1)

Resets the connection of the TCP Device [318] with the Device ID [2.1] of your Pandoras Box Backup connection. In the status bar of PB and the Devices Tree, you will see that the device is dis- and reconnected.
The IP address and Domain for the "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ResetTcpDeviceConnectionMaster

ResetTcpDeviceConnectionMaster(SiteID,DeviceID)

Example:
ResetTcpDeviceConnectionMaster(3,1)

Resets the connection of the TCP Device [318] with the Device ID [3.1] of your Pandoras Box Master connection. In the status bar of PB and the Devices Tree, you will see that the device is dis- and reconnected.
The IP address and Domain for the "Master Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256].

ResourceAdd

ResourceAdd(FilePath,SiteID,FolderID,FileID)

Example:
ResourceAdd("C:\Christie\content\Video.mpg",3,4,2)

This adds a file from a remote or local node (i.e. Client or Master) to the main folder of your Pandoras Box project.
The file "Video.mpg" is loaded from the Site with ID 3 from the path "C:\Christie\content". It is added to the project folder and assigned with Folder and File ID 4,2. If you do not need a Folder/File ID you may set it to "0,0" in the command.

ResourceAddFileFromLocal

ResourceAddFileFromLocal(FilePath)

Example:
ResourceAddFileFromLocal("C:\Christie\content\Image.png")

This adds a file from your local node (i.e. the Master) to the main folder of your Pandoras Box project.
The file "Image.png" is loaded from the path "C:\Christie\content" and added to the project folder.

ResourceAddFileFromLocalToProjectPath

ResourceAddFileFromLocalToProjectPath(FilePath,ProjectPath)

Example:
ResourceAddFileFromLocalToProjectPath("C:\Christie\content\Image.png","background\images")

This adds a file from your local node (i.e. the Master) to a subfolder of your Pandoras Box project.
The file "Image.png" is loaded from the path "C:\Christie\content" and added to the subfolder "images" in the subfolder "background" within the project folder.

ResourceAddFolder

ResourceAddFolder(FolderName)

Example:
ResourceAddFolder("loops")

This adds a subfolder to your Pandoras Box project.
A new subfolder named "loops" is generated.

ResourceAddFolderFromLocal

ResourceAddFolderFromLocal(FilePath)

Example:
ResourceAddFolderFromLocal("C:\Christie\content\testpattern")

This adds a folder from your local node (i.e. the Master) to the main folder of your Pandoras Box project.
The folder "testpattern" including all files is loaded from the path "C:\Christie\content" and added to the project folder.

ResourceAddFolderFromLocalByID

ResourceAddFolderFromLocalByID(FolderPath,FolderID,FileID)

Example:
ResourceAddFolderFromLocalByID("C:\Christie\content\testpattern",5,1)

This adds a folder from your local node (i.e. the Master) to the main folder of your Pandoras Box project and assigns Folder/File IDs.
The folder "testpattern" including all files is loaded from the path "C:\Christie\content" and added to the project folder. In addition all files are numbered consecutively with a Folder/File ID starting from 5,1.

ResourceAddFolderFromLocalToProjectPath

ResourceAddFolderFromLocalToProjectPath(FilePath,ProjectPath)

Example:
ResourceAddFolderFromLocalToProjectPath("C:\Christie\content\testpattern","background\images")

This adds a folder from your local node (i.e. the Master) to a subfolder of your Pandoras Box project.
The folder "testpattern" including all files is loaded from the path "C:\Christie\content" and added to the subfolder "images" in the subfolder "background" within the project folder.

ResourceAddFolderFromLocalToProjectPathByID

ResourceAddFolderFromLocalToProjectPathByID(FolderPath,ProjectPath,FolderID,FileID)

Example:
ResourceAddFolderFromLocalToProjectPathByID("C:\Christie\content\testpattern","background\images",5,1)

This adds a folder from your local node (i.e. the Master) to a subfolder of your Pandoras Box project and assigns Folder/File IDs.
The folder "testpattern" including all files is loaded from the path "C:\Christie\content" and added to the subfolder "images" in the subfolder "background" within the project folder. In addition all files are numbered consecutively with a Folder/File ID starting from 5,1.

ResourceAddFolderToPath

ResourceAddFolderToPath(FolderName,ProjectPath)

Example:
ResourceAddFolderToPath("loops","background\images")

This adds a subfolder to an existing subfolder in your Pandoras Box project.
A new subfolder named "loops" is generated in the subfolder "images" in the subfolder "background" within the project folder.

ResourceAddImageSeq

ResourceAddImageSeq(FolderPath,FPS,SiteID,FolderID,FileID)

Example:
ResourceAddImageSeq("C:\ImgSeq\Intro",25,1,2,3)

This adds the image sequence saved under "C:\ImgSeq\Intro" on the hard disc of Site 1 to the main folder in the Pandoras Box Project tab. The frame rate of the image sequence is set to 25 frames/second and the File and Folder ID is [2,3]. If you do not need a Folder and File ID you may set it to "0,0" in the command.

ResourceAddImageSeqFromLocal

ResourceAddImageSeqFromLocal(FolderPath,FPS)

Example:
ResourceAddImageSeqFromLocal("C:\ImgSeq\Intro",25)

This adds the image sequence saved under "C:\ImgSeq\Intro" on the hard disc of the local Site to the main folder in the Pandoras Box Project tab. The frame rate of the image sequence is set to 25 frames/second.

ResourceAddImageSeqFromLocalByID

ResourceAddImageSeqFromLocalByID(FolderPath,FPS,FolderID,FileID)

Example:
ResourceAddImageSeqFromLocalByID("C:\ImgSeq\Intro",25,2,3)

This adds the image sequence saved under "C:\ImgSeq\Intro" on the hard disc of the local Site to the main folder in the Pandoras Box Project tab. The frame rate of the image sequence is set to 25 frames/second and the File and Folder ID is [2,3].

ResourceAddImageSeqFromLocalToProjectPath

ResourceAddImageSeqFromLocalToProjectPath(FolderPath,FPS,ProjectPath)

Example:
ResourceAddImageSeqFromLocalToProjectPath("C:\ImgSeq\Intro",25,"Image Sequences")

This adds the image sequence saved under "C:\ImgSeq\Intro" on the hard disc of the local Site to the folder "Image Sequences" in the Pandoras Box Project tab. The frame rate of the image sequence is set to 25 frames/second.

ResourceAddImageSeqFromLocalToProjectPathByID

ResourceAddImageSeqFromLocalToProjectPathByID(FolderPath,FPS,ProjectPath,FolderID,FileID)

Example:
ResourceAddImageSeqFromLocalToProjectPathByID("C:\ImgSeq\Intro",25,"Image Sequences",2,3)

This adds the image sequence saved under "C:\ImgSeq\Intro" on the hard disc of the local Site to the folder "Image Sequences" in the Pandoras Box Project tab. The frame rate of the image sequence is set to 25 frames/second and the File and Folder ID is [2,3].

ResourceAddImageSeqToProjectPath

ResourceAddImageSeqToProjectPath(FolderPath,FPS,SiteID,ProjectPath,FolderID,FileID)

Example:
ResourceAddImageSeqToProjectPath("C:\ImgSeq\Intro",25,1,"Image Sequences",2,3)

This adds the image sequence saved under "C:\ImgSeq\Intro" on the hard disc of Site 1 to the sub folder "Image Sequences" in the Pandoras Box Project tab. The frame rate of the image sequence is set to 25 frames/second and the File and Folder ID is [2,3]. If you do not need a Folder and File ID you may set it to "0,0" in the command.

ResourceAddToProjectPath

ResourceAddToProjectPath(FilePath,SiteID,FolderID,FileID,ProjectPath)

Example:
ResourceAddToProjectPath("C:\Christie\content\Video.mpg",3,4,2,background\images")

This adds a file from a remote or local node (i.e. Client or Master) to a subfolder of your Pandoras Box project. The file "Video.mpg" is loaded from the Site with ID 3 from the path "C:\Christie\content". It is added to the project folder and assigned with Folder and File ID 4,2. If you do not need a Folder/File ID you may set it to "0,0" in the command.

ResourceAttachByID

ResourceAttachByID(FolderID,FileID,SiteID,FilePath)

Example:
ResourceAttachByID(2,4,3,"C:\Christie\content\image2.png")

This attaches a file link in your Pandoras Box project so that the specified node loads another file than the one the Master system calls, e.g. as an active value or from the timeline. The node can be a remote or local node (i.e. Client or Master). See the topic "Attaching Files [195]".
In the example, the Site with ID 3 (e.g. a Server) loads the file "image2.png" from the path "C:\Christie\content" as soon as the Master system calls the file with Folder and File ID 2,4.

ResourceAttachByPath

ResourceAttachByPath(ProjectPath,SiteID,FilePath)

Example:
ResourceAttachByPath("backgrounds\images\image1.png",3,"C:\Christie\content\image2.png")

This attaches a file link in your Pandoras Box project so that the specified node loads another file than the one the Master system calls, e.g. as an active value or from the timeline. The node can be a remote or local node (i.e. Client or Master). See the topic "Attaching Files [195]".
In the example, the Site with ID 3 (e.g. a Server) loads the file "image2.png" from the path "C:\Christie\content" as soon as the Master system calls the file "image1.png" that is located in the subfolder "images" in the subfolder "background" within the project folder.

ResourceBrowserClearCacheByID

ResourceBrowserClearCacheByID(FolderID,FileID)

Example:
ResourceBrowserClearCacheByID(2,3)

This clears the browser cache from the Browser Asset with the Folder and File ID [2,3] in the Pandoras Box Project tab. Clear the web cache to renew stored document information including images. Alternatively, click "Clear Cache" in the Browser Inspector [198].

ResourceBrowserClearCacheByName

ResourceBrowserClearCacheByName(BrowserNamePath)

Example:
ResourceBrowserClearCacheByName("file:///C:\website\index.html")

This clears the browser cache from the Browser Asset "file:///C:\website\index.html" in the main folder in the Pandoras Box Project tab. Clear the web cache to renew stored document information including images. Alternatively, click "Clear Cache" in the Browser Inspector [198].

ResourceBrowserSetURLbyID

ResourceBrowserSetURLbyID(FolderID,FileID,URL)

Example:
ResourceBrowserSetURLbyID(2,4,"www.coolux.de")

In PB, the Browser Asset with Folder and File ID 2,4 changes its URL to "www.coolux.de".

ResourceBrowserSetURLbyName

ResourceBrowserSetURLbyName(BrowserNamePath,URL)

Example:
ResourceBrowserSetURLbyName("coolux.de","https://www.christiepandorasbox.com/")

In PB, the Browser Asset that is displayed with the name "coolux.de" in the Project tab, changes its URL to "https://www.christiepandorasbox.com/".

ResourceCanvasClearByID

ResourceCanvasClearByID(FolderID,FileID)

Example:
ResourceCanvasClearByID(2,4)

This clears the Canvas Asset [280] in your Pandoras Box project so that it is empty.

ResourceCreatePBPlaylistByID

ResourceCreatePBPlaylistByID(FolderID,FileID)

Example:
ResourceCreatePBPlaylistByID(1,2)

This creates a new empty Pandoras Box playlist [239] and assigns the folder and file ID [1,2].

ResourceCreatePBPlaylistByPath

ResourceCreatePBPlaylistByPath(ProjectPath)

Example:
ResourceCreatePBPlaylistByPath("Playlist Test\First")

This creates a new empty Pandoras Box playlist [239] and adds it to the subfolder "First" in the subfolder "Playlist Test" within the project folder.

ResourceCreatePBPlaylistByPathWithID

ResourceCreatePBPlaylistByPathWithID(ProjectPath,FolderID,FileID)

Example:
ResourceCreatePBPlaylistByPathWithID("Playlist Test\Second",4,8)

This creates an new empty Pandoras Box playlist [239]. It is added to the subfolder "Second" in the subfolder "Playlist Test" within the project folder and then assigned with the folder and file ID [4,8].

ResourceCreatePBPlaylistFromFolderByPath

ResourceCreatePBPlaylistFromFolderByPath(PlaylistPath,SourceFolderPath)

Example:
ResourceCreatePBPlaylistFromFolderByPath("Playlist Test\First","Content\Medialoops")

This creates a new Pandoras Box playlist [239] and adds it to the subfolder "First" in the subfolder "Playlist Test" within the project folder. The content from "Content\Medialoops" will be added to the playlist.

ResourceCreatePBPlaylistFromFolderByPathWithID

ResourceCreatePBPlaylistFromFolderByPathWithID(PlaylistPath,SourceFolderPath,FolderID,FileID)

Example:
ResourceCreatePBPlaylistFromFolderByPathWithID("Playlist Test\First","Content\Medialoops",3,5)

This creates a new Pandoras Box playlist [239] and adds it to the subfolder "First" in the subfolder "Playlist Test" within the project folder. The content from "Content\Medialoops" will be added to the playlist and the folder and file ID [3,5] is assigned.

ResourceCreateText

ResourceCreateText(FolderID,FileID,Text)

Example:
ResourceCreateText(2,10,"design2")

Creates a new text asset with the folder and file ID [2,10] in PB with the text "design2".

ResourceDetachByID

ResourceDetachByID(FolderID,FileID,SiteID)

Example:
ResourceDetachByID(2,4,3)

This detaches a file link in your Pandoras Box project so that the specified node does not load this file anymore when the Master system calls it, e.g. as an active value or from the timeline. It is now possible to attach another file that should be loaded instead. The node can be a remote or local node (i.e. Client or Master). See the topic "Attaching Files [195]".
In the example, the Site with ID 3 (e.g. a Server) does not load the file with Folder and File ID 2,4 any more.

ResourceDetachByPath

ResourceDetachByPath(ProjectPath,SiteID)

Example:
ResourceDetachByPath("backgrounds\images\image1.png",3)

This detaches a file link in your Pandoras Box project so that the specified node does not load this file anymore when the Master system calls it, e.g. as an active value or from the timeline. It is now possible to attach another file that should be loaded instead. The node can be a remote or local node (i.e. Client or Master). See the topic "Attaching Files [195]".
In the example, the Site with ID 3 (e.g. a Server) does not load the file "image1.png" any more. The file is located in the subfolder "images" in the subfolder "background" within the project folder.

ResourceEncodeFile

ResourceEncodeFile(FileName,Preset)

Example:
ResourceEncodeFile("Videos\loop.wmv","Auto")

This sends the file "loop.wmv" from the folder "Videos" in the Pandoras Box Project tab to the internal Encoder Extension [100] and encodes it with the Preset named "Auto". The encoded file is saved in the original path. Obviously, the Encoder Extension needs to be available in the PB Master.

ResourceEncodeFileByID

ResourceEncodeFileByID(FolderID,FileID,Preset)

Example:
ResourceEncodeFileByID(1,2,"Auto")

This sends the file with the File and Folder ID [1,2] in the Pandoras Box Project tab to the internal Encoder Extension [100] and encodes it with the Preset named "Auto". The encoded file is saved in the original path. Obviously, the Encoder Extension needs to be available in the PB Master.

ResourceEncodeFileByIDToPath

ResourceEncodeFileByIDToPath(FolderID,FileID,TargetPath,Preset)

Example:
ResourceEncodeFileByIDToPath(1,2,"C:\Encoded","Auto")

This sends the file with the File and Folder ID [1,2] in the Pandoras Box Project tab to the internal Encoder Extension [100] and encodes it with the Preset named "Auto". The encoded file is saved in a new path under "C:\Encoded". Obviously, the Encoder Extension needs to be available in the PB Master.

ResourceEncodeFileToPath

ResourceEncodeFileToPath(FileName,TargetPath,Preset)

Example:
ResourceEncodeFileToPath("Videos\loop.wmv","C:\Encoded","Auto")

This sends the file "loop.wmv" from the folder "Videos" in the Pandoras Box Project tab to the internal Encoder Extension [100] and encodes it with the Preset named "Auto". The encoded file is saved in a new path under "C:\Encoded". Obviously, the Encoder Extension needs to be available in the PB Master.

ResourceExportCanvasByID

ResourceExportCanvasByID(FolderID,FileID,FileName)

Example:
ResourceExportCanvasByID(1,2,"C:\Christie\content\Canvas_1.png")

This exports the Canvas resource with the Folder and File ID [1,2] as a PNG image including the alpha channel with the name "Canvas_1" to the existing path "C:\Christie\content".
Available image file formats are: PNG, JPEG, DDS, TIFF, Bitmap and DPX.

ResourceExportCanvasByName

ResourceExportCanvasByName(ProjectPath,FileName)

Example:
ResourceExportCanvasByName("Canvas\Tree.ccm","C:\Christie\content\Tree_1.png")

This exports the Canvas resource "Tree.ccm" from the folder Canvas as PNG image including Alpha channel with the name "Tree_1" to the existing path "C:\Christie\content".
Available image file formats are: PNG, JPEG, DDS, TIFF, Bitmap and DPX.

ResourceExportSeq

ResourceExportSeq(SeqID,FileName,Preset,StartHH,MM,SS,FF,StopHH,MM,SS,FF)

Example:
ResourceExportSeq(2,"C:\Christie\content\export\001","HD 1280x720",00,05,22,12,00,15,23,13)

This records what you have programmed on a sequence using the Video Export [135] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Sequence with ID 2 is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\export". "HD 1280x720" is used as the Encoder Setting. The recording starts at 00:05:22:12 and ends at 00:15:23:13 (hh:mm:ss:ff). Afterwards the recorded file is added to the Pandoras Box project automatically.

ResourceExportSeqF

ResourceExportSeqF(SeqID,FileName,Preset,StartHH:MM:SS:FF,StopHH:MM:SS:FF)

Example:
ResourceExportSeqF(2,"C:\Christie\content\export\001","HD 1280x720","00:05:22:12","00:15:23:13")

This records what you have programmed on a sequence using the Video Export [135] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Sequence with ID 2 is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\export". "HD 1280x720" is used as the Encoder Setting. The recording starts at 00:05:22:12 and ends at 00:15:23:13 (hh:mm:ss:ff). Afterwards the recorded file is added to the Pandoras Box project automatically.
With this command both timecodes can also be stored and recalled using string variables [1900].

ResourceGetTextFromLabel

ResourceGetTextFromLabel(LabelID,FolderID,FileID)

Example:
ResourceGetTextFromLabel(3,2,9)

Sets the text of the text asset with the Folder and File ID [2.9] in PB to the text of label 3 in WD.

ResourceGetTextFromLabelUnicode

ResourceGetTextFromLabelUnicode(LabelID,FolderID,FileID)

Example:
ResourceGetTextFromLabelUnicode(3,2,9)

Sets the text of the text asset with the Folder and File ID [2.9] in PB to the text of label 3 in WD.
Using this Unicode-command, text strings in any language format can be sent to Pandoras Box text assets.

ResourceGetTextFromTextbox

ResourceGetTextFromTextbox(TextboxID,FolderID,FileID)

Example:
ResourceGetTextFromTextbox(4,2,10)

Sets the text of the text asset with the Folder and File ID [2.10] in PB to the text of textbox 4 in WD.

ResourceGetTextFromTextboxUnicode

ResourceGetTextFromTextboxUnicode(TextboxID,FolderID,FileID)

Example:
ResourceGetTextFromTextboxUnicode(4,2,10)

Sets the text of the text asset with the Folder and File ID [2.10] in PB to the text of textbox 4 in WD.
Using this Unicode-command text strings in any language format can be sent to Pandoras Box text assets.

ResourceMove

ResourceMove(Content,ProjectPath)

Example:
ResourceMove("Image.png","background\images")

This moves the file "Image.png" that is already part of your Pandoras Box project to the subfolder "images" in the subfolder "background" within the project folder.
If there is more than one file in the project that holds this name, the first file is taken.

ResourcePBPlaylistAddItemByID

ResourcePBPlaylistAddItemByID(PlaylistFolderID,PlaylistFileID,ItemFolderID,ItemFileID)

Example:
ResourcePBPlaylistAddItemByID(4,5,1,8)

This adds the media file with the folder and file ID [1,8] to the end of the Pandoras Box playlist [239] with the folder and file ID [4,5].

ResourcePBPlaylistAddItemByPath

ResourcePBPlaylistAddItemByPath(PlaylistPath,ItemPath)

Example:
ResourcePBPlaylistAddItemByPath("Playlist Test\Playlist 1","Gradients\Calibrate.png")

This adds the media file "Calibrate.png" to the end of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder,

ResourcePBPlaylistInsertItemByID

ResourcePBPlaylistInsertItemByID(PlaylistFolderID,PlaylistFileID,ItemFolderID,ItemFileID,Index)

Example:
ResourcePBPlaylistInsertItemByID(4,5,1,8,3)

This inserts the media file with the folder and file ID [1,8] to the Pandoras Box playlist [239] with the folder and file ID [4,5] as index 3. All following items increase their indices by one, e.g. the former item number 3 becomes item number 4.

ResourcePBPlaylistInsertItemByPath

ResourcePBPlaylistInsertItemByPath(PlaylistPath,ItemPath,Index)

Example:
ResourcePBPlaylistInsertItemByPath("Playlist Test\Playlist 1","Gradients\Calibrate.png",3)

This inserts the media file "Calibrate.png" to the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder. The new media is assigned with index 3. All following items increase their indices by one, e.g. the item number 3 becomes item number 4.

ResourcePBPlaylistRemoveAllItemByID

ResourcePBPlaylistRemoveAllItemByID(FolderID,FileID)

Example:
ResourcePBPlaylistRemoveAllItemByID(1,9)

This removes the content from the Pandoras Box playlist [239] with the folder and file ID [1,9].

ResourcePBPlaylistRemoveAllItemByPath

ResourcePBPlaylistRemoveAllItemByPath(PlaylistPath)

Example:
ResourcePBPlaylistRemoveAllItemByPath("Playlist Test\Playlist 1")

This removes the content from the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder.

ResourcePBPlaylistRemoveItemByID

ResourcePBPlaylistRemoveItemByID(FolderID,FileID,RemoveID)

Example:
ResourcePBPlaylistRemoveItemByID(1,9,5)

This removes the item with index number 5 from the Pandoras Box playlist [239] with the folder and file ID [1,9]. All following items decrease their indices by one, e.g. item number 6 becomes item number 5.

ResourcePBPlaylistRemoveItemByPath

ResourcePBPlaylistRemoveItemByPath(PlaylistPath,RemoveID)

Example:
ResourcePBPlaylistRemoveItemByPath("Playlist Test\Playlist 1",5)

This removes the item with index number 5 from the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder. All following items decrease their indices by one, e.g. item number 6 becomes item number 5.

ResourcePBPlaylistSetItemDurationByID

ResourcePBPlaylistSetItemDurationByID(PlaylistFolderID,PlaylistFileID,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemDurationByID(2,1,8,00,05,22,12)

This sets the duration of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemDurationByIDF

ResourcePBPlaylistSetItemDurationByIDF(PlaylistFolderID,PlaylistFileID,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemDurationByIDF(2,1,8,"00:05:22:12")

This sets the duration of the item with index 8 of the Pandoras Box playlist with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemDurationByPath

ResourcePBPlaylistSetItemDurationByPath(PlaylistPath,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemDurationByPath("Playlist Test\Playlist 1",8,00,05,22,12)

This sets the duration of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemDurationByPathF

ResourcePBPlaylistSetItemDurationByPathF(PlaylistPath,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemDurationByPathF("Playlist Test\Playlist 1",8,"00:05:22:12")

This sets the duration of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemFadeOutByID

ResourcePBPlaylistSetItemFadeOutByID(PlaylistFolderID,PlaylistFileID,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemFadeOutByID(2,1,8,00,05,22,12)

This sets the fade out time of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemFadeOutByIDF

ResourcePBPlaylistSetItemFadeOutByIDF(PlaylistFolderID,PlaylistFileID,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemFadeOutByIDF(2,1,8,"00:05:22:12")

This sets the fade out time of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemFadeOutByPath

ResourcePBPlaylistSetItemFadeOutByPath(PlaylistPath,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemFadeOutByPath("Playlist Test\Playlist 1",8,00,05,22,12)

This sets the fade out time of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemFadeOutByPathF

ResourcePBPlaylistSetItemFadeOutByPathF(PlaylistPath,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemFadeOutByPathF("Playlist Test\Playlist 1",8,"00:05:22:12")

This sets the fade out time of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemIndexByID

ResourcePBPlaylistSetItemIndexByID(PlaylistFolderID,PlaylistFileID,OldIndex,NewIndex)

Example:
ResourcePBPlaylistSetItemIndexByID(2,1,7,4)

This sets the item with index 7 to index 4 of the Pandoras Box playlist [239] with the folder and file ID [2,1]. All original items 4-6 increase their indices by one, e.g. the item of index 4 moves forward to index 5.

ResourcePBPlaylistSetItemIndexByPath

ResourcePBPlaylistSetItemIndexByPath(PlaylistPath,OldIndex,NewIndex)

Example:
ResourcePBPlaylistSetItemIndexByPath("Playlist Test\Playlist 1",7,4)

This sets the item with index 7 to index 4 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder. All original items 4-6 increase their indices by one, e.g. the item of index 4 moves forward to index 5.

ResourcePBPlaylistSetItemInPointByID

ResourcePBPlaylistSetItemInPointByID(PlaylistFolderID,PlaylistFileID,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemInPointByID(2,1,8,00,05,22,12)

This sets the inpoint time of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemInPointByIDF

ResourcePBPlaylistSetItemInPointByIDF(PlaylistFolderID,PlaylistFileID,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemInPointByIDF(2,1,8,"00:05:22:12")

This sets the inpoint time of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemInPointByPath

ResourcePBPlaylistSetItemInPointByPath(PlaylistPath,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemInPointByPath("Playlist Test\Playlist 1",8,00,05,22,12)

This sets the inpoint time of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemInPointByPathF

ResourcePBPlaylistSetItemInPointByPathF(PlaylistPath,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemInPointByPathF("Playlist Test\Playlist 1",8,"00:05:22:12")

This sets the inpoint time of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemOutPointByID

ResourcePBPlaylistSetItemOutPointByID(PlaylistFolderID,PlaylistFileID,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemOutPointByID(2,1,8,00,05,22,12)

This sets the outpoint time of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemOutPointByIDF

ResourcePBPlaylistSetItemOutPointByIDF(PlaylistFolderID,PlaylistFileID,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemOutPointByIDF(2,1,8,"00:05:22:12")

This sets the outpoint time of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemOutPointByPath

ResourcePBPlaylistSetItemOutPointByPath(PlaylistPath,Index,HH,MM,SS,FF)

Example:
ResourcePBPlaylistSetItemOutPointByPath("Playlist Test\Playlist 1",8,00,05,22,12)

This sets the outpoint time of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff).

ResourcePBPlaylistSetItemOutPointByPathF

ResourcePBPlaylistSetItemOutPointByPathF(PlaylistPath,Index,HH:MM:SS:FF)

Example:
ResourcePBPlaylistSetItemOutPointByPathF("Playlist Test\Playlist 1",8,"00:05:22:12")

This sets the outpoint time of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

ResourcePBPlaylistSetItemTransitionFXByID

ResourcePBPlaylistSetItemTransitionFXByID(PlaylistFolderID,PlaylistFileID,Index,FXID)

Example:
ResourcePBPlaylistSetItemTransitionFXByID(2,1,8,1)

This sets the Transition FX of the item with index 8 of the Pandoras Box playlist [239] with the folder and file ID [2,1] to "TransitionFX 001". If you like to change it back to the default "Crossfade", use 0.

ResourcePBPlaylistSetItemTransitionFXByPath

ResourcePBPlaylistSetItemTransitionFXByPath(PlaylistPath,Index,FXID)

Example:
ResourcePBPlaylistSetItemTransitionFXByPath("Playlist Test\Playlist 1",8,1)

This sets the Transition FX of the item with index 8 of the Pandoras Box playlist [239] "Playlist 1" of the subfolder "Playlist Test" within the project folder to "TransitionFX 001". If you like to change it back to the default "Crossfade", use 0.

ResourceRecordLiveInput

ResourceRecordLiveInput(InputName,FileName,Preset,HH,MM,SS,FF)

Example:
ResourceRecordLiveInput("Integrated Webcam-2","C:\Christie\content\recordings\001","HD 1280x720",00,05,22,12)

This records a Live Input using the Video Recording [137] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Live Input with the name "Integrated Webcam-2" is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\recordings". "HD 1280x720" is used as the Encoder Setting. The total length of the file is 00:05:22:12 (hh:mm:ss:ff). Afterwards the recorded file is added to the Pandoras Box project automatically.

ResourceRecordLiveInputByID

ResourceRecordLiveInputByID(FolderID,FileID,FileName,Preset,HH,MM,SS,FF)

Example:
ResourceRecordLiveInputByID(2,4,"C:\Christie\content\recordings\001","HD 1280x720",00,05,22,12)

This records a Live Input using the Video Recording [137] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Live Input with the Folder and File ID 2,4 is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\recordings". "HD 1280x720" is used as the Encoder Setting. The total length of the file is 00:05:22:12 (hh:mm:ss:ff). Afterwards the recorded file is added to the Pandoras Box project automatically.

ResourceRecordLiveInputByIDF

ResourceRecordLiveInputByIDF(FolderID,FileID,FileName,Preset,HH:MM:SS:FF)

Example:
ResourceRecordLiveInputByIDF(2,4,"C:\Christie\content\recordings\001","HD 1280x720","00:05:22:12")

This records a Live Input using the Video Recording [137] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Live Input with the Folder and File ID 2,4 is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\recordings". "HD 1280x720" is used as the Encoder Setting. The total length of the file is 00:05:22:12 (hh:mm:ss:ff). Afterwards the recorded file is added to the Pandoras Box project automatically.
With this command the timecode can also be stored and recalled using a string variable [1900].

ResourceRecordLiveInputByIDStart

ResourceRecordLiveInputByIDStart(FolderID,FileID,FileName,Preset)

Example:
ResourceRecordLiveInputByIDStart(2,4,"C:\Christie\content\recordings\001","HD 1280x720",00,05,22,12)

This starts to record a Live Input using the Video Recording [137] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Live Input with the Folder and File ID 2,4 is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\recordings". "HD 1280x720" is used as the Encoder Setting. The total length of the file depends on when the recording is stopped again. Afterwards the recorded file is added to the Pandoras Box project automatically.

ResourceRecordLiveInputF

ResourceRecordLiveInputF(InputName,FileName,Preset,HH:MM:SS:FF)

Example:
ResourceRecordLiveInputF("Integrated Webcam-2","C:\Christie\content\recordings\001","HD 1280x720",00:05:22:12)

This records a Live Input using the Video Recording [137] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Live Input with the name "Integrated Webcam-2" is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\recordings". "HD 1280x720" is used as the Encoder Setting. The total length of the file is 00:05:22:12 (hh:mm:ss:ff). Afterwards the recorded file is added to the Pandoras Box project automatically.
With this command the timecode can also be stored and recalled using a string variable [1900].

ResourceRecordLiveInputStart

ResourceRecordLiveInputStart(InputName,FileName,Preset)

Example:
ResourceRecordLiveInputStart("Integrated Webcam-2","C:\Christie\content\recordings\001","HD 1280x720",00,05,22,12)

This starts to record a Live Input using the Video Recording [137] feature in the Pandoras Box project. The feature is limited to some Pandoras Box products [62].
The Live Input with the name "Integrated Webcam-2" is recorded and saved with the file name "001.m2v" under the path "C:\Christie\content\recordings". "HD 1280x720" is used as the Encoder Setting. The total length of the file depends on when the recording is stopped again. Afterwards the recorded file is added to the Pandoras Box project automatically.

ResourceRecordLiveInputStop

ResourceRecordLiveInputStop

Example:
ResourceRecordLiveInputStop

This stops to record a Live Input using the Video Recording [137] feature in the Pandoras Box project.

ResourceReloadMedia

ResourceReloadMedia(FolderID,FileID)

Example:
ResourceReloadMedia(2,5)

Reloads the media file with the Folder and File ID [2,5] in the Pandoras Box project. In PB, this option can be found in the File Inspector [191].

ResourceReloadMesh

ResourceReloadMesh(FolderID,FileID)

Example:
ResourceReloadMesh(3,9)

Reloads the mesh file with the Folder and File ID [3,9] in the Pandoras Box project. In PB, this option can be found in the File Inspector [191].

ResourceRemove

ResourceRemove(Name)

Example:
ResourceRemove("Image.png")

This removes the file "Image.png" from your Pandoras Box project. It does not matter whether the file is in the main folder or a subfolder but if there is more than one file in the project that holds this name, the first file is deleted.

ResourceRemoveAll

ResourceRemoveAll(RemoveFolderStructure)

Example:
ResourceRemoveAll(True)

This removes all files from your Pandoras Box project including the folder structure. When "False" is chosen, all files from the folders are deleted but the folders stay.

ResourceRemoveFolder

ResourceRemoveFolder(ProjectPath\)

Example:
ResourceRemoveFolder("Videos\Background")

This removes the sub folder "Background" from the folder "Videos" in the Pandoras Box Project tab.

ResourceRemoveInconsistent

ResourceRemoveInconsistent

Example:
ResourceRemoveInconsistent

Removes all inconsistent files from the Pandoras Box Master system. In PB, this option can be found in the Folder Inspector [197].

ResourceRemoveMedia

ResourceRemoveMedia(FolderID,FileID)

Example:
ResourceRemoveMedia(2,5)

Removes the media file with the Folder and File ID [2,5] from the PB project.

ResourceRemoveMesh

ResourceRemoveMesh(FolderID,FileID)

Example:
ResourceRemoveMesh(3,9)

Removes the mesh file with the Folder and File ID [3,9] from the Pandoras Box Project Tab [278].

ResourceResize

ResourceResize(FilePath,Width,Height)

Example:
ResourceResize("C:\Christie\content\image2.png",256,256)

This command does not necessarily apply to a Pandoras Box project. It simply resizes an image (here: the file named "image2.png" saved under the path "C:\Christie\content") on the hard drive and saves it under the same name but with the different size. The image from the example would have a new height and width of 256 pixels. If the image is part of a Pandoras Box project it then needs to be reloaded to read-out the new size. The reload happens automatically with the default options and a Revision 9088 or higher. When using an older revision, please reload the file manually, e.g. by using the File Inspector [191] or reloading the project.

ResourceSetAlphaChannel

ResourceSetAlphaChannel(FolderID,FileID,State)(True or False)

Example:
ResourceSetAlphaChannel(2,5,False)

Disables the alpha channel of the media file with the Folder and File ID [2,5] in the File Inspector [191] in Pandoras Box.

Use "True" if movie type supports embedded alpha.

ResourceSetAnisotropicFiltering

ResourceSetAnisotropicFiltering(FolderID,FileID,State)(True or False)

Example:
ResourceSetAnisotropicFiltering(2,5,False)

Deactivates anisotropic filtering (pixel smoothing for real-time video and image scaling) for the media file with the Folder and File ID [2,5] in the File Inspector [191] in Pandoras Box.

ResourceSetDeinterlacing

ResourceSetDeinterlacing(FolderID,FileID,State)(0,1,2,3)

Example:
ResourceSetDeinterlacing(2,5,3)

Activates the deinterlacing mode "fieldblending" for the media file with the Folder and File ID [2,5] in the File Inspector [191] in Pandoras Box.

Deinterlacing Modes:
0 = Off
1 = Top Field
2 = Bottom Field
3 = Fieldblending

ResourceSetFrameBlending

ResourceSetFrameBlending(FolderID,FileID,State)(True or False)

Example:
ResourceSetFrameBlending(2,5,True)

Activates frameblending for the media file with the Folder and File ID [2,5] in the File Inspector [191] in Pandoras Box. Choose this option for slow motion or smooth motion frame-adaptive frameblending and framerate conversion.

ResourceSetID

ResourceSetID(ProjectPath,FolderID,FileID)

Example:
ResourceSetID("backgrounds\images",5,1)

This applies to the subfolder "images" in the subfolder "background" within the Pandoras Box project folder. All files included in the subfolder are assigned consecutively with a Folder and File ID starting from 5,1.

If you like to a assign a Folder/File ID to a single file instead of an entire folder, specify "ProjectPath" as "backgrounds\images\image1.png".

ResourceSetMpegColorSpace

ResourceSetMpegColorSpace(FolderID,FileID,State)(True or False)

Example:
ResourceSetMpegColorSpace(2,5,True)

Enables the optimization of MPEG Colorspace for the media file with the Folder and File ID [2,5] in the File Inspector [191] in Pandoras Box.

ResourceSetText

ResourceSetText(FolderID,FileID,Text)

Example:
ResourceSetText(2,9,"design")

Sets the text of text asset [2,9] in PB to "design".

ResourceSetTextCentered

ResourceSetTextCentered(FolderID,FileID)

Example:
ResourceSetTextCentered(1,2)

Centers the text of text asset [1,2] in PB on its texture.
This command is available from PB Rev. 4982 or higher.

ResourceSetTextFullSize

ResourceSetTextFullSize(FolderID,FileID)

Example:
ResourceSetTextFullSize(1,2)

Removes the center-option of the text asset [1,2] in PB.
This command is available from PB Rev. 4982 or higher.

ResourceSetTextSize

ResourceSetTextSize(FolderID,FileID,Width,Height)

Example:
ResourceSetTextSize(1,2,800,600)

Sets the texture size of text asset [1,2] in PB to 800x600 px.
This command is available from PB Rev. 4982 or higher.

ResourceSetTextStyle

ResourceSetTextStyle(FolderID,FileID,Font,Size,Style,Alignment,Red,Green,Blue)

Example:
ResourceSetTextStyle(1,2,"Times New Roman",24,"Bold","Left",255,0,0)

Sets the Text Asset [1,2] to the following properties:
Font: Time New Roman, Font Size: 24, Font Style: Bold, Alignment: Left, Color: Red.

This command is available from PB Rev. 4982 or higher.

Style options: Regular, Bold, Italic, BoldItalic, Underline.
Alignment options: Left, Center, Right.

ResourceSetTextUnicode

ResourceSetTextUnicode(FolderID,FileID,Text)

Example:
ResourceSetTextUnicode(1,2,"?a?? µ??a")

Sets the text of text asset [1,2] in PB to "?a?? µ??a".
Using this unicode-command, text strings in any language format can be sent to PB.

ResourceSetUnderscan

ResourceSetUnderscan(FolderID,FileID,State)(True or False)

Example:
ResourceSetUnderscan(2,5,True)

Enables Underscan for the media file with the Folder and File ID [2,5] in the File Inspector [191] in Pandoras Box.

ResourceSpreadAll

ResourceSpreadAll

Example:
ResourceSpreadAll

Spreads all resources in PB.

ResourceSpreadMedia

ResourceSpreadMedia(FolderID,FileID)

Example:
ResourceSpreadMedia(2,10)

Spreads the media file with the Folder and File ID [2,10] in PB.

ResourceSpreadMesh

ResourceSpreadMesh(FolderID,FileID)

Example:
ResourceSpreadMesh(3,7)

Spreads the mesh file with the Folder and File ID [3,7] in PB.

ResourceSpreadToSite

ResourceSpreadToSite(ProjectPath,SiteID)

Example:
ResourceSpreadToSite("Images\HD",2)

Spreads all resources which are located under the PB project path "Images\HD" to the Site with ID 2.

ResourceSpreadToSiteById

ResourceSpreadToSiteById(FolderID,FileID,SiteID)

Example:
ResourceSpreadToSiteById(2,1,4)

Spreads the resource with the Folder and File ID [2,1] to the Site 4.

S S

ScriptCancel

ScriptCancel(Filter(optional))

Example:
ScriptCancel("Page1")

The first example shows, how to cancel (i.e. stop) all scripts that are currently running and which were started from a page in WD called "Page1". Likewise can you cancel all scripts from a WD window, e.g. "Window1".

In general, the command can be used in many ways as you see in the other examples. The "Running Scripts [1511]" dialog (from the Scripting menu) might be helpful as it shows which scripts are currently running and how their so called context is, meaning what started them, e.g. a Custom Script Button [822] on a certain page, a Macro or Function [1897], a Script Output node [1207], etc.
If you are looking for a solution to cancel a script by itself, use the "Exit" statement as explained in the "Programming Statements [1886]" > "Exit [1892]".

Example2:
ScriptCancel("FaderM")

The second example cancels a specific script, in this case the Macro with the name "FaderM", no matter how many times it was started. However, if you are using this example inside the Macro itself (in this case "FaderM") it would cancel the instance that was started before but not the current one. Keep in mind: a script cannot cancel itself with this command, just other ones.
If you like to cancel a script that belongs to a Custom Script Button, use a filter like: "Window1/Page1/CustomScript9 [ClickScript]". The filter is displayed in the "Running Scripts" dialog or can be checked with the member "`Context.Name`" first.

Example3:
Label1.Text = ScriptCancel("FaderM")

Following the second example, here, Label1 displays how many scripts where canceled. You could also use the Debug Logger [812] instead: `DebugMessage(ScriptCancel("FaderM"))`.

Example4:
ScriptCancel()

If no filter is specified, the current context is applied automatically. So if this command is executed from a Macro, the context is the Macro name but if it is executed from a Custom Script Button, it would refer to the

button's script name. If you write "ScriptCancel()" at the beginning of a script and execute this script twice, the second call would cancel the first one because it is a previous instance of itself and then start the script again. As already explained, with this command a script cancels other instances but not itself.

ScriptCancelAll

ScriptCancelAll

Example:
ScriptCancelAll

Cancels all running scripts, no matter from which Macros, Function [1897], Custom Script Button or other script field.

ScriptError

ScriptError(ErrorMessage)

Example:
ScriptError("The value you"ve entered is not valid, please enter a Double or Integer value between 10 and 80.")

Displays your own error messages like "The value you've entered is not valid, please enter a Double or Integer value between 10 and 80." in the Debug Logger [812]. This helps to find errors in your programmed scripts.

This feature is currently not available for the WD Free Version.

SeqAddJumpCue

SeqAddJumpCue(SeqID,CueID,HH,MM,SS,FF,Name)

Example:
SeqAddJumpCue(1,5,00,02,22,12,"Scene2")

A Jump Cue with the ID 5 will be added to Sequence 1 at the timecode 00:02:22:12. It will be named "Scene2".

SeqAddJumpCueF

SeqAddJumpCueF(SeqID,CueID,HH:MM:SS:FF,Name)

Example:
SeqAddJumpCueF(1,6,"00:01:20:15","Scene2")

A Jump Cue with the ID 6 will be added to Sequence 1 at the timecode 00:01:20:15. It will be named "Scene2". With this command, the timecode can also be stored and recalled using a string variable [1900].

SeqAddPauseCue

SeqAddPauseCue(SeqID,CueID,HH,MM,SS,FF,Name)

Example:
SeqAddPauseCue(1,2,00,01,22,15,"Scene4")

A Pause Cue with the ID 2 will be added in Sequence 1 at the timecode 00:01:22:15. It will be named "Scene4.

SeqAddPauseCueF

SeqAddPauseCueF(SeqID,CueID,HH:MM:SS:FF,Name)

Example:
SeqAddPauseCueF(1,2,"00:01:22:15","Scene4")

A Pause Cue with the ID 2 will be added in Sequence 1 at the timecode 00:01:22:15. It will be named "Scene4. With this command, the timecode can also be stored and recalled using a string variable [1900].

SeqAddPlayCue

SeqAddPlayCue(SeqID,CueID,HH,MM,SS,FF,Name)

Example:
SeqAddPlayCue(1,5,00,01,22,15,"Scene3")

A Play Cue with the ID 5 will be added in Sequence 1 at the timecode 00:01:22:15. It will be named "Scene3".

SeqAddPlayCueF

SeqAddPlayCueF(SeqID,CueID,HH:MM:SS:FF,Name)

Example:
SeqAddPlayCueF(1,5,"00:01:22:15","Scene3")

A Play Cue with the ID 5 will be added in Sequence 1 at the timecode 00:01:22:15. It will be named "Scene3". With this command, the timecode can also be stored and recalled using a string variable [1900].

SeqAddWaitCue

SeqAddWaitCue(SeqID,CueID,HH,MM,SS,FF,Name)

Example:
SeqAddWaitCue(2,7,00,01,30,10,"Pause1")

A Wait Cue with the ID 7 will be added to Sequence 2 at the timecode 00:01:30:10. It will be named "Pause1".

SeqAddWaitCueF

SeqAddWaitCueF(SeqID,CueID,HH:MM:SS:FF,Name)

Example:
SeqAddWaitCueF(1,4,"00:05:11:13","Pause2")

A Wait Cue with the ID 4 will be added to Sequence 1 at the timecode 00:05:11:13. It will be named "Pause2". With this command, the timecode can also be stored and recalled using a string variable [1900].

SeqContentAtTime

SeqContentAtTime(SiteID,DevID,SeqID,HH,MM,SS,FF,FolderID,FileID)

Example:
SeqContentAtTime(1,3,1,00,05,22,12,1,1)

If a clip container exists at timecode 00:05:22:12 in Sequence 1, the content inside this clip container will get exchanged by the content with the Folder and File ID 1,1 and it also will be re-synced.

SeqContentAtTimeF

SeqContentAtTimeF(SiteID,DevID,SeqID,HH:MM:SS:FF,FolderID,FileID)

Example:
SeqContentAtTimeF(1,3,1,"00:05:22:12",1,1)

If a clip container exists at timecode 00:05:22:12 in Sequence 1, the content inside this clip container will get exchanged by the content with the Folder and File ID 1,1 and it also will be re-synced.
The timecode can also be stored and recalled using a string variable [1900].

SeqContentReplaceAtTime

SeqContentReplaceAtTime(Site,Device,SeqID,HH,MM,SS,FF,FolderID,FileID)

Example:
SeqContentReplaceAtTime(1,3,2,00,05,22,12,4,6)

This applies to layer 3 of Site 1 (e.g. a Server) in your Pandoras Box project. The container in Sequence 2 at timecode 00:05:22:12 (hh,mm,ss,ff) is filled with the media file that holds the Folder and File ID 4,6. Please note, that this command requires an existing container whereto it can drop the media file.

SeqContentReplaceAtTimeByPath

SeqContentReplaceAtTimeByPath(Site,Device,SeqID,HH,MM,SS,FF,ProjectPath)

Example:
SeqContentReplaceAtTimeByPath(2,1,1,00,05,22,12,"Testpattern/Calibrate.png")

This applies to layer 3 of Site 1 (e.g. a Server) in your Pandoras Box project. The container in Sequence 2 at timecode 00:05:22:12 (hh,mm,ss,ff) is filled with the media file "Calibrate.png" from the subfolder "Testpattern" within the project folder.

If there is more than one file in the subfolder that holds this name, the first file is taken. If the file is not in a special subfolder, simply use "Calibrate.png". Please note that "ProjectPath" is case-sensitive and that this command requires an existing container whereto it can drop the media file.

SeqContentReplaceAtTimeByPathF

SeqContentReplaceAtTimeByPathF(Site,Device,SeqID,HH:MM:SS:FF,ProjectPath)

Example:
SeqContentReplaceAtTimeByPathF(2,1,1,"00:05:22:12","Testpattern/Calibrate.png")

This applies to layer 3 of Site 1 (e.g. a Server) in your Pandoras Box project. The container in Sequence 2 at timecode 00:05:22:12 (hh:mm:ss:ff) is filled with the media file "Calibrate.png" from the subfolder "Testpattern" within the project folder.

If there is more than one file in the subfolder that holds this name, the first file is taken. If the file is not in a special subfolder, simply use "Calibrate.png". Please note that "ProjectPath" is case-sensitive and that this command requires an existing container whereto it can drop the media file.
The timecode can also be stored and recalled using a string variable [1900].

SeqContentReplaceAtTimeF

SeqContentReplaceAtTimeF(Site,Device,SeqID,HH:MM:SS:FF,FolderID,FileID)

Example:
SeqContentReplaceAtTimeF(1,3,2,"00:05:22:12",4,6)

This applies to layer 3 of Site 1 (e.g. a Server) in your Pandoras Box project. The container in Sequence 2 at timecode 00:05:22:12 (hh:mm:ss:ff) is filled with the media file that holds the Folder and File ID 4,6. Please note, that this command requires an existing container whereto it can drop the media file.
The timecode can also be stored and recalled using a string variable[1900].

SeqDeleteAllCues

SeqDeleteAllCues(SeqID)

Example:
SeqDeleteAllCues(1)

All Cues in Sequence 1 will be deleted.

SeqDeleteCue

SeqDeleteCue(SeqID,CueID)

Example:
SeqDeleteCue(1,4)

Cue 4 in Sequence 1 will be deleted.

SeqEdit

SeqEdit(ID)

Example:
SeqEdit(2)

This command toggles Sequence with ID 2 into the Sequence tab of Pandoras Box so that it is visible and editable.

SeqGotoCue

SeqGotoCue(SeqID,CueID)

Example:
SeqGotoCue(2,3)

Sets the nowpointer of sequence 2 to the third cue.

SeqLastCue

SeqLastCue(SeqID)

Example:
SeqLastCue(1)

Sets the nowpointer of sequence 1 to the last cue.

SeqLastFrame

SeqLastFrame(SeqID)

Example:
SeqLastFrame(2)

Sets the nowpointer of sequence 2 to the last frame.

SeqNextCue

SeqNextCue(SeqID)

Example:
SeqNextCue(1)

Sets the nowpointer of sequence 1 to the next cue.

SeqNextFrame

SeqNextFrame(SeqID)

Example:
SeqNextFrame(2)

Sets the nowpointer of sequence 2 to the next frame ahead.

SeqResetJumps

SeqResetJumps(SeqID,CueID)

Example:
SeqResetJumps(2,3)

This applies to Sequence 2 in Pandoras Box. The "Repeat Count" of the Cue with ID 3 is reset to the value it was programmed with. In PB, this can be done in the Cue Inspector [208].

SeqSetCueJumpCount

SeqSetCueJumpCount(SeqID,CueID,Count)

Example:
SeqSetCueJumpCount(2,3,5)

This applies to Sequence 2 in Pandoras Box. The "Repeat Count" of the Cue with ID 3 is set to 5. In PB, this can be done in the Cue Inspector [208].

SeqSetCueJumpTargetTime

SeqSetCueJumpTargetTime(SeqID,CueID,HH,MM,SS,FF)

Example:
SeqSetCueJumpTargetTime(2,3,00,05,22,12)

This applies to Sequence 2 in Pandoras Box. The "Jump Target" of the Cue with ID 3 is set to 00:05:22:12 (hh:mm:ss:ff). Please note that the Cue must be a Jump Cue. In PB, this can be done in the Cue Inspector [208].

SeqSetCueJumpTargetTimeF

SeqSetCueJumpTargetTimeF(SeqID,CueID,HH:MM:SS:FF)

Example:
SeqSetCueJumpTargetTimeF(2,3,"00:05:22:12")

This applies to Sequence 2 in Pandoras Box. The "Jump Target" of the Cue with ID 3 is set to 00:05:22:12 (hh:mm:ss:ff). In PB, this can be done in the Cue Inspector [208].
The timecode can also be stored and recalled using a string variable [1900]. Please note that the Cue must be a Jump Cue.

SeqSetCueName

SeqSetCueName(SeqID,CueID,Name1)

Example:
SeqSetCueName(1,3,"Scene4")

This renames the Cue 3 of Sequence 1 in Pandoras Box to "Scene4". In PB, this can be done in the Cue Inspector [208].

SeqSetCuePlayMode

SeqSetCuePlayMode(SeqID,CueID,Play/Pause/Stop/Jump/Wait)

Example:
SeqSetCuePlayMode(1,17,"Pause")

Sets the play mode of cue 17 in Sequence 1 to "Pause".
The following cue play modes are available: Play, Pause, Stop, Jump and Wait.

SeqSetCueWaitTime

SeqSetCueWaitTime(SeqID,CueID,HH,MM,SS,FF)

Example:
SeqSetCueWaitTime(2,3,00,05,22,12)

This applies to Sequence 2 in Pandoras Box. The "Wait Time" of the Cue with ID 3 is set to 00:05:22:12 (hh:mm:ss:ff). In PB, this can be done in the Cue Inspector [208].

SeqSetCueWaitTimeF

SeqSetCueWaitTimeF(SeqID,CueID,HH:MM:SS:FF)

Example:
SeqSetCueWaitTimeF(2,3,"00:05:22:12")

This applies to Sequence 2 in Pandoras Box. The "Wait Time" of the Cue with ID 3 is set to 00:05:22:12 (hh:mm:ss:ff). The timecode can also be stored and recalled using a string variable [1900]. In PB, this can be done in the Cue Inspector [208].

SeqSetDoNotIgnoreNextCue

SeqSetDoNotIgnoreNextCue(SeqID)

Example:
SeqSetDoNotIgnoreNextCue(1)

The next cue in Sequence 1 will not be ignored .

SeqSetIgnoreNextCue

SeqSetIgnoreNextCue(SeqID)

Example:
SeqSetIgnoreNextCue(1)

The next cue in Sequence 1 will be ignored (cue turns orange).

SeqSetNextCuePlayMode

SeqSetNextCuePlayMode(SeqID,Play/Pause/Stop/Jump/Wait)

Example:
SeqSetNextCuePlayMode(1,"Pause")

Sets the play mode of the next cue in Sequence 1 to "Pause".
The following cue play modes are available: Play, Pause, Stop, Jump and Wait.

SeqSetState

SeqSetState(SeqID,"Play"/"Pause"/"Stop")

Example:
SeqSetState(2,"Play")

Sets sequence 2 to Play.

Example 2:
SeqSetState(2,"Pause")
Sets sequence 2 to Pause.

Example 3:
SeqSetState(2,"Stop")
Sets sequence 2 to Stop.

SeqSetTimecode

SeqSetTimecode(SeqID,HH,MM,SS,FF)

Example:
SeqSetTimecode(2,00,05,22,12)

Sets sequence 2 within the Pandoras Box project to the timecode 00:05:22:12 (hh:mm:ss:ff)

SeqSetTimecodeF

SeqSetTimecodeF(SeqID,HH:MM:SS:FF)

Example:
SeqSetTimecodeF(2,"00:05:22:12")

Sets sequence 2 within the Pandoras Box project to the timecode 00:05:22:12 (hh:mm:ss:ff). With this command the timecode can also be stored and recalled using a string variable [1900].

SeqSetTimecodeMode

SeqSetTimecodeMode(SeqID,"None"/"Send"/"Receive")

Example:
SeqSetTimecodeMode(2,"None")

Sets the timecode mode of sequence 2 in Pandoras Box to "None". In PB, this can be done in the Sequence Inspector [204].

Example 2:
SeqSetTimecodeMode(2,"Send")
Sets the timecode mode of sequence 2 to "Send".

Example 3:
SeqSetTimecodeMode(2,"Receive")
Sets the timecode mode of sequence 2 in Pandoras Box to "Receive". In PB, this can be done in the Sequence Inspector [204].

SeqSetTimecodeOffset

SeqSetTimecodeOffset(SeqID,HH,MM,SS,FF)

Example:
SeqSetTimecodeOffset(2,00,05,22,12)

Sets an offset of 00:05:22:12 (hh:mm:ss:ff) to the timecode of sequence 2.

SeqSetTimecodeOffsetF

SeqSetTimecodeOffsetF(SeqID,HH:MM:SS:FF)

Example:
SeqSetTimecodeOffsetF(2,"00:05:22:12")

Sets an offset of 00:05:22:12 (hh:mm:ss:ff) to the timecode of sequence 2. The timecode can also be stored and recalled using a string variable [1900].

SeqSetTimecodeRelative

SeqSetTimecodeRelative(SeqID,HH,MM,SS,FF)

Example:
SeqSetTimecodeRelative(2,00,05,22,12)

Adds an time interval of 00:05:22:12 (hh:mm:ss:ff) to the timecode of sequence 2 within the Pandoras Box project. If the timecode was at 00:01:02:03, it is now at 00:06:24:15.

Example2:
SeqSetTimecodeRelative(2,00,-05,-22,-12)

The second example subtracts the same time interval again. The command is the same but in front of each time value there is a minus (-) symbol.

SeqSetTimecodeRelativeF

SeqSetTimecodeRelativeF(SeqID,HH:MM:SS:FF)

Example:
SeqSetTimecodeRelativeF(2,"00:05:22:12")

Adds an time interval of 00:05:22:12 (hh:mm:ss:ff) to the timecode of sequence 2 within the Pandoras Box project. If the timecode was at 00:01:02:03, it is now at 00:06:24:15.

Example2:
SeqSetTimecodeRelativeF(2,varstring)

Instead of typing the timecode, you could also use a string variable [1900] if it has the correct format of course.

Example3:
SeqSetTimecodeRelativeF(2,"00:-05:-22:-12")

The third example subtracts the same time interval as in the first example. The command is the same but in front of each time value there is a minus (-) symbol.

SeqSetTimecodeStopAction

SeqSetTimecodeStopAction(SeqID,"Stop"/"Pause"/"Continue")

Example:
SeqSetTimecodeStopAction(1,"Stop")

Stops sequence 1 in Pandoras Box if the incoming timecode stops. In PB, this can be done in the Sequence Inspector [204].

Example 2:
SeqSetTimecodeStopAction(1,"Pause")
Pauses sequence 1 in Pandoras Box if the incoming timecode stops. In PB, this can be done in the Sequence Inspector [204].

Example 3:
SeqSetTimecodeStopAction(1,"Continue")
Sequence 1 in Pandoras Box will continue playing if the incoming timecode stops. In PB, this can be done in the Sequence Inspector [204].

SeqStoreActive

SeqStoreActive(SeqID)

Example:
SeqStoreActive(2)

Stores all active parameters in Pandoras Box to sequence 2.

SeqStoreActiveDevice

SeqStoreActiveDevice(SeqID,SiteID,DeviceID)

Example:
SeqStoreActiveDevice(1,2,4)

Stores all active parameters of layer 4 of site 2 in Pandoras Box to sequence 1 to the position of the nowpointer.

SeqStoreActiveParameter

SeqStoreActiveParameter(SeqID,SiteID,DeviceID,Parameter)

Example:
SeqStoreActiveParameter(1,2,5,"Opacity")

Stores all active "Opacity" parameters of layer 4 of site 2 in Pandoras Box to sequence 1 to the position of the nowpointer.

SeqStoreActiveSite

SeqStoreActiveSite(SeqID,SiteID)

Example:
SeqStoreActiveSite(2,1)

Stores all active parameters of site 1 in Pandoras Box to sequence 2 to the position of the nowpointer.

SeqStoreActiveToTime

SeqStoreActiveToTime(SeqID,HH,MM,SS,FF)

Example:
SeqStoreActiveToTime(2,00,05,22,12)

Stores all active parameters in Pandoras Box to sequence 2 at timecode 00:05:22:12 (hh,mm,ss,ff).

SeqStoreActiveToTimeF

SeqStoreActiveToTimeF(SeqID,HH:MM:SS:FF)

Example:
SeqStoreActiveToTimeF(2,"00:05:22:12")

Stores all active parameters in Pandoras Box to sequence 2 at timecode 00:05:22:12 (hh,mm,ss,ff). The timecode can also be stored and recalled using a string variable [1900].

SetAudioMaster

SetAudioMaster(SiteID)

Example:
SetAudioMaster(2)

Ticks the checkbox "Use Audio Clock As Master" for Site 2 in the Pandoras Box Configuration tab under ASIO Audio [166].

SetExcelCellValue

SetExcelCellValue(File,Table,Cell,Value)

Example:
SetExcelCellValue("C:\Data\excel.xls","Sheet1","C7",42)

Sets the value of Excel document "excel.xls", table "Sheet1", cell C7 to 42.

Note: The document has to be an XLS file and has to be closed during function call. If it is opened, a Windows dialog will open and propose to save a copy of the file.

SetPBPlaylistAllItemsDurationByID

SetPBPlaylistAllItemsDurationByID(Folder,File,HH,MM,SS,FF)

Example:
SetPBPlaylistAllItemsDurationByID(2,1,00,05,22,12)

This sets the duration of all items of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff).

SetPBPlaylistAllItemsDurationByIDF

SetPBPlaylistAllItemsDurationByIDF(Folder,File,HH:MM:SS:FF)

Example:
SetPBPlaylistAllItemsDurationByIDF(2,1,"00:05:22:12")

This sets the duration of all items of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff). With this command, the timecode can also be stored and recalled using a string variable [1900].

SetPBPlaylistItemDurationByID

SetPBPlaylistItemDurationByID(Folder,File,ItemID,HH,MM,SS,FF)

Example:
SetPBPlaylistItemDurationByID(2,1,8,00,05,22,12)

This sets the duration of the item with index 7(!) of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff). Note that ItemID 0 refers to the PB index 1!

SetPBPlaylistItemDurationByIDF

SetPBPlaylistItemDurationByIDF(Folder,File,ItemID,HH:MM:SS:FF)

Example:
SetPBPlaylistItemDurationByIDF(2,1,8,"00:05:22:12")

This sets the duration of the item with index 7(!) of the Pandoras Box playlist [239] with the folder and file ID [2,1] to 00:05:22:12 (hh:mm:ss:ff). Note that ItemID 0 refers to the PB index 1!
With this command, the timecode can also be stored and recalled using a string variable [1900].

ShareLayerTexture

ShareLayerTexture(SiteID,SourceDevice,TargetDevice)

Example:
ShareLayerTexture(3,1,2)

This shares the texture from Layer 1 of Site 3 with Layer 2. Note that Texture Sharing is only available for PB Servers.

ShareLayerTextureByName

ShareLayerTextureByName(SiteID,SourceDevice,TargetDevice,TargetParam)

Example:
ShareLayerTextureByName(3,1,2,"Quad Media Overlay|Media1")

This shares the texture from Layer 1 of Site 3 with Layer 2, but not as the main media but the first media of the effect named "Quad Media Overlay". Note that Texture Sharing is only available for PB Servers.

Please note that "TargetParam" is case-sensitive and consists of the name of the effect [327] followed by the character pipe (vertical bar) and the name of the media field (in most cases simply "Media").

ShowEnvironmentVariables

ShowEnvironmentVariables

Example:
ShowEnvironmentVariables

Opens a dialog showing the current Windows environment variables, to ease remote debugging for example.

More information can be found under:
https://en.wikipedia.org/wiki/Environment_variable
http://www.7tutorials.com/simple-questions-what-are-environment-variables

SiteAcceptDmxById

SiteAcceptDmxById(SiteID,"On"/"Off")

Example:
SiteAcceptDmxById(2,"On")

Patches Site 2 in the Patch tab [228] so that it can be controlled via an attached DMX / Art-Net device.

Example 2:
SiteAcceptDmxById(2,"Off")

Unpatches Site 2 in the Patch tab [228] so that it cannot be controlled via an attached DMX / Art-Net device anymore.

Please note:
Art-Net Input needs to be enabled in Configuration Tab [139] in order to remote control the Site via Art-Net.

To change the Devices DMX / Art-Net start address (Channel, Art-Net Subnet and Universe) use either the Patch Tab in PB or the WD command DeviceSetDmxAddress [1534].

SMPTELinkModeNone

SMPTELinkModeNone

Example:
SMPTELinkModeNone

Sets the SMPTE Link Mode in the Connection Manager [1258] to "None".

SMPTELinkModeReceive

SMPTELinkModeReceive

Example:
SMPTELinkModeReceive

Sets the SMPTE Link Mode in the Connection Manager [1258] to "Receive".

SMPTELinkReConnect

SMPTELinkReConnect

Example:
SMPTELinkReConnect

This re-connects your SMPTE Link connection in the Connection Manager [1258].

SMSClearList

SMSClearList

Example:
SMSClearList

Clears the SMS list within the SMS Settings Tool [1494].

SMSSend

SMSSend(PhoneNumber,Message)

Example:
SMSSend(00492211306540,"Test")

Sends an SMS with the message "Test" to the phone number "00492211306540".

Setup your GSM modem in the SMS Settings Tool [1494] first to be able to receive and send SMS via the Widget Designer.

SMSSendDisable

SMSSendDisable

Example:
SMSSendDisable

When this command is executed, no SMS will be send out. Use the script SMSSendEnable to enable this function again.

Setup your GSM modem in the SMS Settings Tool [1494] first to be able to receive and send SMS via the Widget Designer.

SMSSendEnable

SMSSendEnable

Example:
SMSSendEnable

Enables the SMSSend function after it was disabled.

Setup your GSM modem in the SMS Settings Tool [1494] first to be able to receive and send SMS via the Widget Designer.

SpareFromSpread

SpareFromSpread(SiteID)

Example:
SpareFromSpread(2)

This activates the option "Spare from Spread" for Site ID 2 in Pandoras Box. The option can be found in the Device Inspector [210] .

StopClientSound

StopClientSound

Example:
StopClientSound

Stops any currently playing sound on the current client (depending on the script's context).

StopSound

StopSound

Example:
StopSound

Stops the currently playing sound that was started via the command PlaySound,Filename [1576].

SystemLock

SystemLock

Example:
SystemLock

This locks the system. You will see the screen were Windows asks for your password to unlock the system again.

SystemLogOff

SystemLogOff

Example:
SystemLogOff

Logs off the user of the local WD computer in a way that you do not have to confirm it.

SystemReboot

SystemReboot

Example:
SystemReboot

Reboots the local WD computer in a way that you do not have to confirm it.

SystemServiceStart

SystemServiceStart(ServiceName)

Example:
SystemServiceStart("Netman")

Starts the stopped Service for Network Connections called "Netman". Please have a look into the Windows Task Manager to get the correct name of the stopped services you want to start. Please enter the Service Name exactly like it is listed under Taskmanager/Services/Name.

SystemServiceStop

SystemServiceStop(ServiceName)

Example:
SystemServiceStop("Netman")

Stops the running Service for Network Connections called "Netman". Please have a look into the Windows Task Manager to get the correct name of the running services you want to stop. Please enter the Service Name exactly like it is listed under Taskmanager/Services/Name.

SystemShutdown

SystemShutdown

Example:
SystemShutdown

Shuts down the local WD computer in a way that you do not have to confirm it.

SystemVolume

SystemVolume(0-100)

Example:
SystemVolume(50)

Sets the System Volume to 50%.

TCPClearBuffer

TCPClearBuffer(ID)

Example:
TCPClearBuffer(1)

Clears the buffer from the TCP Connection with ID 1 from the Connection Manager [1258]

TCPInject

TCPInject(ID,Message)

Example:
TCPInject(1,"Play")

Injects the message "Play" directly into the stream of the TCP Connection with ID 1, without waiting for other messages or packages to be finished.

The TCP Connection needs to be defined first in the Connection Manager [1258].

Incoming or outgoing values in TCP-, UDP- and COM Port nodes can be entered as string, decimal or hexadecimal values as explained in the chapter Syntax TCP-/UDP-/Serial Messages [944].

Example:
Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

For two commands - e.g. a carriage return and a line feed - you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

TCPSend

TCPSend(ID,Message)

Example:
TCPSend(1,"Play")

Sends the ASCII message "Play" via the TCP Connection with the ID 1 in the Connection Manager [1258].

Incoming or outgoing values in TCP-, UDP- and COM Port nodes can be entered as string, decimal or hexadecimal values as explained in the chapter Syntax TCP-/UDP-/Serial Messages [944].

Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

For two commands - e.g. a carriage return and a line feed - you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

TCPSendDec

TCPSendDec(ID,Message)

Example:
TCPSendDec(1,"72 105 33")

Sends the message "Hi!" in decimal values via the TCP Connection with the ID 1 in the Connection Manager [1258]. For more information see the chapter "Syntax TCP-/UDP-/Serial Messages" [944].

TCPSendHex

TCPSendHex(ID,Message)

Example:
TCPSendHex(1,"48 69 21")

Sends the message "Hi!" in hexadecimal values via the TCP Connection with the ID 1 in the Connection Manager [1258]. For more information see the chapter "Syntax TCP-/UDP-/Serial Messages" [944].

TCPSendToIP

TCPSendToIP(ID,IP,Message)

Example:
TCPSendToIP(1,"192.168.1.38","Play")

Sends the message "Play" via the TCP Connection with ID 1 from the Connection Manager [1258] to IP address 192.168.1.38.

Incoming or outgoing values in TCP-, UDP- and COM Port nodes can be entered as string, decimal or hexadecimal values as explained in the chapter Syntax TCP-/UDP-/Serial Messages [944].

Example:
Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

For two commands - e.g. a carriage return and a line feed - you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

TCPStart

TCPStart(ID)

Example:
TCPStart(1)

Starts the TCP Connection with the ID 1 in the Connection Manager [1258].

TCPStartAll

TCPStartAll

Example:
TCPStartAll

Starts all TCP Connections in the Connection Manager [1258].

TCPStop

TCPStop(ID)

Example:
TCPStop(1)

Stops the TCP Connection with the ID 1 in the Connection Manager [1258].

TCPStopAll

TCPStopAll

Example:
TCPStopAll

Stops all TCP Connections in the Connection Manager [1258].

ToggleFullScreen

ToggleFullScreen(SiteID)

Example:
ToggleFullScreen(4)

Switches the site 4 from windowed to fullscreen mode in Pandoras Box project.

U U

UDPClearBuffer

UDPClearBuffer(ID)

Example:
UDPClearBuffer(1)

Clears the buffer from the UDP Connection with ID 1 from the Connection Manager [1258]

UDPInject

UDPInject(ID,Message)

Example:
UDPInject(1,"Play")

Injects the message "Play" directly into the stream of the UDP Connection with ID 1, without waiting for other messages or packages to be finished.

The UDP Connection needs to be defined first in the Connection Manager [1258].

Incoming or outgoing values in TCP-, UDP- and COM Port nodes can be entered as string, decimal or hexadecimal values as explained in the chapter Syntax TCP-/UDP-/Serial Messages [944].

Example:
Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

For two commands - e.g. a carriage return and a line feed - you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

UDPSend

UDPSend(ID,Message)

Example:
UDPSend(1,"Play")

Sends the message "Play" via the UDP Connection with the ID 1 in the Connection Manager [1258].

Incoming or outgoing values in TCP-, UDP- and COM Port nodes can be entered as string, decimal or hexadecimal values as explained in the chapter Syntax TCP-/UDP-/Serial Messages [944].

Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

For two commands - e.g. a carriage return and a line feed - you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

UDPSendDec

UDPSendDec(ID,Message)

Example:
UDPSendDec(1,"72 105 33")

Sends the message "Hi!" in decimal values via the TCP Connection with the ID 1 in the Connection Manager [1258]. For more information see the chapter "Syntax TCP-/UDP-/Serial Messages" [944].

UDPSendHex

UDPSendHex(ID,Message)

Example:
UDPSendHex(1,"48 69 21")

Sends the message "Hi!" in hexadecimal values via the UDP Connection with the ID 1 in the Connection Manager [1258]. For more information see the chapter "Syntax TCP-/UDP-/Serial Messages" [944].

UDPSendToPort

UDPSendToPort(Port,Message)

Example:
UDPSendToPort(10000,"Play")

Sends the message "Play" to the port 10000 via an UDP Connection .

Incoming or outgoing values in TCP-, UDP- and COM Port nodes can be entered as string, decimal or hexadecimal values as explained in the chapter Syntax TCP-/UDP-/Serial Messages [944].

Example:
Use [d13] to enter a carriage return as a decimal value.
Use [h0D] to enter a carriage return as a hexadecimal value.
Use [CR] to enter a carriage return as a ASCII symbol.

For two commands - e.g. a carriage return and a line feed - you may use this syntax: [d13 d10] respectively [h0D h0A] or [CR LF]. Mixed values are also possible, e.g: Example String[CR d10].

UDPStart

UDPStart(ID)

Example:
UDPStart(1)

Starts the UDP Connection with the ID 1 in the Connection Manager [1258].

UDPStartAll

UDPStartAll

Example:
UDPStartAll

Starts all UDP Connections in the Connection Manager [1258].

UDPStop

UDPStop(ID)

Example:
UDPStop(1)

Stops the UDP Connection with the ID 1 in the Connection Manager [1258].

UDPStopAll

UDPStopAll

Example:
UDPStopAll

Stops all UDP Connections in the Connection Manager [1258].

V V

VAdd

VAdd(VarName,Value1,Value2)

Example:
VAdd("varRes",Counter1,Counter2)

Adds the variables "Counter1" and "Counter2" and assigns this result to the existing variable [1900] "varRes". You could of course also use integer or double values instead of the "Counter" variables.

Example2:
varRes += 20

This example shows the faster and easier syntax "Variable += Value". It adds the value "20" to the current value of "varRes" and assigns the result to the same variable.

Example3:
varRes = Math.Max(1, 2)

The last example shows how to use the Math object [1924] for more complex mathematical functions. "Max" for example returns the larger of two specified numbers (or variables).

To monitor all variables, their type and value, please refer to the Variable List [1900].

VArrayAppend

VArrayAppend(VarName,Value)

Example:
VArrayAppend("var_List",99)

Adds 99 as an element at the end of the variable [1900] "var_List" that was defined beforehand with the List type. Note that the type of the sub element is automatically adjusted.

VArrayGetFileNames

VArrayGetFileNames(VarName,Folder)

Example:
VArrayGetFileNames("var_List","C:\Videos")

Returns the names (including paths) of all files saved in the folder "Videos" to the variable [1900] "var_List" that was defined beforehand with the List type. Note that the amount and type of the sub elements are automatically adjusted.

For example, the variable (with separator "|") could return:
C:\Videos\1.m2v|C:\Videos\2.m2v|C:\Videos\3.m2v|

In case the specified folder is empty or includes only sub folders, the List is cleared.

VArrayGetFilePathsFromFileDialog

VArrayGetFilePathsFromFileDialog(VarName)

Example:
VArrayGetFilePathsFromFileDialog("var_StringArray")

Opens an explorer dialog for the user to select multiple files. By clicking "Open" the file paths are written to the variable [1900] array "var_List" that was defined beforehand.

VArrayGetFilePathsFromFileDialogWithPath

VArrayGetFilePathsFromFileDialogWithPath(Path,VarName)

Example:
VArrayGetFilePathsFromFileDialogWithPath("C:\Program Files\Christie,"var_List")

Opens an explorer dialog displaying the specified folder for the user to select multiple files. By clicking "Open" the file paths are written to the variable [1900] array "var_List" that was defined beforehand.

VArrayGetListViewRow

VArrayGetListViewRow(Varname,ID,Row)

Example:
VArrayGetListViewRow("var_List",2,3)

Imports all values from row 3 of ListView 890 ID 2 to variable 1900 array "var_List" that was defined beforehand.

VArrayGetNodeOutputValue

VArrayGetNodeOutputValue(VarName,NodeID,ParamID)

Example:
VArrayGetNodeOutputValue("varList",5,2)

Assigns the second output value of the node with ID 5 to the List variable 1900 "varList" that was defined beforehand. The second output is also the second entry in the drop-down list when choosing a value in a connected node.
Note: Choose an Input or Filter Node that delivers a list, e.g. a Watch Folder input node 1037.

VArrayPrepend

VArrayPrepend(VarName,Value)

Example:
VArrayPrepend("var_List",99)

Adds 99 as an element at the beginning of the variable 1900 "var_List" that was defined beforehand with the List type. Note that the type of the sub element is automatically adjusted.

VArrayRemoveFirst

VArrayRemoveFirst(VarName)

Example:
VArrayRemoveFirst("var_List")

Removes the first element of the variable 1900 array "var_List" that was defined beforehand with the List type.

VArrayRemoveLast

VArrayRemoveLast(VarName)

Example:
VArrayRemoveLast("var_List")

Removes the last element of the variable 1900 array "var_List" that was defined beforehand with the List type.

VArrayResize

VArrayResize(VarName,Size)

Example:
VArrayResize("var_List",3)

Resizes the variable 1900 array "var_List", that was defined beforehand with the List type, to 3 empty elements, i.e. already existing data is cleared. Use the command `VArrayResizePreserve` if you like to keep existing data.

VArrayResizePreserve

VArrayResizePreserve(VarName,Size)

Example:
VArrayResizePreserve("var_List",3)

Resizes the variable [1900] array "var_List", that was defined beforehand with the List type, to 3 elements and preserves the already existing data. If the original array is larger than the resized one, the redundant elements are deleted. If it is smaller, the missing elements are filled with empty spaces.

VArrayReverse

VArrayReverse(VarName)

Example:
VArrayReverse("var_List")

Reverses the order of elements of the variable [1900] array "var_List" that was defined beforehand with the List type.

VArraySort

VArraySort(VarName)

Example:
VArraySort("var_List")

Sorts the elements of the variable [1900] array "var_List" alphabetically, e.g. ,|.|1|9|a|A|z|Z

VCreate

VCreate(VarName,Value)

Example:
VCreate("varWeight",73.3)

Creates a global variable [1900] called "varWeight" with the type Double and the value 73.3. The variable type (String, Integer, Double, Boolean, Date, List, Color, Json) is assigned automatically. In some cases so called members need to be used to specify the type, e.g. VCreate("varColor","#000FFF".ToColor), see more in the chapter Data Type Specific Members [1914].

The best practice is, to declare global variables in one script but use their values in separate scripts. The reason is, that scripts are compiled before they are executed and global variables cannot be declared and used in one compilation. This is when local variables [1903] should be used. For example:

```
var myLocal = "SomeText"  //this declares a local variable
Label4.Text = myLocal     //this assigns the value to a Label which would not be possible with a global variable if it was declared in the same script
```

VDelete

VDelete(Name)

Example:
VDelete("City")

Deletes the variable [1900] with the name "City".

VDeleteAll

VDeleteAll

Example:
VDeleteAll

Deletes all existing variables [1900] inside WD.

VDivide

VDivide(VarName,Value1,Value2)

Example:
VDivide("varRes",Counter1,Counter2)

Divides the variable "Counter1" by the variable "Counter2" and assigns the result to the existing variable [1900] "varRes". You could of course also use integer or double values instead of the "Counter" variables.

Example2:
varRes /= 20

This example shows the faster and easier syntax "Variable /= Value". It divides the current value of "varRes" with 20 and assigns the result to the same variable.

Example3:
varRes = Math.Max(1, 2)

The last example shows how to use the Math object [1924] for more complex mathematical functions. "Max" for example returns the larger of two specified numbers (or variables).

VExecuteAsScript

VExecuteAsScript(VarName)

Example:
VExecuteAsScript("var_String")

Executes the content of the string-type variable [1900] "var_String" as a script.
E.g., if "var_String" has the following content, the variable "var_Count" increases one step
var_Count += 1
or
VAdd("var_Int",var_Int,1)

VFade

VFade(VarName,StartValue,EndValue,Time)

Example:
VFade("var_test",25,196,10)

Sets the variable [1900] "var_test" to the value 25 and fades it to 196 within 10 seconds. The variable must be defined beforehand and with the type integer and double.

Note: The fade itself is not completely linear, it starts slowly, maintains its speed in the middle and slows down again at the end!

VFadeTo

VFadeTo(VarName,Value,Time)

Example:
VFadeTo("var_test",314,10)

Fades variable [1900] "var_test" to the value 314 within 10 seconds. The variable must be defined beforehand and with the type integer and double.

Note: The fade itself is not completely linear, it starts slowly, maintains its speed in the middle and slows down again at the end!

VGetArraySize

VGetArraySize(VarName,ArrName)

Example:
VGetArraySize("var_Count","var_List")

Returns the absolute number of elements of the variable [1900] array "var_List" to the variable "var_Count" that was defined beforehand with the Integer type. E.g. "var_List" = [1,2,3] -> "var_Count" = 3.

VGetArrayValue

VGetArrayValue(VarName,Index,ArrName)

Example:
VGetArrayValue("var_Value",7,"var_List")

Returns the value of the 8th element from the variable [1900] array "var_List" to the variable "var_Value". Whilst "var_List" was defined with the List type, the type of "var_Value" needs to match the extracted array data (Integer, String, etc.).

Note: The index starts with 0 -> Index 7 matches the 8th element.

VGetAssetDurationHMSByID

VGetAssetDurationHMSByID(VarName,FolderID,FileID)

Example:
VGetAssetDurationHMSByID("var_String",1,3)

Returns the duration (in timecode format hh:mm:ss) of the media file with the Folder and File ID [1,3] in the Pandoras Box Project Tab to the variable [1900] "var_String" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetDurationHMSByPath

VGetAssetDurationHMSByPath(VarName,ProjectPath)

Example:
VGetAssetDurationHMSByPath("var_String","files\blue_lines.m2v")

Returns the duration (in timecode format hh:mm:ss) of the media file "blue_lines.m2v", located in the folder "files" in the Pandoras Box Project Tab, to the variable [1900] "var_String" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetDurationHMSFByID

VGetAssetDurationHMSFByID(VarName,FolderID,FileID)

Example:
VGetAssetDurationHMSFByID("var_String",1,2)

Returns the duration (in timecode format hh:mm:ss:ff) of the media file with the Folder and File ID [1,2] in the Pandoras Box Project Tab to the variable "var_String" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetDurationHMSFByPath

VGetAssetDurationHMSFByPath(VarName,ProjectPath)

Example:
VGetAssetDurationHMSFByPath("var_String","files\blue_lines.m2v")

Returns the duration (in timecode format hh:mm:ss:ff) of the media file "blue_lines.m2v", located in the folder "files" in the Pandoras Box Project Tab, to the variable "var_String" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetDurationTotalFramesById

VGetAssetDurationTotalFramesById(VarName,FolderID,FileID)

Example:
VGetAssetDurationTotalFramesById("var_Int,1,5)

Returns the total number of frames of the media file with the Folder and File ID [1,5] in the Pandoras Box Project Tab to variable "var_Int" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetDurationTotalFramesByPath

VGetAssetDurationTotalFramesByPath(VarName,ProjectPath)

Example:
VGetAssetDurationTotalFramesByPath("var_Int","files\yellow_lines.m2v")

Returns the total number of frames of the media file "yellow_lines.m2v", located in the folder "files" in the Pandoras Box Project Tab, to the variable "var_String" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetDurationTotalSecByID

VGetAssetDurationTotalSecByID(VarName,FolderID,FileID)

Example:
VGetAssetDurationTotalSecByID("var_Value",1,3)

Returns the duration (in seconds) of the media file with the Folder and File ID [1,3] in the Pandoras Box Project Tab to the variable [1900] "var_Value" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetDurationTotalSecByPath

VGetAssetDurationTotalSecByPath(VarName,ProjectPath)

Example:
VGetAssetDurationTotalSecByPath("var_Value","files\blue_lines.m2v")

Returns the duration (in seconds) of the media file "blue_lines.m2v", located in the folder "files" in the Pandoras Box Project Tab, to the variable [1900] "var_Value" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetMediaFileNameByID

VGetAssetMediaFileNameByID(VarName,FolderID,FileID)

Example:
VGetAssetMediaFileNameByID("Var_Name",1,5)

Returns the name (string) of the media file with the Folder and File ID [1,5] in the Pandoras Box Project Tab to the variable "Var_Name" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetMediaHeightByID

VGetAssetMediaHeightByID(VarName,FolderID,FileID)

Example:
VGetAssetMediaHeightByID("var_Value",1,3)

Returns the height (in pixels) of the media file with the Folder and File ID [1,3] in the Pandoras Box Project Tab to the variable [1900] "var_Value" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetMediaHeightByName

VGetAssetMediaHeightByName(VarName,AssetName)

Example:
VGetAssetMediaHeightByName("var_Value","files\blue_lines.m2v")

Returns the height (in pixels) of the media file "blue_lines.m2v", located in the folder "files" in the Pandoras Box Project Tab, to the variable [1900] "var_Value" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetMediaProjectFileNameByID

VGetAssetMediaProjectFileNameByID(VarName,FolderID,FileID)

Example:
VGetAssetMediaProjectFileNameByID("Var_ProjectPath",1,3)

Returns the Pandoras Box project path (string) of the media file with the Folder and File ID [1,3] in the Pandoras Box Project Tab to the variable "Var_ProjectPath" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetMediaProjectFolderByID

VGetAssetMediaProjectFolderByID(VarName,FolderID,FileID)

Example:
VGetAssetMediaProjectFolderByID("Var_Folder",1,1)

Returns the specific Folder Name & Media Name (string) as a path with the Folder and File ID [1,3] in the Pandoras Box Project Tab to the variable "Var_Folder" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetMediaWidthByID

VGetAssetMediaWidthByID(VarName,FolderID,FileID)

Example:
VGetAssetMediaWidthByID("var_Value",1,3)

Returns the width (in pixels) of the media file with the Folder and File ID [1,3] in the Pandoras Box Project Tab to the variable [1900] "var_Value" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetAssetMediaWidthByName

VGetAssetMediaWidthByName(VarName,AssetName)

Example:
VGetAssetMediaWidthByName("var_Value","files\blue_lines.m2v")

Returns the width (in pixels) of the media file "blue_lines.m2v", located in the folder "files" in the Pandoras Box Project Tab, to the variable [1900] "var_Value" that was defined beforehand.

Note: All VGetAsset... commands access cached information that is stored (and updated) through the command PBProjectRefresh [1570]. Execute it once before running a VGetAsset command and any time the project has changed.

VGetContainsText

VGetContainsText(VarName,SourceString,CompareString)

Example:
VGetContainsText("var_true","Hello World!","llo W")

Checks if the compare string ("llo W") matches with any string inside the source string ("Hello World!") and returns "1" for a true argument and "0" for a false one through the string or integer variable [1900] "var_true" that was defined beforehand.

Note: This function is case-sensitive and space characters count.

VGetCSVFileColCount

VGetCSVFileColCount(VarName,Filename,Separator(optional))

Example:
VGetCSVFileColCount("var_Count","C:\Data\Matrix.csv",";")

Returns the amount of columns of the specified CSV file to the variable [1900] "var_Count" that was defined beforehand.

The separator is optional and can be any symbol. Most programs like Microsoft Excel use the semicolon ";" by default.

VGetCSVFileRowCount

VGetCSVFileRowCount(VarName,Filename)

Example:
VGetCSVFileRowCount("var_Count","C:\Data\Matrix.csv")

Returns the amount of rows of the specified CSV file to the variable [1900] "var_Count" that was defined beforehand.

VGetDeviceMediaFileID

VGetDeviceMediaFileID(VarName,SiteID,DeviceID)

Example:
VGetDeviceMediaFileID("var_Number",1,2)

Returns the File ID of the media file of Pandoras Box, Site 1, Layer 2 to variable [1900] "var_Number" that was defined beforehand with the Double or Integer type.
In PB, you can assign a Folder and File ID using the File Inspector [191].

VGetDeviceMediaFolderID

VGetDeviceMediaFolderID(VarName,SiteID,DeviceID)

Example:
VGetDeviceMediaFolderID("var_Number",1,2)

Returns the Folder ID of the media file of Pandoras Box, Site 1, Layer 2 to variable [1900] "var_Number" that was defined beforehand with the Double or Integer type.
In PB, you can assign a Folder and File ID using the File Inspector [191].

VGetDeviceMediaName

VGetDeviceMediaName(VarName,SiteID,DeviceID)

Example:
VGetDeviceMediaName("var_String",1,2)

Returns the name of the media file of Pandoras Box, Site 1, Layer 2 to the string-type variable [1900] "var_String" that was defined beforehand.

Note: The file name does not include the extension (.png / .avi / .wav ...)

VGetDeviceMediaPath

VGetDeviceMediaPath(VarName,SiteID,DeviceID)

Example:
VGetDeviceMediaPath("var_String",1,2)

Returns the path of the media file of Pandoras Box, Site 1, Layer 2 to the string-type variable [1900] "var_String" that was defined beforehand.

VGetDeviceObjectFileID

VGetDeviceObjectFileID(VarName,SiteID,DeviceID)

Example:
VGetDeviceObjectFileID("var_Number",1,2)

Returns the File ID of the object file of Pandoras Box, Site 1, Layer 2 to variable [1900] "var_Number" that was defined beforehand with the Double or Integer type.
In PB, you can assign a Folder and File ID using the File Inspector [191].

VGetDeviceObjectFolderID

VGetDeviceObjectFolderID(VarName,SiteID,DeviceID)

Example:
VGetDeviceObjectFolderID("var_Number",1,2)

Returns the Folder ID of the object file of Pandoras Box, Site 1, Layer 2 to variable [1900] "var_Number" that was defined beforehand with the Double or Integer type.
In PB, you can assign a Folder and File ID using the File Inspector [191].

VGetDeviceObjectPath

VGetDeviceObjectPath(VarName,SiteID,DeviceID)

Example:
VGetDeviceObjectPath("var_String",1,2)

Returns the path of the object file of Pandoras Box, Site 1, Layer 2 to the string-type variable [1900] "var_String" that was defined beforehand.

VGetDeviceParam

VGetDeviceParam(VarName,SiteID,DeviceID,ParamName)

Example:
VGetDeviceParam("var_Number",1,2,"Opacity")

Returns the value of the parameter "Opacity" of Pandoras Box, Site 1, Layer 2 to variable "var_Number" that was defined beforehand with the Double or Integer type.

Note: The parameter name is case sensitive! You can copy all names including the syntax for an Effect or Particle System from the chapter Parameter List [1514].

VGetEndsWithText

VGetEndsWithText(VarName,SourceString,CompareString)

Example:
VGetEndsWithText("var_true","Hello World!","orld!")

Checks if the compare string ("orld!") matches with the end of the source string ("Hello World!") and returns "1" for a true argument and "0" for a false one through the string or integer variable [1900] "var_true" that was defined beforehand.

Note: This function is case-sensitive and space characters count.

VGetExcelCellValue

VGetExcelCellValue(VarName,File,Table,Cell)

Example:
VGetExcelCellValue("var_Value","C:\Data\excel.xls","Sheet1","C7")

Assigns the value of cell C7 from Excel document "excel.xls", table "Sheet1" to the variable [1900] "var_Value" that was defined beforehand.

Note: The document has to be an XLS file.

VGetFaderVal

VGetFaderVal(VarName,FaderID)

Example:
VGetFaderVal("Counter",1)

Applies the value of Fader 1 to the variable [1900] "Counter".

VGetFileCount

VGetFileCount(VarName,Folder)

Example:
VGetFileCount("var_Count","C:\Videos")

Returns the amount of files saved in the folder "Videos" to variable [1900] "var_Count" that was defined beforehand with the Integer or Double type. Note that sub folders are not counted, but system or hidden files are counted which includes for example a "thumbs.db" file.

VGetFileNameFromPath

VGetFileNameFromPath(VarName,Path)

Example:
VGetFileNameFromPath("var_String","C:\Program Files\Christie\Widget Designer 6.0 Rev 4498\logs\wd_gui_log.txt")

Writes the name of the file described by the path into the string-type variable [1900] "var_String" that was defined beforehand.

VGetFilePathFromFileDialog

VGetFilePathFromFileDialog(VarName)

Example:
VGetFilePathFromFileDialog("var_String")

Opens an explorer dialog for the user to select one file. By clicking "Open" the file path is written into the string-type variable [1900] "var_String" that was defined beforehand.

VGetFilePathFromFileDialogWithPath

VGetFilePathFromFileDialogWithPath(Path,VarName)

Example:
VGetFilePathFromFileDialogWithPath("C:\Program Files\Christie,"var_String")

Opens an explorer dialog displaying the specified folder for the user to select one file. By clicking "Open" the file path is written into the string-type variable [1900] "var_String" that was defined beforehand.

VGetFolderPathFromFileDialog

VGetFolderPathFromFileDialog(VarName)

Example:
VGetFolderPathFromFileDialog("var_String")

Opens an explorer dialog for the user to select a folder. By clicking "Open" the folder path is written into the string-type variable [1900] "var_String" that was defined beforehand.

VGetFunctionResult

VGetFunctionresult(VarName,FunctionName,FunctionArguments (optional))

Example:
VGetFunctionresult("return","FaderFunction",1,5)

Executes the example function [1897] "FaderFunction" and passes 1 and 5 as input arguments. The return value of "FaderFunction" is applied to the variable "return".

VGetInputboxText

VGetInputboxText(VarName,ID)

Example:
VGetInputboxText("var_String",3)

Returns the text of the Inputbox [886] 3 to the variable [1900] "var_String" that was defined beforehand.

vGetJsonItem

vGetJsonItem(ToVarName,JsonItem,FromVarName)

Example:
VGetJsonItem("varJson", "products","varJsonExample")

Loads the Json data from the Json variable [1926] "varJsonExample", searches for "products" and stores the related keys and values into the Json variable "varJson" which could be for example:
{"media_server":"Pandoras Box","projector":"Boxer","video_wall":"MicroTiles"}.

VGetLabelText

VGetLabelText(VarName,LabelID)

Example:
VGetLabelText("City",2)

Applies the text of Label 2 to the variable [1900] "City".

VGetListViewAverageCol

VGetListViewAverageCol(VarName,ID,Col)

Example:
VGetListViewAverageCol("var_Value",1,2)

Calculates the arithmetic mean of all values located in column 2 in ListView [890] 1 and assigns the result to the variable [1900] "var_Value" that was defined beforehand.

Note: If the variable is an integer-type, the result is being rounded!

VGetListViewAverageRow

VGetListViewAverageRow(VarName,ID,Row)

Example:
VGetListViewAverageRow("var_Value",1,2)

Calculates the arithmetic mean of all values located in row 2 in ListView 890 1 and assigns the result to the variable 1900 "var_Value" that was defined beforehand.

Note: If the variable is an integer-type, the result is being rounded!

VGetListViewCell

VGetListViewCell(VarName,ID,Column,Row)

Example:
VGetListViewCell("var_Value",1,2,3)

Searches for the cell located in column 2, row 3 in ListView 890 1. Its value is assigned to the variable 1900 "var_Value" that was defined beforehand.

VGetListViewSearchCell

VGetListViewSearchCell(ID,Value,VarNameForColumn,VarNameForRow)

Example:
VGetListViewSearchCell(1,3.142,"var_Col","var_Row")

Searches for a cell with the value 3.142 in ListView 890 1 and assigns its position (number of column and number of row) to the variables 1900 "var_Col" and "var_Row" that were defined beforehand.

Note: If there is no such value in the Listview, the command will return -1 / -1.

VGetListViewSearchDown

VGetListViewSearchDown(VarName,ID,Value,Offset(optional))

Example:
VGetListViewSearchDown("var_Value",1,13,2)

Searches for a cell with the value 13 in ListView 890 1 and looks for the cell three rows down (1 cell down + 2 cells offset). Its value is assigned to the variable 1900 "var_Value" that was defined beforehand.

E.g, if Value 13 was found in (any column) row 1, the command returns the value of (same column) row 4.

Example 2:
VGetListViewSearchDown("var_Value",1,13)
Without the optional "offset" parameter, the command returns the value of the neighbor cell.

VGetListViewSearchLeft

VGetListViewSearchLeft(VarName,ID,Value,Offset(optional))

Example:
VGetListViewSearchLeft("var_Value",1,13,2)

Searches for a cell with the value 13 in ListView 890 1 and looks for the cell three columns to the left (1 cell to the left + 2 cells offset). Its value is assigned the variable 1900 "var_Value" that was defined beforehand.

E.g, if Value 13 was found in column 4 (any row), the command returns the value of column 1 (same row)

Example 2:
VGetListViewSearchLeft("var_Value",1,13)
Without the optional "offset" parameter, the command returns the value of the neighbor cell.

VGetListViewSearchRight

VGetListViewSearchRight(VarName,ID,Value,Offset(optional))

Example:
VGetListViewSearchRight("var_Value",1,13,2)

Searches for a cell with the value 13 in ListView 890 1 and looks for the cell three columns to the right (1 cell to the right + 2 cells offset). Its value is assigned the variable 1900 "var_Value" that was defined beforehand.

E.g, if Value 13 was found in column 1 (any row), the command returns the value of column 4 (same row).

Example 2:
VGetListViewSearchRight("var_Value",1,13)
Without the optional "offset" parameter, the command returns the value of the neighbor cell.

VGetListViewSearchUp

VGetListViewSearchUp(VarName,ID,Value,Offset(optional))

Example:
VGetListViewSearchUp("var_Value",1,13,2)

Searches for a cell with the value 13 in ListView 890 1 and looks for the cell three rows up (1 cell up + 2 cells offset). Its value is assigned to the variable 1900 "var_Value" that was defined beforehand.

E.g, if Value 13 was found in (any column) row 4, the command returns the value of (same column) row 1.

Example 2:
VGetListViewSearchUp("var_Value",1,13)
Without the optional "offset" parameter, the command returns the value of the neighbor cell.

VGetListViewSelectedRow

VGetListViewSelectedRow(VarName,ID)

Example:
VGetListViewSelectedRow("var_Int",1)

Assigns the number of the currently selected row of ListView 890 1 to variable 1900 "var_Int" that was defined beforehand.

VGetListViewSumCol

VGetListViewSumCol(VarName,ID,Col)

Example:
VGetListViewSumCol("var_Value",1,2)

Calculates the sum of all values located in column 2 in ListView 890 1 and assigns the result to the variable 1900 "var_Value" that was defined beforehand.

Note: If the variable is an integer-type, the result is being rounded!

VGetListViewSumRow

VGetListViewSumRow(VarName,ID,Row)

Example:
VGetListViewSumRow("var_Value",1,2)

Calculates the sum of all values located in row 2 in ListView 890 1 and assigns the result to the variable 1900 "var_Value" that was defined beforehand.

Note: If the variable is an integer-type, the result is being rounded!

VGetNodeOutputValue

VGetNodeOutputValue(Varname,NodeID,ParamID)

Example:
VGetNodeOutputValue("var_Value",1,2)

Assigns the second output value of the node with ID 1 to the variable 1900 "var_Value" that was defined beforehand. The second output is also the second entry in the drop-down list when choosing a value in a connected node.

Note: Please use the correct type of variable matching the node output type.

VGetNodeVal

VGetNodeVal(VarName,NodeID,ParamID)

Example:
VGetNodeVal("Counter",10,1)

Applies the value of Node 10, Parameter 1 to the variable 1900 "Counter".

VGetPBPlaylistItemCountByID

VGetPBPlaylistItemCountByID(VarName,PlaylistFolderID,PlaylistFileID)

Example:
VGetPBPlaylistItemCountByID("var1",2,1)

This returns the total count of items of the Pandoras Box playlist 239 with the folder and file ID [2,1]. The (string or number) Variable 1900 "var1" in Widget Designer will be assigned with this number.

VGetPBPlaylistItemCountByPath

VGetPBPlaylistItemCountByPath(VarName,PlaylistPath)

Example:
VGetPBPlaylistItemCountByPath("var1","Playlist Test\Playlist 1")

This returns the total count of items of the Pandoras Box playlist 239 "Playlist 1" of the subfolder "Playlist Test" within the project folder. The (string or number) Variable 1900 "var1" in Widget Designer will be assigned with this number.

VGetRSSUrlItemBody

VGetRSSUrlItemBody(VarName,UrlID,ItemID)

Example:
VGetRSSUrlItemBody("var_String",1,14)

Assigns the body of RSS item 14 of the RSS feed with ID 1 to the string-type variable [1900] "var_String" that was defined beforehand. RSS news feeds can be accessed via the RSS Settings Tool [1493].

VGetRSSUrlItemCount

VGetRSSUrlItemCount(VarName,UrlID)

Example:
VGetRSSUrlItemCount("var_Count",1)

Counts the number of RSS items within the RSS feed with ID 1 and assigns the result to the variable [1900] "var_Count" that was defined beforehand. RSS news feeds can be accessed via the RSS Settings Tool [1493].

VGetRSSUrlItemFeed

VGetRSSUrlItemFeed(VarName,UrlID,ItemID)

Example:
VGetRSSUrlItemFeed("var_String",1,14)

Assigns the title and body of RSS item 14 of the RSS feed with ID 1 to the string-type variable [1900] "var_String" that was defined beforehand. RSS news feeds can be accessed via the RSS Settings Tool [1493].

VGetRSSUrlItemTitle

VGetRSSUrlItemTitle(VarName,UrlID,ItemID)

Example:
VGetRSSUrlItemTitle("var_String",1,14)

Assigns the title of RSS item 14 of the RSS feed with ID 1 to the string-type variable [1900] "var_String" that was defined beforehand. RSS news feeds can be accessed via the RSS Settings Tool [1493].

VGetSeqCueName

VGetSeqCueName(VarName,SeqID,CueID)

Example:
VGetSeqCueName("Counter",1,3)

This applies the name of Cue 3 from the Sequence 1 in Pandoras Box to the variable in Widget Designer named "Counter". If the cue name includes letters, the Variable's type must be "String". To monitor all variables, their type and value, please refer to the Variable List [1900].

VGetSeqState

VGetSeqState(VarName,SeqID)

Example:
VGetSeqState("Counter",2)

This applies the current playback state "Play", "Pause" or "Stop" of the Sequence 1 in Pandoras Box to the variable in Widget Designer which is named "Counter" . To monitor all variables, their type and value, please refer to the Variable List [1900].

VGetSeqTime

VGetSeqTime(VarName,SeqID)

Example:
VGetSeqTime("Counter",1)

This applies the timecode (e.g. 00:01:12:23) of Sequence 1 in Pandoras Box to the variable in Widget Designer which is named "Counter" . The type of the variable must be "String". To monitor all variables, their type and value, please refer to the Variable List [1900].

VGetSiteConnectionBackup

VGetSiteConnectionBackup(VarName,SiteID)

Example:
VGetSiteConnectionBackup("var_Number",4)

Checks if Site 4 of Pandoras Box (Backup) is connected and returns "1" for a true argument and "0" for a false one through the variable [1900] "var_true" that was defined beforehand with the Integer or Double type.

The IP address for the PB Backup connection can be entered in the dialog Pandoras Box Network Configuration [1256].

VGetSiteConnectionMaster

VGetSiteConnectionMaster(VarName,SiteID)

Example:
VGetSiteConnectionMaster("var_Number",4)

Checks if Site 4 of Pandoras Box (Master) is connected and returns "1" for a true argument and "0" for a false one through the variable [1900] "var_Number" that was defined beforehand with the Integer or Double type.

The IP address for the PB Master connection can be entered in the dialog Pandoras Box Network Configuration [1256].

VGetStartsWithText

VGetStartsWithText(VarName,SourceString,CompareString)

Example:
VGetStartsWithText("var_true","Hello World!","Hell")

Checks if the compare string ("Hell") matches with the beginning of the source string ("Hello World!") and returns "1" for a true argument and "0" for a false one through the string or integer variable [1900] "var_true" that was defined beforehand.

Note: This function is case-sensitive and space characters count.

VGetStringLength

VGetStringLength(VarName,String)

Example:
VGetStringLength("var_length","I like trains")

Counts the amount of characters of the source string "I like trains" (= 13) and writes the result into the variable [1900] "var_lenghth" that was defined beforehand. Space characters count as well.

VGetStringReplace

VGetStringReplace(VarName,String,OldString,NewString)

Example:
VGetStringReplace("var_String","Hello World!","llo Wor","lium Go")

Sets the string-type variable to the string "Hello World!" and replaces the part "llo Wor" (if it exists) with "lium Go". New Value of var_String: "Helium Gold!". Space characters count as well.

VGetSubString

VGetSubString(VarName,String,StartIndex,Length)

Example:
VGetSubString("var_String","I like trains",3,8)

Shortens the string "I like trains" and writes the result into the string-type variable [1900] "var_String" that was defined before. In the example, the sub string has the length of 8 characters and begins with the 4th letter of the source string ("ike trai"). The start index 0 would refer to the first character.

VGetSubStringTrimStart

VGetSubStringTrimStart(VarName,String,StartIndex)

Example:
VGetSubStringTrimStart("var_String","Hello World!",7)

Trims the first 7 characters (including spaces etc.) of the string "Hello World!".
The remaining string ("orld!") is written into the string-type variable [1900] "var_String" that was defined beforehand.

VGetTextboxText

VGetTextboxText(VarName,TextboxID)

Example:
VGetTextboxText("V_Test",1)

Applies the text of Textbox 1 to the variable "V_Test".

To monitor all variables, their type and value, please refer to the Variable List [1900].

VLock

VLock(VarName)

Example:
VLock(VarName)

Locks the variable [1900] "var_test" so that its value cannot be edited.

VMultiply

VMultiply(VarName,Value1,Value2)

Example:
VMultiply("varRes",Counter1,Counter2)

Multiplies the values of the variables "Counter1" and "Counter2" and passes the result to the existing variable [1900] "varRes". You could of course also use integer or double values instead of the "Counter" variables.

Example2:
varRes *= 20

This example shows the faster and easier syntax "Variable *= Value". It multiplies the current value of "varRes" with the value "20" and assigns the result to the same variable.

Example3:
varRes = Math.Max(1, 2)

The last example shows how to use the Math object [1924] for more complex mathematical functions. "Max" for example returns the larger of two specified numbers (or variables).

To monitor all variables, their type and value, please refer to the Variable List [1900].

VPow

VPow(VarName,Value1,Value2)

Example:
VPow("varRes",Counter1,Counter2)

Raises the variable "Counter1" to the power of "Counter2" and assigns the result to the existing variable "varRes". You could of course also use integer or double values instead of the "Counter" variables.

Example2:
varRes = Math.Pow(5,2)

This example shows how to use the Math object [1924] for the same approach; it also offers other more complex mathematical functions.

To monitor all variables, their type and value, please refer to the Variable List [1900].

VReplaceAsciiNumberByHexCode

VReplaceAsciiNumberByHexCode(VarName,String,Delimter)

Example:
VReplaceAsciiNumberByHexCode("var_String","1 64 256 hi 11",":")

Replaces any ASCII number in the entered string by its corresponding HEX code and separates them with the separator character ":". Other characters simply remain in the string.

The result (:31 :36:34 :32:35:36 hi :31:31) is returned to the variable [1900] "var_String" that was defined beforehand with the string type.

VReplaceNumberByHexCode

VReplaceNumberByHexCode(VarName,String,Delimter,LeadingZeros)

Example:
VReplaceNumberByHexCode("var_String","1 64 256 hi 11",":",2)

Replaces any number in the entered string by its corresponding HEX code, adds up to 2 leading zeros and separates the result with the separator character ":". Other characters than numbers simply remain in the string.

The result (:001 :040 :100 hi :00B) is returned to the variable 1900 "var_String" that was defined beforehand with the string type.

VSetArrayFromSplitString

VSetArrayFromSplitString(VarName,String,Separator)

Example:
VSetArrayFromSplitString("var_List","fl;ol;lo;pi;ng",";")

Splits the specified string wherever the separator character ";" occurs. The resulting sub strings are returned as elements of the variable 1900 "var_List" that was defined beforehand with the List type. Note that the amount and type of the sub elements are automatically adjusted. The separator can be any single character including comma "," and space character " ".

VSetArrayFromTextBox

VSetArrayFromTextBox(VarName,TextBoxID)

Example:
VSetArrayFromTextBox("var_List",1)

Loads the content of Textbox 923 1 into the variable 1900 "var_List" that was defined beforehand with the List type. Please note that the defined separator is not recognized automatically! If you like to pass several sub elements to the list variable, you have two options:
Either, you separate elements with an empty new line. The amount and type of the sub elements are automatically adjusted.
Or, you write everything in one line using the separator as usually and execute another command which is: `VSetArrayFromSplitString("var_List",TextBox1.Text,"|")` or shorter: `varlist = TextBox1.Text.Split("|")`

VSetArrayRNDUniqueInt

VSetArrayRNDUniqueInt('VarName',StartVal,EndVal)

Example:
VSetArrayRNDUniqueInt("var_List",0,100)

Fills the variable 1900 array "var_List", that was defined beforehand with the List type, with random integer values from 1 to 100, no value appears twice.

Note: The array's size must be larger than the number of possible (unique) values!

VSetArrayValue

VSetArrayValue(VarName,Index,Value)

Example:
VSetArrayValue("var_List",7,444)

Sets the value of the 8th element in the variable [1900] array "var_List", that was defined beforehand with the List type, to the value 444. The type of the sub element is automatically adjusted.

Note: The index starts with 0 -> Index 7 matches the 8th element.

VSetValueFromSplitString

VSetValueFromSplitString(VarName,String,Separator,Index)

Example:
VSetValueFromSplitString("var_Value","00:10:30:42",":",2)

Splits the specified string wherever the separator character ":" occurs. The resulting sub string with index 2 is returned to the variable [1900] "var_Value" that was defined beforehand with the type that matches the string data (Integer, String, etc.). As indexing starts with 0, var_Value = 30.

VSqrt

VSqrt(VarName,Value)

Example:
VSqrt("varRes",Counter1)

Calculates the square root of the variable "Counter1" and assigns the result to the existing variable [1900] "varRes". You could of course also use an integer or double value instead of the "Counter" variable.

Example2:
varRes = Math.Sqrt(25)

This example shows how to use the Math object [1924] for the same approach; it also offers other more complex mathematical functions.

To monitor all variables, their type and value, please refer to the Variable List [1900].

VSubtract

VSubtract(VarName,Value1,Value2)

Example:
VSubtract("varRes",Counter1,Counter2)

Subtracts the variable "Counter2" from "Counter1" and assigns the result to the existing variable [1900] "varRes". You could of course also use integer or double values instead of the "Counter" variables.

Example2:
varRes -= 20

This example shows the faster and easier syntax "Variable -= Value". It subtracts the value "20" from the current value of "varRes" and assigns the result to the same variable.

Example3:
varRes = Math.Max(1, 2)

The last example shows how to use the Math object [1924] for more complex mathematical functions. "Max" for example returns the larger of two specified numbers (or variables).

To monitor all variables, their type and value, please refer to the Variable List [1900].

VUnLock

VUnLock(VarName)

Example:
VUnLock(VarName)

Unlocks the variable [1900] "var_test" so that its value can be edited.

VValue

VValue(VarName,Value)

Example:
VValue("varRes",20)

Applies the value "20" to the variable [1900] "varRes".

Example2:
varRes = 20

The second example shows how to use the syntax "Variable = Value" for the same approach.

VValueDate

VValueDate(VarName)

Example:
VValueDate("var_String")

Returns the current date (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type.

Example2:
var_String = Now.ToString

The second example shows how to use the global variable "Now" for the same approach but in a different format.

VValueDateAddDays

VValueDateAddDays(VarName,Days)

Example:
VValueDateAddDays("var_String",5)

This adds five days to the current date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type.

Note: To subtract please use negative values.

VValueDateAddDaysToDate

VValueDateAddDaysToDate(VarName,DD,MM,YYYY,Days)

Example:
VValueDateAddDaysToDate("var_String",25,01,2010,5)

This adds five days to the specified date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type. When executing this example, the result would be: 30/01/2010

Note: To subtract please use negative values. If you prefer to enter your date in the dd/mm/yyyy format too, use the VValueDateAddDaysToDateF [1644] command.

VValueDateAddDaysToDateF

VValueDateAddDaysToDateF(VarName,DD/MM/YYYY,Days)

Example:
VValueDateAddDaysToDateF("var_String","25/01/2010",5)

This adds five days to the specified date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type. When executing this example, the result would be: 30/01/2010

Note: To subtract please use negative values. If you prefer to enter your date as separate values (i.e. dd,mm,yyyy), use the VValueDateAddDaysToDate [1643] command.

VValueDateAddMonths

VValueDateAddMonths(VarName,Months)

Example:
VValueDateAddMonths("var_String",5)

This adds five months to the current date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type.

Note: To subtract please use negative values.

VValueDateAddMonthsToDate

VValueDateAddMonthsToDate(VarName,DD,MM,YYYY,Months)

Example:
VValueDateAddMonthsToDate("var_String",25,01,2010,5)

This adds five months to the specified date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type. When executing this example, the result would be: 25/06/2010

Note: To subtract please use negative values. If you prefer to enter your date in the dd/mm/yyyy format too, use the VValueDateAddMonthsToDateF [1644] command.

VValueDateAddMonthsToDateF

VValueDateAddMonthsToDateF(VarName,DD/MM/YYYY,Months)

Example:
VValueDateAddMonthsToDateF("var_String","25/01/2010",5)

This adds five months to the specified date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type. When executing this example, the result would be: 25/06/2010

Note: To subtract please use negative values. If you prefer to enter your date as separate values (i.e. dd,mm,yyyy), use the VValueDateAddMonthsToDate [1644] command.

VValueDateAddYears

VValueDateAddYears(VarName,Years)

Example:
VValueDateAddYears("var_String",5)

This adds five years to the current date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type.

Note: To subtract please use negative values.

VValueDateAddYearsToDate

VValueDateAddYearsToDate(VarName,DD,MM,YYYY,Years)

Example:
VValueDateAddYearsToDate("var_String",25,01,2010,5)

This adds five years to the specified date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type. When executing this example, the result would be: 25/01/2015

Note: To subtract please use negative values. If you prefer to enter your date in the dd/mm/yyyy format too, use the VValueDateAddYearsToDateF [1645] command.

VValueDateAddYearsToDateF

VValueDateAddYearsToDateF(VarName,DD/MM/YYYY,Years)

Example:
VValueDateAddYearsToDateF("var_String","25/01/2010",5)

This adds five years to the specified date and returns the result (in the format dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type. When executing this example, the result would be: 25/01/2015

Note: To subtract please use negative values. If you prefer to enter your date as separate values (i.e. dd,mm,yyyy), use the VValueDateAddYearsToDate [1645] command.

VValueDateF

VValueDateF(VarName)

Example:
VValueDateF("var_String")

Returns the current date (format: dd-mm-yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type.

VValueDateTime

VValueDateTime(VarName)

Example:
VValueDateTime("var_String")

Returns the current time and date (format: hh:mm:ss dd/mm/yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type.

VValueDateTimeF

VValueDateTimeF(VarName)

Example:
VValueDateTimeF("var_String")

Returns the current time and date (format: hh-mm-ss_dd-mm-yyyy) to the variable [1900] "var_String" that was defined beforehand with the string type.

VValueRandom

VValueRandom(VarName)

Example:
VValueRandom("var_Double")

Assigns a random value to the variable [1900] "var_Double" that was defined beforehand with the Integer or Double type. A double-type variable ranges between 0 and 1 and an integer-type variable between 0 and 100.

VValueRandomMinMax

VValueRandomMinMax(VarName,Min,Max)

Example:
VValueRandomMinMax("var_Double",10,100)

Assigns a random value between 10 and 100 to the variable [1900] "var_Double" that was defined beforehand with the Integer or Double type.

VValueRound

VValueRound(VarName,DecimalPlaces)

Example:
VValueRound("var_Double",2)

Rounds the double-type variable [1900] "var_Double" to two digits. E.g: 3.1415926 -> 3.14

VValueTime

VValueTime(VarName)

Example:
VValueTime("var_String")

Returns the current time (format: hh:mm:ss) to the variable 1900 "var_String" that was defined beforehand with the string type.

VValueTimeAddHours

VValueTimeAddHours(VarName,Hour)

Example:
VValueTimeAddHours("var_String",3)

Returns the current time (format: hh:mm:ss) plus three hours to the variable 1900 "var_String" that was defined beforehand with the string type.

Note: To subtract please use negative values.

VValueTimeAddMinutes

VValueTimeAddMinutes(VarName,Min)

Example:
VValueTimeAddMinutes("var_String",3)

Returns the current time (format: hh:mm:ss) plus three minutes to the variable 1900 "var_String" that was defined beforehand with the string type.

Note: To subtract please use negative values.

VValueTimeAddSeconds

VValueTimeAddSeconds(VarName,Sec)

Example:
VValueTimeAddSeconds("var_String",10)

Returns the current time (format: hh:mm:ss) plus ten seconds to the variable 1900 "var_String" that was defined beforehand with the string type.

Note: To subtract please use negative values.

VValueTimeF

VValueTimeF(VarName)

Example:
VValueTimeF("var_String")

Returns the current time (format: hh-mm-ss) to the variable 1900 "var_String" that was defined beforehand with the string type.

W W

WakeOnLan

WakeOnLan(MacAddress)

Example:
WakeOnLan("08-00-20-AE-FD-7E")

Turns on or wakes up the computer with the MacAddress 08-00-20-AE-FD-7E.

Please note:
Precondition for using WakeOnLan is that the ethernet adapter of the computer you want to wake up supports WakeOnLan with Magic Packet. Please confirm these settings in the ethernet cards properties in the computers device manager.
If you are using more than one ethernet adapter / connection, consider using the alternative command "WakeOnLanByAdapter [1648]".

WakeOnLanByAdapter

WakeOnLanByAdapter(MacAddress,NicAdapterName)

Example:
WakeOnLanByAdapter("08-00-20-AE-FD-7E","Lan1")

Turns on or wakes up the computer with the MacAddress 08-00-20-AE-FD-7E. The command is sent out via NIC adapter "Lan1" of the local PC.

Please note:
Precondition for using WakeOnLan is that the ethernet adapter of the computer you want to wake up supports WakeOnLan with Magic Packet. Please confirm these settings in the ethernet cards properties in the computers device manager.

WakeOnLanByPort

WakeOnLanByPort(MacAddress,Port)

Example:
WakeOnLanByPort("08-00-20-AE-FD-7E",9)

Turns on or wakes up the computer with the MacAddress 08-00-20-AE-FD-7E. In contrast to the "WakeOnLan" command, this one allows to specify the destination port which is usually 7 or 9. If you are using more than one ethernet adapter / connection, consider using the alternative command "WakeOnLanByAdapter [1648]".

Please note:
Precondition for using WakeOnLan is that the ethernet adapter of the computer you want to wake up supports WakeOnLan with Magic Packet. Please confirm these settings in the ethernet cards properties in the computers device manager.

WDAirScanConnect

WDAirScanConnect(IP(optional))

Example:
WDAirScanConnect("2.0.0.50")

This connects to the AirScan with IP 2.0.0.50.

Example 2:
WDAirScanConnect()

Use this to connect to the AirScan with the IP entered in the AirScan Tool [1277] dialog.

WDAirScanDamping

WDAirScanDamping(Value)

Example:
WDAirScanDamping(0.5)

Changes the value "Damping" inside the AirScan Tool [1277] to 0.5.

WDAirScanDisableTouch

WDAirScanDisableTouch

Example:
WDAirScanDisableTouch

Disables the option inside the AirScan Tool [1277] to send touch points to other controls like the Multi-Touch Panel [899]. Note that there is no check box for that option.

WDAirScanDisconnect

WDAirScanDisconnect

Example:
WDAirScanDisconnect

This disconnects the currently connected AirScan. Use `WDAirScanConnect(IP(optional))` [1648] to re-connect.

WDAirScanEnableTouch

WDAirScanEnableTouch

Example:
WDAirScanEnableTouch

Enables the option inside the AirScan Tool [1277] to send touch points to other controls like the Multi-Touch Panel [899]. Note that there is no check box for that option.

WDAirScanGap

WDAirScanGap(Value)

Example:
WDAirScanGap(20)

Changes the value "Gap" inside the AirScan Tool [1277] to 20.

WDAirScanMaxDelta

WDAirScanMaxDelta(Value)

Example:
WDAirScanMaxDelta(50)

Changes the value "Max Delta" inside the AirScan Tool [1277] to 50.

WDAirScanMotion

WDAirScanMotion(Value)

Example:
WDAirScanMotion(0.5)

Changes the value "Motion" (motion damping) inside the AirScan Tool [1277] to 0.5.

WDAirScanMotionPredictionDisable

WDAirScanMotionPredictionDisable

Example:
WDAirScanMotionPredictionDisable

Disables the option "Motion Prediction" inside the AirScan Tool [1277].

WDAirScanMotionPredictionEnable

WDAirScanMotionPredictionEnable

Example:
WDAirScanMotionPredictionEnable

Enables the option "Motion Prediction" inside the AirScan Tool [1277].

WDAirScanMotionPredictionFactor

WDAirScanMotionPredictionFactor(Value)

Example:
WDAirScanMotionPredictionFactor(0.5)

Changes the value "Motion Prediction" inside the AirScan Tool [1277] to 0.5.

WDAirScanMouseClickDefault

WDAirScanMouseClickDefault

Example:
WDAirScanMouseClickDefault

This command defines when a mouse click (controlled via AirScan) should be executed.
"Default" generates mouse down on enter and mouse up on leave.

Please note:
The option Mouse Click in the AirScan Tool [1277] needs to be enabled!

WDAirScanMouseClickDisabled

WDAirScanMouseClickDisabled

Example:
WDAirScanMouseClickDisabled

With this option the mouse cursor (controlled via the AirScan [1277]) does not generate clicks.

WDAirScanMouseClickEnabled

WDAirScanMouseClickEnabled

Example:
WDAirScanMouseClickEnabled

With this option the mouse cursor (controlled via the AirScan [1277]) generates clicks.

WDAirScanMouseClickOnEnter

WDAirScanMouseClickOnEnter

Example:
WDAirScanMouseClickOnEnter

This command defines when a mouse click (controlled via AirScan) should be executed.
"OnEnter": the click is generated on mouse enter.

Please note:
The option Mouse Click in the AirScan Tool [1277] needs to be enabled!

WDAirScanMouseClickOnLeave

WDAirScanMouseClickOnLeave

Example:
WDAirScanMouseClickOnLeave

This command defines when a mouse click (controlled via AirScan) should be executed.

"OnLeave": the click is generated on mouse leave.

Please note:
The option Mouse Click in the AirScan Tool [1277] needs to be enabled!

WDAirScanMouseDisabled

WDAirScanMouseDisabled

Example:
WDAirScanMouseDisabled

Disables the AirScan [1277] to control the mouse.

WDAirScanMouseDownOnMoveDisabled

WDAirScanMouseDownOnMoveDisabled

Example:
WDAirScanMouseDownOnMoveDisabled

If this option is disabled in the AirScan Tool [1277], left mouse down on move is not active.

WDAirScanMouseDownOnMoveEnabled

WDAirScanMouseDownOnMoveEnabled

Example:
WDAirScanMouseDownOnMoveEnabled

If this option is enabled in the AirScan Tool [1277], left mouse down on move is always active.

WDAirScanMouseEnabled

WDAirScanMouseEnabled

Example:
WDAirScanMouseEnabled

Enables the AirScan [1277] to control the mouse.

WDAirScanMultiPointModeDisabled

WDAirScanMultiPointModeDisabled

Example:
WDAirScanMultiPointModeDisabled

Disables the MultiPointMode in the AirScan Tool [1277], so that the 2-Point-Mode will be used.

WDAirScanMultiPointModeEnabled

WDAirScanMultiPointModeEnabled

Example:
WDAirScanMultiPointModeEnabled

Enables the MultiPointMode in the AirScan Tool [1277].

WDAirScanPointCount

WDAirScanPointCount(PointCount)

Example:
WDAirScanPointCount(12)

Changes the amount of detected touch points inside the AirScan Tool [1277] to 12.

WDAirScanPointIDOffset

WDAirScanPointIDOffset(Value)

Example:
WDAirScanPointIDOffset(2)

Changes the value "Point ID Offset" inside the AirScan Tool [1277] to 2.

WDAirScanStart

WDAirScanStart(Value)

Example:
WDAirScanStart(50)

Changes the value "Start" inside the AirScan Tool [1277] to 50.

WDAirScanStop

WDAirScanStop(Value)

Example:
WDAirScanStop(250)

Changes the value "Stop" inside the AirScan Tool [1277] to 250.

WDAirScanTrim

WDAirScanTrim(Value)

Example:
WDAirScanTrim(25)

Changes the value "Trim" inside the AirScan Tool [1277] to 25.

WDAirScanTUIODisable

WDAirScanTUIODisable

Example:
WDAirScanTUIODisable

Disables the option "TUIO" inside the AirScan Tool [1277].

WDAirScanTUIOEnable

WDAirScanTUIOEnable

Example:
WDAirScanTUIOEnable

Changes the amount of detected touch points inside the AirScan Tool [1277] to 12.

WDAirScanVCount

WDAirScanVCount(Value)

Example:
WDAirScanVCount(5)

Changes the value "VCount" inside the AirScan Tool [1277] to 5.

WDAirScanXOffset

WDAirScanXOffset(Value)

Example:
WDAirScanXOffset(1025)

Changes the value "X Offset" inside the AirScan Tool [1277] to 50.

WDAirScanYOffset

WDAirScanYOffset(Value)

Example:
WDAirScanYOffset(769)

Changes the value "Y Offset" inside the AirScan Tool [1277] to 50.

WDAnalogClockCssStyleDisable

WDAnalogClockCssStyleDisable(ID,StyleID)

Example:
WDAnalogClockCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the AnalogClock [819] with ID 5.

WDAnalogClockCssStyleEdit

WDAnalogClockCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDAnalogClockCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the AnalogClock [819] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDAnalogClockCssStyleEnable

WDAnalogClockCssStyleEnable(ID,StyleID)

Example:
WDAnalogClockCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the AnalogClock [819] with ID 5.

WDAnalogClockFix

WDAnalogClockFix(ID)

Example:
WDAnalogClockFix(5)

This activates the option "Fix" in the Item Properties of the AnalogClock [819] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDAnalogClockGetCaptionFontFamily

WDAnalogClockGetCaptionFontFamily(ID)

Example:
varString = WDAnalogClockGetCaptionFontFamily(5)

This refers to the AnalogClock [819] with ID 5 and returns the caption's current font family name as a string. The "Caption" is the optional text that appears in the center of the clock.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another AnalogClock via a script like:
`WDAnalogClockSetCaptionFontFamily(5,WDAnalogClockGetCaptionFontFamily(1))`
or simply display it in the Debug Logger [812]:
`DebugMessage(WDAnalogClockGetCaptionFontFamily(5))`

WDAnalogClockGetCaptionFontSize

WDAnalogClockGetCaptionFontSize(ID)

Example:
varDouble = WDAnalogClockGetCaptionFontSize(5)

This refers to the AnalogClock [819] with ID 5 and returns the caption's current font size in "pt" as a double. The "Caption" is the optional text that appears in the center of the clock.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another AnalogClock via a script like:
`WDAnalogClockSetCaptionFontSize(5,WDAnalogClockGetCaptionFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDAnalogClockGetCaptionFontSize(5))`

WDAnalogClockGetLabelFontFamily

WDAnalogClockGetLabelFontFamily(ID)

Example:
varString = WDAnalogClockGetLabelFontFamily(5)

This refers to the AnalogClock [819] with ID 5 and returns the current font family name from the labels as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another AnalogClock via a script like:
`WDAnalogClockSetLabelFontFamily(5,WDAnalogClockGetLabelFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDAnalogClockGetLabelFontFamily(5))`

WDAnalogClockGetLabelFontSize

WDAnalogClockGetLabelFontSize(ID)

Example:
varDouble = WDAnalogClockGetLabelFontSize(5)

This refers to the AnalogClock [819] with ID 5 and returns the current font size from the labels in "pt" as a double. The labels refer only to the numbers 1-12, whilst "Caption" refers to the optional text appearing in the center of the clock.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another AnalogClock via a script like:
`WDAnalogClockSetLabelFontSize(5,WDAnalogClockGetLabelFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDAnalogClockGetLabelFontSize(5))`

WDAnalogClockGetLocationLeft

WDAnalogClockGetLocationLeft(ID)

Example:
WDAnalogClockGetLocationLeft(5)

This refers to the AnalogClock [819] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDAnalogClockGetLocationLeft(1)`

WDAnalogClockGetLocationTop

WDAnalogClockGetLocationTop(ID)

Example:
WDAnalogClockGetLocationTop(5)

This refers to the AnalogClock [819] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDAnalogClockGetLocationTop(1)`

WDAnalogClockGetSizeHeight

WDAnalogClockGetSizeHeight(ID)

Example:
WDAnalogClockGetSizeHeight(5)

This refers to the AnalogClock [819] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDAnalogClockGetSizeHeight(1)`

WDAnalogClockGetSizeWidth

WDAnalogClockGetSizeWidth(ID)

Example:
WDAnalogClockGetSizeWidth(5)

This refers to the AnalogClock [819] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDAnalogClockGetSizeWidth(1)`

WDAnalogClockLocation

WDAnalogClockLocation(ID,X,Y)

Example:
WDAnalogClockLocation(5,100,200)

Sets the position of the AnalogClock [819] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDAnalogClockLocationLeft

WDAnalogClockLocationLeft(ID,X)

Example:
WDAnalogClockLocationLeft(5,100)

Sets the position of the AnalogClock [819] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDAnalogClockLocationTop

WDAnalogClockLocationTop(ID,Y)

Example:
WDAnalogClockLocationTop(5,200)

Sets the position of the AnalogClock [819] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDAnalogClockSetCaptionFontFamily

WDAnalogClockSetCaptionFontFamily(ID,FontName)

Example:
WDAnalogClockSetCaptionFontFamily(5,"Arial")

This refers to the AnalogClock [819] with ID 5 and sets the caption's font family to "Arial". The "Caption" is the optional text that appears in the center of the clock.
If you like to copy and paste the font from a Widget, e.g. another AnalogClock, you can use a script like:
`WDAnalogClockSetCaptionFontFamily(5,WDAnalogClockGetCaptionFontFamily(1))`

WDAnalogClockSetCaptionFontSize

WDAnalogClockSetCaptionFontSize(ID,FontSize)

Example:
WDAnalogClockSetCaptionFontSize(5,20)

This refers to the AnalogClock [819] with ID 5 and sets the caption's font size to 20pt. The "Caption" is the optional text that appears in the center of the clock.
If you like to copy and paste the font size from a Widget, e.g. another AnalogClock, you can use a script like:
`WDAnalogClockSetCaptionFontSize(5,WDAnalogClockGetCaptionFontSize(1))`

WDAnalogClockSetLabelFontFamily

WDAnalogClockSetLabelFontFamily(ID,FontName)

Example:
WDAnalogClockSetLabelFontFamily(5,"Arial")

This refers to the AnalogClock [819] with ID 5 and sets the font family for the labels to "Arial". The labels refer only to the numbers 1-12, whilst "Caption" refers to the optional text appearing in the center of the clock.

If you like to copy and paste the font from a Widget, e.g. another AnalogClock, you can use a script like:
`WDAnalogClockSetLabelFontFamily(5,WDAnalogClockGetLabelFontFamily(1))`

WDAnalogClockSetLabelFontSize

WDAnalogClockSetLabelFontSize(ID,FontSize)

Example:
WDAnalogClockSetLabelFontSize(5,20)

This refers to the AnalogClock [819] with ID 5 and sets the font size for the labels to 20pt. The labels refer only to the numbers 1-12, whilst "Caption" refers to the optional text appearing in the center of the clock.
If you like to copy and paste the font size from a Widget, e.g. another AnalogClock, you can use a script like:
`WDAnalogClockSetLabelFontSize(5,WDAnalogClockGetLabelFontSize(1))`

WDAnalogClockSize

WDAnalogClockSize(ID,Width,Height)

Example:
WDAnalogClockSize(5,100,40)

Sets the size of the AnalogClock [819] with ID 5 to a width of 100px and a height of 40px.

WDAnalogClockSizeHeight

WDAnalogClockSizeHeight(ID,Height)

Example:
WDAnalogClockSizeHeight(5,40)

Sets the size of the AnalogClock [819] with ID 5 to a height of 40px but remains the current width.

WDAnalogClockSizeWidth

WDAnalogClockSizeWidth(ID,Width)

Example:
WDAnalogClockSizeWidth(5,100)

Sets the size of the AnalogClock [819] with ID 5 to a width of 100px but remains the current height.

WDAnalogClockUnfix

WDAnalogClockUnfix(ID)

Example:
WDAnalogClockUnfix(5)

This deactivates the option "Fix" in the Item Properties of the AnalogClock [819] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDArtNetRecorderAddTextFromLabel

WDArtNetRecorderAddTextFromLabel(ID,LabelID)

Example:
WDArtNetRecorderAddTextFromLabel(6,3)

Adds the text from Label 3 to the current Label of the Art-Net Recorder [841] button with ID 6.

WDArtNetRecorderClick

WDArtNetRecorderClick(ID)

Example:
WDArtNetRecorderClick(5)

Clicks the Art-Net Recorder [841] button with ID 5.

If you like to execute this command addressing many buttons, the chapter "For loop" and "Project and Context Member" show some interesting examples.

WDArtnetRecorderClickImage

WDArtnetRecorderClickImage(ID,File)

Example:
WDArtnetRecorderClickImage(1,"C:\Christie\content\buttonImage10.jpg")

Changes the look of the Art-Net Recorder [841] Button with ID 1 - it loads the image "buttonImage10.jpg" that was saved under "C:\Christie\content" and displays it when the Art-Net Recorder is in the mode "Click".

WDArtNetRecorderClickImageResource

WDArtNetRecorderClickImageResource(ID,ResourceName)

Example:
WDArtNetRecorderClickImageResource(5,"Default\Button\Lock")

This sets the image for the clicked status of the Art-Net Recorder [841] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDArtNetRecorderCssStyleDisable

WDArtNetRecorderCssStyleDisable(ID,StyleID)

Example:
WDArtNetRecorderCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Art-Net Recorder [841] button with ID 5.

WDArtNetRecorderCssStyleEdit

WDArtNetRecorderCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDArtNetRecorderCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Art-Net Recorder [841] button with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDArtNetRecorderCssStyleEnable

WDArtNetRecorderCssStyleEnable(ID,StyleID)

Example:
WDArtNetRecorderCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Art-Net Recorder [841] button with ID 5.

WDArtNetRecorderFix

WDArtNetRecorderFix(ID)

Example:
WDArtNetRecorderFix(5)

This activates the option "Fix" in the Item Properties of the Art-Net Recorder [841] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDArtNetRecorderGetLocationLeft

WDArtNetRecorderGetLocationLeft(ID)

Example:
WDArtNetRecorderGetLocationLeft(5)

This refers to the Art-Net Recorder [841] button with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetRecorderGetLocationLeft(1)`

WDArtNetRecorderGetLocationTop

WDArtNetRecorderGetLocationTop(ID)

Example:
WDArtNetRecorderGetLocationTop(5)

This refers to the Art-Net Recorder [841] button with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetRecorderGetLocationTop(1)`

WDArtNetRecorderGetSizeHeight

WDArtNetRecorderGetSizeHeight(ID)

Example:
WDArtNetRecorderGetSizeHeight(5)

This refers to the Art-Net Recorder [841] button with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetRecorderGetSizeHeight(1)`

WDArtNetRecorderGetSizeWidth

WDArtNetRecorderGetSizeWidth(ID)

Example:
WDArtNetRecorderGetSizeWidth(5)

This refers to the Art-Net Recorder [841] button with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetRecorderGetSizeWidth(1)`

WDArtNetRecorderGotoFrame

WDArtNetRecorderGotoFrame(ID,Frame)

Example:
WDArtNetRecorderGotoFrame(1,330)

Sets the Art-Net Recorder [841] Button 1 to the frame 330 of the last recorded Art-Net Sequence.

WDArtNetRecorderHighlightImage

WDArtNetRecorderHighlightImage(ID,Existing file)

Example:
WDArtNetRecorderHighlightImage(5,"C:\Christie\clicked.png")

This sets the image for the highlighted status of the Art-Net Recorder [841] button with ID 5 to the image saved under the specified path.

WDArtNetRecorderHighlightImageResource

WDArtNetRecorderHighlightImageResource(ID,Text)

Example:
WDArtNetRecorderHighlightImageResource(5,"Default\Button\Lock")

This sets the image for the highlighted status of the Art-Net Recorder [841] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDArtNetRecorderLabel

WDArtNetRecorderLabel(ID,Text)

Example:
WDArtNetRecorderLabel(5,"Show")

Labels the Art-Net Recorder [841] button with ID 5 with the word "Show".

WDArtNetRecorderLabelColor

WDArtNetRecorderLabelColor(ID,R,G,B)

Example:
WDArtNetRecorderLabelColor(5,255,90,0)

Changes the label color of the Art-Net Recorder [841] button with ID 5 to orange (Red=255, Green=90, Blue=0). The values for R,G,B range from 0 to 255.

WDArtNetRecorderLocation

WDArtNetRecorderLocation(ID,X,Y)

Example:
WDArtNetRecorderLocation(5,100,200)

Sets the position of the Art-Net Recorder [841] button with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDArtNetRecorderLocationLeft

WDArtNetRecorderLocationLeft(ID,X)

Example:
WDArtNetRecorderLocationLeft(5,100)

Sets the position of the Art-Net Recorder [841] button with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDArtNetRecorderLocationTop

WDArtNetRecorderLocationTop(ID,Y)

Example:
WDArtNetRecorderLocationTop(5,200)

Sets the position of the Art-Net Recorder [841] button with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDArtNetRecorderLoop

WDArtNetRecorderLoop(ID)

Example:
WDArtNetRecorderLoop(2)

Sets the playmode of the Art-Net sequence recorded with Art-Net Recorder [841] Button 2 to Loop.

WDArtNetRecorderPause

WDArtNetRecorderPause(ID)

Example:
WDArtNetRecorderPause(2)

Pauses the playing Art-Net sequence recorded with Art-Net Recorder [841] Button 2.

WDArtNetRecorderPlay

WDArtNetRecorderPlay(ID)

Example:
WDArtNetRecorderPlay(2)

Plays the Art-Net sequence recorded with Art-Net Recorder [841] Button 2 from it's current time.

WDArtNetRecorderPlayOnce

WDArtNetRecorderPlayOnce(ID)

Example:
WDArtNetRecorderPlayOnce(2)

Sets the playmode of the Art-Net sequence recorded with Art-Net Recorder [841] Button 2 to PlayOnce.

WDArtNetRecorderReleaseImage

WDArtNetRecorderReleaseImage(ID,Existing file)

Example:
WDArtNetRecorderReleaseImage(5,"C:\Christie\clicked.png")

This sets the image for the released status of the Art-Net Recorder [841] button with ID 5 to the image saved under the specified path.

WDArtNetRecorderReleaseImageResource

WDArtNetRecorderReleaseImageResource(ID,Text)

Example:
WDArtNetRecorderReleaseImageResource(5,"Default\Button\Lock")

This sets the image for the released status of the Art-Net Recorder [841] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDArtNetRecorderRewind

WDArtNetRecorderRewind(ID)

Example:
WDArtNetRecorderRewind(2)

Rewinds the Art-Net sequence recorded with Art-Net Recorder [841] Button 2 to it's inpoint. When the sequence is currently playing, it won't be stopped but starts from it's beginning again.

WDArtNetRecorderSize

WDArtNetRecorderSize(ID,Width,Height)

Example:
WDArtNetRecorderSize(5,100,40)

Sets the size of the Art-Net Recorder [841] button with ID 5 to a width of 100px and a height of 40px.

WDArtNetRecorderSizeHeight

WDArtNetRecorderSizeHeight(ID,Height)

Example:
WDArtNetRecorderSizeHeight(5,40)

Sets the size of the Art-Net Recorder [841] button with ID 5 to a height of 40px but remains the current width.

WDArtNetRecorderSizeWidth

WDArtNetRecorderSizeWidth(ID,Width)

Example:
WDArtNetRecorderSizeWidth(5,100)

Sets the size of the Art-Net Recorder [841] button with ID 5 to a width of 100px but remains the current height.

WDArtNetRecorderStart

WDArtNetRecorderStart(ID)

Example:
WDArtNetRecorderStart(1)

Starts the recording of Art-Net Recorder [841] Button 1.

WDArtNetRecorderStop

WDArtNetRecorderStop(ID)

Example:
WDArtNetRecorderStop(1)

Stops the recording of Art-Net Recorder [841] Button 1.

WDArtNetRecorderTextFromLabel

WDArtNetRecorderTextFromLabel(ID,LabelID)

Example:
WDArtNetRecorderTextFromLabel(6,3)

Labels the Art-Net Recorder [841] button with ID 6 with the text from Label 3.

WDArtNetRecorderTint

WDArtNetRecorderTint(ID,R,G,B)

Example:
WDArtNetRecorderTint(5,255,180,0)

Changes the button color of the Art-Net Recorder [841] button with ID 5 to orange (Red=255, Green=180, Blue=0). The values for R,G,B range from 0 to 255.

WDArtNetRecorderUnfix

WDArtNetRecorderUnfix(ID)

Example:
WDArtNetRecorderUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Art-Net Recorder [841] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDArtNetSnapshotActivate

WDArtNetSnapshotActivate(ID)

Example:
WDArtNetSnapshotActivate(1)

Executes the Art-Net Snapshot [839] Button 1. This command does stop current fades of other running Art-Net snapshots beforehand. This is an advantage when using multiple buttons in the *same* universe. Art-Net always transmitts the entire universe, not only programmed channels (not specified channels=0). If current fades are not stopped, all channels would bounce between the fade value and 0.
When multiple snapshots apply to *different* universes, this command has the disadvantage to stop a channel's fade in between without setting it to a new value. In that scenario, please use the command `WDArtNetSnapshotActivateAllFade,ID`.

WDArtNetSnapshotActivateAllFade

WDArtNetSnapshotActivateAllFade(ID)

Example:
WDArtNetSnapshotActivateAllFade(1)

Executes the Art-Net Snapshot [839] Button 1. This command does not stop current fades of other running Art-Net snapshots beforehand. This is an advantage when using multiple buttons in *different* universes.
When multiple buttons apply to the *same* universe, please use the command `WDArtNetSnapshotActivate,ID` as this stops all fades beforehand.

WDArtNetSnapshotAddTextFromLabel

WDArtNetSnapshotAddTextFromLabel(ID,LabelID)

Example:
WDArtNetSnapshotAddTextFromLabel(6,3)

Adds the text from Label 3 to the current Label of the Art-Net Snapshot [839] button with ID 6.

WDArtNetSnapshotCapture

WDArtNetSnapshotCapture(ID)

Example:
WDArtNetSnapshotCapture(1)

Art-Net Snapshot [839] Button 1 captures the current Art-Net status of the Art-Net Subnet and Universe set up in the Button Properties and overwrites the Art-Net values stored to this button before. If you like to merge the values (HTP), use the command `WDArtNetSnapshotHTPMerge(ID,Subnet,Universe)`

WDArtNetSnapshotClear

WDArtNetSnapshotClear(ID)

Example:
WDArtNetSnapshotClear(1)

This clears the programmed channel(s) from Art-Net Snapshot [839] Button 1. If the button is then clicked, all values for the specified universe would fade to 0.

WDArtNetSnapshotClick

WDArtNetSnapshotClick(ID)

Example:
WDArtNetSnapshotClick(5)

Clicks the Art-Net Snapshot [839] button with ID 5.

If you like to execute this command addressing many buttons, the chapter "For loop" and "Project and Context Member" show some interesting examples.

WDArtnetSnapshotClickImage

WDArtnetSnapshotClickImage(ID,File)

Example:
WDArtnetSnapshotClickImage(1,"C:\Christie\content\buttonImage10.jpg")

Changes the look of the Art-Net Snapshot [839] Button with ID 1 - it loads the image "buttonImage10.jpg" that was saved under "C:\Christie\content" and displays it when the Art-Net Snapshot is in the mode "Click".

WDArtNetSnapshotClickImageResource

WDArtNetSnapshotClickImageResource(ID,ResourceName)

Example:
WDArtNetSnapshotClickImageResource(5,"Default\Button\Lock")

This sets the image for the clicked status of the Art-Net Snapshot [839] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDArtNetSnapshotCssStyleDisable

WDArtNetSnapshotCssStyleDisable(ID,StyleID)

Example:
WDArtNetSnapshotCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Art-Net Snapshot [839] button with ID 5.

WDArtNetSnapshotCssStyleEdit

WDArtNetSnapshotCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDArtNetSnapshotCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Art-Net Snapshot [839] button with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDArtNetSnapshotCssStyleEnable

WDArtNetSnapshotCssStyleEnable(ID,StyleID)

Example:
WDArtNetSnapshotCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Art-Net Snapshot [839] button with ID 5.

WDArtNetSnapshotCut

WDArtNetSnapshotCut(ID)

Example:
WDArtNetSnapshotCut(1)

This applies the stored values of Art-Net Snapshot [839] Button 1 directly without fading.

WDArtNetSnapshotFix

WDArtNetSnapshotFix(ID)

Example:
WDArtNetSnapshotFix(5)

This activates the option "Fix" in the Item Properties of the Art-Net Snapshot [839] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDArtNetSnapshotGetLocationLeft

WDArtNetSnapshotGetLocationLeft(ID)

Example:
WDArtNetSnapshotGetLocationLeft(5)

This refers to the Art-Net Snapshot [839] button with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetSnapshotGetLocationLeft(1)`

WDArtNetSnapshotGetLocationTop

WDArtNetSnapshotGetLocationTop(ID)

Example:
WDArtNetSnapshotGetLocationTop(5)

This refers to the Art-Net Snapshot [839] button with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetSnapshotGetLocationTop(1)`

WDArtNetSnapshotGetSizeHeight

WDArtNetSnapshotGetSizeHeight(ID)

Example:
WDArtNetSnapshotGetSizeHeight(5)

This refers to the Art-Net Snapshot [839] button with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetSnapshotGetSizeHeight(1)`

WDArtNetSnapshotGetSizeWidth

WDArtNetSnapshotGetSizeWidth(ID)

Example:
WDArtNetSnapshotGetSizeWidth(5)

This refers to the Art-Net Snapshot [839] button with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDArtNetSnapshotGetSizeWidth(1)`

WDArtNetSnapshotHighlightImage

WDArtNetSnapshotHighlightImage(ID,Existing file)

Example:
WDArtNetSnapshotHighlightImage(5,"C:\Christie\clicked.png")

This sets the image for the highlighted status of the Art-Net Snapshot [839] button with ID 5 to the image saved under the specified path.

WDArtNetSnapshotHighlightImageResource

WDArtNetSnapshotHighlightImageResource(ID,Text)

Example:
WDArtNetSnapshotHighlightImageResource(5,"Default\Button\Lock")

This sets the image for the highlighted status of the Art-Net Snapshot [839] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDArtNetSnapshotHTPMerge

WDArtNetSnapshotHTPMerge(ID,Subnet,Universe)

Example:
WDArtNetSnapshotHTPMerge(1,2,3)

Art-Net Snapshot [839] Button 1 captures the current Art-Net status of the Art-Net Subnet 2 and Universe 3 and merges the status with the status stored to this button before. The merge uses the method HTP: highest takes precedence, meaning that the largest value is stored. If you like to overwrite the values (i.e. LTP), use the command `WDArtNetSnapshotCapture(ID)`.

WDArtNetSnapshotLabel

WDArtNetSnapshotLabel(ID,Text)

Example:
WDArtNetSnapshotLabel(5,"Show")

Labels the Art-Net Snapshot [839] button with ID 5 with the word "Show".

WDArtNetSnapshotLabelColor

WDArtNetSnapshotLabelColor(ID,R,G,B)

Example:
WDArtNetSnapshotLabelColor(5,255,90,0)

Changes the label color of the Art-Net Snapshot [839] button with ID 5 to orange (Red=255, Green=90, Blue=0). The values for R,G,B range from 0 to 255.

WDArtNetSnapshotLocation

WDArtNetSnapshotLocation(ID,X,Y)

Example:
WDArtNetSnapshotLocation(5,100,200)

Sets the position of the Art-Net Snapshot [839] button with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDArtNetSnapshotLocationLeft

WDArtNetSnapshotLocationLeft(ID,X)

Example:
WDArtNetSnapshotLocationLeft(5,100)

Sets the position of the Art-Net Snapshot [839] button with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDArtNetSnapshotLocationTop

WDArtNetSnapshotLocationTop(ID,Y)

Example:
WDArtNetSnapshotLocationTop(5,200)

Sets the position of the Art-Net Snapshot [839] button with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDArtNetSnapshotReleaseImage

WDArtNetSnapshotReleaseImage(ID,Existing file)

Example:
WDArtNetSnapshotReleaseImage(5,"C:\Christie\clicked.png")

This sets the image for the released status of the Art-Net Snapshot [839] button with ID 5 to the image saved under the specified path.

WDArtNetSnapshotReleaseImageResource

WDArtNetSnapshotReleaseImageResource(ID,Text)

Example:
WDArtNetSnapshotReleaseImageResource(5,"Default\Button\Lock")

This sets the image for the released status of the Art-Net Snapshot [839] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDArtNetSnapshotSet16Bit

WDArtNetSnapshotSet16Bit(ID,Ch,Val)

Example:
WDArtNetSnapshotSet16Bit(1,50,255)

This command allows to change the stored value of the 16Bit Art-Net channel 50 to the value 255 in the Art-Net Snapshot [839] Button 1

WDArtNetSnapshotSet8Bit

WDArtNetSnapshotSet8Bit(ID,Ch,Val)

Example:
WDArtNetSnapshotSet8Bit(1,50,255)

This command allows to change the stored value of the 8Bit Art-Net channel 50 to the value 255 in the Art-Net Snapshot [839] Button 1.

WDArtNetSnapshotSetInputUniverse

WDArtNetSnapshotSetInputUniverse(ID,Subnet,Universe)

Example:
WDArtNetSnapshotSetInputUniverse(5,2,3)

Changes the input settings for the Art-Net Snapshot [839] button with ID 5 so that the subnet is 2 and the universe 3.

WDArtNetSnapshotSetOutputUniverse

WDArtNetSnapshotSetOutputUniverse(ID,Subnet,Universe)

Example:
WDArtNetSnapshotSetInputUniverse(5,6,7)

Changes the output settings for the Art-Net Snapshot [839] button with ID 5 so that the subnet is 6 and the universe 7.

WDArtNetSnapshotSize

WDArtNetSnapshotSize(ID,Width,Height)

Example:
WDArtNetSnapshotSize(5,100,40)

Sets the size of the Art-Net Snapshot [839] button with ID 5 to a width of 100px and a height of 40px.

WDArtNetSnapshotSizeHeight

WDArtNetSnapshotSizeHeight(ID,Height)

Example:
WDArtNetSnapshotSizeHeight(5,40)

Sets the size of the Art-Net Snapshot [839] button with ID 5 to a height of 40px but remains the current width.

WDArtNetSnapshotSizeWidth

WDArtNetSnapshotSizeWidth(ID,Width)

Example:
WDArtNetSnapshotSizeWidth(5,100)

Sets the size of the Art-Net Snapshot [839] button with ID 5 to a width of 100px but remains the current height.

WDArtNetSnapshotStopAllFades

WDArtNetSnapshotStopAllFades

Example:
WDArtNetSnapshotStopAllFades

This stops all fades executed through currently running Art-Net Snapshot [839] Buttons. The programmed channel(s) simply stop at their current value.

WDArtNetSnapshotStopFade

WDArtNetSnapshotStopFade(ID)

Example:
WDArtNetSnapshotStopFade(1)

This stops the fade executed from Art-Net Snapshot [839] Button 1. The programmed channel(s) simply stop at their current value.

WDArtNetSnapshotTextFromLabel

WDArtNetSnapshotTextFromLabel(ID,LabelID)

Example:
WDArtNetSnapshotTextFromLabel(6,3)

Labels the Art-Net Snapshot [839] button with ID 6 with the text from Label 3.

WDArtNetSnapshotTint

WDArtNetSnapshotTint(ID,R,G,B)

Example:
WDArtNetSnapshotTint(5,255,180,0)

Changes the button color of the Art-Net Snapshot [839] button with ID 5 to orange (Red=255, Green=180, Blue=0). The values for R,G,B range from 0 to 255.

WDArtNetSnapshotUnfix

WDArtNetSnapshotUnfix(ID)

Example:
WDArtNetSnapshotUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Art-Net Snapshot [839] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDBackupPBBackupMode

WDBackupPBBackupMode

Example:
WDBackupPBBackupMode

This sets the Pandoras Box master, which is connected as the Backup Connection in Widget Designer from Live Mode to Backup Mode.
The IP address and Domain for the "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256]. See also the topic "Backup [123]" in Pandoras Box for more information.

To reverse this action, execute the command `WDBackupPBTakeoverAllClients` [1672]. The respective command for the Pandoras Box master connection would be `WDMasterPBBackupMode` [1762]

WDBackupPBTakeoverAllClients

WDBackupPBTakeoverAllClients

Example:
WDBackupPBTakeoverAllClients

This sets the Pandoras Box master, which is connected as the Backup Connection in Widget Designer from Backup Mode to Live Mode and takes over all clients which are in the same domain.
The IP address and Domain for the "Backup Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256]. See also the topic "Backup [123]" in Pandoras Box for more information.

To reverse this action, execute the command `WDBackupPBBackupMode` [1672]. The respective command for the Pandoras Box master connection would be `WDMasterPBTakeoverAllClients` [1762]

WDCameraPointTrackerBlur

WDCameraPointTrackerBlur(Value)

Example:
WDCameraPointTrackerBlur(2)

This sets the option "Blur" in the section "Image Processing" in the Camera Tracker[1291] dialog to the value 2.

WDCameraPointTrackerColor

WDCameraPointTrackerColor(R,G,B)

Example:
WDCameraPointTrackerColor(0,255,0)

This sets the color for Color Tracking in the section "Image Processing" in the Camera Tracker[1291] dialog to pure green. Note that the option "Color Tracking" needs to be enabled too, e.g. via the command `WDCameraPointTrackerEnableColorTracking`.

WDCameraPointTrackerDamping

WDCameraPointTrackerDamping(Value)

Example:
WDCameraPointTrackerDamping(0.25)

This sets the option "Damping" in the section "Point Tracker" in the Camera Tracker[1291] dialog to the value 0.25.

WDCameraPointTrackerDisableAverageColor

WDCameraPointTrackerDisableAverageColor

Example:
WDCameraPointTrackerDisableAverageColor

This disables the option "Output Average Color" in the section "Color Processing" in the Camera Tracker[1291] dialog.

WDCameraPointTrackerDisableColorTracking

WDCameraPointTrackerDisableColorTracking

Example:
WDCameraPointTrackerDisableColorTracking

This enables the option "Color Tracking" in the section "Image Processing" in the Camera Tracker[1291] dialog.

WDCameraPointTrackerDisableInvert

WDCameraPointTrackerDisableInvert

Example:
WDCameraPointTrackerDisableInvert

This disables the option "Invert" in the section "Image Processing" in the Camera Tracker[1291] dialog.

WDCameraPointTrackerDisableLookup

WDCameraPointTrackerDisableLookup

Example:
WDCameraPointTrackerDisableLookup

This disables the option "Use Lookup" in the section "Image Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerDisableMask

WDCameraPointTrackerDisableMask

Example:
WDCameraPointTrackerDisableMask

This disables the option "Use Mask" in the section "Image Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerDisableMotionPrediction

WDCameraPointTrackerDisableMotionPrediction

Example:
WDCameraPointTrackerDisableMotionPrediction

This disables the option "Motion Prediction" in the section "Point Tracker" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerDisableMuteUI

WDCameraPointTrackerDisableMuteUI

Example:
WDCameraPointTrackerDisableMuteUI

This disables the option "Mute UI" in the general section in the Camera Tracker [1291] dialog.

WDCameraPointTrackerDisableProcessing

WDCameraPointTrackerDisableProcessing

Example:
WDCameraPointTrackerDisableProcessing

This disables the processing from the Camera Tracker. To enable it again, please use the command `WDCameraPointTrackerEnableProcessing`. Note that there is no option to en-, or disable processing within the Camera Tracker dialog. If you are interested to save performance of the Camera Tracker, you can also use the option "Mute UI" (or the command `WDCameraPointTrackerEnableMuteUI)` which stops the rendering of the camera image in the dialog but not the data processing.

WDCameraPointTrackerEnableAverageColor

WDCameraPointTrackerEnableAverageColor

Example:
WDCameraPointTrackerEnableAverageColor

This enables the option "Output Average Color" in the section "Color Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerEnableColorTracking

WDCameraPointTrackerEnableColorTracking

Example:
WDCameraPointTrackerEnableColorTracking

This enables the option "Color Tracking" in the section "Image Processing" in the Camera Tracker [1291] dialog.
Note that the color for "Color Tracking" can be set too, e.g. via the command `WDCameraPointTrackerColor`.

WDCameraPointTrackerEnableInvert

WDCameraPointTrackerEnableInvert

Example:
WDCameraPointTrackerEnableInvert

This enables the option "Invert" in the section "Image Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerEnableLookup

WDCameraPointTrackerEnableLookup

Example:
WDCameraPointTrackerEnableLookup

This enables the option "Use Lookup" in the section "Image Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerEnableMask

WDCameraPointTrackerEnableMask

Example:
WDCameraPointTrackerEnableMask

This enables the option "Use Mask" in the section "Image Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerEnableMotionPrediction

WDCameraPointTrackerEnableMotionPrediction

Example:
WDCameraPointTrackerEnableMotionPrediction

This enables the option "Motion Prediction" in the section "Point Tracker" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerEnableMuteUI

WDCameraPointTrackerEnableMuteUI

Example:
WDCameraPointTrackerEnableMuteUI

This enables the option "Mute UI" in the general section in the Camera Tracker [1291] dialog.
If you are interested to save performance of the Camera Tracker, you can also use the command

`WDCameraPointTrackerDisableProcessing` which stops not only the rendering but the entire data processing.

WDCameraPointTrackerEnableProcessing

WDCameraPointTrackerEnableProcessing

Example:
WDCameraPointTrackerEnableProcessing

This enables the processing from the Camera Tracker, if it was disabled before with the command `WDCameraPointTrackerDisableProcessing`. Note that there is no option to en-, or disable processing within the Camera Tracker dialog.

WDCameraPointTrackerIDStart

WDCameraPointTrackerIDStart(Value)

Example:
WDCameraPointTrackerIDStart(2)

This sets the option "ID Start" in the general section in the Camera Tracker [1291] dialog to the value 2.

WDCameraPointTrackerInflate

WDCameraPointTrackerInflate(Value)

Example:
WDCameraPointTrackerInflate(10)

This sets the option "Inflate" in the section "Image Processing" in the Camera Tracker [1291] dialog to the value 10.

WDCameraPointTrackerMaxDelta

WDCameraPointTrackerMaxDelta(Value)

Example:
WDCameraPointTrackerMaxDelta(50)

This sets the option "Max Delta" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value 50.

WDCameraPointTrackerMaxHeight

WDCameraPointTrackerMaxHeight(Value)

Example:
WDCameraPointTrackerMaxHeight(50)

This sets the option "Max Height" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value 50.

WDCameraPointTrackerMaxWidth

WDCameraPointTrackerMaxWidth(Value)

Example:
WDCameraPointTrackerMaxWidth(50)

This sets the option "Max Width" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value 50.

WDCameraPointTrackerMinHeight

WDCameraPointTrackerMinHeight(Value)

Example:
WDCameraPointTrackerMinHeight(10)

This sets the option "Min Height" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value 10.

WDCameraPointTrackerMinWidth

WDCameraPointTrackerMinWidth(Value)

Example:
WDCameraPointTrackerMinWidth(10)

This sets the option "Min Width" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value 10.

WDCameraPointTrackerMotionFactor

WDCameraPointTrackerMotionFactor(Value)

Example:
WDCameraPointTrackerMotionFactor(0.25)

This sets the option "Factor" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value 0.25.

WDCameraPointTrackerMotionModeBottom

WDCameraPointTrackerMotionModeBottom

Example:
WDCameraPointTrackerMotionModeBottom

This sets the option "Mode" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value "Bottom".

WDCameraPointTrackerMotionModeCenter

WDCameraPointTrackerMotionModeCenter

Example:
WDCameraPointTrackerMotionModeCenter

This sets the option "Mode" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value "Center".

WDCameraPointTrackerMotionModeTop

WDCameraPointTrackerMotionModeTop

Example:
WDCameraPointTrackerMotionModeTop

This sets the option "Mode" in the section "Point Tracker" in the Camera Tracker [1291] dialog to the value "Top".

WDCameraPointTrackerPoints

WDCameraPointTrackerPoints(Value)

Example:
WDCameraPointTrackerPoints(5)

This sets the option "ID Start" in the general section in the Camera Tracker [1291] dialog to the value 2.

WDCameraPointTrackerShrink

WDCameraPointTrackerShrink(Value)

Example:
WDCameraPointTrackerShrink(2)

This sets the option "Shrink" in the section "Image Processing" in the Camera Tracker [1291] dialog to the value 2.

WDCameraPointTrackerThreshold

WDCameraPointTrackerThreshold(Value)

Example:
WDCameraPointTrackerThreshold(50)

This sets the option "Threshold" in the section "Image Processing" in the Camera Tracker [1291] dialog to the value 50.

WDCameraPointTrackerTouchOutDisableInvX

WDCameraPointTrackerTouchOutDisableInvX

Example:
WDCameraPointTrackerTouchOutDisableInvX

This disables the option "Inv X" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutDisableInvY

WDCameraPointTrackerTouchOutDisableInvY

Example:
WDCameraPointTrackerTouchOutDisableInvY

This disables the option "Inv Y" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutDisableSwap

WDCameraPointTrackerTouchOutDisableSwap

Example:
WDCameraPointTrackerTouchOutDisableSwap

This disables the option "Swap" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutDisableTUIO

WDCameraPointTrackerTouchOutDisableTUIO

Example:
WDCameraPointTrackerTouchOutDisableTUIO

This disables the option "TUIO Output" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutEnableInvX

WDCameraPointTrackerTouchOutEnableInvX

Example:
WDCameraPointTrackerTouchOutEnableInvX

This enables the option "Inv X" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutEnableInvY

WDCameraPointTrackerTouchOutEnableInvY

Example:
WDCameraPointTrackerTouchOutEnableInvY

This enables the option "Inv Y" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutEnableSwap

WDCameraPointTrackerTouchOutEnableSwap

Example:
WDCameraPointTrackerTouchOutEnableSwap

This enables the option "Swap" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutEnableTUIO

WDCameraPointTrackerTouchOutEnableTUIO

Example:
WDCameraPointTrackerTouchOutEnableTUIO

This enables the option "TUIO Output" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog.

WDCameraPointTrackerTouchOutMutePointTime

WDCameraPointTrackerTouchOutMutePointTime(Value)

Example:
WDCameraPointTrackerTouchOutMutePointTime(12.5)

This sets the option "Mute Point Time" in the section "Touch Output Processing" in the Camera Tracker [1291] dialog to the value 12.5.

WDCameraPointTrackerTouchOutRangeHeight

WDCameraPointTrackerTouchOutRangeHeight(Value)

Example:
WDCameraPointTrackerTouchOutRangeHeight(1080)

This sets the option "h" (height of range) in the section "Touch Output Processing" in the Camera Tracker [1291] dialog to the value 1080.

WDCameraPointTrackerTouchOutRangeWidth

WDCameraPointTrackerTouchOutRangeWidth(Value)

Example:
WDCameraPointTrackerTouchOutRangeWidth(1920)

This sets the option "w" (width of range) in the section "Touch Output Processing" in the Camera Tracker [1291] dialog to the value 1080.

WDCameraPointTrackerTouchOutRangeX

WDCameraPointTrackerTouchOutRangeX(Value)

Example:
WDCameraPointTrackerTouchOutRangeX(100)

This sets the option "x" (horizontal offset of range) in the section "Touch Output Processing" in the Camera Tracker [1291] dialog to the value 100.

WDCameraPointTrackerTouchOutRangeY

WDCameraPointTrackerTouchOutRangeY(Value)

Example:
WDCameraPointTrackerTouchOutRangeY(100)

This sets the option "y" (vertical offset of range) in the section "Touch Output Processing" in the Camera Tracker [1291] dialog to the value 100.

WDCameraPointTrackerTouchOutResolutionHeight

WDCameraPointTrackerTouchOutResolutionHeight(Value)

Example:
WDCameraPointTrackerTouchOutResolutionHeight(768)

This sets the option "h" (height of resolution) in the section "Touch Output Processing" in the Camera Tracker [1291] dialog to the value 768.

WDCameraPointTrackerTouchOutResolutionWidth

WDCameraPointTrackerTouchOutResolutionWidth(Value)

Example:
WDCameraPointTrackerTouchOutResolutionWidth(1024)

This sets the option "w" (width of resolution) in the section "Touch Output Processing" in the Camera Tracker [1291] dialog to the value 1024.

WDClose

WDClose

Example:
WDClose

Closes the current WD project. A pop-up dialog is going to ask you if you want to save the current changes.

WDCloseSilent

WDCloseSilent

Example:
WDCloseSilent

Closes the current WD project without saving it (there will be no pop-up dialog asking to save the project).

WDColorPickerCssStyleDisable

WDColorPickerCssStyleDisable(ID,StyleID)

Example:
WDColorPickerCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Color Picker [880] with ID 5.

WDColorPickerCssStyleEdit

WDColorPickerCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDColorPickerCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Color Picker [880] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDColorPickerCssStyleEnable

WDColorPickerCssStyleEnable(ID,StyleID)

Example:
WDColorPickerCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Color Picker [880] with ID 5.

WDColorPickerDevices

WDColorPickerDevices(ID,1.1 1.2 1.3)

Example:
WDColorPickerDevices(6,"1.1 2.4")

Transfers the RGB-values of the Color Picker [880] with the ID 6 in WD to the Color FX fader 1-3 of layers 1,1 and 2,4 in PB (before version 5!).

WDColorPickerFix

WDColorPickerFix(ID)

Example:
WDColorPickerFix(5)

This activates the option "Fix" in the Item Properties of the Color Picker [880] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDColorPickerGetLocationLeft

WDColorPickerGetLocationLeft(ID)

Example:
WDColorPickerGetLocationLeft(5)

This refers to the Color Picker [880] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDColorPickerGetLocationLeft(1)`

WDColorPickerGetLocationTop

WDColorPickerGetLocationTop(ID)

Example:
WDColorPickerGetLocationTop(5)

This refers to the Color Picker [880] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDColorPickerGetLocationTop(1)`

WDColorPickerGetSizeHeight

WDColorPickerGetSizeHeight(ID)

Example:
WDColorPickerGetSizeHeight(5)

This refers to the Color Picker [880] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDColorPickerGetSizeHeight(1)`

WDColorPickerGetSizeWidth

WDColorPickerGetSizeWidth(ID)

Example:
WDColorPickerGetSizeWidth(5)

This refers to the Color Picker [880] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDColorPickerGetSizeWidth(1)`

WDColorPickerLocation

WDColorPickerLocation(ID,X,Y)

Example:
WDColorPickerLocation(5,100,200)

Sets the position of the Color Picker [880] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDColorPickerLocationLeft

WDColorPickerLocationLeft(ID,X)

Example:
WDColorPickerLocationLeft(5,100)

Sets the position of the Color Picker [880] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDColorPickerLocationTop

WDColorPickerLocationTop(ID,Y)

Example:
WDColorPickerLocationTop(5,200)

Sets the position of the Color Picker [880] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDColorPickerSetHSB

WDColorPickerSetHSB(ID,Hue,Saturation,Brightness)

Example:
WDColorPickerSetHSB(1,200,255,190)

Changes the color of the Color Picker [880] with the ID 6 in WD to purple (Hue=200, Saturation=255, Brightness=190). The values for H,S,B range from 0 to 255. Alternatively, you can specify the color with the command `WDColorPickerSetRGB`.

WDColorPickerSetRGB

WDColorPickerSetRGB(ID,Red,Green,Blue)

Example:
WDColorPickerSetRGB(1,150,0,200)

Changes the color of the Color Picker [880] with the ID 6 in WD to purple (Red=150, Green=0, Blue=200). The values for R,G,B range from 0 to 255. Alternatively, you can specify the color with the command `WDColorPickerSetHSB`.

WDColorPickerSize

WDColorPickerSize(ID,Width,Height)

Example:
WDColorPickerSize(5,100,40)

Sets the size of the Color Picker [880] with ID 5 to a width of 100px and a height of 40px.

WDColorPickerSizeHeight

WDColorPickerSizeHeight(ID,Height)

Example:
WDColorPickerSizeHeight(5,40)

Sets the size of the Color Picker [880] with ID 5 to a height of 40px but remains the current width.

WDColorPickerSizeWidth

WDColorPickerSizeWidth(ID,Width)

Example:
WDColorPickerSizeWidth(5,100)

Sets the size of the Color Picker [880] with ID 5 to a width of 100px but remains the current height.

WDColorPickerUnfix

WDColorPickerUnfix(ID)

Example:
WDColorPickerUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Color Picker [880] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDConnect

WDConnect(IP,Domain)

Example:
WDConnect("10.169.10.65",0)

Connects the Widget Designer to Pandoras Box with the IP address 10.169.10.65. The connection can also be set up via Connections > PB Configuration [1256] in WD.

WDConnectBackup

WDConnectBackup

Example:
WDConnectBackup

Connects the Widget Designer to the Pandoras Box Backup after it was disconnected. See the topic PB Network Configuration 1256 in WD for more information.

WDControlBringToFront

WDControlBringToFront(ControlName)

Example:
WDControlBringToFront("Customscript2")

Changes the Z-Order and brings Custom Script Button 2 to the front. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties.

WDControlCopy

WDControlCopy(Key,X,Y,Page,New KeyWD(optional),New ID(optional))

Example:
WDControlCopy("Fader1",100,200,"Page1")

The first example uses only the mandatory parameters. It copies the Widget with the name "Fader1" to the position 100px horizontally and 200px vertically of page "Page1". The new Fader gets the next available ID and the according name, e.g. "Fader2".

Example2:
WDControlCopy("Fader1",100,200,"Page1","Mixer",101)

The second example adds the optional parameters and sets the name of the new Widget to "Mixer" and the ID to 101.

WDControlCreate

WDControlCreate(Type,X,Y,PageName,Width(optional),Height(optional),Name(optional),ID(optional))

Example:
WDControlCreate("Fader_H",100,200,"Page1")

The first example uses only the mandatory parameters. It creates a new horizontal Fader at the position 100px horizontally and 200px vertically on page "Page1" with the initial width and height values and assigns the next available ID and the according name, e.g. "Fader3".

Example2:
WDControlCreate("Fader_H",100,200,"Page1",100,30,"Mixer",101)

The second example adds the optional parameters. The width and height of the Fader are set to 100 and 30, the name is changed to "Mixer" and the ID to 101.

WDControlDelete

WDControlDelete(Key)

Example:
WDControlDelete("CustomScript1")

This deletes the Widget with the name "CustomScript1" from the project.

WDControlFix

WDControlFix(ControlName)

Example:
WDControlFix("CustomScript5")

This activates the option "Fix" within the Item Properties of CustomScript Button 5. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDControlHide

WDControlHide(ControlName)

Example:
WDControlHide("Customscript2")

Hides Custom Script Button 2 outside of its page context. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties.

This way you can hide a control that originates from another page. This may be done e.g. as On Page 805 Enter or Leave script.

WDControlSendToBack

WDControlSendToBack(ControlName)

Example:
WDControlSendToBack("Customscript2")

Changes the Z-Order and sends Custom Script Button 2 to the back. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties.

WDControlSetPage

WDControlSetPage(ControlName,Page)

Example:
WDControlSetPage("CustomScript5","Page2")

This moves the Widget CustomScript Button 5 to Page2. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties. The name of the Page is also suggested and can be changed in the Page Properties 805.

WDControlSetPosition

WDControlSetPosition(ControlName,X,Y)

Example:
WDControlSetPosition("CustomScript5",100,-10)

Sets the position of CustomScript Button 5 to 100px horizontally and 200px vertically. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties.
The position is set absolutely, whereas 0,0 is the top left corner of the Page.

Alternatively, the command `WDControlSetPositionRelative` sets the relative position.

WDControlSetPositionRelative

WDControlSetPositionRelative(ControlName,X,Y)

Example:
WDControlSetPositionRelative("CustomScript5",100,)

Sets the position of CustomScript Button 5. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties.
The command adds 100 pixels to the current horizontal position (the widget moves to the right) and subtracts 10 pixels from the current vertical position (the widget moves up).
Alternatively, the command `WDControlSetPosition` sets the absolute position.

WDControlSetSize

WDControlSetSize(ControlName,Width,Height)

Example:
WDControlSetSize("CustomScript5",100,10)

Sets the size of CustomScript Button 5 to a width of 100px and a height of 10px. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties.

WDControlShow

WDControlShow(ControlName)

Example:
WDControlShow("Customscript2")

Shows the Custom Script Button 2 outside of its page context. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties.

This way you can show a control that originates from another page. This may be done e.g. as On Page [805] Enter or Leave script.

WDControlUnFix

WDControlUnFix(ControlName)

Example:
WDControlUnFix("CustomScript5")

This deactivates the option "Fix" within the Item Properties of CustomScript Button 5. The name of a Widget is suggested in the Script Assistant and can be changed in the Widget's Item Properties. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDCueControlCssStyleDisable

WDCueControlCssStyleDisable(ID,StyleID)

Example:
WDCueControlCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the CueControl [831] with ID 5.

WDCueControlCssStyleEdit

WDCueControlCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDCueControlCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the CueControl [831] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDCueControlCssStyleEnable

WDCueControlCssStyleEnable(ID,StyleID)

Example:
WDCueControlCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the CueControl [831] with ID 5.

WDCueControlFix

WDCueControlFix(ID)

Example:
WDCueControlFix(5)

This activates the option "Fix" in the Item Properties of the CueControl [831] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDCueControlGetFontFamily

WDCueControlGetFontFamily(ID)

Example:
varString = WDCueControlGetFontFamily(5)

This refers to the CueControl [831] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another CueControl via a script like:
`WDCueControlSetFontFamily(5,WDCueControlGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDCueControlGetFontFamily(5))`

WDCueControlGetFontSize

WDCueControlGetFontSize(ID)

Example:
varDouble = WDCueControlGetFontSize(5)

This refers to the CueControl [831] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another CueControl via a script like:
`WDCueControlSetFontSize(5,WDCueControlGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDCueControlGetFontSize(5))`

WDCueControlGetLocationLeft

WDCueControlGetLocationLeft(ID)

Example:
WDCueControlGetLocationLeft(5)

This refers to the CueControl [831] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCueControlGetLocationLeft(1)`

WDCueControlGetLocationTop

WDCueControlGetLocationTop(ID)

Example:
WDCueControlGetLocationTop(5)

This refers to the CueControl [831] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCueControlGetLocationTop(1)`

WDCueControlGetSizeHeight

WDCueControlGetSizeHeight(ID)

Example:
WDCueControlGetSizeHeight(5)

This refers to the CueControl [831] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCueControlGetSizeHeight(1)`

WDCueControlGetSizeWidth

WDCueControlGetSizeWidth(ID)

Example:
WDCueControlGetSizeWidth(5)

This refers to the CueControl [831] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCueControlGetSizeWidth(1)`

WDCueControlLocation

WDCueControlLocation(ID,X,Y)

Example:
WDCueControlLocation(5,100,200)

Sets the position of the CueControl [831] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDCueControlLocationLeft

WDCueControlLocationLeft(ID,X)

Example:
WDCueControlLocationLeft(5,100)

Sets the position of the CueControl [831] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDCueControlLocationTop

WDCueControlLocationTop(ID,Y)

Example:
WDCueControlLocationTop(5,200)

Sets the position of the CueControl [831] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDCueControlSetFontFamily

WDCueControlSetFontFamily(ID,FontName)

Example:
WDCueControlSetFontFamily(5,"Arial")

This refers to the CueControl [831] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another CueControl, you can use a script like:
`WDCueControlSetFontFamily(5,WDCueControlGetFontFamily(1))`

WDCueControlSetFontSize

WDCueControlSetFontSize(ID,FontSize)

Example:
WDCueControlSetFontSize(5,20)

This refers to the CueControl [831] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another CueControl, you can use a script like:
`WDCueControlSetFontSize(5,WDCueControlGetFontSize(1))`

WDCueControlSize

WDCueControlSize(ID,Width,Height)

Example:
WDCueControlSize(5,100,40)

Sets the size of the CueControl [831] with ID 5 to a width of 100px and a height of 40px.

WDCueControlSizeHeight

WDCueControlSizeHeight(ID,Height)

Example:
WDCueControlSizeHeight(5,40)

Sets the size of the CueControl [831] with ID 5 to a height of 40px but remains the current width.

WDCueControlSizeWidth

WDCueControlSizeWidth(ID,Width)

Example:
WDCueControlSizeWidth(5,100)

Sets the size of the CueControl [831] with ID 5 to a width of 100px but remains the current height.

WDCueControlUnfix

WDCueControlUnfix(ID)

Example:
WDCueControlUnfix(5)

This deactivates the option "Fix" in the Item Properties of the CueControl [831] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDCustomScriptAddTextFromLabel

WDCustomScriptAddTextFromLabel(CustomScriptID,ID)

Example:
WDCustomScriptAddTextFromLabel(6,3)

Adds the text from Label [888] 6 to the Custom Script Button [822] with the ID 3.

WDCustomScriptApplyCITPThumbnail

WDCustomScriptApplyCITPThumbnail(ID,FolderID,FileID)

Example:
WDCustomScriptApplyCITPThumbnail(5,1,2)

This depicts the thumbnail from the File and Folder ID 1,2 on Custom Script Button [822] 5. Note that the thumbnail overlays images for the clicked, released or highlight status and can only be reset with the command `WDCustomScriptResetCITPThumbnail`. CITP Thumbnail Exchange can be setup under Connections > PB Configuration [1256]. In Pandoras Box, please go to the Configuration tab > Remote Control Protocols [148].

WDCustomScriptClick

WDCustomScriptClick(ID)

Example:
WDCustomScriptClick(5)

Clicks the Custom Script Button [822] with the ID 5. If the type of Custom Script Button 5 is "Toggle" the command toggles the button.

If you like to execute this command addressing many buttons, the chapter "For loop [1890]" and "Project and Context Member [1906]" show some interesting examples.

Overview of similar commands:
WDCustomScriptClick [1691]: state+script, always
WDCustomScriptForcePressed [1693]/-Released: state+script, if in other mode
WDCustomScriptSetState [1697]: state without script
WDCustomScriptExecuteScript [1693]: script without state

WDCustomScriptClickImage

WDCustomScriptClickImage(ID,File)

Example:
WDCustomScriptClickImage(5,"C:\Christie\clicked.png")

This sets the image for the clicked status of Custom Script Button [822] 5 to the image saved under the specified path.

WDCustomScriptClickImageResource

WDCustomScriptClickImageResource(ID,ResourceName)

Example:
WDCustomScriptClickImageResource(5,"Default\Button\Lock")

This sets the image for the clicked status of Custom Script Button [822] 5 to the image saved under the specified name in the "Image Resource Manager [1509]" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDCustomScriptCssStyleDisable

WDCustomScriptCssStyleDisable(ID,StyleID)

Example:
WDCustomScriptCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the CustomScript [822] button with ID 5.

WDCustomScriptCssStyleEdit

WDCustomScriptCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDCustomScriptCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the CustomScript [822] button with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDCustomScriptCssStyleEnable

WDCustomScriptCssStyleEnable(ID,StyleID)

Example:
WDCustomScriptCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the CustomScript [822] button with ID 5.

WDCustomScriptExecuteScript

WDCustomScriptExecuteScript(ID,Script)

Example:
WDCustomScriptExecuteScript(5,"Press")

This command executes the "Press" script of Custom Script Button [822] with ID 5 without changing the "Pressed" or "Released" mode itself.
These are the available script names: Enter, Leave, Press, Release

If you like to execute this command addressing many buttons, the chapter "For loop [1890]" and "Project and Context Member [1906]" show some interesting examples.

Overview of similar commands:
WDCustomScriptClick [1691]: state+script, always
WDCustomScriptForcePressed [1693]/-Released: state+script, if in other mode
WDCustomScriptSetState [1697]: state without script
WDCustomScriptExecuteScript [1693]: script without state

WDCustomScriptFix

WDCustomScriptFix(ID)

Example:
WDCustomScriptFix(3)

Sets the option "Fix" for the Custom Script Button [822] with the ID 3, so that this CS Button is shown on every page inside WD now.

WDCustomScriptForcePressed

WDCustomScriptForcePressed(ID)

Example:
WDCustomScriptForcePressed(5)

This command works only for Custom Script Buttons [822] with the Type "Toggle". If the Custom Script Button with ID 5 is in the Released mode, the command sets it to the Pressed mode and executes the associated "On Press" script. If it is already in the Pressed mode, the command has no effect.

If you like to execute this command addressing many buttons, the chapter "For loop [1890]" and "Project and Context Member [1906]" show some interesting examples.

Overview of similar commands:
WDCustomScriptClick [1691]: state+script, always
WDCustomScriptForcePressed [1693]/-Released: state+script, if in other mode
WDCustomScriptSetState [1697]: state without script
WDCustomScriptExecuteScript [1693]: script without state

WDCustomScriptForceReleased

WDCustomScriptForceReleased(ID)

Example:
WDCustomScriptForceReleased(5)

This command works only for Custom Script Buttons [822] with the Type "Toggle". If the Custom Script Button with ID 5 is in the Pressed mode, the command sets it to the Released mode and executes the associated "On Release" script. If it is already in the Released mode, the command has no effect.

If you like to execute this command addressing many buttons, the chapter "For loop [1890]" and "Project and Context Member [1906]" show some interesting examples.

Overview of similar commands:
WDCustomScriptClick [1691]: state+script, always
WDCustomScriptForcePressed [1693]/-Released: state+script, if in other mode
WDCustomScriptSetState [1697]: state without script
WDCustomScriptExecuteScript [1693]: script without state

WDCustomScriptGetLocationLeft

WDCustomScriptGetLocationLeft(ID)

Example:
WDCustomScriptGetLocationLeft(5)

This refers to the Custom Script Button [822] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCustomScriptGetLocationLeft(1)`

WDCustomScriptGetLocationTop

WDCustomScriptGetLocationTop(ID)

Example:
WDCustomScriptGetLocationTop(5)

This refers to the Custom Script Button [822] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCustomScriptGetLocationTop(1)`

WDCustomScriptGetSizeHeight

WDCustomScriptGetSizeHeight(ID)

Example:
WDCustomScriptGetSizeHeight(5)

This refers to the Custom Script Button [822] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCustomScriptGetSizeHeight(1)`

WDCustomScriptGetSizeWidth

WDCustomScriptGetSizeWidth(ID)

Example:
WDCustomScriptGetSizeWidth(5)

This refers to the Custom Script Button [822] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDCustomScriptGetSizeWidth(1)`

WDCustomScriptHighlightImage

WDCustomScriptHighlightImage(ID,File)

Example:
WDCustomScriptHighlightImage(5,"C:\Christie\highlight.png")

This sets the image for the highlight status of Custom Script Button [822] 5 to the image saved under the specified path.

WDCustomScriptHighlightImageResource

WDCustomScriptHighlightImageResource(ID,ResourceName)

Example:
WDCustomScriptHighlightImageResource(5,"Default\Button_MouseOver.png")

This sets the image for the highlight status of Custom Script Button [822] 5 to the image saved under the specified name in the "Image Resource Manager [1509]" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDCustomScriptLabel

WDCustomScriptLabel(ID,Text)

Example:
WDCustomScriptLabel(3,"Flip")

Labels the Custom Script Button [822] with the ID 3 with the word "Flip".
Other similar commands would be: WDCustomScriptTextFromLabel [1698] and WDCustomScriptLabelColor [1695].

WDCustomScriptLabelColor

WDCustomScriptLabelColor(ID,R,G,B)

Example:
WDCustomScriptLabelColor(3,255,90,0)

Changes the label color of the Custom Script Button [822] with the ID 3 to orange (Red=255, Green=90, Blue=0). The values for R,G,B range from 0 to 255.
Other similar commands would be: WDCustomScriptTextFromLabel [1698] and WDCustomScriptLabel [1695].

WDCustomScriptLocation

WDCustomScriptLocation(ID,X,Y)

Example:
WDCustomScriptLocation(5,100,200)

Sets the position of the CustomScript [822] button with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDCustomScriptLocationLeft

WDCustomScriptLocationLeft(ID,X)

Example:
WDCustomScriptLocationLeft(5,100)

Sets the position of the CustomScript [822] button with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDCustomScriptLocationTop

WDCustomScriptLocationTop(ID,Y)

Example:
WDCustomScriptLocationTop(5,200)

Sets the position of the CustomScript [822] button with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDCustomScriptMuteClickDisabled

WDCustomScriptMuteClickDisabled(ID)

Example:
WDCustomScriptMuteClickDisabled(5)

This deactivates the option "Mute Click Script" within the Item Properties of Custom Script Button [822] 5.

WDCustomScriptMuteClickEnabled

WDCustomScriptMuteClickEnabled(ID)

Example:
WDCustomScriptMuteClickEnabled(5)

This activates the option "Mute Click Script" within the Item Properties of Custom Script Button [822] 5 to protect it from accidental double-clicks.

WDCustomScriptMuteClickTime

WDCustomScriptMuteClickTime(ID,Time)

Example:
WDCustomScriptMuteClickTime(5,500)

This sets the time for the option "Mute Click Script" within the Item Properties of Custom Script Button [822] 5 to 500ms.

WDCustomScriptReleaseImage

WDCustomScriptReleaseImage(ID,File)

Example:
WDCustomScriptReleaseImage(5,"C:\Christie\released.png")

This sets the image for the released status of Custom Script Button [822] 5 to the image saved under the specified path.

WDCustomScriptReleaseImageResource

WDCustomScriptReleaseImageResource(ID,ResourceName)

Example:
WDCustomScriptReleaseImageResource(5,"Default\Button\Unlock")

This sets the image for the released status of Custom Script Button [822] 5 to the image saved under the specified name in the "Image Resource Manager [1509]" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDCustomScriptResetCITPThumbnail

WDCustomScriptResetCITPThumbnail(ID)

Example:
WDCustomScriptResetCITPThumbnail(ID)

This resets the CITP Thumbnail from Custom Script Button [822] 5 which was applied with the command `WDCustomScriptApplyCITPThumbnail`.

WDCustomScriptSetState

WDCustomScriptSetState(ID,State)

Example:
WDCustomScriptSetState(5,"Pressed")

This command works only for Custom Script Buttons [822] with the Type "Toggle". If the Custom Script Button with ID 5 is in the Released mode, the command sets it to the Pressed mode but does not executes the associated "On Press" script. If it is already in the Pressed mode, the command has no effect.

Example 2:
WDCustomScriptSetState(5,"Released")

This command works only for Custom Script Buttons with the Type "Toggle". If the Custom Script Button with ID 5 is in the Pressed mode, the command sets it to the Released mode but does not executes the associated "On Release" script. If it is already in the Released mode, the command has no effect.

If you like to execute this command addressing many buttons, the chapter "For loop [1890]" and "Project and Context Member [1906]" show some interesting examples.

Overview of similar commands:
WDCustomScriptClick [1691]: state+script, always
WDCustomScriptForcePressed [1693]/-Released: state+script, if in other mode
WDCustomScriptSetState [1697]: state without script
WDCustomScriptExecuteScript [1693]: script without state

WDCustomScriptSize

WDCustomScriptSize(ID,Width,Height)

Example:
WDCustomScriptSize(5,100,40)

Sets the size of the CustomScript [822] button with ID 5 to a width of 100px and a height of 40px.

WDCustomScriptSizeHeight

WDCustomScriptSizeHeight(ID,Height)

Example:
WDCustomScriptSizeHeight(5,40)

Sets the size of the CustomScript [822] button with ID 5 to a height of 40px but remains the current width.

WDCustomScriptSizeWidth

WDCustomScriptSizeWidth(ID,Width)

Example:
WDCustomScriptSizeWidth(5,100)

Sets the size of the CustomScript [822] button with ID 5 to a width of 100px but remains the current height.

WDCustomScriptTextFromLabel

WDCustomScriptTextFromLabel(CustomScriptID,ID)

Example:
WDCustomScriptTextFromLabel(2,4)

Labels the Custom Script Button [822] with the ID 4 with the text from Label [888] 2.
Other similar commands would be: WDCustomScriptLabel [1695] and WDCustomScriptLabelColor [1695].

WDCustomScriptTint

WDCustomScriptTint(ID,Red,Green,Blue)

Example:
WDCustomScriptTint(5,0,255,0)

This tints the Custom Script Button [822] with the ID 5 in green.

WDCustomScriptUnfix

WDCustomScriptUnfix(ID)

Example:
WDCustomScriptUnfix(3)

Disables the option "Fix" for the Custom Script Button [822] with the ID 3, so that this CS Button is not shown on every page inside WD any more.

WDDigitalDisplayCssStyleDisable

WDDigitalDisplayCssStyleDisable(ID,StyleID)

Example:
WDDigitalDisplayCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Digital Display [854] with ID 5.

WDDigitalDisplayCssStyleEdit

WDDigitalDisplayCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDDigitalDisplayCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Digital Display [854] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDDigitalDisplayCssStyleEnable

WDDigitalDisplayCssStyleEnable(ID,StyleID)

Example:
WDDigitalDisplayCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Digital Display [854] with ID 5.

WDDigitalDisplayFix

WDDigitalDisplayFix(ID)

Example:
WDDigitalDisplayFix(5)

This activates the option "Fix" in the Item Properties of the Digital Display [854] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDDigitalDisplayGetLocationLeft

WDDigitalDisplayGetLocationLeft(ID)

Example:
WDDigitalDisplayGetLocationLeft(5)

This refers to the Digital Display [854] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDigitalDisplayGetLocationLeft(1)`

WDDigitalDisplayGetLocationTop

WDDigitalDisplayGetLocationTop(ID)

Example:
WDDigitalDisplayGetLocationTop(5)

This refers to the Digital Display [854] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDigitalDisplayGetLocationTop(1)`

WDDigitalDisplayGetSizeHeight

WDDigitalDisplayGetSizeHeight(ID)

Example:
WDDigitalDisplayGetSizeHeight(5)

This refers to the Digital Display [854] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDigitalDisplayGetSizeHeight(1)`

WDDigitalDisplayGetSizeWidth

WDDigitalDisplayGetSizeWidth(ID)

Example:
WDDigitalDisplayGetSizeWidth(5)

This refers to the Digital Display [854] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDigitalDisplayGetSizeWidth(1)`

WDDigitalDisplayLocation

WDDigitalDisplayLocation(ID,X,Y)

Example:
WDDigitalDisplayLocation(5,100,200)

Sets the position of the Digital Display [854] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDDigitalDisplayLocationLeft

WDDigitalDisplayLocationLeft(ID,X)

Example:
WDDigitalDisplayLocationLeft(5,100)

Sets the position of the Digital Display [854] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDDigitalDisplayLocationTop

WDDigitalDisplayLocationTop(ID,Y)

Example:
WDDigitalDisplayLocationTop(5,200)

Sets the position of the Digital Display [854] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDDigitalDisplaySize

WDDigitalDisplaySize(ID,Width,Height)

Example:
WDDigitalDisplaySize(5,100,40)

Sets the size of the Digital Display [854] with ID 5 to a width of 100px and a height of 40px.

WDDigitalDisplaySizeHeight

WDDigitalDisplaySizeHeight(ID,Height)

Example:
WDDigitalDisplaySizeHeight(5,40)

Sets the size of the Digital Display [854] with ID 5 to a height of 40px but remains the current width.

WDDigitalDisplaySizeWidth

WDDigitalDisplaySizeWidth(ID,Width)

Example:
WDDigitalDisplaySizeWidth(5,100)

Sets the size of the Digital Display [854] with ID 5 to a width of 100px but remains the current height.

WDDigitalDisplayUnfix

WDDigitalDisplayUnfix(ID)

Example:
WDDigitalDisplayUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Digital Display [854] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDDisconnect

WDDisconnect

Example:
WDDisconnect

Disconnects Widget Designer from both the Pandoras Box Master and Backup. To connect to PB Master or Client System again, please go to Connections > PB Configuration [1256] in WD or use the WD command `WDConnect` [1684] or `WDConnectBackup` [1685].

WDDisconnectBackup

WDDisconnectBackup

Example:
WDDisconnectBackup

Disconnects the Widget Designer from Pandoras Box Backup. See the topic PB Network Configuration [1256] in WD for more information.

WDDisconnectMaster

WDDisconnectMaster

Example:
WDDisconnectMaster

Disconnects the Widget Designer from Pandoras Box Master. See the topic PB Network Configuration [1256] in WD for more information.

WDDownloadThumbnails

WDDownloadThumbnails

Example:
WDDownloadThumbnails

Downloads all Thumbnails from the connected PB.
Please make sure that the function to download Thumbnails [148] from PB is enabled!

WDDrawingCanvasClear

WDDrawingCanvasClear(ID)

Example:
WDDrawingCanvasClear(1)

Clears the interface of the Drawing Canvas [893] with the ID 1.

WDDrawingCanvasCssStyleDisable

WDDrawingCanvasCssStyleDisable(ID,StyleID)

Example:
WDDrawingCanvasCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Drawing Canvas [893] with ID 5.

WDDrawingCanvasCssStyleEdit

WDDrawingCanvasCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDDrawingCanvasCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Drawing Canvas [893] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDDrawingCanvasCssStyleEnable

WDDrawingCanvasCssStyleEnable(ID,StyleID)

Example:
WDDrawingCanvasCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Drawing Canvas [893] with ID 5.

WDDrawingCanvasFix

WDDrawingCanvasFix(ID)

Example:
WDDrawingCanvasFix(5)

This activates the option "Fix" in the Item Properties of the Drawing Canvas [893] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDDrawingCanvasGetLocationLeft

WDDrawingCanvasGetLocationLeft(ID)

Example:
WDDrawingCanvasGetLocationLeft(5)

This refers to the Drawing Canvas [893] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDrawingCanvasGetLocationLeft(1)`

WDDrawingCanvasGetLocationTop

WDDrawingCanvasGetLocationTop(ID)

Example:
WDDrawingCanvasGetLocationTop(5)

This refers to the Drawing Canvas [893] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDrawingCanvasGetLocationTop(1)`

WDDrawingCanvasGetSizeHeight

WDDrawingCanvasGetSizeHeight(ID)

Example:
WDDrawingCanvasGetSizeHeight(5)

This refers to the Drawing Canvas [893] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDrawingCanvasGetSizeHeight(1)`

WDDrawingCanvasGetSizeWidth

WDDrawingCanvasGetSizeWidth(ID)

Example:
WDDrawingCanvasGetSizeWidth(5)

This refers to the Drawing Canvas [893] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDrawingCanvasGetSizeWidth(1)`

WDDrawingCanvasLocation

WDDrawingCanvasLocation(ID,X,Y)

Example:
WDDrawingCanvasLocation(5,100,200)

Sets the position of the Drawing Canvas [893] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDDrawingCanvasLocationLeft

WDDrawingCanvasLocationLeft(ID,X)

Example:
WDDrawingCanvasLocationLeft(5,100)

Sets the position of the Drawing Canvas [893] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDDrawingCanvasLocationTop

WDDrawingCanvasLocationTop(ID,Y)

Example:
WDDrawingCanvasLocationTop(5,200)

Sets the position of the Drawing Canvas [893] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDDrawingCanvasSetBGCol

WDDrawingCanvasSetBGCol(ID,R,G,B)

Example:
WDDrawingCanvasSetBGCol(1,255,0,100)

Changes the current background color of the Drawing Canvas with the ID 1 to the color R=255,G=0,B=100 (pink).

WDDrawingCanvasSetPenCol

WDDrawingCanvasSetPenCol(ID,R,G,B)

Example:
WDDrawingCanvasSetPenCol(1,255,0,0)

Changes the current pen color of the Drawing Canvas with the ID 1 to the color R=255,G=0,B=0 (red).

WDDrawingCanvasSetPenSize

WDDrawingCanvasSetPenSize(ID,Size)

Example:
WDDrawingCanvasSetPenSize(1,16)

Changes the current pen size of the Drawing Canvas with the ID 1 to the size 16.

WDDrawingCanvasSize

WDDrawingCanvasSize(ID,Width,Height)

Example:
WDDrawingCanvasSize(5,100,40)

Sets the size of the Drawing Canvas [893] with ID 5 to a width of 100px and a height of 40px.

WDDrawingCanvasSizeHeight

WDDrawingCanvasSizeHeight(ID,Height)

Example:
WDDrawingCanvasSizeHeight(5,40)

Sets the size of the Drawing Canvas [893] with ID 5 to a height of 40px but remains the current width.

WDDrawingCanvasSizeWidth

WDDrawingCanvasSizeWidth(ID,Width)

Example:
WDDrawingCanvasSizeWidth(5,100)

Sets the size of the Drawing Canvas [893] with ID 5 to a width of 100px but remains the current height.

WDDrawingCanvasUnfix

WDDrawingCanvasUnfix(ID)

Example:
WDDrawingCanvasUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Drawing Canvas [893] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDDropDownGetFontFamily

WDDropDownGetFontFamily(ID)

Example:
varString = WDDropDownGetFontFamily(5)

This refers to the Drop Down List [868] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Drop Down List via a script like:
`WDDropDownSetFontFamily(5,WDDropDownGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDDropDownGetFontFamily(5))`

WDDropDownGetFontSize

WDDropDownGetFontSize(ID)

Example:
varDouble = WDDropDownGetFontSize(5)

This refers to the Drop Down List [868] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Drop Down List via a script like:
`WDDropDownSetFontSize(5,WDDropDownGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDDropDownGetFontSize(5))`

WDDropDownListCssStyleDisable

WDDropDownListCssStyleDisable(ID,StyleID)

Example:
WDDropDownListCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Drop Down List [868] with ID 5.

WDDropDownListCssStyleEdit

WDDropDownListCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDDropDownListCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Drop Down List [868] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDDropDownListCssStyleEnable

WDDropDownListCssStyleEnable(ID,StyleID)

Example:
WDDropDownListCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Drop Down List [868] with ID 5.

WDDropDownListFix

WDDropDownListFix(ID)

Example:
WDDropDownListFix(5)

This activates the option "Fix" in the Item Properties of the Drop Down List [868] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDDropDownListGetTitle

WDDropDownListGetTitle(ID)

Example:
varString = WDDropDownListGetTitle(5)

This refers to the Drop Down List [868] with ID 5 and returns its current title as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. a Label [888] via the script: `Label1.Text = WDDropDownListGetTitle(5)`
or simply display it in the Debug Logger [812]: `DebugMessage(WDDropDownListGetTitle(5))`

WDDropDownListGetLocationLeft

WDDropDownListGetLocationLeft(ID)

Example:
WDDropDownListGetLocationLeft(5)

This refers to the Drop Down List [868] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDropDownListGetLocationLeft(1)`

WDDropDownListGetLocationTop

WDDropDownListGetLocationTop(ID)

Example:
WDDropDownListGetLocationTop(5)

This refers to the Drop Down List [868] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDropDownListGetLocationTop(1)`

WDDropDownListGetSizeHeight

WDDropDownListGetSizeHeight(ID)

Example:
WDDropDownListGetSizeHeight(5)

This refers to the Drop Down List [868] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDropDownListGetSizeHeight(1)`

WDDropDownListGetSizeWidth

WDDropDownListGetSizeWidth(ID)

Example:
WDDropDownListGetSizeWidth(5)

This refers to the Drop Down List [868] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDDropDownListGetSizeWidth(1)`

WDDropDownListLocation

WDDropDownListLocation(ID,X,Y)

Example:
WDDropDownListLocation(5,100,200)

Sets the position of the Drop Down List [868] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDDropDownListLocationLeft

WDDropDownListLocationLeft(ID,X)

Example:
WDDropDownListLocationLeft(5,100)

Sets the position of the Drop Down List [868] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDDropDownListLocationTop

WDDropDownListLocationTop(ID,Y)

Example:
WDDropDownListLocationTop(5,200)

Sets the position of the Drop Down List [868] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDDropDownListSetBackgroundColor

WDDropDownListSetBackgroundColor(ID,R,G,B)

Example:
WDDropDownListSetBackgroundColor(1,0,120,150)

This changes the background color of the Drop Down List [868] with ID 1 to a greenish blue (Red=0, Green=120, Blue=150). The values for R,G,B range from 0 to 255.

WDDropDownListSetIndex

WDDropDownListSetIndex(ID,Value)

Example:
WDDropDownListSetIndex(1,3)

In the DropDownList [868] with ID 1, the entry element with index 3 is selected. Note: The indexing starts with "0".

WDDropDownListSetItemsFromArray

WDDropDownListSetItemsFromArray(ID,VarName)

Example:
WDDropDownListSetItemsFromArray(1,"var_array")

Fills the elements of DropDownList 868 with ID 1 with the values from the array-type Variable 1900 "var_array".

WDDropDownListSetItemsFromExcel

WDDropDownListSetItemsFromExcel(ID,File,Table,CellStart,CellEnd)

Example:
WDDropDownListSetItemsFromExcel(1,"C:\Christie\content\drop_down.xls","Sheet1","A1","A4")

Fills the entry elements of DropDownList 868 with ID 1 with the values from cell A1 to A4, saved in the Excel file "drop_down.xls", Sheet1.
Note: The Excel document has to be an XLS file. The file may take some seconds to load.

WDDropDownListSetText

WDDropDownListSetText(ID,Text)

Example:
WDDropDownListSetText(1,"Load File")

Sets the default text of DropDownList 868 with ID 1 to "Load File".

WDDropDownListSetTextColor

WDDropDownListSetTextColor(ID,R,G,B)

Example:
WDDropDownListSetTextColor(1,255,150,0)

This changes the text color of the Drop Down List 868 with ID 1 to orange (Red=255, Green=150, Blue=0). The values for R,G,B range from 0 to 255.

WDDropDownListSetTitle

WDDropDownListSetTitle(ID,Title)

Example:
varString = WDDropDownListSetTitle(5,"Choose...")

This refers to the Drop Down List 868 with ID 5 and changes the title to "Choose...".

WDDropDownListSize

WDDropDownListSize(ID,Width,Height)

Example:
WDDropDownListSize(5,100,40)

Sets the size of the Drop Down List 868 with ID 5 to a width of 100px and a height of 40px.

WDDropDownListSizeHeight

WDDropDownListSizeHeight(ID,Height)

Example:
WDDropDownListSizeHeight(5,40)

Sets the size of the Drop Down List [868] with ID 5 to a height of 40px but remains the current width.

WDDropDownListSizeWidth

WDDropDownListSizeWidth(ID,Width)

Example:
WDDropDownListSizeWidth(5,100)

Sets the size of the Drop Down List [868] with ID 5 to a width of 100px but remains the current height.

WDDropDownListUnfix

WDDropDownListUnfix(ID)

Example:
WDDropDownListUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Drop Down List [868] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDDropDownSetFontFamily

WDDropDownSetFontFamily(ID,FontName)

Example:
WDDropDownSetFontFamily(5,"Arial")

This refers to the Drop Down List [868] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Drop Down List, you can use a script like:
`WDDropDownSetFontFamily(5,WDDropDownGetFontFamily(1))`

WDDropDownSetFontSize

WDDropDownSetFontSize(ID,FontSize)

Example:
WDDropDownSetFontSize(5,20)

This refers to the Drop Down List [868] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Drop Down List, you can use a script like:
`WDDropDownSetFontSize(5,WDDropDownGetFontSize(1))`

WDEncoderCssStyleDisable

WDEncoderCssStyleDisable(ID,StyleID)

Example:
WDEncoderCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Encoder [877] with ID 5.

WDEncoderCssStyleEdit

WDEncoderCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDEncoderCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Encoder [877] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDEncoderCssStyleEnable

WDEncoderCssStyleEnable(ID,StyleID)

Example:
WDEncoderCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Encoder [877] with ID 5.

WDEncoderFactor

WDEncoderFactor(ID,Value)

Example:
WDEncoderFactor(7,5)

Sets the factor of Encoder [877] 7 to the value 5.

WDEncoderFix

WDEncoderFix(ID)

Example:
WDEncoderFix(5)

This activates the option "Fix" in the Item Properties of the Encoder [877] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDEncoderGetLocationLeft

WDEncoderGetLocationLeft(ID)

Example:
WDEncoderGetLocationLeft(5)

This refers to the Encoder [877] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEncoderGetLocationLeft(1)`

WDEncoderGetLocationTop

WDEncoderGetLocationTop(ID)

Example:
WDEncoderGetLocationTop(5)

This refers to the Encoder [877] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEncoderGetLocationTop(1)`

WDEncoderGetSizeHeight

WDEncoderGetSizeHeight(ID)

Example:
WDEncoderGetSizeHeight(5)

This refers to the Encoder [877] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEncoderGetSizeHeight(1)`

WDEncoderGetSizeWidth

WDEncoderGetSizeWidth(ID)

Example:
WDEncoderGetSizeWidth(5)

This refers to the Encoder [877] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEncoderGetSizeWidth(1)`

WDEncoderGoDown

WDEncoderGoDown(ID,Value)

Example:
WDEncoderGoDown(7,10)

Subtracts immediately 10 values from the value of Encoder [877] 7.

WDEncoderGoUp

WDEncoderGoUp(ID,Value)

Example:
WDEncoderGoUp(7,10)

Adds immediately 10 values to the value of Encoder [877] 7.

WDEncoderInfoMidiInputDisable

WDEncoderInfoMidiInputDisable(ID)

Example:
WDEncoderInfoMidiInputDisable(1)

Disables the Midi Input in the Item Properties of the Encoder [877] with ID 1.

WDEncoderInfoMidiInputEnable

WDEncoderInfoMidiInputEnable(ID)

Example:
WDEncoderInfoMidiInputEnable(1)

Enables the Midi Input in the Item Properties of the Encoder [877] with ID 1.

WDEncoderInfoMidiOutputDisable

WDEncoderInfoMidiOutputDisable(ID)

Example:
WDEncoderInfoMidiOutputDisable(1)

Disables the Midi Output in the Item Properties of the Encoder [877] with ID 1.

WDEncoderInfoMidiOutputEnable

WDEncoderInfoMidiOutputEnable(ID)

Example:
WDEncoderInfoMidiOutputEnable(1)

Enables the Midi Output in the Item Properties of the Encoder [877] with ID 1.

WDEncoderLocation

WDEncoderLocation(ID,X,Y)

Example:
WDEncoderLocation(5,100,200)

Sets the position of the Encoder [877] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDEncoderLocationLeft

WDEncoderLocationLeft(ID,X)

Example:
WDEncoderLocationLeft(5,100)

Sets the position of the Encoder [877] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDEncoderLocationTop

WDEncoderLocationTop(ID,Y)

Example:
WDEncoderLocationTop(5,200)

Sets the position of the Encoder [877] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDEncoderSize

WDEncoderSize(ID,Width,Height)

Example:
WDEncoderSize(5,100,40)

Sets the size of the Encoder [877] with ID 5 to a width of 100px and a height of 40px.

WDEncoderSizeHeight

WDEncoderSizeHeight(ID,Height)

Example:
WDEncoderSizeHeight(5,40)

Sets the size of the Encoder [877] with ID 5 to a height of 40px but remains the current width.

WDEncoderSizeWidth

WDEncoderSizeWidth(ID,Width)

Example:
WDEncoderSizeWidth(5,100)

Sets the size of the Encoder [877] with ID 5 to a width of 100px but remains the current height.

WDEncoderUnfix

WDEncoderUnfix(ID)

Example:
WDEncoderUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Encoder [877] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDEncoderValue

WDEncoderValue(ID,Value)

Example:
WDEncoderValue(7,33)

Sets the Encoder [877] 7 to value 33 immediately.

WDEnterKioskView

WDEnterKioskModeView

Example:
WDEnterKioskView

This command activates the Kiosk View which removes all bars and menus and resizes the main GUI to fullscreen. You can also press the shortcut [F11] to toggle the Kiosk view.

The Kiosk View gives you the largest available space to arrange any controls forming an individual user interface. There are also the commands WDLeaveKioskView[1753] and WDToggleKioskView[1844]

WDEventCreate

WDEventCreate(EventName)

Example:
WDEventCreate("Show")

Creates the Event[1483] "Show".

WDEventCreateWithCategory

WDEventCreateWithCategory(EventName,Category)

Example:
WDEventCreateWithCategory("Show","Start")

Creates the Event[1483] "Show" with the Category "Start".

WDEventDelete

WDEventDelete(EventName)

Example:
WDEventDelete("Show")

Deletes the Event[1483] "Show".

WDEventDeleteAll

WDEventDeleteAll

Example:
WDEventDeleteAll

Deletes all Events[1483].

WDEventDeleteCategory

WDEventDeleteCategory(Category)

Example:
WDEventDeleteCategory("Start")

Deletes all Events[1483] with the category "Start".

WDEventDeleteObsoleteItems

WDEventDeleteObsoleteItems

Example:
WDEventDeleteObsoleteItems

Deletes passed Events [1483].

WDEventEndDate

WDEventEndDate(EventName,YYYY,MM,DD,HH,MM,SS)

Example:
WDEventEndDate("Show",2015,12,24,20,00,00)

Sets the end date of the Event [1483] "Show" to 8 p.m. at Christmas 2015.

WDEventEveryDay

WDEventEveryDay(EventName)

Example:
WDEventEveryDay("Show")

Repeats the Event [1483] "Show" every day.

WDEventEveryHour

WDEventEveryHour(EventName)

Example:
WDEventEveryHour("Show")

Repeats the Event [1483] "Show" every hour.

WDEventEveryMinute

WDEventEveryMinute(EventName)

Example:
WDEventEveryMinute("Show")

Repeats the Event [1483] "Show" every minute.

WDEventEveryMonth

WDEventEveryMonth(EventName)

Example:
WDEventEveryMonth("Show")

Repeats the Event [1483] "Show" every month.

WDEventEverySecond

WDEventEverySecond(EventName)

Example:
WDEventEverySecond("Show")

Repeats the Event [1483] "Show" every second.

WDEventEveryWeek

WDEventEveryWeek(EventName)

Example:
WDEventEveryWeek("Show")

Repeats the Event 1483 "Show" every week.

WDEventEveryYear

WDEventEveryYear(EventName)

Example:
WDEventEveryYear("Show")

Repeats the Event 1483 "Show" every year.

WDEventModeEndDate

WDEventModeEndDate(EventName)

Example:
WDEventModeEndDate("Show")

The option "Count" of the Event 1483 "Show" is set to "End Date". The Event will end at the specified end date.

WDEventModeEndless

WDEventModeEndless(EventName)

Example:
WDEventModeEndless("Show")

The option "Count" of the Event 1483 "Show" is set to "Endless". The Event will not end at the specified end date and is repeated endlessly.

WDEventModeMaxCount

WDEventModeMaxCount(EventName)

Example:
WDEventModeEndless("Show")

The option "Repeat Count Mode" of the Event 1483 "Show" is set to "Maximum Count". The Event will end after the specified amount of iterations has been reached.

WDEventMute

WDEventMute(EventName)

Example:
WDEventMute("Show")

Disables/Mutes the Event 1483 "Show".

WDEventRepeatInterval

WDEventRepeatInterval(EventName,IntervalNumber)

Example:
WDEventRepeatInterval("Show",5)

Sets the interval of the Event 1483 "Show" to 5 units (e.g. seconds, minutes, hours...).

WDEventRepeatMaximum

WDEventRepeatMaximum(EventName,NumberOfIterations)

Example:
WDEventRepeatMaximum("Show",5)

The option "Repeat Max" of the Event 1483 "Show" is set to 5 iterations.

WDEventScript

WDEventScript(EventName,Script)

Example:
WDEventScript("Show","WDFaderUp(1,3)")

Applies the command "WDFaderUp(1,3) to the Event 1483 "Show".

WDEventSingleEvent

WDEventSingleEvent(EventName)

Example:
WDEventSingleEvent("Show")

The Event 1483 "Show" will not be repeated, it is a single event.

WDEventStartDate

WDEventStartDate(EventName,YYYY,MM,DD,HH,MM,SS)

Example:
WDEventStartDate("Show",2015,12,24,20,00,00)

Sets the start date of the Event 1483 "Show" to 8 p.m. at Christmas 2015.

WDEventUnMute

WDEventUnMute(EventName)

Example:
WDEventUnMute("Show")

Activates / Unmutes the Event 1483 "Show".

WDEventViewerCssStyleDisable

WDEventViewerCssStyleDisable(ID,StyleID)

Example:
WDEventViewerCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the EventViewer [871] with ID 5.

WDEventViewerCssStyleEdit

WDEventViewerCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDEventViewerCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the EventViewer [871] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDEventViewerCssStyleEnable

WDEventViewerCssStyleEnable(ID,StyleID)

Example:
WDEventViewerCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the EventViewer [871] with ID 5.

WDEventViewerFix

WDEventViewerFix(ID)

Example:
WDEventViewerFix(5)

Sets the option "Fix" for the EventViewer with ID 5, so that this EventViewer [871] is shown on every page inside WD now.

WDEventViewerGetLocationLeft

WDEventViewerGetLocationLeft(ID)

Example:
WDEventViewerGetLocationLeft(5)

This refers to the EventViewer [871] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEventViewerGetLocationLeft(1)`

WDEventViewerGetLocationTop

WDEventViewerGetLocationTop(ID)

Example:
WDEventViewerGetLocationTop(5)

This refers to the EventViewer [871] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEventViewerGetLocationTop(1)`

WDEventViewerGetSizeHeight

WDEventViewerGetSizeHeight(ID)

Example:
WDEventViewerGetSizeHeight(5)

This refers to the EventViewer [871] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEventViewerGetSizeHeight(1)`

WDEventViewerGetSizeWidth

WDEventViewerGetSizeWidth(ID)

Example:
WDEventViewerGetSizeWidth(5)

This refers to the EventViewer [871] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDEventViewerGetSizeWidth(1)`

WDEventViewerLocation

WDEventViewerLocation(ID,X,Y)

Example:
WDEventViewerLocation(5,100,200)

Sets the position of the EventViewer [871] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDEventViewerLocationLeft

WDEventViewerLocationLeft(ID,X)

Example:
WDEventViewerLocationLeft(5,100)

Sets the position of the EventViewer [871] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDEventViewerLocationTop

WDEventViewerLocationTop(ID,Y)

Example:
WDEventViewerLocationTop(5,200)

Sets the position of the EventViewer [871] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDEventViewerSize

WDEventViewerSize(ID,Width,Height)

Example:
WDEventViewerSize(5,1000,400)

Sets the size of the EventViewer [871] with ID 5 to a width of 1000px and a height of 400px. Note that the EventViewer has a minimal width of 920px and a height of 350px.

WDEventViewerSizeHeight

WDEventViewerSizeHeight(ID,Height)

Example:
WDEventViewerSizeHeight(5,400)

Sets the size of the EventViewer [871] with ID 5 to a height of 400px but remains the current width. Note that the EventViewer has a minimal width of 920px and a height of 350px.

WDEventViewerSizeWidth

WDEventViewerSizeWidth(ID,Width)

Example:
WDEventViewerSizeWidth(5,1000)

Sets the size of the EventViewer [871] with ID 5 to a width of 1000px but remains the current height. Note that the EventViewer has a minimal width of 920px and a height of 350px.

WDEventViewerUnfix

WDEventViewerUnfix(ID)

Example:
WDEventViewerUnfix(5)

Disables the option "Fix" for the EventViewer with ID 5, so that this EventViewer [871] is not shown on every page inside WD any more.

WDFaderAbortAllFades

WDFaderAbortAllFades

Example:
WDFaderAbortAllFades

Abort all current fades from Faders [874].

WDFaderAbortFade

WDFaderAbortFade(ID)

Example:
WDFaderAbortFade(1)

Aborts the fade of Fader [874] 1.

WDFaderBGImage

WDFaderBGImage(ID,File)

Example:
WDFaderBGImage(1,"C:\Christie\content\faderBG.png")

Changes the look of Fader [874] 1 - it loads the image "faderBG.png" that was saved under "C:\Christie\content" as the new background image.

WDFaderCssStyleDisable

WDFaderCssStyleDisable(ID,StyleID)

Example:
WDFaderCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Fader [874] with ID 5.

WDFaderCssStyleEdit

WDFaderCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDFaderCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Fader [874] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDFaderCssStyleEnable

WDFaderCssStyleEnable(ID,StyleID)

Example:
WDFaderCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Fader [874] with ID 5.

WDFaderDown

WDFaderDown(ID,Seconds)

Example:
WDFaderDown(7,2)

Fades Fader [874] 7 in 2sec down to its minimum value.

WDFaderFix

WDFaderFix(ID)

Example:
WDFaderFix(2)

Sets the option "Fix" for Fader 2, so that this Fader [874] is shown on every page inside WD now.

WDFaderGetLocationLeft

WDFaderGetLocationLeft(ID)

Example:
WDFaderGetLocationLeft(5)

This refers to the Fader [874] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDFaderGetLocationLeft(1)`

WDFaderGetLocationTop

WDFaderGetLocationTop(ID)

Example:
WDFaderGetLocationTop(5)

This refers to the Fader [874] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDFaderGetLocationTop(1)`

WDFaderGetSizeHeight

WDFaderGetSizeHeight(ID)

Example:
WDFaderGetSizeHeight(5)

This refers to the Fader [874] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDFaderGetSizeHeight(1)`

WDFaderGetSizeWidth

WDFaderGetSizeWidth(ID)

Example:
WDFaderGetSizeWidth(5)

This refers to the Fader [874] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDFaderGetSizeWidth(1)`

WDFaderGoDown

WDFaderGoDown(ID,Value)

Example:
WDFaderGoDown(7,10)

Subtracts immediately 10 values from the value of Fader [874] 7.

WDFaderGoUp

WDFaderGoUp(ID,Value)

Example:
WDFaderGoUp(7,10)

Adds immediately 10 values to the value of Fader [874] 7.

WDFaderHandleImage

WDFaderHandleImage(ID,File)

Example:
WDFaderHandleImage(1,"C:\Christie\content\faderHandle.png")

Changes the look of the Fader [874] with ID 1 - it loads the image "faderHandle.png" that was saved under "C:\Christie\content" as the new handle image.

WDFaderInfoMidiInputDisable

WDFaderInfoMidiInputDisable(ID)

Example:
WDFaderInfoMidiInputDisable(3)

Disables the MIDI input of Fader [874] 3.

WDFaderInfoMidiInputEnable

WDFaderInfoMidiInputEnable(ID)

Example:
WDFaderInfoMidiInputEnable(3)

Enables the MIDI input of Fader [874] 3.

WDFaderInfoMidiOutputDisable

WDFaderInfoMidiOutputDisable(ID)

Example:
WDFaderInfoMidiOutputDisable(3)

Disables the MIDI output of Fader [874] 3.

WDFaderInfoMidiOutputEnable

WDFaderInfoMidiOutputEnable(ID)

Example:
WDFaderInfoMidiOutputEnable(3)

Enables the MIDI output of Fader [874] 3.

WDFaderLocation

WDFaderLocation(ID,X,Y)

Example:
WDFaderLocation(5,100,200)

Sets the position of the Fader [874] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDFaderLocationLeft

WDFaderLocationLeft(ID,X)

Example:
WDFaderLocationLeft(5,100)

Sets the position of the Fader [874] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDFaderLocationTop

WDFaderLocationTop(ID,Y)

Example:
WDFaderLocationTop(5,200)

Sets the position of the Fader [874] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDFaderSize

WDFaderSize(ID,Width,Height)

Example:
WDFaderSize(5,100,40)

Sets the size of the Fader [874] with ID 5 to a width of 100px and a height of 40px.

WDFaderSizeHeight

WDFaderSizeHeight(ID,Height)

Example:
WDFaderSizeHeight(5,40)

Sets the size of the Fader [874] with ID 5 to a height of 40px but remains the current width.

WDFaderSizeWidth

WDFaderSizeWidth(ID,Width)

Example:
WDFaderSizeWidth(5,100)

Sets the size of the Fader [874] with ID 5 to a width of 100px but remains the current height.

WDFaderUnfix

WDFaderUnfix(ID)

Example:
WDFaderUnfix(2)

Disables the option "Fix" for the Fader 2, so that this Fader [874] is not shown on every page inside WD any more.

WDFaderUp

WDFaderUp(ID,Seconds)

Example:
WDFaderUp(7,2)

Fades Fader [874] 7 in 2sec up to its maximum value.

WDFaderValue

WDFaderValue(ID,Value)

Example:
WDFaderValue(7,33)

Sets the Fader [874] 7 to value 33 immediately.

WDFadeToRelativeValue

WDFadeToRelativeValue(ID,Seconds,Value)

Example:
WDFadeToRelativeValue(6,13,41)

Fades Fader [874] 6 in 13sec to the result of its current value plus 41.

WDFadeToRelativeValueSmooth

WDFadeToRelativeValueSmooth(ID,Time,Value)

Example:
WDFadeToRelativeValueSmooth(6,13,41)

Fades Fader [874] 6 smoothly in 13sec to the result of its current value plus 41. "Smoothly" means that the motion is eased in and out, on start and stop of each fade.

WDFadeToValue

WDFadeToValue(ID,Seconds,Value)

Example:
WDFadeToValue(1,4,200)

Fades Fader [874] 1 in 4s to the value 200.

WDFadeToValueSmooth

WDFadeToValueSmooth(ID,Time,Value)

Example:
WDFadeToValueSmooth(2,5,170)

Fades Fader [874] 2 smoothly in 5sec up to value of 170. "Smoothly" means that the motion is eased in and out, on start and stop of each fade.

WDFunction

WDFunction(FunctionName,Function Parameters(optional))

Example:
WDFunction("Fader Jump",5)

Executes Function "Fader Jump" and passes "5" as input parameter, if it was created in the Function Tool [1897] before.

WDGaugeCssStyleDisable

WDGaugeCssStyleDisable(ID,StyleID)

Example:
WDGaugeCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Gauge [856] with ID 5.

WDGaugeCssStyleEdit

WDGaugeCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDGaugeCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Gauge [856] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDGaugeCssStyleEnable

WDGaugeCssStyleEnable(ID,StyleID)

Example:
WDGaugeCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Gauge [856] with ID 5.

WDGaugeFix

WDGaugeFix(ID)

Example:
WDGaugeFix(5)

This activates the option "Fix" in the Item Properties of the Gauge [856] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDGaugeGetCaptionFontFamily

WDGaugeGetCaptionFontFamily(ID)

Example:
varString = WDGaugeGetCaptionFontFamily(5)

This refers to the Gauge [856] with ID 5 and returns the caption`s current font family name as a string. The "Caption" is the optional text that appears in the center of the Gauge.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Gauge via a script like:
`WDGaugeSetCaptionFontFamily(5,WDGaugeGetCaptionFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDGaugeGetCaptionFontFamily(5))`

WDGaugeGetCaptionFontSize

WDGaugeGetCaptionFontSize(ID)

Example:
varDouble = WDGaugeGetCaptionFontSize(5)

This refers to the Gauge [856] with ID 5 and returns the caption`s current font size in "pt" as a double. The "Caption" is the optional text that appears in the center of the Gauge.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Gauge via a script like:
`WDGaugeSetCaptionFontSize(5,WDGaugeGetCaptionFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDGaugeGetCaptionFontSize(5))`

WDGaugeGetLabelFontFamily

WDGaugeGetLabelFontFamily(ID)

Example:
varString = WDGaugeGetLabelFontFamily(5)

This refers to the Gauge [856] with ID 5 and returns its current font family name from the labels as a string. The labels refer only to the numbers, whilst "Caption" refers to the optional text appearing in the center of the Gauge.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Gauge via a script like:
`WDGaugeSetLabelFontFamily(5,WDGaugeGetLabelFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDGaugeGetLabelFontFamily(5))`

WDGaugeGetLabelFontSize

WDGaugeGetLabelFontSize(ID)

Example:
varDouble = WDGaugeGetLabelFontSize(5)

This refers to the Gauge [856] with ID 5 and returns its current font size from the labels in "pt" as a double. The labels refer only to the numbers, whilst "Caption" refers to the optional text appearing in the center of the

Gauge.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Gauge via a script like:
`WDGaugeSetLabelFontSize(5,WDGaugeGetLabelFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDGaugeGetLabelFontSize(5))`

WDGaugeGetLocationLeft

WDGaugeGetLocationLeft(ID)

Example:
WDGaugeGetLocationLeft(5)

This refers to the Gauge [856] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGaugeGetLocationLeft(1)`

WDGaugeGetLocationTop

WDGaugeGetLocationTop(ID)

Example:
WDGaugeGetLocationTop(5)

This refers to the Gauge [856] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGaugeGetLocationTop(1)`

WDGaugeGetSizeHeight

WDGaugeGetSizeHeight(ID)

Example:
WDGaugeGetSizeHeight(5)

This refers to the Gauge [856] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGaugeGetSizeHeight(1)`

WDGaugeGetSizeWidth

WDGaugeGetSizeWidth(ID)

Example:
WDGaugeGetSizeWidth(5)

This refers to the Gauge [856] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGaugeGetSizeWidth(1)`

WDGaugeLocation

WDGaugeLocation(ID,X,Y)

Example:
WDGaugeLocation(5,100,200)

Sets the position of the Gauge [856] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDGaugeLocationLeft

WDGaugeLocationLeft(ID,X)

Example:
WDGaugeLocationLeft(5,100)

Sets the position of the Gauge [856] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDGaugeLocationTop

WDGaugeLocationTop(ID,Y)

Example:
WDGaugeLocationTop(5,200)

Sets the position of the Gauge [856] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDGaugeSetCaptionFontFamily

WDGaugeSetCaptionFontFamily(ID,FontName)

Example:
WDGaugeSetCaptionFontFamily(5,"Arial")

This refers to the Gauge [856] with ID 5 and sets the caption`s font family to "Arial". The "Caption" is the optional text that appears in the center of the Gauge.
If you like to copy and paste the font from a Widget, e.g. another Gauge, you can use a script like:
`WDGaugeSetCaptionFontFamily(5,WDGaugeGetCaptionFontFamily(1))`

WDGaugeSetCaptionFontSize

WDGaugeSetCaptionFontSize(ID,FontSize)

Example:
WDGaugeSetCaptionFontSize(5,20)

This refers to the Gauge [856] with ID 5 and sets the caption`s font size to 20pt. The "Caption" is the optional text that appears in the center of the Gauge.
If you like to copy and paste the font size from a Widget, e.g. another Gauge, you can use a script like:
`WDGaugeSetCaptionFontSize(5,WDGaugeGetCaptionFontSize(1))`

WDGaugeSetLabelFontFamily

WDGaugeSetLabelFontFamily(ID,FontName)

Example:
WDGaugeSetLabelFontFamily(5,"Arial")

This refers to the Gauge [856] with ID 5 and sets the font family for the labels to "Arial". The labels refer only to the numbers, whilst "Caption" refers to the optional text appearing in the center of the Gauge.

If you like to copy and paste the font from a Widget, e.g. another Gauge, you can use a script like:
`WDGaugeSetLabelFontFamily(5,WDGaugeGetLabelFontFamily(1))`

WDGaugeSetLabelFontSize

WDGaugeSetLabelFontSize(ID,FontSize)

Example:
WDGaugeSetLabelFontSize(5,20)

This refers to the Gauge [856] with ID 5 and sets from the font size for the labels to 20pt. The labels refer only to the numbers, whilst "Caption" refers to the optional text appearing in the center of the Gauge.
If you like to copy and paste the font size from a Widget, e.g. another Gauge, you can use a script like:
`WDGaugeSetLabelFontSize(5,WDGaugeGetLabelFontSize(1))`

WDGaugeSize

WDGaugeSize(ID,Width,Height)

Example:
WDGaugeSize(5,100,40)

Sets the size of the Gauge [856] with ID 5 to a width of 100px and a height of 40px.

WDGaugeSizeHeight

WDGaugeSizeHeight(ID,Height)

Example:
WDGaugeSizeHeight(5,40)

Sets the size of the Gauge [856] with ID 5 to a height of 40px but remains the current width.

WDGaugeSizeWidth

WDGaugeSizeWidth(ID,Width)

Example:
WDGaugeSizeWidth(5,100)

Sets the size of the Gauge [856] with ID 5 to a width of 100px but remains the current height.

WDGaugeUnfix

WDGaugeUnfix(ID)

Example:
WDGaugeUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Gauge [856] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDGraphDisplayCssStyleDisable

WDGraphDisplayCssStyleDisable(ID,StyleID)

Example:
WDGraphDisplayCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the GraphDisplay [858] with ID 5.

WDGraphDisplayCssStyleEdit

WDGraphDisplayCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDGraphDisplayCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the GraphDisplay [858] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDGraphDisplayCssStyleEnable

WDGraphDisplayCssStyleEnable(ID,StyleID)

Example:
WDGraphDisplayCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the GraphDisplay [858] with ID 5.

WDGraphDisplayFix

WDGraphDisplayFix(ID)

Example:
WDGraphDisplayFix(5)

This activates the option "Fix" in the Item Properties of the GraphDisplay [858] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDGraphDisplayGetFontFamily

WDGraphDisplayGetFontFamily(ID)

Example:
varString = WDGraphDisplayGetFontFamily(5)

This refers to the GraphDisplay [858] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another GraphDisplay via a script like:
`WDGraphDisplaySetFontFamily(5,WDGraphDisplayGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDGraphDisplayGetFontFamily(5))`

WDGraphDisplayGetFontSize

WDGraphDisplayGetFontSize(ID)

Example:
varDouble = WDGraphDisplayGetFontSize(5)

This refers to the GraphDisplay [858] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another GraphDisplay via a script like:
`WDGraphDisplaySetFontSize(5,WDGraphDisplayGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDGraphDisplayGetFontSize(5))`

WDGraphDisplayGetLocationLeft

WDGraphDisplayGetLocationLeft(ID)

Example:
WDGraphDisplayGetLocationLeft(5)

This refers to the GraphDisplay [858] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGraphDisplayGetLocationLeft(1)`

WDGraphDisplayGetLocationTop

WDGraphDisplayGetLocationTop(ID)

Example:
WDGraphDisplayGetLocationTop(5)

This refers to the GraphDisplay [858] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGraphDisplayGetLocationTop(1)`

WDGraphDisplayGetSizeHeight

WDGraphDisplayGetSizeHeight(ID)

Example:
WDGraphDisplayGetSizeHeight(5)

This refers to the GraphDisplay [858] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGraphDisplayGetSizeHeight(1)`

WDGraphDisplayGetSizeWidth

WDGraphDisplayGetSizeWidth(ID)

Example:
WDGraphDisplayGetSizeWidth(5)

This refers to the GraphDisplay [858] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDGraphDisplayGetSizeWidth(1)`

WDGraphDisplayLocation

WDGraphDisplayLocation(ID,X,Y)

Example:
WDGraphDisplayLocation(5,100,200)

Sets the position of the GraphDisplay [858] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDGraphDisplayLocationLeft

WDGraphDisplayLocationLeft(ID,X)

Example:
WDGraphDisplayLocationLeft(5,100)

Sets the position of the GraphDisplay [858] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDGraphDisplayLocationTop

WDGraphDisplayLocationTop(ID,Y)

Example:
WDGraphDisplayLocationTop(5,200)

Sets the position of the GraphDisplay [858] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDGraphDisplaySetFontFamily

WDGraphDisplaySetFontFamily(ID,FontName)

Example:
WDGraphDisplaySetFontFamily(5,"Arial")

This refers to the GraphDisplay [858] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another GraphDisplay, you can use a script like:
`WDGraphDisplaySetFontFamily(5,WDGraphDisplayGetFontFamily(1))`

WDGraphDisplaySetFontSize

WDGraphDisplaySetFontSize(ID,FontSize)

Example:
WDGraphDisplaySetFontSize(5,20)

This refers to the GraphDisplay [858] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another GraphDisplay, you can use a script like:
`WDGraphDisplaySetFontSize(5,WDGraphDisplayGetFontSize(1))`

WDGraphDisplaySize

WDGraphDisplaySize(ID,Width,Height)

Example:
WDGraphDisplaySize(5,100,40)

Sets the size of the GraphDisplay [858] with ID 5 to a width of 100px and a height of 40px.

WDGraphDisplaySizeHeight

WDGraphDisplaySizeHeight(ID,Height)

Example:
WDGraphDisplaySizeHeight(5,40)

Sets the size of the GraphDisplay [858] with ID 5 to a height of 40px but remains the current width.

WDGraphDisplaySizeWidth

WDGraphDisplaySizeWidth(ID,Width)

Example:
WDGraphDisplaySizeWidth(5,100)

Sets the size of the GraphDisplay [858] with ID 5 to a width of 100px but remains the current height.

WDGraphDisplayUnfix

WDGraphDisplayUnfix(ID)

Example:
WDGraphDisplayUnfix(5)

This deactivates the option "Fix" in the Item Properties of the GraphDisplay [858] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDHTTPRequest

WDHTTPRequest(Request)

Example:
WDHTTPRequest("http://192.168.50.40/cgi-bin/shutter_on")

This allows to send single HTTP commands via scripts directly to a web server. It is not necessary to establish the connection from Widget Designer to the external web server using the Connection Manager but they need to be connected via a switch for example.
The HTTP Request is useful when working with an external web server. It does not apply to the integrated WD Web Server feature [1929]. As well, you may control external devices, e.g. a projector supporting the HTTP protocol. This is an alternative to controlling them via a RS232 / Serial protocol.
If the external web server requires a username and / or a passwort and supports a HTTP Basic Authentication (in most browsers that gives you a pop-up window asking for credentials) then you can embed the user in the URL like this

```
WDHTTPRequest("http://USERNAME@192.168.50.40/cgi-bin/shutter_on")
WDHTTPRequest("http://USERNAME:PASSWORD@192.168.50.40/cgi-bin/shutter_on")
```

USERNAME and PASSWORD have to be replaced. Alternatively, you can also use the command `WDHTTPRequestWithLogin` [1735].

The commands `HTTPRequestToVar` [1544] and `HTTPRequestToVarWithLogin` [1545] send a request and write the answer from the web server to a variable in WD.

WDHTTPRequestWithLogin

WDHTTPRequestWithLogin(Username,Password,URL)

Example:
WDHTTPRequestWithLogin("User1","Password1","http://192.168.50.40/cgi-bin/shutter_on")

This allows to send single HTTP commands via scripts directly to a web server including required login information. It is not necessary to establish the connection from Widget Designer to the external web server using the Connection Manager but they need to be connected via a switch for example.

The HTTP Request is useful when working with an external web server. It does not apply to the integrated WD Web Server feature. As well, you may control external devices, e.g. a projector supporting the HTTP protocol. This is an alternative to controlling them via a RS232 / Serial protocol.

Use this command if the external web server requires a username and / or a passwort and supports a HTTP Basic Authentication (in most browsers that gives you a pop-up window asking for credentials). Use the command WDHTTPRequest 1735 if no login information is needed.

The commands HTTPRequestToVar 1544 and HTTPRequestToVarWithLogin 1545 send a request and write the answer from the web server to a variable in WD.

WDImageLoaderAddTextFromLabel

WDImageLoaderAddTextFromLabel(ID,LabelID)

Example:
WDImageLoaderAddTextFromLabel(6,3)

Adds the text from Label 3 to the current Label of the Image Loader 833 button with ID 6.

WDImageLoaderClick

WDImageLoaderClick(ID)

Example:
WDImageLoaderClick(5)

Clicks the Image Loader 833 button with ID 5.

If you like to execute this command addressing many buttons, the chapter "For loop" and "Project and Context Member" show some interesting examples.

WDImageLoaderClickImage

WDImageLoaderClickImage(ID,File)

Example:
WDImageLoaderClickImage(1,"C:\Christie\content\buttonImage10.jpg")

Changes the look of the Image Loader Button 833 with ID 1 - it loads the image "buttonImage10.jpg" that was saved under "C:\Christie\content" and displays it when the Image Loader is in the mode "Click".

WDImageLoaderClickImageResource

WDImageLoaderClickImageResource(ID,ResourceName)

Example:
WDImageLoaderClickImageResource(5,"Default\Button\Lock")

This sets the image for the clicked status of the Image Loader [833] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDImageLoaderCssStyleDisable

WDImageLoaderCssStyleDisable(ID,StyleID)

Example:
WDImageLoaderCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Image Loader [833] button with ID 5.

WDImageLoaderCssStyleEdit

WDImageLoaderCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDImageLoaderCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Image Loader [833] button with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDImageLoaderCssStyleEnable

WDImageLoaderCssStyleEnable(ID,StyleID)

Example:
WDImageLoaderCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Image Loader [833] button with ID 5.

WDImageLoaderFix

WDImageLoaderFix(ID)

Example:
WDImageLoaderFix(5)

This activates the option "Fix" in the Item Properties of the Image Loader [833] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDImageLoaderGetLocationLeft

WDImageLoaderGetLocationLeft(ID)

Example:
WDImageLoaderGetLocationLeft(5)

This refers to the Image Loader [833] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDImageLoaderGetLocationLeft(1)`

WDImageLoaderGetLocationTop

WDImageLoaderGetLocationTop(ID)

Example:
WDImageLoaderGetLocationTop(5)

This refers to the Image Loader [833] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDImageLoaderGetLocationTop(1)`

WDImageLoaderGetSizeHeight

WDImageLoaderGetSizeHeight(ID)

Example:
WDImageLoaderGetSizeHeight(5)

This refers to the Image Loader [833] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDImageLoaderGetSizeHeight(1)`

WDImageLoaderGetSizeWidth

WDImageLoaderGetSizeWidth(ID)

Example:
WDImageLoaderGetSizeWidth(5)

This refers to the Image Loader [833] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDImageLoaderGetSizeWidth(1)`

WDImageLoaderHighlightImage

WDImageLoaderHighlightImage(ID,Existing file)

Example:
WDImageLoaderHighlightImage(5,"C:\Christie\clicked.png")

This sets the image for the highlighted status of the Image Loader [833] button with ID 5 to the image saved under the specified path.

WDImageLoaderHighlightImageResource

WDImageLoaderHighlightImageResource(ID,Text)

Example:
WDImageLoaderHighlightImageResource(5,"Default\Button\Lock")

This sets the image for the highlighted status of the Image Loader [833] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDImageLoaderLabel

WDImageLoaderLabel(ID,Text)

Example:
WDImageLoaderLabel(5,"Show")

Labels the Image Loader [833] button with ID 5 with the word "Show".

WDImageLoaderLabelColor

WDImageLoaderLabelColor(ID,R,G,B)

Example:
WDImageLoaderLabelColor(5,255,90,0)

Changes the label color of the Image Loader [833] button with ID 5 to orange (Red=255, Green=90, Blue=0). The values for R,G,B range from 0 to 255.

WDImageLoaderLocation

WDImageLoaderLocation(ID,X,Y)

Example:
WDImageLoaderLocation(5,100,200)

Sets the position of the Image Loader [833] button with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDImageLoaderLocationLeft

WDImageLoaderLocationLeft(ID,X)

Example:
WDImageLoaderLocationLeft(5,100)

Sets the position of the Image Loader [833] button with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDImageLoaderLocationTop

WDImageLoaderLocationTop(ID,Y)

Example:
WDImageLoaderLocationTop(5,200)

Sets the position of the Image Loader [833] button with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDImageLoaderReleaseImage

WDImageLoaderReleaseImage(ID,Existing file)

Example:
WDImageLoaderReleaseImage(5,"C:\Christie\clicked.png")

This sets the image for the released status of the Image Loader [833] button with ID 5 to the image saved under the specified path.

WDImageLoaderReleaseImageResource

WDImageLoaderReleaseImageResource(ID,Text)

Example:
WDImageLoaderReleaseImageResource(5,"Default\Button\Lock")

This sets the image for the released status of the Image Loader [833] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDImageLoaderSize

WDImageLoaderSize(ID,Width,Height)

Example:
WDImageLoaderSize(5,100,40)

Sets the size of the Image Loader [833] button with ID 5 to a width of 100px and a height of 40px.

WDImageLoaderSizeHeight

WDImageLoaderSizeHeight(ID,Height)

Example:
WDImageLoaderSizeHeight(5,40)

Sets the size of the Image Loader [833] button with ID 5 to a height of 40px but remains the current width.

WDImageLoaderSizeWidth

WDImageLoaderSizeWidth(ID,Width)

Example:
WDImageLoaderSizeWidth(5,100)

Sets the size of the Image Loader [833] button with ID 5 to a width of 100px but remains the current height.

WDImageLoaderTextFromLabel

WDImageLoaderTextFromLabel(ID,LabelID)

Example:
WDImageLoaderTextFromLabel(6,3)

Labels the Image Loader [833] button with ID 6 with the text from Label 3.

WDImageLoaderTint

WDImageLoaderTint(ID,R,G,B)

Example:
WDImageLoaderTint(5,255,180,0)

Changes the button color of the Image Loader [833] button with ID 5 to orange (Red=255, Green=180, Blue=0). The values for R,G,B range from 0 to 255.

WDImageLoaderUnfix

WDImageLoaderUnfix(ID)

Example:
WDImageLoaderUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Image Loader [833] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDInputboxAddTextFromLabel

WDInputboxAddTextFromLabel(ID,LabelID)

Example:
WDInputboxAddTextFromLabel(1,4)

Adds the text from Label [888] 4 to the Input Box [886] with the ID 1.

WDInputboxAddTextFromLastSms

WDInputboxAddTextFromLastSms(ID)

Example:
WDInputboxAddTextFromLastSms(3)

Adds the text from the last incoming SMS [1494] to the Input Box [886] with the ID 3.

WDInputboxAddTextFromRssFeed

WDInputboxAddTextFromRssFeed(ID,RSSFeedID)

Example:
WDInputboxAddTextFromRssFeed(2,12)

Adds the text from the RSS [1493] feed with the ID 12 to the Input Box [886] with the ID 2.

WDInputboxAddTextFromRssFeedBody

WDInputboxAddTextFromRssFeedBody(ID,RSSFeedID)

Example:
WDInputboxAddTextFromRssFeedBody(3,8)

Adds the text from the RSS [1493] feed body with the ID 8 to the Input Box [886] with the ID 3.

WDInputboxAddTextFromRssFeedTitle

WDInputboxAddTextFromRssFeedTitle(ID,RSSFeedID)

Example:
WDInputboxAddTextFromRssFeedTitle(2,5)

Adds the text from the RSS [1493] feed title with the ID 5 to the Input Box [886] with the ID 2.

WDInputboxAddTextFromSms

WDInputboxAddTextFromSms(ID,SMSID)

Example:
WDInputboxAddTextFromSms(1,8)

Adds the text from the SMS [1494] with the ID 8 to the Input Box [886] with the ID 1.

WDInputboxAddTextFromTextBox

WDInputboxAddTextFromTextBox(ID,TextBoxID)

Example:
WDInputboxAddTextFromTextBox(2,7)

Adds the text from the Text Box [923] with the ID 7 to the Input Box [886] with the ID 2.

WDInputboxAppend

WDInputboxAppend(ID)

Example:
WDInputBoxAppend(2,"Hello")

Adds the text "Hello" to the Input Box [886] 2.

Example 2:
WDInputBoxAppend(2,varString)

The second example uses a variable [1900] name. In this case, the variable's content/value is added to the Input Box [886].

WDInputboxAppendFromFile

WDInputboxAppendFromFile(ID,File)

Example:
WDInputboxAppendFromFile(2,"C:\Christie\content\myFile.txt")

Adds the text from the file "myFile.txt" which is located under the path "C:\Christie\content" to the Input Box [886] 2.

In case the imported data does not show the same letters etc., check the encoding of your file. Widget Designer imports all common encodings except ANSI. That is: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8.

WDInputboxAppendTextOnly

WDInputboxAppendTextOnly(ID)

Example:
WDInputBoxAppendTextOnly(2,"Hello")

Adds the text "Hello" to the Input Box 886 2.

Example 2:
WDInputBoxAppendTextOnly(2,varString)

The second example uses a variable 1900 name. In this case, the variable's content/value is added to the Input Box 886.

WDInputboxAppendToFile

WDInputboxAppendToFile(ID,File)

Example:
WDInputboxAppendToFile(1,"C:\Christie\content\myFile.txt")

Adds the text from the Input Box 886 1 to the file "myFile.txt" which is located under the path "C:\Christie\content".

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.

WDInputboxClear

WDInputboxClear(ID)

Example:
WDInputboxClear(1)

Clears all text from the Input Box 886 1.

WDInputBoxCssStyleDisable

WDInputBoxCssStyleDisable(ID,StyleID)

Example:
WDInputBoxCssStyleDisable(5,2)

Disables the CSS Style 814 with ID 2 in the Item Properties of the Input Box 886 with ID 5.

WDInputBoxCssStyleEdit

WDInputBoxCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDInputBoxCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style 814 with ID 2 in the Item Properties of the Input Box 886 with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDInputBoxCssStyleEnable

WDInputBoxCssStyleEnable(ID,StyleID)

Example:
WDInputBoxCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Input Box [886] with ID 5.

WDInputboxExecuteAsScript

WDInputboxExecuteAsScript(ID)

Example:
WDInputboxExecuteAsScript(2)

Executes the text in Input Box [886] 1 as a command. You may use the command `WDInputboxAppend` [1742] to compose the script from single parts.

WDInputBoxFix

WDInputBoxFix(ID)

Example:
WDInputBoxFix(5)

This activates the option "Fix" in the Item Properties of the Input Box [886] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDInputboxFocus

WDInputboxFocus(ID)

Example:
WDInputboxFocus(1)

This sets the Input Box [886] with ID 1 into the focus, meaning that the next keyboard input will be added to this Input Box [886].

WDInputboxFromFile

WDInputboxFromFile(ID,File)

Example:
WDInputboxFromFile(2,"C:\Christie\content\name.txt")

Adds the text from the text file "name.txt" which is located under the path "C:\Christie\content" to the Input Box [886] 2.

In case the imported data does not show the same letters etc., check the encoding of your file. Widget Designer imports all common encodings except ANSI. That is: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8.

WDInputBoxGetFontFamily

WDInputBoxGetFontFamily(ID)

Example:
varString = WDInputBoxGetFontFamily(5)

This refers to the Input Box [886] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Input Box via a script like:
`WDInputBoxSetFontFamily(5,WDInputBoxGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDInputBoxGetFontFamily(5))`

WDInputBoxGetFontSize

WDInputBoxGetFontSize(ID)

Example:
varDouble = WDInputBoxGetFontSize(5)

This refers to the Input Box [886] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Input Box via a script like:
`WDInputBoxSetFontSize(5,WDInputBoxGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDInputBoxGetFontSize(5))`

WDInputBoxGetLocationLeft

WDInputBoxGetLocationLeft(ID)

Example:
WDInputBoxGetLocationLeft(5)

This refers to the Input Box [886] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDInputBoxGetLocationLeft(1)`

WDInputBoxGetLocationTop

WDInputBoxGetLocationTop(ID)

Example:
WDInputBoxGetLocationTop(5)

This refers to the Input Box [886] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDInputBoxGetLocationTop(1)`

WDInputBoxGetSizeHeight

WDInputBoxGetSizeHeight(ID)

Example:
WDInputBoxGetSizeHeight(5)

This refers to the Input Box [886] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDInputBoxGetSizeHeight(1)`

WDInputBoxGetSizeWidth

WDInputBoxGetSizeWidth(ID)

Example:
WDInputBoxGetSizeWidth(5)

This refers to the Input Box [886] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDInputBoxGetSizeWidth(1)`

WDInputBoxLocation

WDInputBoxLocation(ID,X,Y)

Example:
WDInputBoxLocation(5,100,200)

Sets the position of the Input Box [886] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDInputBoxLocationLeft

WDInputBoxLocationLeft(ID,X)

Example:
WDInputBoxLocationLeft(5,100)

Sets the position of the Input Box [886] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDInputBoxLocationTop

WDInputBoxLocationTop(ID,Y)

Example:
WDInputBoxLocationTop(5,200)

Sets the position of the Input Box [886] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDInputboxRefresh

WDInputboxRefresh(ID)

Example:
WDInputboxRefresh(3)

Updates the text of Input Box [886] 3. This is useful when the Input Box [886] Input Node is used and the Input Box [886] ID inside the node is changed via a command.

WDInputBoxSetBackgroundColor

WDInputBoxSetBackgroundColor(ID,R,G,B)

Example:
WDInputBoxSetBackgroundColor(1,0,120,150)

This changes the background color of the Input Box [886] with ID 1 to a greenish blue (Red=0, Green=120, Blue=150). The values for R,G,B range from 0 to 255.

WDInputBoxSetFontFamily

WDInputBoxSetFontFamily(ID,FontName)

Example:
WDInputBoxSetFontFamily(5,"Arial")

This refers to the Input Box [886] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Input Box, you can use a script like:
`WDInputBoxSetFontFamily(5,WDInputBoxGetFontFamily(1))`

WDInputBoxSetFontSize

WDInputBoxSetFontSize(ID,FontSize)

Example:
WDInputBoxSetFontSize(5,20)

This refers to the Input Box [886] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Input Box, you can use a script like:
`WDInputBoxSetFontSize(5,WDInputBoxGetFontSize(1))`

WDInputBoxSetTextColor

WDInputBoxSetTextColor(ID,R,G,B)

Example:
WDInputBoxSetTextColor(1,255,150,0)

This changes the text color of the Input Box [886] with ID 1 to orange (Red=255, Green=150, Blue=0). The values for R,G,B range from 0 to 255.

WDInputBoxSize

WDInputBoxSize(ID,Width,Height)

Example:
WDInputBoxSize(5,100,40)

Sets the size of the Input Box [886] with ID 5 to a width of 100px and a height of 40px.

WDInputBoxSizeHeight

WDInputBoxSizeHeight(ID,Height)

Example:
WDInputBoxSizeHeight(5,40)

Sets the size of the Input Box [886] with ID 5 to a height of 40px but remains the current width.

WDInputBoxSizeWidth

WDInputBoxSizeWidth(ID,Width)

Example:
WDInputBoxSizeWidth(5,100)

Sets the size of the Input Box [886] with ID 5 to a width of 100px but remains the current height.

WDInputboxToFile

WDInputboxToFile(ID,File)

Example:
WDInputboxToFile(2,"C:\Christie\content\myFile.txt")

This creates a new text file "myFile" under the path "C:\Christie\content" and adds the text from Input Box [886] with the ID 2 to it.

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.

WDInputboxTrimEndToTotalCount

WDInputboxTrimEndToTotalCount(ID,Count)

Example:
WDInputboxTrimEndToTotalCount(1,5)

This removes all characters from Input Box [886] with ID 1, except the first 5 characters.

WDInputBoxUnfix

WDInputBoxUnfix(ID)

Example:
WDInputBoxUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Input Box [886] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDKeyboardShortcutsDisable

WDKeyboardShortcutsDisable

Example:
WDKeyboardShortcutsDisable

Disables the use of Keyboard Shortcuts set up in the Keyboard Shortcut Editor [1484].

WDKeyboardShortcutsEnable

WDKeyboardShortcutsEnable

Example:
WDKeyboardShortcutsEnable

Enables the use of Keyboard Shortcuts set up in the Keyboard Shortcut Editor [1484].

WDLabelCssStyleDisable

WDLabelCssStyleDisable(ID,StyleID)

Example:
WDLabelCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Label [888] with ID 5.

WDLabelCssStyleEdit

WDLabelCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDLabelCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Label [888] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDLabelCssStyleEnable

WDLabelCssStyleEnable(ID,StyleID)

Example:
WDLabelCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Label [888] with ID 5.

WDLabelFix

WDLabelFix(ID)

Example:
WDLabelFix(5)

This activates the option "Fix" in the Item Properties of the Label [888] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDLabelFlashInterval

WDLabelFlashInterval(ID,Value)

Example:
WDLabelFlashInterval(10,2000)

Sets the flash interval of Label [888] 10 to 2 seconds (2000 ms).

WDLabelGetFontFamily

WDLabelGetFontFamily(ID)

Example:
varString = WDLabelGetFontFamily(5)

This refers to the Label [888] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Label via a script like:
`WDLabelSetFontFamily(5,WDLabelGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDLabelGetFontFamily(5))`

WDLabelGetFontSize

WDLabelGetFontSize(ID)

Example:
varDouble = WDLabelGetFontSize(5)

This refers to the Label [888] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Label via a script like:
`WDLabelSetFontSize(5,WDLabelGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDLabelGetFontSize(5))`

WDLabelGetLocationLeft

WDLabelGetLocationLeft(ID)

Example:
WDLabelGetLocationLeft(5)

This refers to the Label [888] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDLabelGetLocationLeft(1)`

WDLabelGetLocationTop

WDLabelGetLocationTop(ID)

Example:
WDLabelGetLocationTop(5)

This refers to the Label [888] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDLabelGetLocationTop(1)`

WDLabelGetSizeHeight

WDLabelGetSizeHeight(ID)

Example:
WDLabelGetSizeHeight(5)

This refers to the Label [888] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDLabelGetSizeHeight(1)`

WDLabelGetSizeWidth

WDLabelGetSizeWidth(ID)

Example:
WDLabelGetSizeWidth(5)

This refers to the Label [888] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDLabelGetSizeWidth(1)`

WDLabelLocation

WDLabelLocation(ID,X,Y)

Example:
WDLabelLocation(5,100,200)

Sets the position of the Label [888] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDLabelLocationLeft

WDLabelLocationLeft(ID,X)

Example:
WDLabelLocationLeft(5,100)

Sets the position of the Label [888] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDLabelLocationTop

WDLabelLocationTop(ID,Y)

Example:
WDLabelLocationTop(5,200)

Sets the position of the Label [888] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDLabelSetFontFamily

WDLabelSetFontFamily(ID,FontName)

Example:
WDLabelSetFontFamily(5,"Arial")

This refers to the Label [888] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Label, you can use a script like:
`WDLabelSetFontFamily(5,WDLabelGetFontFamily(1))`

WDLabelSetFontSize

WDLabelSetFontSize(ID,FontSize)

Example:
WDLabelSetFontSize(5,20)

This refers to the Label [888] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Label, you can use a script like:
`WDLabelSetFontSize(5,WDLabelGetFontSize(1))`

WDLabelSize

WDLabelSize(ID,Width,Height)

Example:
WDLabelSize(5,100,40)

Sets the size of the Label [888] with ID 5 to a width of 100px and a height of 40px.

WDLabelSizeHeight

WDLabelSizeHeight(ID,Height)

Example:
WDLabelSizeHeight(5,40)

Sets the size of the Label [888] with ID 5 to a height of 40px but remains the current width.

WDLabelSizeWidth

WDLabelSizeWidth(ID,Width)

Example:
WDLabelSizeWidth(5,100)

Sets the size of the Label [888] with ID 5 to a width of 100px but remains the current height.

WDLabelStartFlash

WDLabelStartFlash(ID)

Example:
WDLabelStartFlash(10)

Will start flashing up Label [888] 10.

WDLabelStopFlash

WDLabelStopFlash(ID)

Example:
WDLabelStopFlash(10)

Will stop flashing up Label [888] 10.

WDLabelText

WDLabelText(ID,Text)

Example:
WDLabelText(3,"Flip")

Changes the text of Label [888] 3 to the word Flip.

WDLabelTextColor

WDLabelTextColor(ID,R,G,B)

Example:
WDLabelTextColor(3,255,90,0)

Changes the color of Label [888] 3 to orange. The values for R,G,B range from 0 to 255.

WDLabelTextSubString

WDLabelTextSubString(ID,StartIndex,Length)

Example:
WDLabelTextSubString(1,2,2)

Removes any text from Label [888] 1 except those letters that are within the series starting at the second index for the length of 2 positions. Note that the indexing starts with "0" at the first symbol.

E.g.: Label text = Dewdrop; resulting sub string = wd

WDLabelTextTrimStart

WDLabelTextTrimStart(ID,Count)

Example:
WDLabelTextTrimStart(5,3)

Removes the first 3 characters of the text of Label [888] 5. E.g.: Label text = Dewdrop; resulting sub string = drop

WDLabelUnfix

WDLabelUnfix(ID)

Example:
WDLabelUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Label [888] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDLeaveKioskView

WDLeaveKioskView

Example:
WDLeaveKioskView

This command deactivates the Kiosk View and activates the Maximized View. All bars and menus are back again and the main GUI is resized to a maximized Window. You can also press the shortcut [F11] to toggle the Kiosk view.

There are also the commands WDEnterKioskView [1714] and WDToggleKioskView [1844]

WDListViewClearCells

WDListViewClearCells(ID)

Example:
WDListViewClearCells(1)

Clears all cells of List View [890] 1.

WDListViewCssStyleDisable

WDListViewCssStyleDisable(ID,StyleID)

Example:
WDListViewCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the List View [890] with ID 5.

WDListViewCssStyleEdit

WDListViewCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDListViewCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the List View [890] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDListViewCssStyleEnable

WDListViewCssStyleEnable(ID,StyleID)

Example:
WDListViewCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the List View [890] with ID 5.

WDListViewDeleteRows

WDListViewDeleteRows(ID,Row,Count)

Example:
WDListViewDeleteRows(1,5,3)

This deletes three rows starting with the 5th row in the List View [890] with ID 1.

WDListViewFix

WDListViewFix(ID)

Example:
WDListViewFix(5)

This activates the option "Fix" in the Item Properties of the List View [890] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDListViewGetCell

WDListViewGetCell(ID,Column,Row)

Example:
varString = WDListViewGetCell(1,2,5)

This command returns the cell value of List View [890] 1 in column 2 and row 5. The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also write it into a Widget, e.g. a Label [888] via the script: `Label1.Text = WDListViewGetCell(1,2,5)`
or simply display it in the Debug Logger [812]: `DebugMessage(WDListViewGetCell(1,2,5))`

WDListViewGetFontFamily

WDListViewGetFontFamily(ID)

Example:
varString = WDListViewGetFontFamily(5)

This refers to the List View [890] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another List View via a script like:
`WDListViewSetFontFamily(5,WDListViewGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDListViewGetFontFamily(5))`

WDListViewGetFontSize

WDListViewGetFontSize(ID)

Example:
varDouble = WDListViewGetFontSize(5)

This refers to the List View [890] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another List View via a script like:
`WDListViewSetFontSize(5,WDListViewGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDListViewGetFontSize(5))`

WDListViewGetLocationLeft

WDListViewGetLocationLeft(ID)

Example:
WDListViewGetLocationLeft(5)

This refers to the List View [890] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0).
You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDListViewGetLocationLeft(1)`

WDListViewGetLocationTop

WDListViewGetLocationTop(ID)

Example:
WDListViewGetLocationTop(5)

This refers to the List View [890] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDListViewGetLocationTop(1)`

WDListViewGetSizeHeight

WDListViewGetSizeHeight(ID)

Example:
WDListViewGetSizeHeight(5)

This refers to the List View [890] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDListViewGetSizeHeight(1)`

WDListViewGetSizeWidth

WDListViewGetSizeWidth(ID)

Example:
WDListViewGetSizeWidth(5)

This refers to the List View [890] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDListViewGetSizeWidth(1)`

WDListViewInsertRows

WDListViewInsertRows(ID,Row,Count)

Example:
WDListViewInsertRows(1,5,3)

This inserts three rows before the 5th row in the List View [890] with ID 1.

WDListViewLoadCSVFile

WDListViewLoadCSVFile(ID,FileName,Separator)

Example:
WDListViewLoadCSVFile(1,"C:\Christie\content\measured_data.csv","|")

Loads the data of the CSV file "measured_data.csv" into the ListView [890] 1 and separates the data sets from the CSV file with a "|" (pipe or vertical line) symbol.

Note: You may use any symbol as a separator, except the space " ".

In case the imported data does not show the same letters etc., check the encoding of your file. With this command, Widget Designer imports all common encodings except ANSI. That is: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8.
The alternative command `WDListViewLoadUCS2CSVFile` [1757], also loads ANSI encoded data

WDListViewLoadCSVFileExt

WDListViewLoadCSVFileExt(ID,FileName,Interval,LoadItemCount,Separator)

Example:
WDListViewLoadCSVFileExt(1,"C:\Christie\content\measured_data.csv",500,500,"|")

.

WDListViewLoadUCS2CSVFile

WDListViewLoadUCS2CSVFile(ID,FileName,Separator)

Example:
WDListViewLoadUCS2CSVFile(1,"C:\Christie\content\measured_data.csv","|")

Loads the data of the CSV file "measured_data.csv" into the ListView [890] 1 and separates the data sets from the CSV file with a "|" (pipe or vertical line) symbol.

Note: You may use any symbol as a separator, except the space " ".

In case the imported data does not show the same letters etc., check the encoding of your file. With this command, Widget Designer imports all common encodings except UTF-8 without a byte order mark (BOM). That is: ANSI, UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8 BOM.
The alternative command `WDListViewLoadCSVFile` [1756], also loads UTF-8 (without BOM) encoded data.

WDListViewLocation

WDListViewLocation(ID,X,Y)

Example:
WDListViewLocation(5,100,200)

Sets the position of the List View [890] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDListViewLocationLeft

WDListViewLocationLeft(ID,X)

Example:
WDListViewLocationLeft(5,100)

Sets the position of the List View [890] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDListViewLocationTop

WDListViewLocationTop(ID,Y)

Example:
WDListViewLocationTop(5,200)

Sets the position of the List View [890] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDListViewNewLine

WDListViewNewLine(ID,Col(optional),Value(optional))

Example:
WDListViewNewLine(1)

The first example uses only the mandatory parameters. It creates a new empty row beneath all other rows at ListView 890 1.

Example2:
WDListViewNewLine(1,2,"hello")

The second example adds the optional parameters and sets the value of column 2 of this new line to "hello".

WDListViewResize

WDListViewResize(ID,ColCount,RowCount)

Example:
WDListViewResize(1,5,4)

Resizes ListView 890 1 to a size of 5 columns and 4 rows.
Note: All existing data will be deleted when resizing the ListView.

WDListViewSaveCsvFile

WDListViewSaveCsvFile(ID,FilePath,Separator)

Example:
WDListViewSaveCsvFile(1,"C:\Christie\content\ListView.csv","-")

Creates a new CSV file "Listview" under the path "C:\Christie\content". The data from separate cells in a row from ListView 890 1 is separated with the - character.

WDListViewSetCell

WDListViewSetCell(ID,Column,Row,Value)

Example:
WDListViewSetCell(1,2,4,"hello")

Sets the value of ListView 890 1, column 2, row 4, to "hello".

WDListViewSetColumnName

WDListViewSetColumnName(ID,Column,Name)

Example:
WDListViewSetColumnName(1,3,"Height")

Sets the name of ListView 890 1 column 3 to "Height".

WDListViewSetFontFamily

WDListViewSetFontFamily(ID,FontName)

Example:
WDListViewSetFontFamily(5,"Arial")

This refers to the List View [890] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another List View, you can use a script like:
`WDListViewSetFontFamily(5,WDListViewGetFontFamily(1))`

WDListViewSetFontSize

WDListViewSetFontSize(ID,FontSize)

Example:
WDListViewSetFontSize(5,20)

This refers to the List View [890] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another List View, you can use a script like:
`WDListViewSetFontSize(5,WDListViewGetFontSize(1))`

WDListViewSetRowName

WDListViewSetRowName(ID,Row,Name)

Example:
WDListViewSetRowName(2,4,"area 4")

This sets the name of row 4 to "area 4" in the ListView [890] with ID 2.

WDListViewSize

WDListViewSize(ID,Width,Height)

Example:
WDListViewSize(5,100,40)

Sets the size of the List View [890] with ID 5 to a width of 100px and a height of 40px.

WDListViewSizeHeight

WDListViewSizeHeight(ID,Height)

Example:
WDListViewSizeHeight(5,40)

Sets the size of the List View [890] with ID 5 to a height of 40px but remains the current width.

WDListViewSizeWidth

WDListViewSizeWidth(ID,Width)

Example:
WDListViewSizeWidth(5,100)

Sets the size of the List View [890] with ID 5 to a width of 100px but remains the current height.

WDListViewUnfix

WDListViewUnfix(ID)

Example:
WDListViewUnfix(5)

This deactivates the option "Fix" in the Item Properties of the List View [890] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDListViewUpdateSource

WDListViewUpdateSource(ID)

Example:
WDListViewUpdateSource(1)

Updates the data from the source Excel file of ListView [890] 1.

WDLoad

WDLoad

Example:
WDLoad

This opens the file dialog where you can select a project you would like to load.

If changes have been made to your current project, Widget Designer asks whether you like to save or discard them before loading the new file.

WDLoadProject

WDLoadProject(File)

Example:
WDLoadProject("C:\Christie\content\test\test.wdp")

Loads the file "test.wdp" from the specified directory.

If changes have been made in your current project, Widget Designer asks whether you like to save or discard them before loading the new file.

WDLoadProjectRestart

WDLoadProjectRestart(File)

Example:
WDLoadProjectRestart("C:\Christie\content\test\test.wdp")

Restarts Widget Designer and loads the file "test.wdp" from the specified directory without saving the current WD project.

With restarting WD you can clean up the entire memory usage from the old project.

WDMacro

WDMacro(MacroName,Macro Parameters(optional))

Example:
WDMacro("Fader Jump",5)

Executes Macro "Fader Jump" and passes "5" as an input parameter. The Macro Tool [1897] allows to create Macros with optional input parameters.

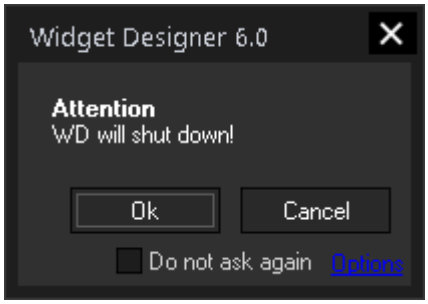
WDMacroBoxOkCancel

WDMacroBoxOkCancel(Title,Content,MacroName1,MacroName2)

Example:
WDMacroBoxOkCancel("Attention!","WD will shut down!","MacroOk","MacroCancel")

Pops up a message box with the title "Attention!" and the text "WD will shut down!". If there are following commands, they will not be executed until the message box is answered.
If you click on "OK", the Macro "MacroOk" will be executed, if you click on "Cancel" the Macro "MacroCancel" will be executed.

Note: The Macros need to be set up in the Macro tool [1897] before.
The command "WDMessageBox [1766]" might also be of interest.



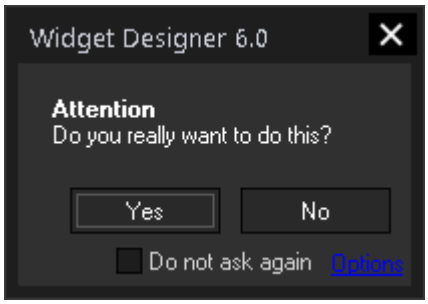
WDMacroBoxYesNo

WDMacroBoxYesNo(Title,Content,MacroName1,MacroName2)

Example:
WDMacroBoxYesNo("Attention","Do you really want to do this?","MacroYes","MacroNo")

Pops up a message box with the title "Attention!" and the text "Do you really want to do this?". If there are following commands, they will not be executed until the message box is answered.
If you click on "Yes", the Macro "MacroYes" will be executed, if you click on "No" the Macro "MacroNo" will be executed.

Note: The Macros need to be set up in the Macro tool [1897] before.
The command "WDMessageBox [1766]" might also be of interest.



WDMasterPBBackupMode

WDMasterPBBackupMode

Example:
WDMasterPBBackupMode

This sets the connected Pandoras Box Master from Live Mode to Backup Mode.
The IP address and Domain for the "Master Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256]. See also the topic "Backup [123]" in Pandoras Box for more information.

To reverse this action, execute the command `WDMasterPBTakeoverAllClients` [1762]. The respective command for the Pandoras Box backup connection would be `WDBackupPBBackupMode` [1672]

WDMasterPBTakeoverAllClients

WDMasterPBTakeoverAllClients

Example:
WDMasterPBTakeoverAllClients

This sets the connected Pandoras Box Master from Backup Mode to Live Mode and takes over all clients which are in the same domain.
The IP address and Domain for the "Master Connection" to Pandoras Box can to be set up in the menu: Connections > PB Configuration [1256]. See also the topic "Backup [123]" in Pandoras Box for more information.

To reverse this action, execute the command `WDMasterPBBackupMode` [1762]. The respective command for the Pandoras Box backup connection would be `WDBackupPBTakeoverAllClients` [1672]

WDMediaControlClearSelection

WDMediaControlClearSelection(MediaControlID)

Example:
WDMediaControlClearSelection(1)

Clears the selection box of the last selected Media in the Widget Designer Media Control [828] Panel with ID 1.
Note: This does not reset the Media on the Pandoras Box Device.

WDMediaControlCssStyleDisable

WDMediaControlCssStyleDisable(ID,StyleID)

Example:
WDMediaControlCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Media Control [828] Panel with ID 5.

WDMediaControlCssStyleEdit

WDMediaControlCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDMediaControlCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Media Control [828] Panel with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDMediaControlCssStyleEnable

WDMediaControlCssStyleEnable(ID,StyleID)

Example:
WDMediaControlCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Media Control [828] Panel with ID 5.

WDMediaControlFix

WDMediaControlFix(ID)

Example:
WDMediaControlFix(5)

This activates the option "Fix" in the Item Properties of the Media Control [828] Panel with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDMediaControlGetLocationLeft

WDMediaControlGetLocationLeft(ID)

Example:
WDMediaControlGetLocationLeft(5)

This refers to the MediaControl [828] Panel with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMediaControlGetLocationLeft(1)`

WDMediaControlGetLocationTop

WDMediaControlGetLocationTop(ID)

Example:
WDMediaControlGetLocationTop(5)

This refers to the MediaControl [828] Panel with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMediaControlGetLocationTop(1)`

WDMediaControlGetSizeHeight

WDMediaControlGetSizeHeight(ID)

Example:
WDMediaControlGetSizeHeight(5)

This refers to the MediaControl [828] Panel with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMediaControlGetSizeHeight(1)`

WDMediaControlGetSizeWidth

WDMediaControlGetSizeWidth(ID)

Example:
WDMediaControlGetSizeWidth(5)

This refers to the MediaControl [828] Panel with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMediaControlGetSizeWidth(1)`

WDMediaControlLocation

WDMediaControlLocation(ID,X,Y)

Example:
WDMediaControlLocation(5,100,200)

Sets the position of the Media Control [828] Panel with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDMediaControlLocationLeft

WDMediaControlLocationLeft(ID,X)

Example:
WDMediaControlLocationLeft(5,100)

Sets the position of the Media Control [828] Panel with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDMediaControlLocationTop

WDMediaControlLocationTop(ID,Y)

Example:
WDMediaControlLocationTop(5,200)

Sets the position of the Media Control [828] Panel with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDMediaControlSetDevices

WDMediaControlSetDevices(MediaControlID,DeviceIDs)

Example:
WDMediaControlSetDevices(5,"3.1 3.2")

This command writes the Site and Device IDs "3.1 3.2" into the Devices field of the Media Control [828] Panel with ID 5. From now on, the selected media will be assigned to the first and second layer of the Site 3 in the

connected Pandoras Box master. As you see in the example, you can not only assign single devices but also multiple ones by separating the next Site and Device ID with a spacebar. The Site and Device ID itself can be separated with a dot or comma.

WDMediaControlSetFolder

WDMediaControlSetFolder(MediaControlID,FolderID)

Example:
WDMediaControlSetFolder(5,2)

This command sets the "Folder ID" option to the Folder ID 2 within the Media Control [828] Panel with ID 5. The Folder ID refers to the downloaded Thumbnail Folder.

WDMediaControlSetIndex

WDMediaControlSetIndex(MediaControlID,Index)

Example:
WDMediaControlSetIndex(5,3)

This command sets the "Index" setting for the Media Control [828] Panel with ID 5. The index defines which thumbnail should be displayed first.

WDMediaControlSetLayout

WDMediaControlSetLayout(MediaControlID,Columns,Rows)

Example:
WDMediaControlSetLayout(5,3,4)

This command changes the layout for the Media Control [828] Panel with ID 5 by setting the "Column" to 3 and "Row" to 4.

WDMediaControlSetNextFolder

WDMediaControlSetNextFolder(MediaControlID)

Example:
WDMediaControlSetNextFolder(5)

This command increases the "Folder ID" setting in the Media Control [828] Panel with ID 5. The Folder ID refers to the downloaded Thumbnail Folder.

WDMediaControlSetPreviousFolder

WDMediaControlSetPreviousFolder(MediaControlID)

Example:
WDMediaControlSetPreviousFolder(5)

This command decreases the "Folder ID" setting in the Media Control [828] Panel with ID 5. The Folder ID refers to the downloaded Thumbnail Folder.

WDMediaControlSize

WDMediaControlSize(ID,Width,Height)

Example:
WDMediaControlSize(5,100,40)

Sets the size of the Media Control [828] Panel with ID 5 to a width of 100px and a height of 40px.

WDMediaControlSizeHeight

WDMediaControlSizeHeight(ID,Height)

Example:
WDMediaControlSizeHeight(5,40)

Sets the size of the Media Control [828] Panel with ID 5 to a height of 40px but remains the current width.

WDMediaControlSizeWidth

WDMediaControlSizeWidth(ID,Width)

Example:
WDMediaControlSizeWidth(5,100)

Sets the size of the Media Control [828] Panel with ID 5 to a width of 100px but remains the current height.

WDMediaControlUnfix

WDMediaControlUnfix(ID)

Example:
WDMediaControlUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Media Control [828] Panel with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDMemoryCleanUp

WDMemoryCleanUp

Example:
WDMemoryCleanUp

This cleans up the non-active processes when having not enough memory to operate at a high speed.

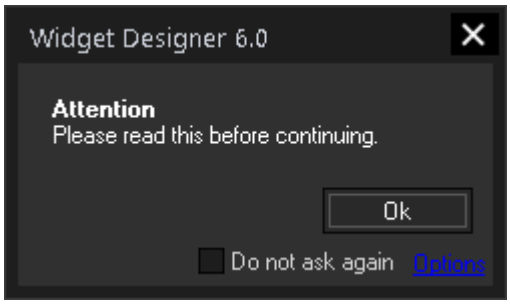
WDMessageBox

WDMessageBox(Title,Content)

Example:
WDMessageBox("Attention!","Please read this before continuing.")

Displays a message box with the title "Attention!" and the text "Please read this before continuing." If there are following commands, they will not be executed until the message box is answered with the "Ok" button.

The commands "WDMacroBoxOkCancel[1761]" or "WDMacroBoxYesNo[1761]" might also be of interest.



WDMultiMouseLinkLoop

WDMultiMouseLinkLoop(ID)

Example:
WDMultiMouseLinkLoop(1)

Sets the video mode of the item in MultiTouch Panel 1, that currently is linked to the mouse, to "Loop".

WDMultiMouseLinkPause

WDMultiMouseLinkPause(ID)

Example:
WDMultiMouseLinkPause(1)

Sets the video mode of the item in MultiTouch Panel 1, that currently is linked to the mouse, to "Pause".

WDMultiMouseLinkPlay

WDMultiMouseLinkPlay(ID)

Example:
WDMultiMouseLinkPlay(1)

Sets the video mode of the item in MultiTouch Panel 1, that currently is linked to the mouse, to "Play".

WDMultiMouseLinkStop

WDMultiMouseLinkStop(ID)

Example:
WDMultiMouseLinkStop(1)

Sets the video mode of the item in MultiTouch Panel 1, that currently is linked to the mouse, to "Stop".

WDMultiTouchApplyDefaults

WDMultiTouchApplyDefaults(ID)

Example:
WDMultiTouchApplyDefaults(1)

Applies the stored default values to the Multitouch Panel 1.
To store new default values to the panel please use the command WDMultiTouchStoreDefaults,ID 1777.
Please note that every Multitouch Panel has it's own default values.

WDMultiTouchBringToFront

WDMultiTouchBringToFront(ID,ItemID)

Example:
WDMultiTouchBringToFront(1,2)

Brings the Multitouch item 2 of Multitouch panel 1 to the front if it is completely or partially covered by another Multitouch item.

WDMultiTouchDisableAirScan

WDMultiTouchDisableAirScan

Example:
WDMultiTouchDisableAirScan

Disables the AirScan for MultiTouch Panel 1.

WDMultiTouchDisableCamera

WDMultiTouchDisableCamera

Example:
WDMultiTouchDisableCamera

Disables the Camera Tool for MultiTouch Panel 1.

WDMultiTouchDisableDrag

WDMultiTouchDisableDrag(ID,ItemID)

Example:
WDMultiTouchDisableDrag(1,2)

Disables the Drag option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchDisableGestures

WDMultiTouchDisableGestures(ID)

Example:
WDMultiTouchDisableGestures(1)

Disables Gestures for Synaptics Touch Pads and Wacom Bamboo Devices for MultiTouch Panel 1.

WDMultiTouchDisableIPhone

WDMultiTouchDisableIPhone

Example:
WDMultiTouchDisableIPhone

Disables the iPhone for MultiTouch Panel 1.

WDMultiTouchDisableKinect

WDMultiTouchDisableKinect

Example:
WDMultiTouchDisableKinect

Disables the Kinect for MultiTouch Panel 1.

WDMultiTouchDisableMouse

WDMultiTouchDisableMouse(ID)

Example:
WDMultiTouchDisableMouse(1)

Disables the Mouse for MultiTouch Panel 1.

WDMultiTouchDisablePan

WDMultiTouchDisablePan(ID,ItemID)

Example:
WDMultiTouchDisablePan(1,2)

Disables the Pan option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchDisableRotate

WDMultiTouchDisableRotate(ID,ItemID)

Example:
WDMultiTouchDisableRotate(1,2)

Disables the Rotate option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchDisableSize

WDMultiTouchDisableSize(ID,ItemID)

Example:
WDMultiTouchDisableSize(1,2)

Disables the Size option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchDisableTouch

WDMultiTouchDisableTouch(ID)

Example:
WDMultiTouchDisableTouch(1)

Disables Touch for MultiTouch Panel 1.

WDMultiTouchDisableZoom

WDMultiTouchDisableZoom(ID,ItemID)

Example:
WDMultiTouchDisableZoom(1,2)

Disables the Zoom option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchDisableZorder

WDMultiTouchDisableZorder(ID)

Example:
WDMultiTouchDisableZorder(1)

Disables the Z-Order option for the MultiTouch Panel with the ID 1.

WDMultiTouchDisplayMode

WDMultiTouchDisplayMode(ID,DisplayMode)

Example:
WDMultiTouchDisplayMode(1,"16:10")

Sets the Display Mode for the MultiTouch Panel with the ID 1 to the aspect ratio 16:10.

The following aspect ratios can be assigned:
4:3, 16:9, 16:10.

WDMultiTouchEnableAirScan

WDMultiTouchEnableAirScan

Example:
WDMultiTouchEnableAirScan

Enables the AirScan for MultiTouch Panel 1.

WDMultiTouchEnableCamera

WDMultiTouchEnableCamera

Example:
WDMultiTouchEnableCamera

Enables the Camera Tool for MultiTouch Panel 1.

WDMultiTouchEnableDrag

WDMultiTouchEnableDrag(ID,ItemID)

Example:
WDMultiTouchEnableDrag(1,2)

Enables the Drag option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchEnableGestures

WDMultiTouchEnableGestures(ID)

Example:
WDMultiTouchEnableGestures(1)

Enables Gestures for Synaptics Touch Pads and Wacom Bamboo Devices for MultiTouch Panel 1.

WDMultiTouchEnableIPhone

WDMultiTouchEnableIPhone

Example:
WDMultiTouchEnableIPhone

Enables the iPhone for MultiTouch Panel 1.

WDMultiTouchEnableKinect

WDMultiTouchEnableKinect

Example:
WDMultiTouchEnableKinect

Enables the Kinect for MultiTouch Panel 1.

WDMultiTouchEnableMouse

WDMultiTouchEnableMouse(ID)

Example:
WDMultiTouchEnableMouse(1)

Enables the Mouse for MultiTouch Panel 1.

WDMultiTouchEnablePan

WDMultiTouchEnablePan(ID,ItemID)

Example:
WDMultiTouchEnablePan(1,2)

Enables the Pan option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchEnableRotate

WDMultiTouchEnableRotate(ID,ItemID)

Example:
WDMultiTouchEnableRotate(1,2)

Enables the Rotate option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchEnableSize

WDMultiTouchEnableSize(ID,ItemID)

Example:
WDMultiTouchEnableSize(1,2)

Enables the Size option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchEnableTouch

WDMultiTouchEnableTouch(ID)

Example:
WDMultiTouchEnableTouch(1)

Enables Touch for MultiTouch Panel 1.

WDMultiTouchEnableZoom

WDMultiTouchEnableZoom(ID,ItemID)

Example:
WDMultiTouchEnableZoom(1,2)

Enables the Zoom option for Item 2 in the MultiTouch Panel with the ID 1.

WDMultiTouchEnableZorder

WDMultiTouchEnableZorder(ID)

Example:
WDMultiTouchEnableZorder(1)

Enables the Z-Order option for the MultiTouch Panel with the ID 1.

WDMultiTouchItemLoop

WDMultiTouchItemLoop(ID,ItemID)

Example:
WDMultiTouchItemLoop(1,4)

Sets the video mode of item 4 in MultiTouch Panel 1 to "Loop".

WDMultiTouchItemPause

WDMultiTouchItemPause(ID,ItemID)

Example:
WDMultiTouchItemPause(1,4)

Sets the video mode of item 4 in MultiTouch Panel 1 to "Pause".

WDMultiTouchItemPlay

WDMultiTouchItemPlay(ID,ItemID)

Example:
WDMultiTouchItemPlay(1,4)

Sets the video mode of item 4 in MultiTouch Panel 1 to "Play".

WDMultiTouchItemStop

WDMultiTouchItemPause(ID,ItemID)

Example:
WDMultiTouchItemStop(1,4)

Sets the video mode of item 4 in MultiTouch Panel 1 to "Stop".

WDMultiTouchLayerMode

WDMultiTouchLayerMode(ID,ItemID,DisplayMode)

Example:
WDMultiTouchLayerMode(1,2,"16:9")

Sets the Layer Mode for Item 2 in the MultiTouch Panel with the ID 1 to the aspect ratio 16:9.

The following aspect ratios can be assigned:
4:3, 16:9, 16:10.

WDMultiTouchPanelCssStyleDisable

WDMultiTouchPanelCssStyleDisable(ID,StyleID)

Example:
WDMultiTouchPanelCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the MultiTouch [899] Panel with ID 5.

WDMultiTouchPanelCssStyleEdit

WDMultiTouchPanelCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDMultiTouchPanelCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the MultiTouch [899] Panel with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDMultiTouchPanelCssStyleEnable

WDMultiTouchPanelCssStyleEnable(ID,StyleID)

Example:
WDMultiTouchPanelCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the MultiTouch [899] Panel with ID 5.

WDMultiTouchPanelFix

WDMultiTouchPanelFix(ID)

Example:
WDMultiTouchPanelFix(5)

This activates the option "Fix" in the Item Properties of the MultiTouch [899] Panel with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDMultiTouchPanelGetLocationLeft

WDMultiTouchPanelGetLocationLeft(ID)

Example:
WDMultiTouchPanelGetLocationLeft(5)

This refers to the MultiTouch [899] Panel with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMultiTouchPanelGetLocationLeft(1)`

WDMultiTouchPanelGetLocationTop

WDMultiTouchPanelGetLocationTop(ID)

Example:
WDMultiTouchPanelGetLocationTop(5)

This refers to the MultiTouch [899] Panel with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMultiTouchPanelGetLocationTop(1)`

WDMultiTouchPanelGetSizeHeight

WDMultiTouchPanelGetSizeHeight(ID)

Example:
WDMultiTouchPanelGetSizeHeight(5)

This refers to the MultiTouch [899] Panel with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMultiTouchPanelGetSizeHeight(1)`

WDMultiTouchPanelGetSizeWidth

WDMultiTouchPanelGetSizeWidth(ID)

Example:
WDMultiTouchPanelGetSizeWidth(5)

This refers to the MultiTouch [899] Panel with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDMultiTouchPanelGetSizeWidth(1)`

WDMultiTouchPanelLocation

WDMultiTouchPanelLocation(ID,X,Y)

Example:
WDMultiTouchPanelLocation(5,100,200)

Sets the position of the MultiTouch [899] Panel with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDMultiTouchPanelLocationLeft

WDMultiTouchPanelLocationLeft(ID,X)

Example:
WDMultiTouchPanelLocationLeft(5,100)

Sets the position of the MultiTouch [899] Panel with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDMultiTouchPanelLocationTop

WDMultiTouchPanelLocationTop(ID,Y)

Example:
WDMultiTouchPanelLocationTop(5,200)

Sets the position of the MultiTouch [899] Panel with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDMultiTouchPanelSize

WDMultiTouchPanelSize(ID,Width,Height)

Example:
WDMultiTouchPanelSize(5,100,40)

Sets the size of the MultiTouch [899] Panel with ID 5 to a width of 100px and a height of 40px.

WDMultiTouchPanelSizeHeight

WDMultiTouchPanelSizeHeight(ID,Height)

Example:
WDMultiTouchPanelSizeHeight(5,40)

Sets the size of the MultiTouch [899] Panel with ID 5 to a height of 40px but remains the current width.

WDMultiTouchPanelSizeWidth

WDMultiTouchPanelSizeWidth(ID,Width)

Example:
WDMultiTouchPanelSizeWidth(5,100)

Sets the size of the MultiTouch [899] Panel with ID 5 to a width of 100px but remains the current height.

WDMultiTouchPanelUnfix

WDMultiTouchPanelUnfix(ID)

Example:
WDMultiTouchPanelUnfix(5)

This deactivates the option "Fix" in the Item Properties of the MultiTouch [899] Panel with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDMultiTouchRefresh

WDMultiTouchRefresh(ID)

Example:
WDMultiTouchRefresh(1)

Refreshes the values of MultiTouch Panel 1 to Pandoras Box.

WDMultiTouchSetActive

WDMultiTouchSetActive(ID,ItemID)

Example:
WDMultiTouchSetActive(1,3)

Enables the Item 3 in the MultiTouch Panel with the ID 1.

WDMultiTouchSetDevice

WDMultiTouchSetDevice(ID,ItemID,SiteID,DeviceID)

Example:
WDMultiTouchSetDevice(1,3,2,5)

Assigns the PB Device 2,5 to the Item 3 in the MultiTouch Panel with the ID 1.

WDMultiTouchSetInActive

WDMultiTouchSetInActive(ID,ItemID)

Example:
WDMultiTouchSetInActive(1,4)

Disables the Item 4 in the MultiTouch Panel with the ID 1.

WDMultiTouchSetPan

WDMultiTouchSetPan(ID,X,Y)

Example:
WDMultiTouchSetPan(1,10,5)

Changes the Pan in the MultiTouch Panel with the ID 1 to the value X=10 and Y=5.

WDMultiTouchSetPos

WDMultiTouchSetPos(ID,ItemID,XPos,YPos)

Example:
WDMultiTouchSetPos(1,3,25,50)

Sets the Item 3 in the MultiTouch Panel with the ID 1 to the Position X=25 and Y=50.

WDMultiTouchSetRot

WDMultiTouchSetRot(ID,ItemID,Rot)

Example:
WDMultiTouchSetRot(1,3,90)

Sets the Item 3 in the MultiTouch Panel with the ID 1 to the rotation value of 90°.

WDMultiTouchSetScale

WDMultiTouchSetScale(ID,ItemID,Scale)

Example:
WDMultiTouchSetScale(1,3,40)

Scales the Item 3 in the MultiTouch Panel with the ID 1 to 40%.

WDMultiTouchSetZoom

WDMultiTouchSetZoom(ID,Zoomfactor)

Example:
WDMultiTouchSetZoom(1,100)

Changes the Zoom in the MultiTouch Panel with the ID 1 to the value 100.

WDMultiTouchStoreDefaults

WDMultiTouchStoreDefaults(ID)

Example:
WDMultiTouchStoreDefaults(1)

Stores all current values of the Multitouch Panel 1 as Defaults. To apply these values later on to the Multitouch Panel again, use the command WDMultiTouchApplyDefaults,ID [1767].
Please note that every Multitouch Panel has it's own default values.

WDNBSBroadcastDisable

WDNBSBroadcastDisable

Example:
WDNBSBroadcastDisable

Disables the Network Broadcast Service (NBS) for Faders [874], Labels [888] and Custom Script Buttons [822].

WDNBSBroadcastEnable

WDNBSBroadcastEnable

Example:
WDNBSBroadcastEnable

Enables the Network Broadcast Service (NBS) for Faders [874], Labels [888] and Custom Script Buttons [822].

WDNBSReceiveDisable

WDNBSReceiveDisable

Example:
WDNBSReceiveDisable

Disables Faders [874], Labels [888] and Custom Script Buttons [822] to receive values via Network Broadcast Service from other controls.

WDNBSReceiveEnable

WDNBSReceiveEnable

Example:
WDNBSReceiveEnable

Enables Faders [874], Labels [888] and Custom Script Buttons [822] to receive values via Network Broadcast Service from other controls.

WDNew

WDNew

Example:
WDNew

Creates a new WD project. If changes have been made in your current project, Widget Designer asks whether you like to save or discard them.

WDNodeConnect

WDNodeConnect(SourceNodeID,TargetNodeID)

Example:
WDNodeConnect(2,5)

Connects the Source Node with the ID 2 to the Target Node with the ID 5.

WDNodeCssStyleDisable

WDNodeCssStyleDisable(ID,StyleID)

Example:
WDNodeCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Node [937] with ID 5.

WDNodeCssStyleEdit

WDNodeCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDNodeCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Node [937] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDNodeCssStyleEnable

WDNodeCssStyleEnable(ID,StyleID)

Example:
WDNodeCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Node [937] with ID 5.

WDNodeDisableOutput

WDNodeDisableOutput(ID)

Example:
WDNodeDisableOutput(35)

Disables Output node 36 (for example an Output Node like a Fader Output).

WDNodeDisableOutputAll

WDNodeDisableOutputAll

Example:
WDNodeDisableOutputAll

Enables all Output Nodes.

WDNodeDisconnect

WDNodeDisconnect(NodeID,SourceNodeID)

Example:
WDNodeDisconnect(3,2)

Disconnects the Node with the ID 3 from the Source Node with the ID 2.

WDNodeEnableOutput

WDNodeEnableOutput(ID)

Example:
WDNodeEnableOutput(35)

Enables Output node 36 (for example an Output Node like a Fader Output).

WDNodeEnableOutputAll

WDNodeEnableOutputAll

Example:
WDNodeEnableOutputAll

Disables all Output Nodes.

WDNodeFix

WDNodeFix(ID)

Example:
WDNodeFix(5)

This activates the option "Fix" in the Item Properties of the Node [937] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDNodeGetLocationLeft

WDNodeGetLocationLeft(ID)

Example:
WDNodeGetLocationLeft(5)

This refers to the Node [937] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDNodeGetLocationLeft(1)`

WDNodeGetLocationTop

WDNodeGetLocationTop(ID)

Example:
WDNodeGetLocationTop(5)

This refers to the Node [937] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDNodeGetLocationTop(1)`

WDNodeGetSizeHeight

WDNodeGetSizeHeight(ID)

Example:
WDNodeGetSizeHeight(5)

This refers to the Node [937] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDNodeGetSizeHeight(1)`

WDNodeGetSizeWidth

WDNodeGetSizeWidth(ID)

Example:
WDNodeGetSizeWidth(5)

This refers to the Node [937] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDNodeGetSizeWidth(1)`

WDNodeLocation

WDNodeLocation(ID,X,Y)

Example:
WDNodeLocation(5,100,200)

Sets the position of the Node [937] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDNodeLocationLeft

WDNodeLocationLeft(ID,X)

Example:
WDNodeLocationLeft(5,100)

Sets the position of the Node [937] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDNodeLocationTop

WDNodeLocationTop(ID,Y)

Example:
WDNodeLocationTop(5,200)

Sets the position of the Node [937] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDNodeProcessOutput

WDNodeProcessOutput(NodeID)

Example:
WDNodeProcessOutput(12)

Per default all output nodes update their value according to the Node Cycle Interval [1510](Tools > Options). This means their entire preceding node chain is run through starting with the input node and passing all possible filter nodes. With this command, output node 12 updates its value (once and independent from the set Node Cycle Interval) by pulling new data from its preceding node chain and sending it to the according output control or connection.

WDNodeRefreshOutput

WDNodeRefreshOutput(NodeID)

Example:
WDNodeRefreshOutput(12)

Per default all output nodes update their value according to the Node Cycle Interval [1510](Tools > Options). This means their entire preceding node chain is run through starting with the input node and passing all possible filter nodes. With this command, output node 12 sends its current value (once and independent from the set Node Cycle Interval) to the according output control or connection. However, it does not pull new data from its preceding node chain, this happens with the next Node Cycle.

Please note:
This command is only valid for the output nodes for a Label, Fader, Bargraph, AngularDisplay, DigitalDisplay. In addition the Script Output Node can be refreshed which executes (depending on the input data) the True or False Script.

WDNodesDisable

WDNodesDisable

Example:
WDNodesDisable

Disables all nodes and releases their connection to other nodes and outputs in WD.

WDNodesEnable

WDNodesEnable

Example:
WDNodesEnable

Enables all nodes in WD if disabled before and re-establishes their connection to other nodes and outputs.

WDNodeSetOutput

WDNodeSetOutput(NodeID,ParamID,Value)

Example:
WDNodeSetOutput(2,1,100)

Sets the first output parameter of Input Node 2 to the value 100.

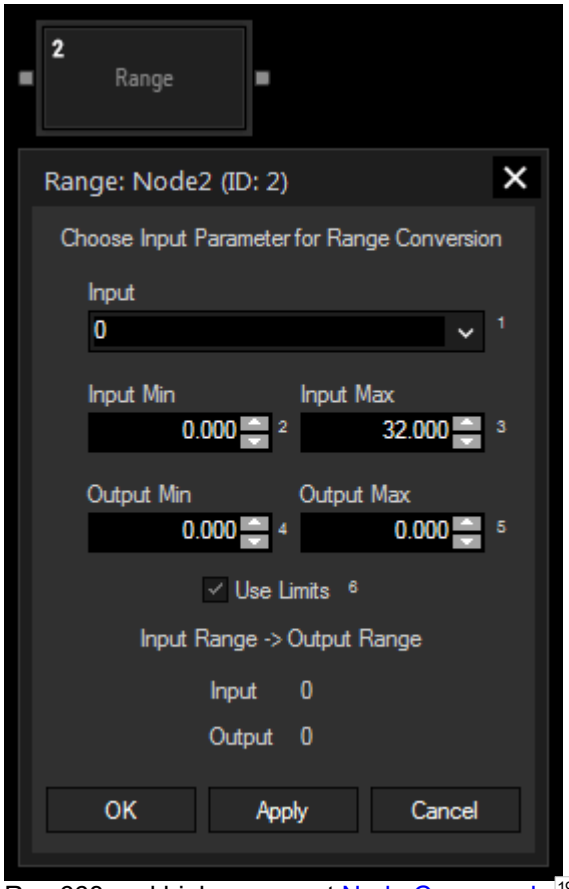
This command allows to set an output parameter of an Input Node to a specific value (applies to TCP, UDP and COM ASCII String / Stream Input Nodes)

WDNodeSetParam

WDNodeSetParam(NodeID,ParamID,Value)

Example:
WDNodeSetParam(2,3,32)

Sets Parameter 3 of Node 2 to the value 32, for example if Node 2 is a Range filter node, the Input Max will be changed to 32, see Picture below.
All parameters with a small number behind can be changed with this command.



Rev 668 and higher support Node Commands[1928] in a more direct form. The above command can be replaced with the more direct command `NodeID.SetParam(ParamID,Value)`. With the numbers from the example, that would be: `Node2.SetParam(3,32)`

WDNodeSetParamAdd

WDNodeSetParamAdd(NodeID,ParamID,Value)

Example:
WDNodeSetParamAdd(3,1,10)

Adds the value 10 to parameter 1 of Node 3, for example if Node 3 is a Value Input Node, its Output value will be increased by 10.

WDNodeSetParamDivide

WDNodeSetParamDivide(NodeID,ParamID,Value)

Example:
WDNodeSetParamDivide(3,1,10)

Divides the value of parameter 1 of Node 3 through 10, for example if Node 3 is a Value Input Node, its Output value will be divided through 10.

WDNodeSetParamFromNodeAdd

WDNodeSetParamFromNodeAdd(NodeID,ParamID,SourceNodeID,SourceNodeParamID)

Example:
WDNodeSetParamFromNodeAdd(4,1,2,3)

Adds the value of parameter 3 of Node 2 to the value of parameter 1 of Node 4.

This way node values can be dynamically changed via commands to any number, without causing infinite loops.

WDNodeSetParamFromNodeDivide

WDNodeSetParamFromNodeDivide(NodeID,ParamID,SourceNodeID,SourceNodeParamID)

Example:
WDNodeSetParamFromNodeDivide(4,1,2,3)

Divides the value of parameter 1 of Node 4 through the value of parameter 3 of Node 2.

This way node values can be dynamically changed via commands to any number, without causing infinite loops.

WDNodeSetParamFromNodeInputValue

WDNodeSetParamFromNodeInputValue(NodeID,ParamID,SourceNodeID,SourceNodeInParamID)

Example:
WDNodeSetParamFromNodeInputValue(7,2,6,1)

This command looks at the input source that is setup for the parameter 1 of the (Filter or Output) Node 6 and copies the current state. Then it pastes this (current!) value permanently to the parameter 2 of the (Filter or Output) Node 7. Node 7 does not need to be connected to the same Input Node as Node 6 is. It will display the value that is received in Node 6 in that moment when the command is executed.

WDNodeSetParamFromNodeMinus

WDNodeSetParamFromNodeMinus(NodeID,ParamID,SourceNodeID,SourceNodeParamID)

Example:
WDNodeSetParamFromNodeMinus(4,1,2,3)

Subtracts the value of parameter 3 of Node 2 from the value of parameter 1 of Node 4.

This way node values can be dynamically changed via commands to any number, without causing infinite loops.

WDNodeSetParamFromNodeMultiply

WDNodeSetParamFromNodeMultiply(NodeID,ParamID,SourceNodeID,SourceNodeParamID)

Example:
WDNodeSetParamFromNodeMultiply(4,1,2,3)

Multiplies the value of parameter 1 of Node 4 with the value of parameter 3 of Node 2.

WDNodeSetParamFromNodeOutputValue

WDNodeSetParamFromNodeOutputValue(NodeID,ParamID,SourceNodeID,SourceNodeOutParamID)

Example:
WDNodeSetParamFromNodeOutputValue(7,2,6,1)

Sets the parameter 2 of Node 7 to the output value of parameter 1 of Node 6.

Allows to combine the current values of Nodes by ID

WDNodeSetParamMinus

WDNodeSetParamMinus(NodeID,ParamID,Value)

Example:
WDNodeSetParamMinus(3,1,10)

Subtracts the value 10 from parameter 1 of Node 3, for example if Node 3 is a Value Input Node, its Output value will be decreased by 10.

WDNodeSetParamMultiply

WDNodeSetParamMultiply(NodeID,ParamID,Value)

Example:
WDNodeSetParamMultiply(3,1,10)

Multiplies the value parameter 1 of Node 3 with the factor 10, for example if Node 3 is a Value Input Node, its Output value will be multiplied by 10.

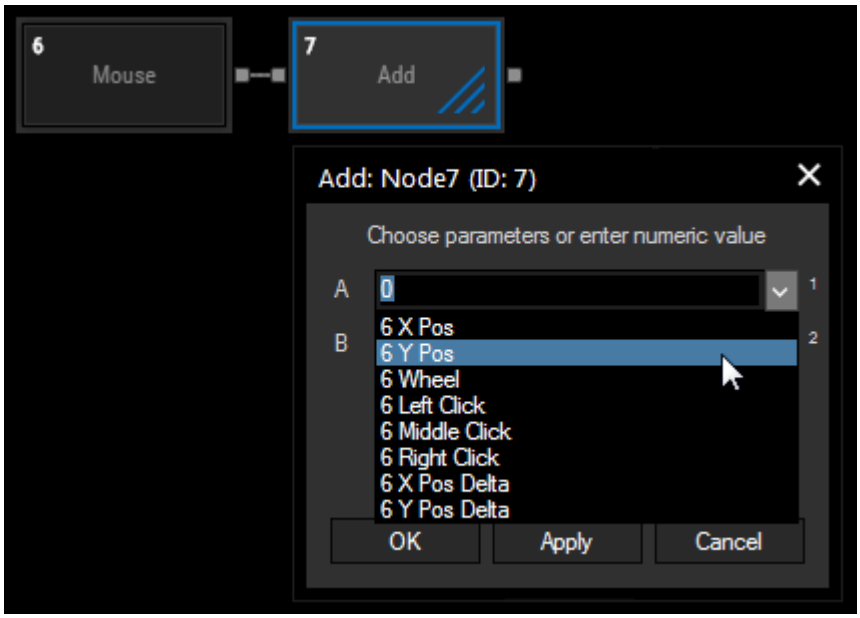
WDNodeSetParamSource

WDNodeSetParamSource(NodeID,ParamID,SourceNodeID,SourceNodeParamID)

Example:
WDNodeSetParamSource(7,1,6,2)

Changes the source for the input parameter if two or more nodes or a single one with several values is connected to a filter or output node.

In this depicted example the input parameter (1) of the Add node (7) will change to the value for Y Position (2) of the Mouse Input node (6).



Rev 668 and higher support Node Commands [1928] in a more direct form. The above command can be replaced with the more direct command `Node2.SetParamSource(ParamID,SourceNodeID,SourceNodeParamID)`. With the numbers from the example, that would be: `Node7.SetParamSource(1,6,2)`

WDNodeSize

WDNodeSize(ID,Width,Height)

Example:
WDNodeSize(5,100,40)

Sets the size of the Node [937] with ID 5 to a width of 100px and a height of 40px.

WDNodeSizeHeight

WDNodeSizeHeight(ID,Height)

Example:
WDNodeSizeHeight(5,40)

Sets the size of the Node [937] with ID 5 to a height of 40px but remains the current width.

WDNodeSizeWidth

WDNodeSizeWidth(ID,Width)

Example:
WDNodeSizeWidth(5,100)

Sets the size of the Node [937] with ID 5 to a width of 100px but remains the current height.

WDNodeUnfix

WDNodeUnfix(ID)

Example:
WDNodeUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Node [937] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDOffsetAllControls

WDOffsetAllControls(X,Y)

Example:
WDOffsetAllControls(100,50)

Moves all Widget Designer control widgets [818] 100px to the right and 50px down.

WDOffsetControl

WDOffsetControl(ControlName,X,Y)

Example:
WDOffsetControl("CustomScript3",100,50)

Moves the button "CustomScript 3" 100px to the right and 50px down.

Note: Per default, the name of the control widget [818] consists of control type (Custom Script, Fader, XY Panel, etc.) and ID without space characters. It is displayed in the Item Properties dialog at the very beginning and can also be changed there.

WDOpacity

WDOpacity(Value 0-1)

Example:
WDOpacity(0.5)

Displays the entire Widget Designer window as a see-through (transparent) window. Insert values between 0 and 1, with 0 being completely transparent and 1 being completely opaque.

WDPageBackgroundBottomColor

WDPageBackgroundBottomColor(Page,R,G,B)

Example:
WDPageBackgroundBottomColor("Page1,150,0,255)

Sets the background bottom color of page "Page1" to an intensive violet (R: 150, G: 0, B: 255). You can change the color manually in the dialog "Page Settings [805]" (Page > Edit Page) where you can also set the Page size which, by the way, is independent from the Window size.

WDPageBackgroundColor

WDPageBackgroundColor(Page,R,G,B)

Example:
WDPageBackgroundColor("Page1,150,0,255)

Sets the background color of page "Page1" to an intensive violet (R: 150, G: 0, B: 255). You can change the color manually in the dialog "Page Settings [805]" (Page > Edit Page).

WDPageBackgroundTopColor

WDPageBackgroundTopColor(Page,R,G,B)

Example:
WDPageBackgroundTopColor("Page1,150,0,255)

Sets the background top color of page "Page1" to an intensive violet (R: 150, G: 0, B: 255). You can change the color manually in the dialog "Page Settings [805]" (Page > Edit Page) where you can also set the Page size which, by the way, is independent from the Window size.

WDPageBackward

WDPageBackward

Example:
WDPageBackward

Goes one page back in history. You can choose the same command from the Pages [805] menu.

WDPageClearHistory

WDPageClearHistory

Example:
WDPageClearHistory

Clears the history, so that "WDPageForward" and "WDPagePrevious" do not call up another page different than the "Default" one.

WDPageCssStyleDisable

WDPageCssStyleDisable(PageName,StyleID)

Example:
WDPageCssStyleDisable("Page5",2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Page [805] with the name "Page5".

WDPageCssStyleEnable

WDPageCssStyleEnable(PageName,StyleID)

Example:
WDPageCssStyleEnable("Page5",2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Page [805] with the name "Page5".

WDPageDisablePassword

WDPageDisablePassword(Page)

Example:
WDPageDisablePassword("Page2")

Disables the password of page "Page2".

WDPageEnablePassword

WDPageEnablePassword(Page)

Example:
WDPageEnablePassword("Page2")

Enables the password of page "Page2". If there was no password set before, nothing has to be entered before clicking on "OK".

WDPageExecuteEnterScript

WDPageExecuteEnterScript(Pagename)

Example:
WDPageExecuteEnterScript("Settings2")

Executes the On Page [805] Enter Script of the page named "Settings2".

WDPageExecuteLeaveScript

WDPageExecuteLeaveScript(Pagename)

Example:
WDPageExecuteLeaveScript("Settings2")

Executes the On Page [805] Leave Script of the page named "Settings2".

WDPageFirst

WDPageFirst

Example:
WDPageFirst

Goes to the first page. You can also choose pages when opening the Pages [805] menu.

WDPageForward

WDPageForward

Example:
WDPageForward

Calls up the next page in history. You can also choose pages when opening the Pages [805] menu.

WDPageGoto

WDPageGoto(Pagename)

Example:
WDPageGoto("Faders")

Changes to page "Faders" (if the page "Faders" was created before).
Please note that this command only works when executed in the respective Window context, i.e. for example the Click Script of a CustomScript button that is part of the Window.
If you require to trigger a page change externally, please use the Window's PageName member:

Window1.PageName = "Faders"

WDPageGotoSilent

WDPageGotoSilent(Pagename)

Example:
WDPageGotoSilent("Faders")

Changes to page "Faders" without executing its page enter script.

Please note that this command only works when executed in the respective Window context, i.e. for example the Click Script of a CustomScript button that is part of the Window.
If you require to trigger a page change externally, please use the Window's PageName member:

Window1.PageName = "Faders"

WDPageLabelConvertToPNG

WDPageLabelConvertToPNG(Page,File,Red,Green,Blue,Alpha)

Example:
WDPageLabelConvertToPNG("Page1","C:\Christie\test1.png",150,0,255,128)

This saves an image with the name "test1.png" under the given path. The image shows all Labels [888] from "Page1" (but no other controls) on a slightly transparent violet background (R: 150, G: 0, B: 255, Alpha: 128) and has the same size as "Page1" does.

WDPageLabelPictureBoxConvertToPNG

WDPageLabelPictureBoxConvertToPNG(Page,File)

Example:
WDPageLabelPictureBoxConvertToPNG("Page1","C:\Christie\test2.png")

This saves an image with the name "test2.png" under the given path. The image shows all Labels [888] and PictureBoxes [911] from "Page1" (but no other controls) and has the same background color(s) [805] and size as "Page1" does.

WDPageLast

WDPageLast

Example:
WDPageLast

Goes to the last page in order. If this command is called when being on the first page, it will stay here. If you like to call the last one in the list, use the command `WDPageLoopPrevious` instead. You can also choose pages when opening the Pages [805] menu.

WDPageLoopNext

WDPageLoopNext

Example:
WDPageLoopNext

Goes to the next page in order. If this command is called when being on the last page, the first one will be called. If you like to stay here, use the command `WDPageNext` instead. You can also choose pages when opening the Pages [805] menu.

WDPageLoopPrevious

WDPageLoopPrevious

Example:
WDPageLoopPrevious

Goes to the previous page in order. This this command is called when being on the first page, the last one will be called. If you like to stay here, use the command `WDPageLast` instead. You can also choose pages when opening the Pages [805] menu.

WDPageNext

WDPageNext

Example:
WDPageNext

Goes to the next page in order. If this command is called when being on the last page, it will stay here. If you like to call the first one, use the command `WDPageLoopNext` instead. You can also choose pages when opening the Pages [805] menu.

WDPagePassword

WDPagePassword(Page,Password)

Example:
WDPagePassword("Page2","PW123")

Sets (or changes) the password of page "Page2" to "PW123". This command does not enable the password function automatically.

WDPagePrevious

WDPagePrevious

Example:
WDPagePrevious

Calls up the last page in history. You can also choose pages when opening the Pages [805] menu.

WDPasswordDialog

WDPasswordDialog(Password,Continue True/False)

Example:
WDPasswordDialog("widget",false)

This command opens a Password Dialog, passwords can only be entered using the displayed keyboard. In case the password is incorrect and the "continue" parameter is set to "false", the script is aborted. If "widget" is typed in correctly, the following script will be executed.

If the "continue" parameter is set to "true", the script will continue regardless of the password being correct or not. The command instead provides a Boolean return value, indicating whether the entered password matches. This Boolean can be written in a variable [1900] and used for an if statement [1886] that executes different scripts for both cases, instead of simply aborting the action.

Example Script:

```
var pwCorrect = false
pwCorrect = WDPasswordDialog("123",true)
if pwCorrect {DebugMessage("Login successful")}
else {DebugMessage("Password invalid")}
```

WDPictureBoxApplyCITPThumbnail

WDPictureBoxApplyCITPThumbnail(ID,FolderID,FileID)

Example:
WDPictureBoxApplyCITPThumbnail(ID,FolderID,FileID)

Applies the CITP Thumbnail with Folder/File ID 1/5 to the PictureBox [911] with ID 1. A connection to the respective Pandoras Box Master application has to be established. The Thumbnails do not have to be exchanged manually between Widget Designer and Pandoras Box.

In Pandoras Box, the Folder and File ID can be applied or changed in the File Inspector [191].

WDPictureBoxCssStyleDisable

WDPictureBoxCssStyleDisable(ID,StyleID)

Example:
WDPictureBoxCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Picture Box [911] with ID 5.

WDPictureBoxCssStyleEdit

WDPictureBoxCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDPictureBoxCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Picture Box [911] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDPictureBoxCssStyleEnable

WDPictureBoxCssStyleEnable(ID,StyleID)

Example:
WDPictureBoxCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Picture Box [911] with ID 5.

WDPictureBoxFile

WDPictureBoxFile(ID,File)

Example:
WDPictureBoxFile(1,"C:\Christie\content\test1.jpg")

Replaces the content of the PictureBox [911] control with the ID 1 with the image "test1.jpg" from the path C:\Christie\content.

WDPictureBoxFix

WDPictureBoxFix(ID)

Example:
WDPictureBoxFix(5)

This activates the option "Fix" in the Item Properties of the Picture Box [911] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDPictureBoxGetLocationLeft

WDPictureBoxGetLocationLeft(ID)

Example:
WDPictureBoxGetLocationLeft(5)

This refers to the PictureBox [911] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPictureBoxGetLocationLeft(1)`

WDPictureBoxGetLocationTop

WDPictureBoxGetLocationTop(ID)

Example:
WDPictureBoxGetLocationTop(5)

This refers to the PictureBox [911] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPictureBoxGetLocationTop(1)`

WDPictureBoxGetSizeHeight

WDPictureBoxGetSizeHeight(ID)

Example:
WDPictureBoxGetSizeHeight(5)

This refers to the PictureBox [911] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPictureBoxGetSizeHeight(1)`

WDPictureBoxGetSizeWidth

WDPictureBoxGetSizeWidth(ID)

Example:
WDPictureBoxGetSizeWidth(5)

This refers to the PictureBox [911] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPictureBoxGetSizeWidth(1)`

WDPictureBoxLoadRecentResource

WDPictureBoxLoadRecentResource(ID)

Example:
WDPictureBoxLoadRecentResource(1)

Loads the last image that was send from with the WD Remote App [1275] to the Image Resource Manager [1509] (Style: User, Control: User) to the PictureBox [911] with ID 1.
If you like to load other images from the Resource Manager, use the command WDPictureBoxLoadResource [1794]

WDPictureBoxLoadResource

WDPictureBoxLoadResource(ID,Resource)

Example:
WDPictureBoxLoadResource(1,"Default\Button\Forward")

Loads the image with the name "Forward" from the Image Resource Manager [1509] (Style: Default, Control: Button) to the PictureBox [911] with ID 1.

Example 2:
WDPictureBoxLoadResource(1,"User\User\DSCN0780.JPG")

This loads the image "nature.jpg" that was either added manually or sent with the WD Remote App [1275], as these images automatically appear under Style: User, Control: User.

If you automatically want to load the last image that was sent, the command WDPictureBoxLoadRecentResource [1794] could be easier to use.

WDPictureBoxLocation

WDPictureBoxLocation(ID,X,Y)

Example:
WDPictureBoxLocation(5,100,200)

Sets the position of the Picture Box [911] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDPictureBoxLocationLeft

WDPictureBoxLocationLeft(ID,X)

Example:
WDPictureBoxLocationLeft(5,100)

Sets the position of the Picture Box [911] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDPictureBoxLocationTop

WDPictureBoxLocationTop(ID,Y)

Example:
WDPictureBoxLocationTop(5,200)

Sets the position of the Picture Box [911] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDPictureBoxRefresh

WDPictureBoxRefresh(ID)

Example:
WDPictureBoxRefresh(ID)

Refreshes the PictureBox [911] 1 and displays the current file. This command is useful if the file is changed occasionally.

WDPictureBoxSaveFile

WDPictureBoxSaveFile(ID,Path)

Example:
WDPictureBoxSaveFile(1,"C:\Christie\content\fromWD.jpg")

This saves the current image displayed by the PictureBox [911] with ID 1 to the path "C:\Christie\content" and names it "fromWD.jpg".
Available image file formats are: PNG, JPEG, DDS, TIFF, Bitmap and DPX.

WDPictureBoxScreenshot

WDPictureBoxScreenshot(ID)

Example:
WDPictureBoxScreenshot(1)

Replaces the content of the PictureBox [911] control with the ID 1 with a screenshot from the current desktop.

WDPictureBoxScreenshotRegion

WDPictureBoxScreenshotRegion(ID,X,Y,Width,Height)

Example:
WDPictureBoxScreenshotRegion(1,100,200,400,300)

Replaces the content of the PictureBox [911] control with the ID 1 with a screenshot of a region from the current desktop. The screenshot regions position starts at pixel 100,200 (X,Y) and is 400 px wide and 300 px high.

WDPictureBoxSize

WDPictureBoxSize(ID,Width,Height)

Example:
WDPictureBoxSize(5,100,40)

Sets the size of the Picture Box [911] with ID 5 to a width of 100px and a height of 40px.

WDPictureBoxSizeHeight

WDPictureBoxSizeHeight(ID,Height)

Example:
WDPictureBoxSizeHeight(5,40)

Sets the size of the Picture Box [911] with ID 5 to a height of 40px but remains the current width.

WDPictureBoxSizeWidth

WDPictureBoxSizeWidth(ID,Width)

Example:
WDPictureBoxSizeWidth(5,100)

Sets the size of the Picture Box [911] with ID 5 to a width of 100px but remains the current height.

WDPictureBoxUnfix

WDPictureBoxUnfix(ID)

Example:
WDPictureBoxUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Picture Box [911] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDPlaylistCssStyleDisable

WDPlaylistCssStyleDisable(ID,StyleID)

Example:
WDPlaylistCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Playlist [913] with ID 5.

WDPlaylistCssStyleEdit

WDPlaylistCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDPlaylistCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Playlist [913] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDPlaylistCssStyleEnable

WDPlaylistCssStyleEnable(ID,StyleID,Timeout(optional))

Example:
WDPlaylistCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Playlist [913] with ID 5.

WDPlaylistFix

WDPlaylistFix(ID)

Example:
WDPlaylistFix(5)

This activates the option "Fix" in the Item Properties of the Playlist [913] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDPlayListGetFontFamily

WDPlayListGetFontFamily(ID)

Example:
varString = WDPlayListGetFontFamily(5)

This refers to the Playlist [913] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Playlist via a script like:
`WDPlayListSetFontFamily(5,WDPlayListGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDPlayListGetFontFamily(5))`

WDPlayListGetFontSize

WDPlayListGetFontSize(ID)

Example:
varDouble = WDPlayListGetFontSize(5)

This refers to the Playlist [913] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Playlist via a script like:
`WDPlayListSetFontSize(5,WDPlayListGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDPlayListGetFontSize(5))`

WDPlaylistGetLocationLeft

WDPlaylistGetLocationLeft(ID)

Example:
WDPlaylistGetLocationLeft(5)

This refers to the Playlist [913] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0).
You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPlaylistGetLocationLeft(1)`

WDPlaylistGetLocationTop

WDPlaylistGetLocationTop(ID)

Example:
WDPlaylistGetLocationTop(5)

This refers to the Playlist [913] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPlaylistGetLocationTop(1)`

WDPlaylistGetSizeHeight

WDPlaylistGetSizeHeight(ID)

Example:
WDPlaylistGetSizeHeight(5)

This refers to the Playlist [913] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPlaylistGetSizeHeight(1)`

WDPlaylistGetSizeWidth

WDPlaylistGetSizeWidth(ID)

Example:
WDPlaylistGetSizeWidth(5)

This refers to the Playlist [913] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPlaylistGetSizeWidth(1)`

WDPlaylistLocation

WDPlaylistLocation(ID,X,Y)

Example:
WDPlaylistLocation(5,100,200)

Sets the position of the Playlist [913] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDPlaylistLocationLeft

WDPlaylistLocationLeft(ID,X)

Example:
WDPlaylistLocationLeft(5,100)

Sets the position of the Playlist [913] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDPlaylistLocationTop

WDPlaylistLocationTop(ID,Y)

Example:
WDPlaylistLocationTop(5,200)

Sets the position of the Playlist [913] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDPlaylistPause

WDPlaylistPause(ID)

Example:
WDPlaylistPause(5)

Pauses the playback of the Playlist [913] with ID 5.

WDPlaylistPlay

WDPlaylistPlay(ID)

Example:
WDPlaylistPlay(5)

Starts the playback of the Playlist [913] with ID 5.

WDPlayListSetFontFamily

WDPlayListSetFontFamily(ID,FontName)

Example:
WDPlayListSetFontFamily(5,"Arial")

This refers to the Playlist [913] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Playlist, you can use a script like:
`WDPlayListSetFontFamily(5,WDPlayListGetFontFamily(1))`

WDPlayListSetFontSize

WDPlayListSetFontSize(ID,FontSize)

Example:
WDPlayListSetFontSize(5,20)

This refers to the Playlist [913] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Playlist, you can use a script like:
`WDPlayListSetFontSize(5,WDPlayListGetFontSize(1))`

WDPlaylistSize

WDPlaylistSize(ID,Width,Height)

Example:
WDPlaylistSize(5,100,40)

Sets the size of the Playlist [913] with ID 5 to a width of 100px and a height of 40px.

WDPlaylistSizeHeight

WDPlaylistSizeHeight(ID,Height)

Example:
WDPlaylistSizeHeight(5,40)

Sets the size of the Playlist [913] with ID 5 to a height of 40px but remains the current width.

WDPlaylistSizeWidth

WDPlaylistSizeWidth(ID,Width)

Example:
WDPlaylistSizeWidth(5,100)

Sets the size of the Playlist [913] with ID 5 to a width of 100px but remains the current height.

WDPlaylistStop

WDPlaylistStop(ID)

Example:
WDPlaylistStop(5)

Stops the playback of the Playlist [913] with ID 5.

WDPlaylistUnfix

WDPlaylistUnfix(ID)

Example:
WDPlaylistUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Playlist [913] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDPowerPointDisplayCssStyleDisable

WDPowerPointDisplayCssStyleDisable(ID,StyleID)

Example:
WDPowerPointDisplayCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the PowerPoint Display [860] with ID 5.

WDPowerPointDisplayCssStyleEdit

WDPowerPointDisplayCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDPowerPointDisplayCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the PowerPoint Display [860] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDPowerPointDisplayCssStyleEnable

WDPowerPointDisplayCssStyleEnable(ID,StyleID)

Example:
WDPowerPointDisplayCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the PowerPoint Display [860] with ID 5.

WDPowerPointDisplayFix

WDPowerPointDisplayFix(ID)

Example:
WDPowerPointDisplayFix(5)

This activates the option "Fix" in the Item Properties of the PowerPoint Display [860] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDPowerPointDisplayGetDevice

WDPowerPointDisplayGetDevice(ID)

Example:
varString = WDPowerPointDisplayGetDevice(5)

This refers to the PowerPoint Display [860] with ID 5 and returns the name of the device from the Configuration dialog [1305] it is currently connected to as a string, e.g. "PPT1".
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. a Label [888] via the script: `Label1.Text = WDPowerPointDisplayGetDevice(5)`
or simply display it in the Debug Logger [812]: `DebugMessage(WDPowerPointDisplayGetDevice(5))`

WDPowerPointDisplayGetLocationLeft

WDPowerPointDisplayGetLocationLeft(ID)

Example:
WDPowerPointDisplayGetLocationLeft(5)

This refers to the PowerPoint Display [860] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPowerPointDisplayGetLocationLeft(1)`

WDPowerPointDisplayGetLocationTop

WDPowerPointDisplayGetLocationTop(ID)

Example:
WDPowerPointDisplayGetLocationTop(5)

This refers to the PowerPoint Display [860] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPowerPointDisplayGetLocationTop(1)`

WDPowerPointDisplayGetMode

WDPowerPointDisplayGetMode(ID)

Example:
varString = WDPowerPointDisplayGetMode(5)

This refers to the PowerPoint Display [860] with ID 5 and returns the mode it is currently set to as a string, e.g. "Current", "Previous" or "Next".
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. a Label [888] via the script: `Label1.Text = WDPowerPointDisplayGetMode(5)`
or simply display it in the Debug Logger [812]: `DebugMessage(WDPowerPointDisplayGetMode(5))`

WDPowerPointDisplayGetSizeHeight

WDPowerPointDisplayGetSizeHeight(ID)

Example:
WDPowerPointDisplayGetSizeHeight(5)

This refers to the PowerPoint Display [860] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPowerPointDisplayGetSizeHeight(1)`

WDPowerPointDisplayGetSizeWidth

WDPowerPointDisplayGetSizeWidth(ID)

Example:
WDPowerPointDisplayGetSizeWidth(5)

This refers to the PowerPoint Display [860] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDPowerPointDisplayGetSizeWidth(1)`

WDPowerPointDisplayLocation

WDPowerPointDisplayLocation(ID,X,Y)

Example:
WDPowerPointDisplayLocation(5,100,200)

Sets the position of the PowerPoint Display [860] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDPowerPointDisplayLocationLeft

WDPowerPointDisplayLocationLeft(ID,X)

Example:
WDPowerPointDisplayLocationLeft(5,100)

Sets the position of the PowerPoint Display [860] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDPowerPointDisplayLocationTop

WDPowerPointDisplayLocationTop(ID,Y)

Example:
WDPowerPointDisplayLocationTop(5,200)

Sets the position of the PowerPoint Display [860] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDPowerPointDisplaySetDevice

WDPowerPointDisplaySetDevice(ID,DeviceName)

Example:
WDPowerPointDisplaySetDevice(5,"PPT2")

This refers to PowerPoint Display [860] with ID 5 and changes the device it should connect to, to "PPT2". Make sure that the Configuration dialog [1305] offers this device name.

WDPowerPointDisplaySetMode

WDPowerPointDisplaySetMode(ID,Mode)

Example:
WDPowerPointDisplaySetMode(1,"Next")

This refers to PowerPoint Display [860] with ID 5 and changes the mode to "Next". The modes "Current", "Next", and "Previous" are available.

WDPowerPointDisplaySize

WDPowerPointDisplaySize(ID,Width,Height)

Example:
WDPowerPointDisplaySize(5,100,40)

Sets the size of the PowerPoint Display [860] with ID 5 to a width of 100px and a height of 40px.

WDPowerPointDisplaySizeHeight

WDPowerPointDisplaySizeHeight(ID,Height)

Example:
WDPowerPointDisplaySizeHeight(5,40)

Sets the size of the PowerPoint Display [860] with ID 5 to a height of 40px but remains the current width.

WDPowerPointDisplaySizeWidth

WDPowerPointDisplaySizeWidth(ID,Width)

Example:
WDPowerPointDisplaySizeWidth(5,100)

Sets the size of the PowerPoint Display [860] with ID 5 to a width of 100px but remains the current height.

WDPowerPointDisplayUnfix

WDPowerPointDisplayUnfix(ID)

Example:
WDPowerPointDisplayUnfix(5)

This deactivates the option "Fix" in the Item Properties of the PowerPoint Display [860] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDProjectorManagerApplyHomeLocation

WDProjectorManagerApplyHomeLocation(ID)

Example:
WDProjectorManagerApplyHomeLocation(ID)

Resets the calibrated points from Output ID 1 in the Projector Calibration Manager [1496] to the stored home location.

WDProjectorManagerApplyRecalOnly

WDProjectorManagerApplyRecalOnly(ID)

Example:
WDProjectorManagerApplyRecalOnly(ID)

Applies the re-calibrated point values from Output ID 1 in the Projector Calibration Manager [1496] to the x-file without calibrating again. This might be useful if the X'-, Y'- and Z'-values were modified manually.

WDProjectorManagerRecalibrate

WDProjectorManagerRecalibrate(ID)

Example:
WDProjectorManagerRecalibrate(ID)

Re-calibrates and applies the changes of Output ID 1 in the Projector Calibration Manager [1496] to the x-file.

WDProjectorManagerRecalOnly

WDProjectorManagerRecalOnly(ID)

Example:
WDProjectorManagerRecalOnly(1)

Re-calibrates the calibration points from Output ID 1 in the Projector Calibration Manager [1496] without applying them to the x-file. This might be useful if the calibration has to be checked before overwriting the x-file.

WDProjectorManagerShowCalibratedPoints

WDProjectorManagerShowCalibratedPoints(ID)

Example:
WDProjectorManagerShowCalibratedPoints(1)

Shows the location of the re-calibrated points from Output ID 1 in the Projector Calibration Manager[1496]. This is useful for checking if the calibration was done accurately e.g. if they match physical reference points or fiber tips.

WDProjectorManagerShowHomePoints

WDProjectorManagerShowHomePoints(ID)

Example:
WDProjectorManagerShowHomePoints(1)

Shows the location of the home points from Output ID 1 in the Projector Calibration Manager[1496]. This is useful for checking if the calibration was done accurately e.g. if they match physical reference points or fiber tips.

WDProjectorManagerStoreHome

WDProjectorManagerStoreHome(ID)

Example:
WDProjectorManagerStoreHome(1)

Calibrates and stores the home points of Output ID 1 in the Projector Calibration Manager[1496].

WDRecalApplyHomeMesh

WDRecalApplyHomeMesh(outputName)

Example:
WDRecalApplyHomeMesh("Output 2.27")

Creates a copy of the home mesh and assigns it to the output "Output 2.27" in the Projector Recalibration[1496] Tool.

WDRecalApplyRecal

WDRecalApplyRecal(outputName)

Example:
WDRecalApplyRecal("Output 2.27")

Uses the Projector Recalibration[1496] Tool to create a copy of the home mesh, assign it to the output "Output 2.27" and transform it according to the offset between home points and located points.

WDRecalDrawHomePoints

WDRecalDrawHomePoints(outputName)

Example:
WDRecalDrawHomePoints("Output 2.27")

Uses the Projector Recalibration[1496] Tool to assign an Editable Canvas to output "Output 2.27" and draws crosshairs on the home point locations.

WDRecalDrawLocatedPoints

WDRecalDrawLocatedPoints(outputName)

Example:
WDRecalDrawLocatedPoints("Output 2.27")

Uses the Projector Recalibration [1496] Tool to assign an Editable Canvas to output "Output 2.27" and draws crosshairs on the last located point locations.

WDRecalLocate

WDRecalLocate(outputName)

Example:
WDRecalLocate("Output 2.27")

Starts the localization process of Output "Output 2.27" in the Projector Recalibration [1496] Tool.

WDRecalLocateAndApply

WDRecalLocateAndApply(outputName)

Example:
WDRecalLocateAndApply("Output 2.27")

Runs the location process of Output "Output 2.27" in the Projector Recalibration [1496] Tool and applies the mesh transformation according to the offset between home points and located points.
This is not the same as running WDRecalLocate followed by WDRecalApplyRecal.

WDReConnect

WDReConnect

Example:
WDReConnect

Reconnects Widget Designer from both the Pandoras Box Master and Backup. See the topic "PB Network Configuration" [1256] for more information.

WDReConnectBackup

WDReConnectBackup

Example:
WDReConnectBackup

Reconnects Widget Designer from Pandoras Box Backup. See the topic "PB Network Configuration" [1256] for more information.

WDReconnectMaster

WDReconnectMaster

Example:
WDReconnectMaster

Reconnects Widget Designer from Pandoras Box Master. See the topic "PB Network Configuration" [1256] for more information.

WDRemoteCOMDisable

WDRemoteCOMDisable

Example:
WDRemoteCOMDisable

Disables the COM Port connection set up via the Remoting Tool [1273].

WDRemoteCOMEnable

WDRemoteCOMEnable

Example:
WDRemoteCOMEnable

Enables the COM Port connection set up via the Remoting Tool [1273].

WDRemoteHTTPDisable

WDRemoteHTTPDisable

Example:
WDRemoteHTTPDisable

Disables the HTTP Listener set up via the Remoting Tool [1273].

WDRemoteHTTPEnable

WDRemoteHTTPEnable

Example:
WDRemoteHTTPEnable

Enables the HTTP Listener set up via the Remoting Tool [1273].

WDRemoteInputBufferClearCommands

WDRemoteInputBufferClearCommands

Example:
WDRemoteInputBufferClearCommands

Clears the Remote Input Buffer in the Remoting Tool [1273].

Note: The option "Buffer Remote Input Commands" has to be enabled in the Remoting Tool.

WDRemoteInputBufferCommandsDisable

WDRemoteInputBufferCommandsDisable

Example:
WDRemoteInputBufferCommandsDisable

Disables the Remote Input Buffer in the Remoting Tool [1273] (i.e. the check box "Buffer Remote Input Commands").

WDRemoteInputBufferCommandsEnable

WDRemoteInputBufferCommandsEnable

Example:
WDRemoteInputBufferCommandsEnable

Enables the Remote Input Buffer in the Remoting Tool [1273] (i.e. the check box "Buffer Remote Input Commands").

WDRemoteInputBufferProcessCommands

WDRemoteInputBufferProcessCommands

Example:
WDRemoteInputBufferProcessCommands

Processes all commands from the Remote Input Buffer fro the Remoting Tool [1273], according to their order.

Note: The option "Buffer Remote Input Commands" has to be enabled in the Remoting Tool.

WDRemoteInputBufferProcessFirstCommand

WDRemoteInputBufferProcessFirstCommand

Example:
WDRemoteInputBufferProcessFirstCommand

Processes the first command from the Remote Input Buffer in the Remoting Tool [1273].

Note: The option "Buffer Remote Input Commands" has to be enabled in the Remoting Tool.

WDRemoteTCPDisable

WDRemoteTCPDisable

Example:
WDRemoteTCPDisable

Disables the TCP connection set up via the Remoting Tool [1273].

WDRemoteTCPEnable

WDRemoteTCPEnable

Example:
WDRemoteTCPEnable

Enables the TCP connection set up via the Remoting Tool [1273].

WDRemoteUDPDisable

WDRemoteUDPDisable

Example:
WDRemoteUDPDisable

Disables the UDP connection set up via the Remoting Tool [1273].

WDRemoteUDPEnable

WDRemoteUDPEnable

Example:
WDRemoteUDPEnable

Enables the UDP connection set up via the Remoting Tool [1273].

WDResourceRecentSave

WDResourceRecentSave(Path)

Example:
WDResourceRecentSave("C:\WDResources")

Takes the last image that was sent from the WD Remote App [1275] to the Image Resource Manager [1509] (Style: User, Control: User) and saves it to the path "C:\WDResources". If you like to load other images from the Resource Manager, use the command `WDResourceSave`.

WDResourceSave

WDResourceSave(Resource,Path)

Example:
WDResourceSave("Default\Button\Forward","C:\WDResources")

This saves the image with the name "Forward" from the Image Resource Manager [1509] (Style: Default, Control: Button) to the path "C:\WDResources".

Example 2:
WDResourceSave("User\User\DSCN0780.JPG","C:\WDResources")

This saves images either added manually or sent with the WD Remote App [1275], as these images automatically appear under Style: User, Control: User. Alternatively, the command `WDResourceRecentSave` could also be of interest.

WDRestart

WDRestart

Example:
WDRestart

This restarts Widget Designer without saving the current WD project and opens a new empty file. With restarting WD you can clean up the entire memory usage from the old project.

WDRssAddFeed

WDRssAddFeed(URL)

Example:
WDRssAddFeed("https://www.christiedigital.com/_layouts/15/christiefeeds.aspx")

Adds the RSS feed [1493] "https://www.christiedigital.com/_layouts/15/christiefeeds.aspx" to the RSS feed list.

WDRssRefreshFeeds

WDRssRefreshFeeds

Example:
WDRssRefreshFeeds

Refreshes the RSS feed [1493] list.

WDRssRemoveFeed

WDRssRemoveFeed(URL)

Example:
WDRssRemoveFeed("https://www.christiedigital.com/_layouts/15/christiefeeds.aspx")

Removes the RSS feed [1493] "https://www.christiedigital.com/_layouts/15/christiefeeds.aspx" from the RSS feed list.

WDSave

WDSave

Example:
WDSave

Saves the current WD project. Please bear in mind that saving the project results in an abortion of ALL running scripts.

WDSaveAs

WDSaveAs

Example:
WDSaveAs

Opens the WD "Save as" dialog. Please bear in mind that saving the project results in an abortion of ALL running scripts.

WDSaveAsCopy

WDSaveAsCopy

Example:
WDSaveAsCopy

This opens the Save As dialog box to save a new version of the existing file. After saving the copy you will keep working in the current file with the current saving path. You might want to save the current file in addition to making a copy. Please bear in mind that saving the project results in an abortion of ALL running scripts.

WDSaveFile

WDSaveFile(Filename)

Example:
WDSaveFile("C:\Christie\content\project_test\project_test.wdp")

Saves the file "project_test.wdp" from the specified directory. Please bear in mind that saving the project results in an abortion of ALL running scripts.

WDScreenshot

WDScreenshot(FileName)

Example:
WDScreenshot("C:\Christie\screenshots\Page1")

Creates a screenshot of the current WD user interface and save it as Page1.png under the path C:\Christie\screenshots.

If you don't specify the file path with this command, the screenshot will be saved as Page1.png in the root folder of your WD project.

WDScreenshotDialog

WDScreenshotDialog

Example:
WDScreenshotDialog

Creates a screenshot of the current WD user interface and opens the Save As Dialog, where you can choose the location you want the screenshot to be saved and its name. It allows you to choose the file type of your screenshot as well (Bitmap, Jpeg or PNG).

WDScreenshotRegion

WDScreenshotRegion(X,Y,Width,Height,FileName)

Example:
WDScreenshotRegion(1,1,400,300,"C:\Christie\screenshots\Test1")

Creates a screenshot that is 400px wide and 300 px high, starting at pixel X=1 and Y=1 of the current screen and saves it as Test1.png under the path C:\Christie\screenshots.

If you don't specify the file path with this command, the screenshot will be saved in the root folder of your WD project.

WDScreenshotRegionDialog

WDScreenshotRegionDialog(X,Y,Width,Height)

Example:
WDScreenshotRegionDialog(1,1,400,300)

Creates a screenshot that is 400px wide and 300 px high, starting at pixel X=1 and Y=1 of the current screen and opens the Save As Dialog, where you can choose the location you want the screenshot to be saved and its name. It allows you to choose the file type of your screenshot as well (Bitmap, Jpeg or PNG).

WDScreenshotRegionWithTime

WDScreenshotRegionWithTime(X,Y,Width,Height,FileName)

Example:
WDScreenshotRegionWithTime(1,1,400,300,"C:\Christie\screenshots\Test1")

Creates a screenshot that is 400px wide and 300 px high, starting at pixel X=1 and Y=1 of the current screen and saves it as e.g. Test1_31_05_2011-16_43_22.png under the path C:\Christie\screenshots.

If you don't specify the file path with this command, the screenshot will be saved in the root folder of your WD project.

WDScreenshotWithTime

WDScreenshotWithTime(FileName)

Example:
WDScreenshotWithTime("C:\Christie\screenshots\Page1")

Creates a screenshot of the current WD user interface and save it e.g. as Page1_31_05_2011-16_43_22.png under the path C:\Christie\screenshots.

If you don't specify the file path with this command, the screenshot will be saved in the root folder of your WD project.

WDScriptTimerCssStyleDisable

WDScriptTimerCssStyleDisable(ID,StyleID)

Example:
WDScriptTimerCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Script Timer [826] button with ID 5.

WDScriptTimerCssStyleEdit

WDScriptTimerCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDScriptTimerCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Script Timer [826] button with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDScriptTimerCssStyleEnable

WDScriptTimerCssStyleEnable(ID,StyleID)

Example:
WDScriptTimerCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Script Timer [826] button with ID 5.

WDScriptTimerFix

WDScriptTimerFix(ID)

Example:
WDScriptTimerFix(5)

This activates the option "Fix" in the Item Properties of the Script Timer [826] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDScriptTimerGetLocationLeft

WDScriptTimerGetLocationLeft(ID)

Example:
WDScriptTimerGetLocationLeft(5)

This refers to the Script Timer [826] button with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScriptTimerGetLocationLeft(1)`

WDScriptTimerGetLocationTop

WDScriptTimerGetLocationTop(ID)

Example:
WDScriptTimerGetLocationTop(5)

This refers to the Script Timer [826] button with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScriptTimerGetLocationTop(1)`

WDScriptTimerGetSizeHeight

WDScriptTimerGetSizeHeight(ID)

Example:
WDScriptTimerGetSizeHeight(5)

This refers to the Script Timer [826] button with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScriptTimerGetSizeHeight(1)`

WDScriptTimerGetSizeWidth

WDScriptTimerGetSizeWidth(ID)

Example:
WDScriptTimerGetSizeWidth(5)

This refers to the Script Timer [826] button with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScriptTimerGetSizeWidth(1)`

WDScriptTimerInterval

WDScriptTimerInterval(ID,Value)

Example:
WDScriptTimerInterval(1,5000)

Changes the Interval Time of the Script Timer [826] with the ID 1 to 5000 ms.

WDScriptTimerLocation

WDScriptTimerLocation(ID,X,Y)

Example:
WDScriptTimerLocation(5,100,200)

Sets the position of the Script Timer [826] button with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDScriptTimerLocationLeft

WDScriptTimerLocationLeft(ID,X)

Example:
WDScriptTimerLocationLeft(5,100)

Sets the position of the Script Timer [826] button with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDScriptTimerLocationTop

WDScriptTimerLocationTop(ID,Y)

Example:
WDScriptTimerLocationTop(5,200)

Sets the position of the Script Timer [826] button with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDScriptTimerRunContinuous

WDScriptTimerRunContinuous(ID)

Example:
WDScriptTimerRunContinuous(1)

Changes the mode of the Script Timer [826] with the ID 1 to run continuously.

WDScriptTimerRunOnce

WDScriptTimerRunOnce(ID)

Example:
WDScriptTimerRunOnce(1)

Changes the mode of the Script Timer [826] with the ID 1 to run only once.

WDScriptTimerSize

WDScriptTimerSize(ID,Width,Height)

Example:
WDScriptTimerSize(5,100,40)

Sets the size of the Script Timer [826] button with ID 5 to a width of 100px and a height of 40px.

WDScriptTimerSizeHeight

WDScriptTimerSizeHeight(ID,Height)

Example:
WDScriptTimerSizeHeight(5,40)

Sets the size of the Script Timer [826] button with ID 5 to a height of 40px but remains the current width.

WDScriptTimerSizeWidth

WDScriptTimerSizeWidth(ID,Width)

Example:
WDScriptTimerSizeWidth(5,100)

Sets the size of the Script Timer [826] button with ID 5 to a width of 100px but remains the current height.

WDScriptTimerStart

WDScriptTimerStart(ID)

Example:
WDScriptTimerStart(1)

Starts the Script Timer [826] with the ID 1. It will keep on running until it is stopped via a command or via a click on it with the mouse (when it is set to run continuously).

WDScriptTimerStop

WDScriptTimerStop(ID)

Example:
WDScriptTimerStop(1)

Stops the Script Timer [826] with the ID 1.

WDScriptTimerUnfix

WDScriptTimerUnfix(ID)

Example:
WDScriptTimerUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Script Timer [826] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDScrollerCssStyleDisable

WDScrollerCssStyleDisable(ID,StyleID)

Example:
WDScrollerCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Scroller [844] with ID 5.

WDScrollerCssStyleEdit

WDScrollerCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDScrollerCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Scroller [844] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDScrollerCssStyleEnable

WDScrollerCssStyleEnable(ID,StyleID)

Example:
WDScrollerCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Scroller [844] with ID 5.

WDScrollerFix

WDScrollerFix(ID)

Example:
WDScrollerFix(5)

This activates the option "Fix" in the Item Properties of the Scroller [844] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDScrollerGetLocationLeft

WDScrollerGetLocationLeft(ID)

Example:
WDScrollerGetLocationLeft(5)

This refers to the Scroller [844] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScrollerGetLocationLeft(1)`

WDScrollerGetLocationTop

WDScrollerGetLocationTop(ID)

Example:
WDScrollerGetLocationTop(5)

This refers to the Scroller [844] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScrollerGetLocationTop(1)`

WDScrollerGetSizeHeight

WDScrollerGetSizeHeight(ID)

Example:
WDScrollerGetSizeHeight(5)

This refers to the Scroller [844] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScrollerGetSizeHeight(1)`

WDScrollerGetSizeWidth

WDScrollerGetSizeWidth(ID)

Example:
WDScrollerGetSizeWidth(5)

This refers to the Scroller [844] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDScrollerGetSizeWidth(1)`

WDScrollerLocation

WDScrollerLocation(ID,X,Y)

Example:
WDScrollerLocation(5,100,200)

Sets the position of the Scroller [844] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDScrollerLocationLeft

WDScrollerLocationLeft(ID,X)

Example:
WDScrollerLocationLeft(5,100)

Sets the position of the Scroller [844] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDScrollerLocationTop

WDScrollerLocationTop(ID,Y)

Example:
WDScrollerLocationTop(5,200)

Sets the position of the Scroller [844] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDScrollerReset

WDScrollerReset(ID)

Example:
WDScrollerReset(1)

Resets Scroller [844] 1 by bringing the first Item to the first position..

WDScrollerSize

WDScrollerSize(ID,Width,Height)

Example:
WDScrollerSize(5,100,40)

Sets the size of the Scroller [844] with ID 5 to a width of 100px and a height of 40px.

WDScrollerSizeHeight

WDScrollerSizeHeight(ID,Height)

Example:
WDScrollerSizeHeight(5,40)

Sets the size of the Scroller [844] with ID 5 to a height of 40px but remains the current width.

WDScrollerSizeWidth

WDScrollerSizeWidth(ID,Width)

Example:
WDScrollerSizeWidth(5,100)

Sets the size of the Scroller [844] with ID 5 to a width of 100px but remains the current height.

WDScrollerUnfix

WDScrollerUnfix(ID)

Example:
WDScrollerUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Scroller [844] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDSetBackupAdapter

WDSetBackupAdapter(Adapter)

Example:
WDSetBackupAdapter("Lan1")

Sets the Lan Adapter to "Lan1" for the "Backup Connection" to Pandoras Box which can be set up in the menu: Connections > PB Configuration [1256].

WDSetLocalAdapter

WDSetLocalAdapter(NetworkAdapterName)

Example:
WDSetLocalAdapter("Lan1")

This command chooses "Lan1" from the drop-down list with available adapter names for the local Adapter of Widget Designer. This setting can be found in the Connection Manager [1258] under IP Info & Setup.

WDSetMasterAdapter

WDSetMasterAdapter(Adapter)

Example:
WDSetMasterAdapter("Lan2")

Sets the Lan Adapter to "Lan2" for the "Master Connection" to Pandoras Box which can be set up in the menu: Connections > PB Configuration [1256].

WDShapeCol

WDShapeCol(ID,R,G,B,A(optional))

Example:
WDShapeCol(2,120,0,255,255)

Sets the Fill and Outline Color of Shape [919] 2 to R: 120, G: 0, B: 255 (deep violett). The last parameter is optional, in the example the alpha channel/opacity is set to 255.

Note: The properties "Fill Color" and "Line Color" have to be enabled.

WDShapeCssStyleDisable

WDShapeCssStyleDisable(ID,StyleID)

Example:
WDShapeCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Shape [919] with ID 5.

WDShapeCssStyleEdit

WDShapeCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDShapeCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Shape [919] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDShapeCssStyleEnable

WDShapeCssStyleEnable(ID,StyleID)

Example:
WDShapeCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Shape [919] with ID 5.

WDShapeDisableFillCol

WDShapeDisableFillCol(ID)

Example:
WDShapeDisableFillCol(2)

Disables the fill color of Shape [919] 2.

WDShapeDisableOutlineCol

WDShapeDisableOutlineCol(ID)

Example:
WDShapeDisableOutlineCol(2)

Disables the colored outline of Shape [919] 2.

WDShapeEnableFillCol

WDShapeEnableFillCol(ID)

Example:
WDShapeEnableFillCol(2)

Enables the fill color of Shape [919] 2.

WDShapeEnableOutlineCol

WDShapeEnableOutlineCol(ID)

Example:
WDShapeEnableOutlineCol(2)

Enables the colored outline of Shape [919] 2.

WDShapeFillCol

WDShapeFillCol(ID,R,G,B)

Example:
WDShapeFillCol(2,120,0,255)

Sets the Fill Color of Shape [919] 2 to R: 120, G: 0, B: 255 (deep violet).

Note: The property "Fill Color" has to be enabled.

WDShapeFix

WDShapeFix(ID)

Example:
WDShapeFix(5)

This activates the option "Fix" in the Item Properties of the Shape [919] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDShapeGetLocationLeft

WDShapeGetLocationLeft(ID)

Example:
WDShapeGetLocationLeft(5)

This refers to the Shape [919] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDShapeGetLocationLeft(1)`

WDShapeGetLocationTop

WDShapeGetLocationTop(ID)

Example:
WDShapeGetLocationTop(5)

This refers to the Shape [919] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDShapeGetLocationTop(1)`

WDShapeGetSizeHeight

WDShapeGetSizeHeight(ID)

Example:
WDShapeGetSizeHeight(5)

This refers to the Shape [919] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDShapeGetSizeHeight(1)`

WDShapeGetSizeWidth

WDShapeGetSizeWidth(ID)

Example:
WDShapeGetSizeWidth(5)

This refers to the Shape [919] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDShapeGetSizeWidth(1)`

WDShapeLocation

WDShapeLocation(ID,X,Y)

Example:
WDShapeLocation(5,100,200)

Sets the position of the Shape [919] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDShapeLocationLeft

WDShapeLocationLeft(ID,X)

Example:
WDShapeLocationLeft(5,100)

Sets the position of the Shape [919] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDShapeLocationTop

WDShapeLocationTop(ID,Y)

Example:
WDShapeLocationTop(5,200)

Sets the position of the Shape [919] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDShapeOpacity

WDShapeOpacity(ID,Alpha)

Example:
WDShapeOpacity(1,128)

Sets the opacity of Shape [919] 1 to 128.

Note: Both the property "Line Color" and "Fill Color" are affected.

WDShapeOutlineCol

WDShapeOutlineCol(ID,R,G,B)

Example:
WDShapeOutlineCol(2,120,0,255)

Sets the Outline Color of Shape [919] 2 to R: 120, G: 0, B: 255 (deep violet).

Note: The property "Line Color" has to be enabled.

WDShapeSize

WDShapeSize(ID,Width,Height)

Example:
WDShapeSize(5,100,40)

Sets the size of the Shape [919] with ID 5 to a width of 100px and a height of 40px.

WDShapeSizeHeight

WDShapeSizeHeight(ID,Height)

Example:
WDShapeSizeHeight(5,40)

Sets the size of the Shape [919] with ID 5 to a height of 40px but remains the current width.

WDShapeSizeWidth

WDShapeSizeWidth(ID,Width)

Example:
WDShapeSizeWidth(5,100)

Sets the size of the Shape [919] with ID 5 to a width of 100px but remains the current height.

WDShapeUnfix

WDShapeUnfix(ID)

Example:
WDShapeUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Shape [919] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDShowControlCount

WDShowControlCount

Example:
WDShowControlCount

This displays the total number of controls (Widgets/Nodes) of your project in a pop-up window.

WDStartPbHealthListener

WDStartPbHealthListener

Example:
WDStartPbHealthListener

This opens the internally used UDP port 8992 that is used to receive the PB Engine Health data which is sent as multicast data to 239.0.0.11. The data is received by the "PB Engine Health" Input node [1057]. You can execute this command when `WDStopPbHealthListener` [1827] was used before.

WDStickyNoteCssStyleDisable

WDStickyNoteCssStyleDisable(ID,StyleID)

Example:
WDStickyNoteCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Sticky Note [921] with ID 5.

WDStickyNoteCssStyleEdit

WDStickyNoteCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDStickyNoteCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Sticky Note [921] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDStickyNoteCssStyleEnable

WDStickyNoteCssStyleEnable(ID,StyleID)

Example:
WDStickyNoteCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Sticky Note [921] with ID 5.

WDStickyNoteFix

WDStickyNoteFix(ID)

Example:
WDStickyNoteFix(5)

This activates the option "Fix" in the Item Properties of the Sticky Note [921] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDStickyNoteGetFontFamily

WDStickyNoteGetFontFamily(ID)

Example:
varString = WDStickyNoteGetFontFamily(5)

This refers to the Sticky Note [921] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Sticky Note via a script like:
`WDStickyNoteSetFontFamily(5,WDStickyNoteGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDStickyNoteGetFontFamily(5))`

WDStickyNoteGetFontSize

WDStickyNoteGetFontSize(ID)

Example:
varDouble = WDStickyNoteGetFontSize(5)

This refers to the Sticky Note [921] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Sticky Note via a script like:
`WDStickyNoteSetFontSize(5,WDStickyNoteGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDStickyNoteGetFontSize(5))`

WDStickyNoteGetLocationLeft

WDStickyNoteGetLocationLeft(ID)

Example:
WDStickyNoteGetLocationLeft(5)

This refers to the Sticky Note [921] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDStickyNoteGetLocationLeft(1)`

WDStickyNoteGetLocationTop

WDStickyNoteGetLocationTop(ID)

Example:
WDStickyNoteGetLocationTop(5)

This refers to the Sticky Note [921] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDStickyNoteGetLocationTop(1)`

WDStickyNoteGetSizeHeight

WDStickyNoteGetSizeHeight(ID)

Example:
WDStickyNoteGetSizeHeight(5)

This refers to the Sticky Note [921] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDStickyNoteGetSizeHeight(1)`

WDStickyNoteGetSizeWidth

WDStickyNoteGetSizeWidth(ID)

Example:
WDStickyNoteGetSizeWidth(5)

This refers to the Sticky Note [921] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDStickyNoteGetSizeWidth(1)`

WDStickyNoteLocation

WDStickyNoteLocation(ID,X,Y)

Example:
WDStickyNoteLocation(5,100,200)

Sets the position of the Sticky Note [921] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDStickyNoteLocationLeft

WDStickyNoteLocationLeft(ID,X)

Example:
WDStickyNoteLocationLeft(5,100)

Sets the position of the Sticky Note [921] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDStickyNoteLocationTop

WDStickyNoteLocationTop(ID,Y)

Example:
WDStickyNoteLocationTop(5,200)

Sets the position of the Sticky Note [921] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDStickyNoteSetFontFamily

WDStickyNoteSetFontFamily(ID,FontName)

Example:
WDStickyNoteSetFontFamily(5,"Arial")

This refers to the Sticky Note [921] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Sticky Note, you can use a script like:
`WDStickyNoteSetFontFamily(5,WDStickyNoteGetFontFamily(1))`

WDStickyNoteSetFontSize

WDStickyNoteSetFontSize(ID,FontSize)

Example:
WDStickyNoteSetFontSize(5,20)

This refers to the Sticky Note [921] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Sticky Note, you can use a script like:
`WDStickyNoteSetFontSize(5,WDStickyNoteGetFontSize(1))`

WDStickyNoteSize

WDStickyNoteSize(ID,Width,Height)

Example:
WDStickyNoteSize(5,100,40)

Sets the size of the Sticky Note [921] with ID 5 to a width of 100px and a height of 40px.

WDStickyNoteSizeHeight

WDStickyNoteSizeHeight(ID,Height)

Example:
WDStickyNoteSizeHeight(5,40)

Sets the size of the Sticky Note [921] with ID 5 to a height of 40px but remains the current width.

WDStickyNoteSizeWidth

WDStickyNoteSizeWidth(ID,Width)

Example:
WDStickyNoteSizeWidth(5,100)

Sets the size of the Sticky Note [921] with ID 5 to a width of 100px but remains the current height.

WDStickyNoteUnfix

WDStickyNoteUnfix(ID)

Example:
WDStickyNoteUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Sticky Note [921] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDStopPbHealthListener

WDStopPbHealthListener

Example:
WDStopPbHealthListener

This closes the internally used UDP port 8992 that is used to receive the PB Engine Health data which is sent as multicast data to 239.0.0.11. The data is received by the "PB Engine Health" Input node [1057]. To re-open the port again, use the command `WDStartPbHealthListener` [1823].

WDTerraDisplayArrayCssStyleDisable

WDTerraDisplayArrayCssStyleDisable(ID,StyleID)

Example:
WDTerraDisplayArrayCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the TerraDisplayArray [863] with ID 5.

WDTerraDisplayArrayCssStyleEdit

WDTerraDisplayArrayCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDTerraDisplayArrayCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the TerraDisplayArray [863] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDTerraDisplayArrayCssStyleEnable

WDTerraDisplayArrayCssStyleEnable(ID,StyleID)

Example:
WDTerraDisplayArrayCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the TerraDisplayArray [863] with ID 5.

WDTerraDisplayArrayFix

WDTerraDisplayArrayFix(ID)

Example:
WDTerraDisplayArrayFix(5)

This activates the option "Fix" in the Item Properties of the TerraDisplayArray [863] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDTerraDisplayArrayGetFontFamily

WDTerraDisplayArrayGetFontFamily(ID)

Example:
varString = WDTerraDisplayArrayGetFontFamily(5)

This refers to the TerraDisplayArray [863] with ID 5 and returns its current font family name as a string. The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Textbox via a script like:
`WDTerraDisplayArraySetFontFamily(5,WDTerraDisplayArrayGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTerraDisplayArrayGetFontFamily(5))`

WDTerraDisplayArrayGetFontSize

WDTerraDisplayArrayGetFontSize(ID)

Example:
varDouble = WDTerraDisplayArrayGetFontSize(5)

This refers to the TerraDisplayArray [863] with ID 5 and returns its current font size in "pt" as a double. The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Textbox via a script like:
`WDTerraDisplayArraySetFontSize(5,WDTerraDisplayArrayGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTerraDisplayArrayGetFontSize(5))`

WDTerraDisplayArrayGetLocationLeft

WDTerraDisplayArrayGetLocationLeft(ID)

Example:
WDTerraDisplayArrayGetLocationLeft(5)

This refers to the TerraDisplayArray [863] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTerraDisplayArrayGetLocationLeft(1)`

WDTerraDisplayArrayGetLocationTop

WDTerraDisplayArrayGetLocationTop(ID)

Example:
WDTerraDisplayArrayGetLocationTop(5)

This refers to the TerraDisplayArray [863] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTerraDisplayArrayGetLocationTop(1)`

WDTerraDisplayArrayGetSizeHeight

WDTerraDisplayArrayGetSizeHeight(ID)

Example:
WDTerraDisplayArrayGetSizeHeight(5)

This refers to the TerraDisplayArray [863] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTerraDisplayArrayGetSizeHeight(1)`

WDTerraDisplayArrayGetSizeWidth

WDTerraDisplayArrayGetSizeWidth(ID)

Example:
WDTerraDisplayArrayGetSizeWidth(5)

This refers to the TerraDisplayArray [863] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTerraDisplayArrayGetSizeWidth(1)`

WDTerraDisplayArrayLocation

WDTerraDisplayArrayLocation(ID,X,Y)

Example:
WDTerraDisplayArrayLocation(5,100,200)

Sets the position of the TerraDisplayArray [863] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDTerraDisplayArrayLocationLeft

WDTerraDisplayArrayLocationLeft(ID,X)

Example:
WDTerraDisplayArrayLocationLeft(5,100)

Sets the position of the TerraDisplayArray [863] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDTerraDisplayArrayLocationTop

WDTerraDisplayArrayLocationTop(ID,Y)

Example:
WDTerraDisplayArrayLocationTop(5,200)

Sets the position of the TerraDisplayArray [863] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDTerraDisplayArraySetFontFamily

WDTerraDisplayArraySetFontFamily(ID,FontName)

Example:
WDTerraDisplayArraySetFontFamily(5,"Arial")

This refers to the TerraDisplayArray [863] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Textbox, you can use a script like:
`WDTerraDisplayArraySetFontFamily(5,WDTerraDisplayArrayGetFontFamily(1))`

WDTerraDisplayArraySetFontSize

WDTerraDisplayArraySetFontSize(ID,FontSize)

Example:
WDTerraDisplayArraySetFontSize(5,20)

This refers to the TerraDisplayArray [863] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Textbox, you can use a script like:
`WDTerraDisplayArraySetFontSize(5,WDTerraDisplayArrayGetFontSize(1))`

WDTerraDisplayArraySize

WDTerraDisplayArraySize(ID,Width,Height)

Example:
WDTerraDisplayArraySize(5,100,40)

Sets the size of the TerraDisplayArray [863] with ID 5 to a width of 100px and a height of 40px.

WDTerraDisplayArraySizeHeight

WDTerraDisplayArraySizeHeight(ID,Height)

Example:
WDTerraDisplayArraySizeHeight(5,40)

Sets the size of the TerraDisplayArray [863] with ID 5 to a height of 40px but remains the current width.

WDTerraDisplayArraySizeWidth

WDTerraDisplayArraySizeWidth(ID,Width)

Example:
WDTerraDisplayArraySizeWidth(5,100)

Sets the size of the TerraDisplayArray[863] with ID 5 to a width of 100px but remains the current height.

WDTerraDisplayArrayUnfix

WDTerraDisplayArrayUnfix(ID)

Example:
WDTerraDisplayArrayUnfix(5)

This deactivates the option "Fix" in the Item Properties of the TerraDisplayArray[863] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDTextboxAddTextFromLabel

WDTextboxAddTextFromLabel(TXTBoxID,LabelID)

Example:
WDTextboxAddTextFromLabel(1,3)

Adds the text from Label 3 to the text of Textbox[923] 1.

WDTextboxAddTextFromLastSMS

WDTextboxAddTextFromLastSMS(TXTBoxID)

Example:
WDTextboxAddTextFromLastSMS(2)

This adds the penultimate incoming SMS message to the Textbox[923] with ID 2.

WDTextboxAddTextFromRSSFeed

WDTextboxAddTextFromRSSFeed(TXTBoxID,RSSFeedID)

Example:
WDTextboxAddTextFromRSSFeed(2,26)

This adds the RSS Feed[1493] including its title and body with ID 26 to the Textbox[923] with ID 2.

WDTextboxAddTextFromRSSFeedBody

WDTextboxAddTextFromRSSFeedBody(TXTBoxID,RSSFeedID)

Example:
WDTextboxAddTextFromRSSFeedBody(2,26)

This adds the RSS Feed[1493] body content with ID 26 to the Textbox[923] with ID 2.

WDTextboxAddTextFromRSSFeedTitle

WDTextboxAddTextFromRSSFeedTitle(TXTBoxID,RSSFeedID)

Example:
WDTextboxAddTextFromRSSFeedTitle(2,26)

This adds the RSS Feed [1493] title with ID 26 to the Textbox [923] with ID 2.

WDTextboxAddTextFromSMS

WDTextboxAddTextFromSMS(TXTBoxID,SMSID)

Example:
WDTextboxAddTextFromSMS(2,26)

This adds the SMS message [1494] with ID 26 to the Textbox [923] with ID 2.

WDTextboxAddTextFromTextbox

WDTextboxAddTextFromTextbox(TXTBoxID,TXTBoxSourceID)

Example:
WDTextboxAddTextFromTextbox(1,3)

Adds the text from Textbox 3 to the text of Textbox [923] 1.

WDTextboxAppend

WDTextboxAppend(ID,Value)

Example:
WDTextboxAppend(1,"Hello")

Adds the text "Hello" to the text already present in Textbox [923] 1.

Example 2:
WDTextboxAppend(1,varString)

The second example uses a variable [1900] name. In this case, the variable's content/value is added to the Textbox [923].

A nice example for this command is to compose a script inside a textbox which is then executed via the command WDTextboxExecuteAsScript(ID) [1834].

WDTextboxAppendFromFile

WDTextboxAppendFromFile(ID,Filename)

Example:
WDTextboxAppendFromFile(3,"C:\Christie\content\Test.txt")

This copies the content from the text file saved under `C:\Christie\content\Test.txt` and appends it to the text in the Widget Designer Textbox [923] with ID 3.

In case the imported data does not show the same letters etc., check the encoding of your file. Widget Designer imports all common encodings except ANSI. That is: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8.

WDTextboxAppendTextOnly

WDTextboxAppendTextOnly(ID,Text)

Example:
WDTextboxAppendTextOnly(1,"Hello")

Adds the text "Hello" to the text already present in Textbox 923 1.

Example 2:
WDTextboxAppendTextOnly(1,varString)

The second example uses a variable 1900 name. In this case, the variable's content/value is added to the Textbox 923.

A nice example for this command is to compose a script inside a textbox which is then executed via the command WDTextboxExecuteAsScript(ID) 1834.

WDTextboxAppendToFile

WDTextboxAppendToFile(ID,Filename)

Example:
WDTextboxAppendToFile(3,"C:\Christie\content\Test.txt")

This copies the text content from the Widget Designer Textbox 923 with ID 3 and appends it to the text from the file saved under the following path: `C:\Christie\content\Test.txt`.

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.

WDTextboxClear

WDTextboxClear(TXTBoxID)

Example:
WDTextboxClear(1)

Clears all text from the Textbox 923 1.

WDTextboxConvertToHorizontalScrolltextPNG

WDTextboxConvertToHorizontalScrolltextPNG(ID,File,Red,Green,Blue,Alpha)

Example:
WDTextboxConvertToHorizontalScrolltextPNG(3,"C:\Christie\content\blue.png",0,0,170,255)

First(this command converts the text of)Textbox 923 with ID 3 to a horizontal scrolltext, meaning that all line ends are deleted. Then it saves the text as an image file to the following path `C:\Christie\content\blue.png`. The image size, text color and other font properties accord to the Textbox font settings in Widget Designer whilst the image background color is set with the command. In the example, the color 0,0,170,255 is chosen for RGBA, resulting in a fully opaque blue background.

WDTextboxConvertToImage

WDTextboxConvertToImage(ID,File)

Example:
WDTextboxConvertToImage(3,"C:\Christie\content\text.jpg")

This command saves the text of Textbox 923 with ID 3 including all line ends as a (jpg, BMP,...) image file to the following path `C:\Christie\content\text.jpg`. The image size matches the Textbox size and the text

color and other font properties accord to the Textbox font settings in Widget Designer. The image background color is always black.

WDTextboxConvertToPNG

WDTextboxConvertToPNG(ID,File)

Example:
WDTextboxConvertToPNG(3,"C:\Christie\content\text.png")

This command saves the text of Textbox [923] with ID 3 including all line ends as a png-image file to the following path `C:\Christie\content\text.png`. The image size matches the Textbox size and the text color and other font properties accord to the Textbox font settings in Widget Designer. The image background color is always black.

WDTextBoxCssStyleDisable

WDTextBoxCssStyleDisable(ID,StyleID)

Example:
WDTextBoxCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Textbox [923] with ID 5.

WDTextBoxCssStyleEdit

WDTextBoxCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDTextBoxCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Textbox [923] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDTextBoxCssStyleEnable

WDTextBoxCssStyleEnable(ID,StyleID)

Example:
WDTextBoxCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Textbox [923] with ID 5.

WDTextboxExecuteAsScript

WDTextboxExecuteAsScript(ID)

Example:
WDTextboxExecuteAsScript(1)

Executes the text in Textbox [923] 1 as a command. You may use the command WDTextboxAppend [1832] to compose the script from single parts.

WDTextBoxFix

WDTextBoxFix(ID)

Example:
WDTextBoxFix(5)

This activates the option "Fix" in the Item Properties of the Textbox [923] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDTextboxFocus

WDTextboxFocus(ID)

Example:
WDTextboxFocus(3)

This sets the Textbox [923] with ID 3 into the focus, meaning that the next keyboard input will be added to this Textbox.

WDTextboxFromFile

WDTextboxFromFile(ID,Filename)

Example:
WDTextboxFromFile(3,"C:\Christie\content\Test.txt")

This copies the content from the text file saved under `C:\Christie\content\Test.txt` and overwrittes the text in the Widget Designer Textbox [923] with ID 3.

In case the imported data does not show the same letters etc., check the encoding of your file. Widget Designer imports all common encodings except ANSI. That is: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8.

WDTextboxGetFontFamily

WDTextboxGetFontFamily(ID)

Example:
varString = WDTextboxGetFontFamily(5)

This refers to the Textbox [923] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Textbox via a script like:
`WDTextboxSetFontFamily(5,WDTextboxGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTextboxGetFontFamily(5))`

WDTextboxGetFontSize

WDTextboxGetFontSize(ID)

Example:
varDouble = WDTextboxGetFontSize(5)

This refers to the Textbox [923] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Textbox via a script like:
`WDTextboxSetFontSize(5,WDTextboxGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTextboxGetFontSize(5))`

WDTextBoxGetLocationLeft

WDTextBoxGetLocationLeft(ID)

Example:
WDTextBoxGetLocationLeft(5)

This refers to the Textbox [923] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextBoxGetLocationLeft(1)`

WDTextBoxGetLocationTop

WDTextBoxGetLocationTop(ID)

Example:
WDTextBoxGetLocationTop(5)

This refers to the Textbox [923] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextBoxGetLocationTop(1)`

WDTextBoxGetSizeHeight

WDTextBoxGetSizeHeight(ID)

Example:
WDTextBoxGetSizeHeight(5)

This refers to the Textbox [923] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextBoxGetSizeHeight(1)`

WDTextBoxGetSizeWidth

WDTextBoxGetSizeWidth(ID)

Example:
WDTextBoxGetSizeWidth(5)

This refers to the Textbox [923] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextBoxGetSizeWidth(1)`

WDTextBoxLocation

WDTextBoxLocation(ID,X,Y)

Example:
WDTextBoxLocation(5,100,200)

Sets the position of the Textbox [923] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDTextBoxLocationLeft

WDTextBoxLocationLeft(ID,X)

Example:
WDTextBoxLocationLeft(5,100)

Sets the position of the Textbox [923] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDTextBoxLocationTop

WDTextBoxLocationTop(ID,Y)

Example:
WDTextBoxLocationTop(5,200)

Sets the position of the Textbox [923] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDTextboxNewline

WDTextboxNewline(ID)

Example:
WDTextboxNewline,1

Starts a new line in the Textbox [923] 1.

This way you may compose a script list inside a textbox that can be executed via the command WDTextboxExecuteAsScript,ID [1834].

WDTextboxRefresh

WDTextboxRefresh(ID)

Example:
WDTextboxRefresh(1)

Updates the text of Textbox [923] 1. This is useful when the Textbox Input Node [1088] is used and the Textbox ID inside the node is changed via a command.

WDTextboxReplaceLineEnd

WDTextboxReplaceLineEnd(ID,Text)

Example:
WDTextboxReplaceLineEnd(1,"FLIP")

This replaces each end of a line in Textbox [923] with ID 1 with the word "FLIP".
First Line
Second Line
Third Line
becomes: first lineFLIPsecond lineFLIPthird line

WDTextboxSetFontFamily

WDTextboxSetFontFamily(ID,FontName)

Example:
WDTextboxSetFontFamily(5,"Arial")

This refers to the Textbox [923] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Textbox, you can use a script like:
`WDTextboxSetFontFamily(5,WDTextboxGetFontFamily(1))`

WDTextboxSetFontSize

WDTextboxSetFontSize(ID,FontSize)

Example:
WDTextboxSetFontSize(5,20)

This refers to the Textbox [923] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Textbox, you can use a script like:
`WDTextboxSetFontSize(5,WDTextboxGetFontSize(1))`

WDTextBoxSize

WDTextBoxSize(ID,Width,Height)

Example:
WDTextBoxSize(5,100,40)

Sets the size of the Textbox [923] with ID 5 to a width of 100px and a height of 40px.

WDTextBoxSizeHeight

WDTextBoxSizeHeight(ID,Height)

Example:
WDTextBoxSizeHeight(5,40)

Sets the size of the Textbox [923] with ID 5 to a height of 40px but remains the current width.

WDTextBoxSizeWidth

WDTextBoxSizeWidth(ID,Width)

Example:
WDTextBoxSizeWidth(5,100)

Sets the size of the Textbox [923] with ID 5 to a width of 100px but remains the current height.

WDTextboxToFile

WDTextboxToFile(ID,Filename)

Example:
WDTextboxToFile(3,"C:\Christie\content\Test.txt")

This copies the text content from the Widget Designer Textbox [923] with ID 3 and overwrittes all text in the text file saved under the following path: `C:\Christie\content\Test.txt`.

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.

WDTextboxTrimEndToTotalCount

WDTextboxTrimEndToTotalCount(ID,Count)

Example:
WDTextboxTrimEndToTotalCount(1,8)

This removes all characters from Textbox [923] with ID 1, except the first 8 characters.

WDTextBoxUnfix

WDTextBoxUnfix(ID)

Example:
WDTextBoxUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Textbox [923] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDTextConvertToColoredQRCodePNG

WDTextConvertToColoredQRCodePNG(Size,R,G,B,A,R,G,B,A,File,Text)

Example:
WDTextConvertToColoredQRCodePNG(5,150,0,255,255,0,255,0,120,"C:\Christie\content\qr_col.png","Hello")

This command converts the text "Hello" to a colored QR code. The QR code size 5 results in an image width and height of 189px, see below. The foreground color is set to 150,0,255,255 for RGBA (a fully opaque purple) and the background to 0,255,0,120, a half transparent green.

The size 0 results in an 1px large image. Every further step adds 47px. Hence, a size of 5 results in 189px as 1px+4*47px=189px.

WDTextConvertToHorizontalScrollTextPNG

WDTextConvertToHorizontalScrollTextPNG(Text(),Font(),File,Red,Green,Blue,Alpha)

Example:
WDTextConvertToHorizontalScrollTextPNG("Hello","Mistral/12/Bold/170/255/170/170","C:\Christie\content\text.png",0,0,170,255)

This command saves the text "Hello" as an image file to the following path `C:\Christie\content\text.png`. The text font is set to "Mistral", the text size to 12pt, the text style to "Bold" and the text color to 170,0,170,170 for RGBA. The image background color is set to 0,0,170,255 for RGBA, resulting in a fully opaque blue background.

Font syntax: Name/Size/Style/R/G/B/A
Style options: Regular,Bold,Italic,Underline or Strikeout

WDTextConvertToQRCodePNG

WDTextConvertToQRCodePNG(File,Text)

Example:
WDTextConvertToQRCodePNG("C:\Christie\content\qr_bw.png","Widget Designer")

This command converts the text "Widget Designer" to a QR code. The black and white QR code is saved as an image file to the following path `C:\Christie\content\qr_bw.png`.

WDTextInputClear

WDTextInputClear(ID)

Example:
WDTextInputClear(1)

Clears all text from the Text Input 926 1.

WDTextInputCssStyleDisable

WDTextInputCssStyleDisable(ID,StyleID)

Example:
WDTextInputCssStyleDisable(5,2)

Disables the CSS Style 814 with ID 2 in the Item Properties of the Text Input 926 with ID 5.

WDTextInputCssStyleEdit

WDTextInputCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDTextInputCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style 814 with ID 2 in the Item Properties of the Text Input 926 with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDTextInputCssStyleEnable

WDTextInputCssStyleEnable(ID,StyleID)

Example:
WDTextInputCssStyleEnable(5,2)

Enables the CSS Style 814 with ID 2 in the Item Properties of the Text Input 926 with ID 5.

WDTextInputFix

WDTextInputFix(ID)

Example:
WDTextInputFix(5)

This activates the option "Fix" in the Item Properties of the Text Input 926 with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDTextInputGetFontFamily

WDTextInputGetFontFamily(ID)

Example:
varString = WDTextInputGetFontFamily(5)

This refers to the Text Input [926] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another Text Input via a script like:
`WDTextInputSetFontFamily(5,WDTextInputGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTextInputGetFontFamily(5))`

WDTextInputGetFontSize

WDTextInputGetFontSize(ID)

Example:
varDouble = WDTextInputGetFontSize(5)

This refers to the Text Input [926] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another Text Input via a script like:
`WDTextInputSetFontSize(5,WDTextInputGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTextInputGetFontSize(5))`

WDTextInputGetLocationLeft

WDTextInputGetLocationLeft(ID)

Example:
WDTextInputGetLocationLeft(5)

This refers to the Text Input [926] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0).
You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextInputGetLocationLeft(1)`

WDTextInputGetLocationTop

WDTextInputGetLocationTop(ID)

Example:
WDTextInputGetLocationTop(5)

This refers to the Text Input [926] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0).
You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextInputGetLocationTop(1)`

WDTextInputGetSizeHeight

WDTextInputGetSizeHeight(ID)

Example:
WDTextInputGetSizeHeight(5)

This refers to the Text Input [926] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextInputGetSizeHeight(1)`

WDTextInputGetSizeWidth

WDTextInputGetSizeWidth(ID)

Example:
WDTextInputGetSizeWidth(5)

This refers to the Text Input [926] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTextInputGetSizeWidth(1)`

WDTextInputGotoLine

WDTextInputGotoLine(ID,SourceID)

Example:
WDTextInputGotoLine(1,2)

Jumps to the line number 2 in Text Input [926] 1.

WDTextInputGotoLineFromTextBox

WDTextInputGotoLineFromTextBox(ID,SourceTextboxID)

Example:
WDTextInputGotoLineFromTextBox(1,2)

Jumps to the line in Text Input [926] 1, that is taken from the numeric entry in Textbox [923] 2.

WDTextInputLast

WDTextInputLast(ID)

Example:
WDTextInputLast(1)

Sets the Text Input [926] to the last available line.

WDTextInputLocation

WDTextInputLocation(ID,X,Y)

Example:
WDTextInputLocation(5,100,200)

Sets the position of the Text Input [926] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDTextInputLocationLeft

WDTextInputLocationLeft(ID,X)

Example:
WDTextInputLocationLeft(5,100)

Sets the position of the Text Input [926] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDTextInputLocationTop

WDTextInputLocationTop(ID,Y)

Example:
WDTextInputLocationTop(5,200)

Sets the position of the Text Input [926] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDTextInputNext

WDTextInputNext(ID)

Example:
WDTextInputNext(1)

Sets the Text Input [926] to the next available line.

WDTextInputReset

WDTextInputReset(ID)

Example:
WDTextInputReset(1)

Sets the Text Input [926] to the first available line.

WDTextInputSetFontFamily

WDTextInputSetFontFamily(ID,FontName)

Example:
WDTextInputSetFontFamily(5,"Arial")

This refers to the Text Input [926] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another Text Input, you can use a script like:
`WDTextInputSetFontFamily(5,WDTextInputGetFontFamily(1))`

WDTextInputSetFontSize

WDTextInputSetFontSize(ID,FontSize)

Example:
WDTextInputSetFontSize(5,20)

This refers to the Text Input [926] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another Text Input, you can use a script like:
`WDTextInputSetFontSize(5,WDTextInputGetFontSize(1))`

WDTextInputSize

WDTextInputSize(ID,Width,Height)

Example:
WDTextInputSize(5,100,40)

Sets the size of the Text Input [926] with ID 5 to a width of 100px and a height of 40px.

WDTextInputSizeHeight

WDTextInputSizeHeight(ID,Height)

Example:
WDTextInputSizeHeight(5,40)

Sets the size of the Text Input [926] with ID 5 to a height of 40px but remains the current width.

WDTextInputSizeWidth

WDTextInputSizeWidth(ID,Width)

Example:
WDTextInputSizeWidth(5,100)

Sets the size of the Text Input [926] with ID 5 to a width of 100px but remains the current height.

WDTextInputUnfix

WDTextInputUnfix(ID)

Example:
WDTextInputUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Text Input [926] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDTextInputUpdate

WDTextInputUpdate(ID)

Example:
WDTextInputUpdate(1)

Updates the current line in Text Input [926] 1 (no effect when Auto-Update is set in the Item Properties of this panel).

WdToggleKioskView

WdToggleKioskView

Example:
WdToggleKioskView

This command toggles the Kiosk View. If activated, all bars and menus are removed and the main GUI is resized to fullscreen. If deactivated, all bars and menus are back again and the main GUI is resized to a

maximized Window. You can also press the shortcut [F11] to toggle the Kiosk view. The Kiosk View gives you the largest available space to arrange any controls forming an individual user interface.

You can also press the shortcut [F11] to toggle the Kiosk view or use the commands WDEnterKioskView[1714] and WDLeaveKioskView[1753]

WDTransparencyKey

WDTransparencyKey(R,G,B)

Example:
WDTransparencyKey(255,255,255)

Sets the Transparency Key to white (Red:255, Green:255, Blue:255) and displays all pixels with this color as see-through (transparent) areas. Note that this effects the entire Widget Designer window. The command WDTransparencyKeyOff deactivates this feature again.

WDTransparencyKeyOff

WDTransparencyKeyOff

Example:
WDTransparencyKeyOff

This deactivates the Transparency Key, when used before with the command WDTransparencyKey.

WDTreeViewCssStyleDisable

WDTreeViewCssStyleDisable(ID,StyleID)

Example:
WDTreeViewCssStyleDisable(5,2)

Disables the CSS Style[814] with ID 2 in the Item Properties of the Tree View[929] with ID 5.

WDTreeViewCssStyleEdit

WDTreeViewCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDTreeViewCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style[814] with ID 2 in the Item Properties of the Tree View[929] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDTreeViewCssStyleEnable

WDTreeViewCssStyleEnable(ID,StyleID)

Example:
WDTreeViewCssStyleEnable(5,2)

Enables the CSS Style[814] with ID 2 in the Item Properties of the Tree View[929] with ID 5.

WDTreeViewFix

WDTreeViewFix(ID)

Example:
WDTreeViewFix(5)

This activates the option "Fix" in the Item Properties of the Tree View [929] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDTreeViewGetFontFamily

WDTreeViewGetFontFamily(ID)

Example:
varString = WDTreeViewGetFontFamily(5)

This refers to the TreeView [929] with ID 5 and returns its current font family name as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. another TreeView via a script like:
`WDTreeViewSetFontFamily(5,WDTreeViewGetFontFamily(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTreeViewGetFontFamily(5))`

WDTreeViewGetFontSize

WDTreeViewGetFontSize(ID)

Example:
varDouble = WDTreeViewGetFontSize(5)

This refers to the TreeView [929] with ID 5 and returns its current font size in "pt" as a double.
The first example shows how to use it with an existing (global) double variable [1900] named "varDouble".

You could also assign it directly to a Widget, e.g. another TreeView via a script like:
`WDTreeViewSetFontSize(5,WDTreeViewGetFontSize(1))`
or simply display it in the Debug Logger [812]: `DebugMessage(WDTreeViewGetFontSize(5))`

WDTreeViewGetLocationLeft

WDTreeViewGetLocationLeft(ID)

Example:
WDTreeViewGetLocationLeft(5)

This refers to the Tree View [929] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0).
You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTreeViewGetLocationLeft(1)`

WDTreeViewGetLocationTop

WDTreeViewGetLocationTop(ID)

Example:
WDTreeViewGetLocationTop(5)

This refers to the Tree View [929] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTreeViewGetLocationTop(1)`

WDTreeViewGetSizeHeight

WDTreeViewGetSizeHeight(ID)

Example:
WDTreeViewGetSizeHeight(5)

This refers to the Tree View [929] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTreeViewGetSizeHeight(1)`

WDTreeViewGetSizeWidth

WDTreeViewGetSizeWidth(ID)

Example:
WDTreeViewGetSizeWidth(5)

This refers to the Tree View [929] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDTreeViewGetSizeWidth(1)`

WDTreeViewLocation

WDTreeViewLocation(ID,X,Y)

Example:
WDTreeViewLocation(5,100,200)

Sets the position of the Tree View [929] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDTreeViewLocationLeft

WDTreeViewLocationLeft(ID,X)

Example:
WDTreeViewLocationLeft(5,100)

Sets the position of the Tree View [929] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDTreeViewLocationTop

WDTreeViewLocationTop(ID,Y)

Example:
WDTreeViewLocationTop(5,200)

Sets the position of the Tree View [929] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDTreeviewRefreshTree

WDTreeviewRefreshTree(ID)

Example:
WDTreeviewRefreshTree(1)

Refreshes the TreeView [929] with ID 1.

WDTreeviewRoot

WDTreeviewRoot(ID,Path)

Example:
WDTreeviewRoot(1,"C:\Christie\content")

Sets the root folder of TreeView [929] with ID 1 to "C:\Christie\content".

Note: The TreeView property "Source" needs to be set to "FileSystem" and the TreeView has to be refreshed with the command `WDTreeviewRefreshTree(1)`.

WDTreeViewSetBackgroundColor

WDTreeViewSetBackgroundColor(ID,R,G,B)

Example:
WDTreeViewSetBackgroundColor(1,0,120,150)

This changes the background color of the TreeView [929] with ID 1 to a greenish blue (Red=0, Green=120, Blue=150). The values for R,G,B range from 0 to 255.

WDTreeViewSetFontFamily

WDTreeViewSetFontFamily(ID,FontName)

Example:
WDTreeViewSetFontFamily(5,"Arial")

This refers to the TreeView [929] with ID 5 and sets its font family to "Arial".
If you like to copy and paste the font from a Widget, e.g. another TreeView, you can use a script like:
`WDTreeViewSetFontFamily(5,WDTreeViewGetFontFamily(1))`

WDTreeViewSetFontSize

WDTreeViewSetFontSize(ID,FontSize)

Example:
WDTreeViewSetFontSize(5,20)

This refers to the TreeView [929] with ID 5 and sets its font size to 20pt.
If you like to copy and paste the font size from a Widget, e.g. another TreeView, you can use a script like:
`WDTreeViewSetFontSize(5,WDTreeViewGetFontSize(1))`

WDTreeViewSetTextColor

WDTreeViewSetTextColor(ID,R,G,B)

Example:
WDTreeViewSetTextColor(1,255,150,0)

This changes the text color of the TreeView [929] with ID 1 to orange (Red=255, Green=150, Blue=0). The values for R,G,B range from 0 to 255.

WDTreeViewSize

WDTreeViewSize(ID,Width,Height)

Example:
WDTreeViewSize(5,100,40)

Sets the size of the Tree View [929] with ID 5 to a width of 100px and a height of 40px.

WDTreeViewSizeHeight

WDTreeViewSizeHeight(ID,Height)

Example:
WDTreeViewSizeHeight(5,40)

Sets the size of the Tree View [929] with ID 5 to a height of 40px but remains the current width.

WDTreeViewSizeWidth

WDTreeViewSizeWidth(ID,Width)

Example:
WDTreeViewSizeWidth(5,100)

Sets the size of the Tree View [929] with ID 5 to a width of 100px but remains the current height.

WDTreeViewUnfix

WDTreeViewUnfix(ID)

Example:
WDTreeViewUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Tree View [929] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVariableAppendLineToFile

WDVariableAppendLineToFile(VarName,FileName)

Example:
WDVariableAppendLineToFile("var_Message","C:\Temp\file.txt")

Appends the value of the variable [1900] "var_Message" as a new line to the text file "C:\Temp\file.txt". If the file does not exist yet, WD creates it.

If you like to append the message without creating a new line, please use the command WDVariableAppendToFile [1850].

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.

WDVariableAppendTimelogLineToFile

WDVariableAppendTimelogLineToFile(VarName,FileName)

Example:
WDVariableAppendTimelogLineToFile(var_Message,"C:\Temp\file.txt")

Appends the current date and time as well as the value of the variable [1900] "var_Message" as a new line to the text file "C:\Temp\file.txt". If the file does not exist yet, WD creates it.

E.g.: Nov 20, 2015 10:42:37 and var_Message = WD Message Test
Result: 20,11,2015,10,42,37,WD Message Test

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.

WDVariableAppendToFile

WDVariableAppendToFile(VarName,FileName)

Example:
WDVariableAppendToFile(var_Message,"C:\Temp\file.txt")

Appends the value of the variable [1900] "var_Message" to the text file "C:\Temp\file.txt". If the file does not exist yet, WD creates it.

If you like to append the message in a new line, please use the command WDVariableAppendLineToFile [1849].

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.

WDVariablesLoadFromJsonFile

WDVariablesLoadFromJsonFile(FilePath)

Example:
WDVariablesLoadFromJsonFile("C:\Christie\content\Items.json")

This command allows to update many variables [1900] (with various types) with only one file. The file needs to include the variable name and its (new) value in the format shown below. Remember to define the variables in Widget Designer before loading the file. The Debug Logger logs a message if the command cannot be executed, e.g. because a value stored in the JSON file does not match the variable type in Widget Designer.

Exemplary content of JSON file:

```
{
"var_string": ["sometext"],
"var_int": [123],
"var_bool": ["False"],
"var_list": ["Item1",2,3],
"var_date": ["2020-07-13 10:30:00.000"]
}
```

Note that color variables are not supported, please use the ToColor member [1916] instead. If you like to assign json variables, please use the command JsonLoadFileToVar [1546].

In case the imported data does not show the same letters etc., check the encoding of your file. Widget Designer imports all common encodings except ANSI. That is: UCS-2 LE / BE (also called Unicode or Unicode big endian) and UTF-8 (with or without BOM).

WDVariableToFile

WDVariableToFile(VarName,Filename)

Example:
WDVariableToFile("var_Message","C:\Temp\file.txt")

Writes the value of the variable [1900] "var_Message" to the text file "C:\Temp\file.txt". If the file does not exist yet, WD creates it.

Currently, the following variables can be exported: String, Integer, Double and Boolean.

In case the exported data does not show the same letters etc., check the encoding of your file. Widget Designer exports data with UTF-8 encoding.Currently the following variable types [1900] can be exported: String, Integer, Double, Boolean

WDVideoControl

WDVideoControl(ID,"Play"/"Loop"/"Pause"/"Stop")

Example:
WDVideoControl(2,"Play")

Plays the file that is loaded into the Videoplayer [932] with the ID 2.

Example 2:
WDVideoControl(2,"Loop")
Loops the file that is played in the Videoplayer [932] with the ID 2.

Example 3:
WDVideoControl(2,"Pause")
Pauses the file that is played in the Videoplayer [932] with the ID 2.

Example 4:
WDVideoControl(2,"Stop")
Stops the file that is played in the Videoplayer [932] with the ID 2.

WDVideoFile

WDVideoFile(ID,File)

Example:
WDVideoFile(2,"C:\Christie\content Pandora\MediaLoops 00201-00300\00214_kiosk.mpg")

Loads the file 00214_kiosk.mpg from the specified directory into the Videoplayer [932] with the ID 2.

WDVideoGotoTime

WDVideoGotoTime(ID,Second)

Example:
WDVideoGotoTime(1,80)

Jumps in the file loaded to Videoplayer [932] 1 to the time 1min 20 sec (80 sec).

WDVideoGotoTimeRelative

WDVideoGotoTimeRelative(ID,Second)

Example:
WDVideoGotoTimeRelative(1,-25)

Jumps in the file loaded to Videoplayer [932] 1 to the time 25 seconds backward.

To jump forward 15 seconds enter WDVideoGotoTimeRelative,1,15

WDVideoInputDisplayCssStyleDisable

WDVideoInputDisplayCssStyleDisable(ID,StyleID)

Example:
WDVideoInputDisplayCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Video Input Display [865] with ID 5.

WDVideoInputDisplayCssStyleEdit

WDVideoInputDisplayCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDVideoInputDisplayCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Video Input Display [865] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDVideoInputDisplayCssStyleEnable

WDVideoInputDisplayCssStyleEnable(ID,StyleID)

Example:
WDVideoInputDisplayCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Video Input Display [865] with ID 5.

WDVideoInputDisplayFix

WDVideoInputDisplayFix(ID)

Example:
WDVideoInputDisplayFix(5)

This activates the option "Fix" in the Item Properties of the Video Input Display [865] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVideoInputDisplayGetFormat

WDVideoInputDisplayGetFormat(ID)

Example:
varString = WDVideoInputDisplayGetFormat(5)

This refers to the Video Input Display [865] with ID 5 and returns its current format (e.g. MJPG 1280x720@30.00p 16:9) as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. a Label [888] via the script: `Label1.Text = WDVideoInputDisplayGetFormat(5)`
or simply display it in the Debug Logger [812]: `DebugMessage(WDVideoInputDisplayGetFormat(5))`

WDVideoInputDisplayGetFormatList

WDVideoInputDisplayGetFormatList(ID)

Example:
varList = WDVideoInputDisplayGetFormatList(5)

This refers to the Video Input Display [865] with ID 5 and returns available formats (e.g. MJPG 1280x720@30.00p 16:9) as a list.
The first example shows how to use it with an existing (global) list variable [1900] named "varList".

You could also write it into a Widget, e.g. a DropDown List [868] via this script and a local variable "temp":

```
var temp = WDVideoInputDisplayGetFormatList(5)
DropDownList1.SetItemsFromArray("temp")
```

or simply display it in the Debug Logger [812]: `DebugMessage(WDVideoInputDisplayGetFormat(5))`

WDVideoInputDisplayGetLocationLeft

WDVideoInputDisplayGetLocationLeft(ID)

Example:
WDVideoInputDisplayGetLocationLeft(5)

This refers to the Video Input Display [865] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoInputDisplayGetLocationLeft(1)`

WDVideoInputDisplayGetLocationTop

WDVideoInputDisplayGetLocationTop(ID)

Example:
WDVideoInputDisplayGetLocationTop(5)

This refers to the Video Input Display [865] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoInputDisplayGetLocationTop(1)`

WDVideoInputDisplayGetSizeHeight

WDVideoInputDisplayGetSizeHeight(ID)

Example:
WDVideoInputDisplayGetSizeHeight(5)

This refers to the Video Input Display [865] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoInputDisplayGetSizeHeight(1)`

WDVideoInputDisplayGetSizeWidth

WDVideoInputDisplayGetSizeWidth(ID)

Example:
WDVideoInputDisplayGetSizeWidth(5)

This refers to the Video Input Display [865] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoInputDisplayGetSizeWidth(1)`

WDVideoInputDisplayGetSource

WDVideoInputDisplayGetSource(ID)

Example:
varString = WDVideoInputDisplayGetSource(5)

This refers to the Video Input Display [865] with ID 5 and returns its current source / device (e.g. Integrated Webcam) as a string.
The first example shows how to use it with an existing (global) string variable [1900] named "varString".

You could also assign it directly to a Widget, e.g. a Label [888] via the script: `Label1.Text = WDVideoInputDisplayGetSource(5)`
or simply display it in the Debug Logger [812]: `DebugMessage(WDVideoInputDisplayGetSource(5))`

WDVideoInputDisplayGetSourceList

WDVideoInputDisplayGetSourceList(ID)

Example:
varList = WDVideoInputDisplayGetSourceList(5)

This refers to the Video Input Display [865] with ID 5 and returns available sources / devices (e.g. Integrated Webcam) as a list.
The first example shows how to use it with an existing (global) list variable [1900] named "varList".

You could also write it into a Widget, e.g. a DropDown List [868] via this script and a local variable "temp":

```
var temp = WDVideoInputDisplayGetSourceList(5)
DropDownList1.SetItemsFromArray("temp")
```

or simply display it in the Debug Logger [812]: `DebugMessage(WDVideoInputDisplayGetSourceList(5))`

WDVideoInputDisplayLocation

WDVideoInputDisplayLocation(ID,X,Y)

Example:
WDVideoInputDisplayLocation(5,100,200)

Sets the position of the Video Input Display [865] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDVideoInputDisplayLocationLeft

WDVideoInputDisplayLocationLeft(ID,X)

Example:
WDVideoInputDisplayLocationLeft(5,100)

Sets the position of the Video Input Display [865] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDVideoInputDisplayLocationTop

WDVideoInputDisplayLocationTop(ID,Y)

Example:
WDVideoInputDisplayLocationTop(5,200)

Sets the position of the Video Input Display [865] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDVideoInputDisplaySaveCurrentImage

WDVideoInputDisplaySaveCurrentImage(ID,Path)

Example:
WDVideoInputDisplaySaveCurrentImage(5,"C:\Christie\content\frame")

This saves the currently displayed image from the Video Input Display [865] with ID 5 in the JPEG format with the name "frame" under the location `C:\Christie\content`.

The following example, shows how to add a time stamp to the file name. First, a local variable [1903] is created holding the path as above. Then the time stamp is appended using the internal variable "Now" (in string format). As it contains a colon, which is not allowed in file names, it is replaced with an underscore using the member "Replace [1914]". The resulting name would be for example: `C:\Christie\content\frame_at_2021-02-10 13_55_50.194`

```
var path = "C:\Christie\content\frame_at_"+Now.ToString.Replace(":","_")
WDVideoInputDisplaySaveCurrentImage(1,path)
```

If you like to use images from the Video Input Display in Pandoras Box, it might be a more convenient workflow to use the Video Snapshot Button [837] instead.

WDVideoInputDisplaySetFormat

WDVideoInputDisplaySetFormat(ID,FormatName)

Example:
WDVideoInputDisplaySetFormat(5,"MJPG 1280x720@30.00p 16:9")

This refers to the Video Input Display [865] with ID 5 and changes the current format to one that is called "MJPG 1280x720@30.00p 16:9" if this format is available in the Properties dialog > Format.
It might be of interest that you can return all available format names via the command `WDVideoInputDisplayGetFormatList` [1853] or set it via `WDVideoInputDisplaySetFormatByIndex` [1855].

WDVideoInputDisplaySetFormatByIndex

WDVideoInputDisplaySetFormatByIndex(ID,Index)

Example:
WDVideoInputDisplaySetFormatByIndex(5,0)

This refers to the Video Input Display [865] with ID 5 and changes the current format to the first entry available in the Properties dialog > Format.
It might be of interest that you can return all available format names via the command WDVideoInputDisplayGetFormatList [1853].

WDVideoInputDisplaySetSource

WDVideoInputDisplaySetSource(ID,SourceName)

Example:
WDVideoInputDisplaySetSource(5,"Integrated Webcam")

This refers to the Video Input Display [865] with ID 5 and changes the current source / device to one that is called "Integrated Webcam" if this source is available in the Properties dialog > Device.
It might be of interest that you can return all available source names via the command WDVideoInputDisplayGetSourceList [1854] or set it via WDVideoInputDisplaySetSourceByIndex [1856].

WDVideoInputDisplaySetSourceByIndex

WDVideoInputDisplaySetSourceByIndex(ID,Index)

Example:
WDVideoInputDisplaySetSourceByIndex(5,0)

This refers to the Video Input Display [865] with ID 5 and changes the current source / device to the first entry available in the Properties dialog > Device.
It might be of interest that you can return all available source names via the command WDVideoInputDisplayGetSourceList [1854].

WDVideoInputDisplaySize

WDVideoInputDisplaySize(ID,Width,Height)

Example:
WDVideoInputDisplaySize(5,100,40)

Sets the size of the Video Input Display [865] with ID 5 to a width of 100px and a height of 40px.

WDVideoInputDisplaySizeHeight

WDVideoInputDisplaySizeHeight(ID,Height)

Example:
WDVideoInputDisplaySizeHeight(5,40)

Sets the size of the Video Input Display [865] with ID 5 to a height of 40px but remains the current width.

WDVideoInputDisplaySizeWidth

WDVideoInputDisplaySizeWidth(ID,Width)

Example:
WDVideoInputDisplaySizeWidth(5,100)

Sets the size of the Video Input Display [865] with ID 5 to a width of 100px but remains the current height.

WDVideoInputDisplayUnfix

WDVideoInputDisplayUnfix(ID)

Example:
WDVideoInputDisplayUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Video Input Display [865] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVideologgerAddComment

WDVideologgerAddComment(Comment)

Example:
WDVideologgerAddComment("On ne voit bien qu'avec le coeur")

Adds the comment "On ne voit bien qu'avec le coeur" to the Video Logger [1507] log with the time stamp when the command was executed.

WDVideologgerAddDevice

WDVideologgerAddDevice(SiteID,DeviceID,MinOpacity)

Example:
WDVideologgerAddDevice(2,1,0)

Adds "Site 2 - Layer 1 - Minimum opacity: 0" to the Video Logger' [1507]s device list.

WDVideologgerAddMediaComment

WDVideologgerAddMediaComment(Comment)

Example:
WDVideologgerAddMediaComment("L'essentiel est invisible pour les yeux")

Adds the comment "L'essentiel est invisible pour les yeux" to the Video Logger [1507] log, together with the current media state according to the Video Logger's device list.

WDVideologgerClearDevices

WDVideologgerClearDevices

Example:
WDVideologgerClearDevices

Clears the Video Logger [1507] device list.

WDVideologgerClearlog

WDVideologgerClearlog

Example:
WDVideologgerClearlog

Clears the Video Logger [1507] log.

WDVideologgerSave

WDVideologgerSave(Path)

Example:
WDVideologgerSave("C:\coolux\wd_video-logger.txt")

Saves a copy of the current Video Logger [1507] log as "vlog1.txt" to the specified path.

WDVideologgerStart

WDVideologgerStart

Example:
WDVideologgerStart

Starts the Video Logger [1507] log.

WDVideologgerStop

WDVideologgerStop

Example:
WDVideologgerStop

Pauses the Video Logger [1507] log.

WDVideoPlayerCssStyleDisable

WDVideoPlayerCssStyleDisable(ID,StyleID)

Example:
WDVideoPlayerCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Video Player [932] with ID 5.

WDVideoPlayerCssStyleEdit

WDVideoPlayerCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDVideoPlayerCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Video Player [932] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDVideoPlayerCssStyleEnable

WDVideoPlayerCssStyleEnable(ID,StyleID)

Example:
WDVideoPlayerCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Video Player [932] with ID 5.

WDVideoPlayerFix

WDVideoPlayerFix(ID)

Example:
WDVideoPlayerFix(5)

This activates the option "Fix" in the Item Properties of the Video Player [932] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVideoPlayerGetLocationLeft

WDVideoPlayerGetLocationLeft(ID)

Example:
WDVideoPlayerGetLocationLeft(5)

This refers to the Video Player [932] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoPlayerGetLocationLeft(1)`

WDVideoPlayerGetLocationTop

WDVideoPlayerGetLocationTop(ID)

Example:
WDVideoPlayerGetLocationTop(5)

This refers to the Video Player [932] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoPlayerGetLocationTop(1)`

WDVideoPlayerGetSizeHeight

WDVideoPlayerGetSizeHeight(ID)

Example:
WDVideoPlayerGetSizeHeight(5)

This refers to the Video Player [932] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoPlayerGetSizeHeight(1)`

WDVideoPlayerGetSizeWidth

WDVideoPlayerGetSizeWidth(ID)

Example:
WDVideoPlayerGetSizeWidth(5)

This refers to the Video Player [932] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoPlayerGetSizeWidth(1)`

WDVideoPlayerLocation

WDVideoPlayerLocation(ID,X,Y)

Example:
WDVideoPlayerLocation(5,100,200)

Sets the position of the Video Player [932] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDVideoPlayerLocationLeft

WDVideoPlayerLocationLeft(ID,X)

Example:
WDVideoPlayerLocationLeft(5,100)

Sets the position of the Video Player [932] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDVideoPlayerLocationTop

WDVideoPlayerLocationTop(ID,Y)

Example:
WDVideoPlayerLocationTop(5,200)

Sets the position of the Video Player [932] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDVideoPlayerSize

WDVideoPlayerSize(ID,Width,Height)

Example:
WDVideoPlayerSize(5,100,40)

Sets the size of the Video Player [932] with ID 5 to a width of 100px and a height of 40px.

WDVideoPlayerSizeHeight

WDVideoPlayerSizeHeight(ID,Height)

Example:
WDVideoPlayerSizeHeight(5,40)

Sets the size of the Video Player [932] with ID 5 to a height of 40px but remains the current width.

WDVideoPlayerSizeWidth

WDVideoPlayerSizeWidth(ID,Width)

Example:
WDVideoPlayerSizeWidth(5,100)

Sets the size of the Video Player [932] with ID 5 to a width of 100px but remains the current height.

WDVideoPlayerUnfix

WDVideoPlayerUnfix(ID)

Example:
WDVideoPlayerUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Video Player [932] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVideoSetOpacity

WDVideoSetOpacity(ID,Value)

Example:
WDVideoSetOpacity(1,125)

Sets the opacity of the WD Videoplayer [932] with ID 1 to 125 which is half transparent. The opacity value ranges from 0 (fully transparent) to 255 (fully opaque).

WDVideoSetVolume

WDVideoSetVolume(ID,Value)

Example:
WDVideoSetVolume(1,5000)

Sets the volume of the WD Videoplayer [932] to 50000. The volume value ranges from 0 to 10000.

WDVideoSnapshotAddTextFromLabel

WDVideoSnapshotAddTextFromLabel(ID,LabelID)

Example:
WDVideoSnapshotAddTextFromLabel(6,3)

Adds the text from Label 3 to the current Label of the Video Snapshot [837] button with ID 6.

WDVideoSnapshotClick

WDVideoSnapshotClick(ID)

Example:
WDVideoSnapshotClick(5)

Clicks the Video Snapshot [837] button with ID 5.

If you like to execute this command addressing many buttons, the chapter "For loop" and "Project and Context Member" show some interesting examples.

WDVideoSnapshotClickImage

WDVideoSnapshotClickImage(ID,File)

Example:
WDVideoSnapshotClickImage(1,"C:\Data\Snapshot_C")

This sets the image for the clicked status of Video Snapshot [837] button 1 to the image saved under the specified path.

WDVideoSnapshotClickImageResource

WDVideoSnapshotClickImageResource(ID,ResourceName)

Example:
WDVideoSnapshotClickImageResource(5,"Default\Button\Lock")

This sets the image for the clicked status of the Video Snapshot [837] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDVideoSnapshotCssStyleDisable

WDVideoSnapshotCssStyleDisable(ID,StyleID)

Example:
WDVideoSnapshotCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Video Snapshot [837] button with ID 5.

WDVideoSnapshotCssStyleEdit

WDVideoSnapshotCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDVideoSnapshotCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Video Snapshot [837] button with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDVideoSnapshotCssStyleEnable

WDVideoSnapshotCssStyleEnable(ID,StyleID)

Example:
WDVideoSnapshotCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Video Snapshot [837] button with ID 5.

WDVideoSnapshotFix

WDVideoSnapshotFix(ID)

Example:
WDVideoSnapshotFix(5)

This activates the option "Fix" in the Item Properties of the Video Snapshot [837] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVideoSnapshotGetLocationLeft

WDVideoSnapshotGetLocationLeft(ID)

Example:
WDVideoSnapshotGetLocationLeft(5)

This refers to the Video Snapshot [837] button with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoSnapshotGetLocationLeft(1)`

WDVideoSnapshotGetLocationTop

WDVideoSnapshotGetLocationTop(ID)

Example:
WDVideoSnapshotGetLocationTop(5)

This refers to the Video Snapshot [837] button with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoSnapshotGetLocationTop(1)`

WDVideoSnapshotGetSizeHeight

WDVideoSnapshotGetSizeHeight(ID)

Example:
WDVideoSnapshotGetSizeHeight(5)

This refers to the Video Snapshot [837] button with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoSnapshotGetSizeHeight(1)`

WDVideoSnapshotGetSizeWidth

WDVideoSnapshotGetSizeWidth(ID)

Example:
WDVideoSnapshotGetSizeWidth(5)

This refers to the Video Snapshot [837] button with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVideoSnapshotGetSizeWidth(1)`

WDVideoSnapshotHighlightImage

WDVideoSnapshotHighlightImage(ID,Existing file)

Example:
WDVideoSnapshotHighlightImage(5,"C:\Christie\clicked.png")

This sets the image for the highlighted status of the Video Snapshot [837] button with ID 5 to the image saved under the specified path.

WDVideoSnapshotHighlightImageResource

WDVideoSnapshotHighlightImageResource(ID,Text)

Example:
WDVideoSnapshotHighlightImageResource(5,"Default\Button\Lock")

This sets the image for the highlighted status of the Video Snapshot [837] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDVideoSnapshotLabel

WDVideoSnapshotLabel(ID,Text)

Example:
WDVideoSnapshotLabel(5,"Show")

Labels the Video Snapshot [837] button with ID 5 with the word "Show".

WDVideoSnapshotLabelColor

WDVideoSnapshotLabelColor(ID,R,G,B)

Example:
WDVideoSnapshotLabelColor(5,255,90,0)

Changes the label color of the Video Snapshot [837] button with ID 5 to orange (Red=255, Green=90, Blue=0). The values for R,G,B range from 0 to 255.

WDVideoSnapshotLocation

WDVideoSnapshotLocation(ID,X,Y)

Example:
WDVideoSnapshotLocation(5,100,200)

Sets the position of the Video Snapshot [837] button with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDVideoSnapshotLocationLeft

WDVideoSnapshotLocationLeft(ID,X)

Example:
WDVideoSnapshotLocationLeft(5,100)

Sets the position of the Video Snapshot [837] button with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDVideoSnapshotLocationTop

WDVideoSnapshotLocationTop(ID,Y)

Example:
WDVideoSnapshotLocationTop(5,200)

Sets the position of the Video Snapshot [837] button with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDVideoSnapshotReleaseImage

WDVideoSnapshotReleaseImage(ID,Existing file)

Example:
WDVideoSnapshotReleaseImage(5,"C:\Christie\clicked.png")

This sets the image for the released status of the Video Snapshot [837] button with ID 5 to the image saved under the specified path.

WDVideoSnapshotReleaseImageResource

WDVideoSnapshotReleaseImageResource(ID,Text)

Example:
WDVideoSnapshotReleaseImageResource(5,"Default\Button\Lock")

This sets the image for the released status of the Video Snapshot [837] button with ID 5 to the image saved under the specified name in the "Image Resource Manager" which is also accessible through the "Res" button in the Item Properties dialog of the button.

WDVideoSnapshotReload

WDVideoSnapshotReload(ID)

Example:
WDVideoSnapshotReload(1)

Refreshes the image sent to Pandoras Box by the Video Snapshot [837] button with ID 1.

WDVideoSnapshotSize

WDVideoSnapshotSize(ID,Width,Height)

Example:
WDVideoSnapshotSize(5,100,40)

Sets the size of the Video Snapshot [837] button with ID 5 to a width of 100px and a height of 40px.

WDVideoSnapshotSizeHeight

WDVideoSnapshotSizeHeight(ID,Height)

Example:
WDVideoSnapshotSizeHeight(5,40)

Sets the size of the Video Snapshot [837] button with ID 5 to a height of 40px but remains the current width.

WDVideoSnapshotSizeWidth

WDVideoSnapshotSizeWidth(ID,Width)

Example:
WDVideoSnapshotSizeWidth(5,100)

Sets the size of the Video Snapshot [837] button with ID 5 to a width of 100px but remains the current height.

WDVideoSnapshotTextFromLabel

WDVideoSnapshotTextFromLabel(ID,LabelID)

Example:
WDVideoSnapshotTextFromLabel(6,3)

Labels the Video Snapshot [837] button with ID 6 with the text from Label 3.

WDVideoSnapshotTint

WDVideoSnapshotTint(ID,Red,Green,Blue)

Example:
WDVideoSnapshotTint(5,255,180,0)

Changes the button color of the Video Snapshot [837] button with ID 5 to orange (Red=255, Green=180, Blue=0). The values for R,G,B range from 0 to 255.

WDVideoSnapshotUnfix

WDVideoSnapshotUnfix(ID)

Example:
WDVideoSnapshotUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Video Snapshot [837] button with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVideoVolumeDown

WDVideoVolumeDown(ID,Value)

Example:
WDVideoVolumeDown(2,10)

Decreases the volume of the Videoplayer [932] with the ID 2 by 10 steps. The value for volume ranges from 0 to 100.

WDVideoVolumeUp

WDVideoVolumeUp(ID,Value)

Example:
WDVideoVolumeUp(2,10)

Increases the volume of the Videoplayer [932] with the ID 2 by 10 steps. The value for volume ranges from 0 to 100.

WDVncPanelConnect

WDVncPanelConnect(ID)

Example:
WDVncPanelConnect(1)

This command starts the VNC session of the VNC Panel [894] with the ID 1.

WDVncPanelCssStyleDisable

WDVncPanelCssStyleDisable(ID,StyleID)

Example:
WDVncPanelCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the VNC Panel [894] with ID 5.

WDVncPanelCssStyleEdit

WDVncPanelCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDVncPanelCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the VNC Panel [894] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDVncPanelCssStyleEnable

WDVncPanelCssStyleEnable(ID,StyleID)

Example:
WDVncPanelCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the VNC Panel [894] with ID 5.

WDVncPanelDisconnect

WDVncPanelDisconnect(VNCPanelID)

Example:
WDVncPanelDisconnect(1)

This command stops the VNC session of the VNC Panel [894] with the ID 1.

WDVncPanelFix

WDVncPanelFix(ID)

Example:
WDVncPanelFix(5)

This activates the option "Fix" in the Item Properties of the VNC Panel [894] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDVncPanelGetLocationLeft

WDVncPanelGetLocationLeft(ID)

Example:
WDVncPanelGetLocationLeft(5)

This refers to the VNC Panel [894] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVncPanelGetLocationLeft(1)`

WDVncPanelGetLocationTop

WDVncPanelGetLocationTop(ID)

Example:
WDVncPanelGetLocationTop(5)

This refers to the VNC Panel [894] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVncPanelGetLocationTop(1)`

WDVncPanelGetSizeHeight

WDVncPanelGetSizeHeight(ID)

Example:
WDVncPanelGetSizeHeight(5)

This refers to the VNC Panel [894] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVncPanelGetSizeHeight(1)`

WDVncPanelGetSizeWidth

WDVncPanelGetSizeWidth(ID)

Example:
WDVncPanelGetSizeWidth(5)

This refers to the VNC Panel [894] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDVncPanelGetSizeWidth(1)`

WDVncPanelLocation

WDVncPanelLocation(ID,X,Y)

Example:
WDVncPanelLocation(5,100,200)

Sets the position of the VNC Panel [894] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDVncPanelLocationLeft

WDVncPanelLocationLeft(ID,X)

Example:
WDVncPanelLocationLeft(5,100)

Sets the position of the VNC Panel [894] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDVncPanelLocationTop

WDVncPanelLocationTop(ID,Y)

Example:
WDVncPanelLocationTop(5,200)

Sets the position of the VNC Panel [894] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDVncPanelSize

WDVncPanelSize(ID,Width,Height)

Example:
WDVncPanelSize(5,100,40)

Sets the size of the VNC Panel [894] with ID 5 to a width of 100px and a height of 40px.

WDVncPanelSizeHeight

WDVncPanelSizeHeight(ID,Height)

Example:
WDVncPanelSizeHeight(5,40)

Sets the size of the VNC Panel [894] with ID 5 to a height of 40px but remains the current width.

WDVncPanelSizeWidth

WDVncPanelSizeWidth(ID,Width)

Example:
WDVncPanelSizeWidth(5,100)

Sets the size of the VNC Panel [894] with ID 5 to a width of 100px but remains the current height.

WDVncPanelUnfix

WDVncPanelUnfix(ID)

Example:
WDVncPanelUnfix(5)

This deactivates the option "Fix" in the Item Properties of the VNC Panel [894] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDWait

WDWait(Seconds)

Example:
WDWait(5.5)

WD waits 5.5 seconds (5500ms) before the next command in the script list is executed.

WDWebBrowserClearCache

WDWebBrowserClearCache

Example:
WDWebBrowserClearCache

Clears the cache of the Web Browser [934].

WDWebBrowserContent

WDWebBrowserContent(BrowserID,Url)

Example:
WDWebBrowserContent(1,"www.coolux.de")

Displays the web site with the address http://www.coolux.de in the Web Browser [934] with the ID 1.

WDWebBrowserCssStyleDisable

WDWebBrowserCssStyleDisable(ID,StyleID)

Example:
WDWebBrowserCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Web Browser [934] with ID 5.

WDWebBrowserCssStyleEdit

WDWebBrowserCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDWebBrowserCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Web Browser [934] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDWebBrowserCssStyleEnable

WDWebBrowserCssStyleEnable(ID,StyleID)

Example:
WDWebBrowserCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Web Browser [934] with ID 5.

WDWebBrowserFix

WDWebBrowserFix(ID)

Example:
WDWebBrowserFix(5)

This activates the option "Fix" in the Item Properties of the Web Browser [934] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDWebBrowserGetLocationLeft

WDWebBrowserGetLocationLeft(ID)

Example:
WDWebBrowserGetLocationLeft(5)

This refers to the Web Browser [934] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWebBrowserGetLocationLeft(1)`

WDWebBrowserGetLocationTop

WDWebBrowserGetLocationTop(ID)

Example:
WDWebBrowserGetLocationTop(5)

This refers to the Web Browser [934] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWebBrowserGetLocationTop(1)`

WDWebBrowserGetSizeHeight

WDWebBrowserGetSizeHeight(ID)

Example:
WDWebBrowserGetSizeHeight(5)

This refers to the Web Browser [934] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWebBrowserGetSizeHeight(1)`

WDWebBrowserGetSizeWidth

WDWebBrowserGetSizeWidth(ID)

Example:
WDWebBrowserGetSizeWidth(5)

This refers to the Web Browser [934] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWebBrowserGetSizeWidth(1)`

WDWebBrowserLocation

WDWebBrowserLocation(ID,X,Y)

Example:
WDWebBrowserLocation(5,100,200)

Sets the position of the Web Browser [934] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDWebBrowserLocationLeft

WDWebBrowserLocationLeft(ID,X)

Example:
WDWebBrowserLocationLeft(5,100)

Sets the position of the Web Browser [934] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDWebBrowserLocationTop

WDWebBrowserLocationTop(ID,Y)

Example:
WDWebBrowserLocationTop(5,200)

Sets the position of the Web Browser [934] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDWebBrowserSize

WDWebBrowserSize(ID,Width,Height)

Example:
WDWebBrowserSize(5,100,40)

Sets the size of the Web Browser [934] with ID 5 to a width of 100px and a height of 40px.

WDWebBrowserSizeHeight

WDWebBrowserSizeHeight(ID,Height)

Example:
WDWebBrowserSizeHeight(5,40)

Sets the size of the Web Browser [934] with ID 5 to a height of 40px but remains the current width.

WDWebBrowserSizeWidth

WDWebBrowserSizeWidth(ID,Width)

Example:
WDWebBrowserSizeWidth(5,100)

Sets the size of the Web Browser [934] with ID 5 to a width of 100px but remains the current height.

WDWebBrowserUnfix

WDWebBrowserUnfix(ID)

Example:
WDWebBrowserUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Web Browser [934] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDWheelAbortAllFades

WDWheelAbortAllFades

Example:
WDWheelAbortAllFades

Aborts all currently running fades on every Wheel [883].

WDWheelAbortFade

WDWheelAbortFade(ID)

Example:
WDWheelAbortFade(3)

Aborts the currently running fade of Wheel [883] 3.

WDWheelCssStyleDisable

WDWheelCssStyleDisable(ID,StyleID)

Example:
WDWheelCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Wheel [883] with ID 5.

WDWheelCssStyleEdit

WDWheelCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDWheelCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the Wheel [883] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDWheelCssStyleEnable

WDWheelCssStyleEnable(ID,StyleID)

Example:
WDWheelCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Wheel [883] with ID 5.

WDWheelDown

WDWheelDown(ID,Value)

Example:
WDWheelDown(5,20)

Fades the current value of Wheel [883] 5 down to the minimum value within 2 seconds.

WDWheelFactor

WDWheelFactor(ID,Value)

Example:
WDWheelFactor(7,5)

Sets the factor of Wheel [883] 7 to the value 5.

WDWheelFix

WDWheelFix(ID)

Example:
WDWheelFix(5)

This activates the option "Fix" in the Item Properties of the Wheel [883] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDWheelGetLocationLeft

WDWheelGetLocationLeft(ID)

Example:
WDWheelGetLocationLeft(5)

This refers to the Wheel [883] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWheelGetLocationLeft(1)`

WDWheelGetLocationTop

WDWheelGetLocationTop(ID)

Example:
WDWheelGetLocationTop(5)

This refers to the Wheel [883] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWheelGetLocationTop(1)`

WDWheelGetSizeHeight

WDWheelGetSizeHeight(ID)

Example:
WDWheelGetSizeHeight(5)

This refers to the Wheel [883] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWheelGetSizeHeight(1)`

WDWheelGetSizeWidth

WDWheelGetSizeWidth(ID)

Example:
WDWheelGetSizeWidth(5)

This refers to the Wheel [883] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDWheelGetSizeWidth(1)`

WDWheelGoDown

WDWheelGoDown(ID,Value)

Example:
WDWheelGoDown(7,10)

Subtracts immediately 10 values from the value of Wheel [883] 7.

WDWheelGoUp

WDWheelGoUp(ID,Value)

Example:
WDWheelGoUp(7,10)

Adds immediately 10 values to the value of Wheel [883] 7.

WDWheelInfoMidiInputDisable

WDWheelInfoMidiInputDisable(ID)

Example:
WDWheelInfoMidiInputDisable(3)

Disables the MIDI input of Wheel [883] 3.

WDWheelInfoMidiInputEnable

WDWheelInfoMidiInputEnable(ID)

Example:
WDWheelInfoMidiInputEnable(3)

Enables the MIDI input of Wheel [883] 3.

WDWheelInfoMidiOutputDisable

WDWheelInfoMidiOutputDisable(ID)

Example:
WDWheelInfoMidiOutputDisable(3)

Disables the MIDI output of Wheel [883] 3.

WDWheelInfoMidiOutputEnable

WDWheelInfoMidiOutputEnable(ID)

Example:
WDWheelInfoMidiOutputEnable(3)

Enables the MIDI output of Wheel [883] 3.

WDWheelLocation

WDWheelLocation(ID,X,Y)

Example:
WDWheelLocation(5,100,200)

Sets the position of the Wheel [883] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDWheelLocationLeft

WDWheelLocationLeft(ID,X)

Example:
WDWheelLocationLeft(5,100)

Sets the position of the Wheel [883] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDWheelLocationTop

WDWheelLocationTop(ID,Y)

Example:
WDWheelLocationTop(5,200)

Sets the position of the Wheel [883] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDWheelSize

WDWheelSize(ID,Width,Height)

Example:
WDWheelSize(5,100,40)

Sets the size of the Wheel [883] with ID 5 to a width of 100px and a height of 40px.

WDWheelSizeHeight

WDWheelSizeHeight(ID,Height)

Example:
WDWheelSizeHeight(5,40)

Sets the size of the Wheel [883] with ID 5 to a height of 40px but remains the current width.

WDWheelSizeWidth

WDWheelSizeWidth(ID,Width)

Example:
WDWheelSizeWidth(5,100)

Sets the size of the Wheel [883] with ID 5 to a width of 100px but remains the current height.

WDWheelToValue

WDWheelToValue(ID,Time,Value)

Example:
WDWheelToValue(5,2,100)

Fades the current value of Wheel [883] 5 to value 100 within 2 seconds.

WDWheelUnfix

WDWheelUnfix(ID)

Example:
WDWheelUnfix(5)

This deactivates the option "Fix" in the Item Properties of the Wheel [883] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDWheelUp

WDWheelUp(ID,Value)

Example:
WDWheelUp(5,2)

Fades the current value of Wheel [883] 5 up to the maximum value within 2 seconds.

WDWheelValue

WDWheelValue(ID,Value)

Example:
WDWheelValue(7,33)

Sets the Wheel [883] 7 to value 33 immediately.

WDWindowCssStyleDisable

WDWindowCssStyleDisable(WindowName,StyleID)

Example:
WDWindowCssStyleDisable("Window5",2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the Window [803] with the name "Window5".

WDWindowCssStyleEnable

WDWindowCssStyleEnable(WindowName,StyleID)

Example:
WDWindowCssStyleEnable("Window5",2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the Window [803] with the name "Window5".

WDWindowFocus

WDWindowFocus(WindowName)

Example:
WDWindowFocus("Window2")

Sets the focus on the Widget Designer Window with the name "Window2". If the cursor was located in that window in a specific Widget before, e.g. an InputBox, it will be located there again.

WDWindowLocation

WDWindowLocation(X,Y)

Example:
WDWindowLocation(500,300)

Locates the Widget Designer window (upper left corner) 500px from the left, and 300px from the top of the screen.

Note: This command does not work in full-screen mode.

WDWindowSize

WDWindowSize(W,H)

Example:
WDWindowSize(1000,600)

Sets the width of the Widget Designer window to 1000px and the height to 600px.

Note: This command does not work in full-screen mode.

WDWindowStateBorderless

WDWindowStateBorderless

Example:
WDWindowStateBorderless

Removes the border of the Widget Designer window in windowed mode.

WDWindowStateMaximized

WDWindowStateMaximized

Example:
WDWindowStateMaximized

Maximizes the Widget Designer Window.

WDWindowStateMinimized

WDWindowStateMinimized

Example:
WDWindowStateMinimized

Minimizes the Widget Designer Window.

WDWindowStateNormal

WDWindowStateNormal

Example:
WDWindowStateNormal

Restores the last size of the Widget Designer window in windowed mode.

WDXyPanelCssStyleDisable

WDXyPanelCssStyleDisable(ID,StyleID)

Example:
WDXyPanelCssStyleDisable(5,2)

Disables the CSS Style [814] with ID 2 in the Item Properties of the XY Panel [896] with ID 5.

WDXyPanelCssStyleEdit

WDXyPanelCssStyleEdit(ID,StyleID,ParamName,Value)

Example:
WDXyPanelCssStyleEdit(5,2,"StartOpacity",50)

Edits the CSS Style [814] with ID 2 in the Item Properties of the XY Panel [896] with ID 5. The parameter "StartOpacity", e.g. from the Style "2D\FadeIn", is set to a value of 150.

WDXyPanelCssStyleEnable

WDXyPanelCssStyleEnable(ID,StyleID)

Example:
WDXyPanelCssStyleEnable(5,2)

Enables the CSS Style [814] with ID 2 in the Item Properties of the XY Panel [896] with ID 5.

WDXyPanelFix

WDXyPanelFix(ID)

Example:
WDXyPanelFix(5)

This activates the option "Fix" in the Item Properties of the XY Panel [896] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WDXyPanelGetLocationLeft

WDXyPanelGetLocationLeft(ID)

Example:
WDXyPanelGetLocationLeft(5)

This refers to the XY Panel [896] with ID 5 and returns its current horizontal location on the Page. The value expresses the distance in pixels from the left border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDXyPanelGetLocationLeft(1)`

WDXyPanelGetLocationTop

WDXyPanelGetLocationTop(ID)

Example:
WDXyPanelGetLocationTop(5)

This refers to the XY Panel [896] with ID 5 and returns its current vertical location on the Page. The value expresses the distance in pixels from the top border. The top left corner of the Page has the coordinates (0,0). You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDXyPanelGetLocationTop(1)`

WDXyPanelGetSizeHeight

WDXyPanelGetSizeHeight(ID)

Example:
WDXyPanelGetSizeHeight(5)

This refers to the XY Panel [896] with ID 5 and returns its current height (i.e. vertical size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDXyPanelGetSizeHeight(1)`

WDXyPanelGetSizeWidth

WDXyPanelGetSizeWidth(ID)

Example:
WDXyPanelGetSizeWidth(5)

This refers to the XY Panel [896] with ID 5 and returns its current width (i.e. horizontal size) in pixels. You can assign the value directly to another Widget or first to a Variable [1900], e.g. `var0 = WDXyPanelGetSizeWidth(1)`

WDXyPanelLocation

WDXyPanelLocation(ID,X,Y)

Example:
WDXyPanelLocation(5,100,200)

Sets the position of the XY Panel [896] with ID 5 to 100px horizontally and 200px vertically. 0,0 is the top left corner of the Page.

WDXyPanelLocationLeft

WDXyPanelLocationLeft(ID,X)

Example:
WDXyPanelLocationLeft(5,100)

Sets the position of the XY Panel [896] with ID 5 to 100px horizontally but remains the current vertical position. 0,0 is the top left corner of the Page.

WDXyPanelLocationTop

WDXyPanelLocationTop(ID,Y)

Example:
WDXyPanelLocationTop(5,200)

Sets the position of the XY Panel [896] with ID 5 to 200px vertically but remains the current horizontal position. 0,0 is the top left corner of the Page.

WDXYPanelResetAngleRelative

WDXYPanelResetAngleRelative(PanelID)

Example:
WDXYPanelResetAngleRelative(1)

Resets the value for the internal Relative X Position inside XY Panel with the ID 1 back to its default value.

WDXYPanelResetDefault

WDXYPanelResetDefault(PanelID)

Example:
WDXYPanelResetDefault(1)

Resets all current values inside XY Panel with the ID 1 back to the default values entered in its Item Properties.

WDXYPanelResetDistanceRelative

WDXYPanelResetDistanceRelative(PanelID)

Example:
WDXYPanelResetDistanceRelative(1)

Resets the value for the internal Relative Distance inside XY Panel with the ID 1 back to its default value.

WDXYPanelResetXPosRelative

WDXYPanelResetXPosRelative(PanelID)

Example:
WDXYPanelResetXPosRelative(1)

Resets the value for the internal Relative X Position inside XY Panel with the ID 1 back to its default value.

WDXYPanelResetYPosRelative

WDXYPanelResetYPosRelative(PanelID)

Example:
WDXYPanelResetYPosRelative(1)

Resets the value for the internal Relative Y Position inside XY Panel with the ID 1 back to its default value.

WDXYPanelSetAngleRelative

WDXYPanelSetAngleRelative(PanelID,Value)

Example:
WDXYPanelSetAngleRelative(1,18000)

Sets the value for the internal Relative Angle inside XY Panel with the ID 1 to 18000.

WDXYPanelSetDistanceRelative

WDXYPanelSetDistanceRelative(PanelID,Value)

Example:
WDXYPanelSetDistanceRelative(1,18000)

Sets the value for the internal Relative Distance inside XY Panel with the ID 1 to 18000.

WDXYPanelSetXDefault

WDXYPanelSetXDefault(PanelID,Value)

Example:
WDXYPanelSetXDefault(1,32000)

Sets the value for the default of Mouse X inside XY Panel with the ID 1 to 32000.

WDXYPanelSetXPosRelative

WDXYPanelSetXPosRelative(PanelID,Value)

Example:
WDXYPanelSetXPosRelative(1,22000)

Sets the value for the internal Relative X Position inside XY Panel with the ID 1 to 22000.

WDXYPanelSetYDefault

WDXYPanelSetYDefault(PanelID,Value)

Example:
WDXYPanelSetYDefault(1,34000)

Sets the value for the default of Mouse Y inside XY Panel with the ID 1 to 34000.

WDXYPanelSetYPosRelative

WDXYPanelSetYPosRelative(PanelID,Value)

Example:
WDXYPanelSetYPosRelative(1,22000)

Sets the value for the internal Relative Y Position inside XY Panel with the ID 1 to 22000.

WDXyPanelSize

WDXyPanelSize(ID,Width,Height)

Example:
WDXyPanelSize(5,100,40)

Sets the size of the XY Panel [896] with ID 5 to a width of 100px and a height of 40px.

WDXyPanelSizeHeight

WDXyPanelSizeHeight(ID,Height)

Example:
WDXyPanelSizeHeight(5,40)

Sets the size of the XY Panel [896] with ID 5 to a height of 40px but remains the current width.

WDXyPanelSizeWidth

WDXyPanelSizeWidth(ID,Width)

Example:
WDXyPanelSizeWidth(5,100)

Sets the size of the XY Panel [896] with ID 5 to a width of 100px but remains the current height.

WDXyPanelUnfix

WDXyPanelUnfix(ID)

Example:
WDXyPanelUnfix(5)

This deactivates the option "Fix" in the Item Properties of the XY Panel [896] with ID 5. "Fix" influences whether the Widget is only visible on the Page it was created on, or on all Pages.

WindowedModeByID

WindowedModeByID(SiteID)

Example:
WindowedModeByID(4)

Switches the site 4 to windowed mode in Pandoras Box project whether it is in fullscreen mode or in windowed mode already.
For addressing the machines combined to a Virtual Site [311], you need to use the Site IDs of the single machines as displayed in the Virtual Site Tab.
To toggle the site back into fullscren, use FullscreenByID [1544]

WindowedModeByIP

WindowedModeByIP(SiteIP)

Example:
WindowedModeByIP("2.0.0.5")

Switches the client with the IP 2.0.0.5 to windowed mode in Pandoras Box project whether it is in fullscreen mode or in windowed mode already.
For addressing the machines combined to a Virtual Site [311], you need to use the IPs of the single machines as displayed in the Virtual Site Tab or the command `WindowedModeByID` and the individual SiteID.
To toggle the site back into fullscreen, use FullscreenByIP [1544]

WindowFocus

WindowFocus(WindowTitleText)

Example:
WindowFocus("test - Paint")

Activates the application with the Window title "test - Paint" and sets it into the focus. The Window title is also shown in the Windows Task Manager. This command could be very useful if you e.g. want to send a Keystroke to an application window that is not focused right now (via the command KeyboardKeyPress(Keycode) [1547]).

Other examples:
WindowFocus("C:\Christie\content\myNotes.txt - Notepad++")
WindowFocus("PB Menu OSD")
WindowFocus("Christie Pandoras Box - Google Chrome")

WindowsDisableTaskManager

WindowsDisableTaskManager

Example:
WindowsDisableTaskManager

This blocks the Windows Task Manager from opening.

WindowsEnableTaskManager

WindowsEnableTaskManager

Example:
WindowsEnableTaskManager

This enables the Windows Task Manager it was blocked before from opening.

WriteEnvironmentVariablesToLog

WriteEnvironmentVariablesToLog

Example:
WriteEnvironmentVariablesToLog

Writes the current Windows environment variables into the Widget Designer's logfile "wd_gui_log.txt", to ease remote debugging for example.

The log file can be found in the installation path, e.g. C:\Program Files\Christie\Widget Designer 6.0 Rev 4498\logs

More information can be found under:
https://en.wikipedia.org/wiki/Environment_variable
http://www.7tutorials.com/simple-questions-what-are-environment-variables

X X

Y Y

Z Z

7.8.1.3 Programming Statements

If else [1886]
Switch Case [1888]
For Loop [1890]

Other advanced script techniques include:
Variables [1900]
Object and Member Notation [1904]
Math Expressions [1894]
Functions and Macros [1897]

For all statements it is strongly recommended to add new lines and also tabs for a clear and easy to read arrangement. Code blocks can be in/outdented using [Tab] or [Shift + Tab].

If you like to comment a line out without getting errors in the Debug Logger [812], use this syntax:

```
// one line that should not be processed

/* several lines
that should not
be processed */
```

Keep in mind, that you can test selected lines [1511] with the according command from the right-click menu available in all scripting fields.

7.8.1.3.1 If Elseif Else

As in other programming languages, the if-then-else statement in Widget Designer can be used to program conditional actions. This means, that you may program to perform a certain action if a condition is "true" and another action if it isn't. The result whether something is "true" is based on a comparison of different values, e.g. "is value A bigger than value B?".

- A value can be a number (double, integer) or text (string); it can be a static value e.g. "10" or it can be a variable [1900] (holding "10").
- The comparison "bigger than" would be expressed with a logical operator ">". Four operators are supported to evaluate whether two values are equal or unequal and in case of a numeric values, if one is bigger or smaller than the other one.
- An entire condition can not only consist of one comparison of two values. It can be extended with a second comparison (of two other values) using the logical "AND" or "OR", e.g. "is value A bigger than value B and at the same time C smaller than D?"
- If a condition is true, the True Script is executed. It starts and ends with curly brackets {} and contains an unlimited number of commands including functions and macros [1897].
- You may decide what happens if the condition is not true. If nothing should take place, the statement simply ends with the curly bracket behind the True Script. If something else should happen, it needs to be announced with the word "Else" followed by the {False Script}
- You can also add several possibilities with the "Elseif" statement, followed by the {Else True Script}
  If you have several "Elseif" statements, the switch case [1888] might be a better option.

Even encapsulated structures can be used, containing further if-else-statements, for-loops [1890] or switch cases [1888].

Please note that you can also use mathematical expressions [1894] as if values.

Basic syntax: If A = B

```
If value A Operator value B {
      True Script
}
Elseif value A Operator value B {
      True Script
}
Else {
      False Script
}
```

The "operator value" could be "=", all available operators are listed further down.

It does not matter whether the curly brackets are in the same or next line, you could even write:

```
If value A Operator value B {True Script}
Else {False Script}
```

Or even shorter without "Else":

```
If value A Operator value B {True Script 1}
If value C Operator value B {True Script 2}
```

But it is strongly recommended to add new lines and also tabs for a clear and easy to read arrangement. Code blocks can be in/out-dented using [Tab] or [Shift + Tab].

If you like to comment a line out without getting errors in the Debug Logger [812], use this syntax:

```
// one line that should not be processed
/* several lines
that should not
be processed */
```

Script Example

This example changes the value of Fader1. The first position is at 128. The second position depends on the variable [1900] Var1. If Var1 equals 100, the fader goes down after waiting 2 seconds. If the variable is not 100, the fader goes up after 2 seconds. No matter what the second position was, the third position at 64 is set after waiting one second.

```
WDFaderValue(1,128)
if Var1 = 100 {
      WDWait(2)
      WDFaderDown(1,1)
}

Else {
      WDWait(2)
      WDFaderUp(1,1)
}
WDWait(1)
WDFaderValue(1,64)
```

The available operators

"=" equals
">" greater than
"<" lower than
"!=" not equal
">=" greater than or equals
"<=" lower than or equals

It is not necessary to put white spaces between values and operators, but advisable for better readability.

If you are simply requesting a Boolean value as if-statement, e.g. `"if var_bool = true {...}"`, you can shorten this to `"if var_bool {...}"`. Using a statement without operator automatically checks if the statement is true.

Syntax for advanced conditions: If A = B AND / OR C = D

Logical AND & OR are also possible in order to combine two sets of conditions:

```
If value A Operator value B AND value C Operator value D {
      True Script
}
Else {
      False Script
}
```

Alternatively the syntax could look as following:

```
If value A Operator value B OR value C Operator value D {
True Script
 }
```

Script Example

This example changes the value of Fader1. The first position is at 128. The second position depends on the variables Var1 and Var2. Only if Var1 equals the word "Hello" and at the same time, Var2 equals the word "World", the fader goes down after waiting 2 seconds. If the variable is not 100, the fader goes up after 2 seconds. No matter what the second position was, the third position at 64 is set after waiting one second.

```
WDFaderValue,1,128
if Var1 = Hello AND Var2 = World {
      WDWait(2)
      WDFaderDown(1,1)
}

Else {
      WDWait(2)
      WDFaderUp(1,1)
}
WDWait(1)
WDFaderValue(1,64)
```

7.8.1.3.2 Switch

The switch statement (also known as multi way branches) can be used to program nested if statements more easily and with a better overview. Please refer to the previous chapter if-then-else statements 1886.

Basically, the statement starts with looking at a specific variable 1900, the so-called "Input Value". It needs to be declared beforehand. The following Cases (i.e command blocks) contain an individual value, the "Case Value". Each Case is evaluated, that is, it is compared whether the Input Value equals Case Value. If this condition is "true", the Case Script is executed.
The first Case could be "If the above variable equals value A then execute script A". All following Cases would be "If it equals my value then execute my script". The entire select-case statement searches for the Cases whose value equals the Input Value (the above variable) and executes the according scripts.

- A Case Value can be a variable itself.
- A Case Script starts and ends with curly brackets {} and contains an unlimited number of commands including functions and macros 1897.
- The number of Cases is unlimited.

- If there is no True Case (command block containing an equal value), an optional default script can be defined. This block starts with "CaseElse".

Even encapsulated structures can be used, containing further switch cases, for-loops [1890] or if-else-statements [1886].

Please note that you can also use mathematical expressions [1894] as Input or Case Values.

Basic syntax

```
Switch YourVariable {

 Case VariableValue1
      Value1 Script

 Case VariableValue2
      Value2 Script

 Case VariableValueN
      ValueN Script

 Case Else
      Default Script
 }
```

It does not matter whether the curly brackets are in the same or next line, but it is strongly recommended to add new lines and also tabs for a clear and easy to read arrangement.
Code blocks can be in/out-dented using [Tab] or [Shift + Tab].

If you like to comment a line out without getting errors in the Debug Logger [812], use this syntax:

```
// one line that should not be processed
/* several lines
that should not
be processed */
```

Script Example

This example is based on the string variable "VarName". It holds the forename of a customer who has logged in before. The idea is, that a Label displays a text with the country, the customer comes from and in addition the Widget Designer calls a page containing regional information and control elements.
Three Cases are programmed, one for "Daniel", one for "Rene" and one for "Rajesh".
Afterwards, a "Case Else" is defined as it is possible that unknown persons log in. In that case, he is routed to a Home Screen and the Label displays "Please register".

```
Switch VarName {

Case "Daniel"
      Label1.Text = "from Finland"
      WDPageGoto("Europe")

Case "Rene"
      Label1.Text = "from France"
      WDPageGoto("Europe")
```

```
Case "Rajesh"
      Label1.Text = "from India"
      WDPageGoto("Asia")

Case Else
      Label1.Text = "Please register..."
      WDPageGoto("Home")
}
```

Alternatively, the "Case Else" block can be eliminated from the statement if it is not needed. Or, the Page "Home" is called at the very beginning of the entire script, even before the main variable is selected. Different solutions are possible and depend on your project.

7.8.1.3.3 For Loop

If you want to repeat a certain script segment several times or based on a time that can change through a variable, then For Loop scripts are ideal to solve such a task efficiently.
Here is a quick example of how this looks like in a WD script:

Basic syntax

```
For i = Value1 to Value2 optional Step Value {
      Script
}
```

General Scripting Hints

It does not matter whether the curly brackets are in the same or next line, but it is recommended to add new lines and also tabs for a clear and easy to read arrangement.
Code blocks can be in/out-dented using [Tab] or [Shift + Tab].

If you like to comment a line out without getting errors in the Debug Logger [812], use this syntax:

```
// one line that should not be processed
/* several lines

that should not
be processed */
```

Next to the Debug Logger [812], the "Running Scripts [1511]" dialog (from the Scripting menu [811]) might be helpful. It shows which scripts are currently running and offers the option to cancel them via the right-click menu. That way, you can cancel a for loop e.g. if it takes too long. Further down, the "break" statement is explained.

Script Example

```
For i = 1 to 10 {
      Label1.Text = Now.ToString
      WDWait(1)
}
```

This script will execute 10 times the code between the curly brackets, i.e. display the current time in Label1 and then wait one second before it does that again.

A special local variable is defined in the beginning and increases automatically in every loop. The default name for the iteration variable is "i" and as you see below, you can use it in your script, e.g. as the text for a label or as the ID for a Fader.

You can use any valid name for the iteration variable, it doesn't necessarily have to be "i", but please do not use already defined variables as iteration variables, too.
If you want to run nested For Loops, you would have to use a different iteration variable for each loop.

As te iteration variable "i" it is an integer, you can also do math with it. In the following example we use the command `WDFaderValue(ID,Value)` to set the Faders 1-10 to a value which is ten times its own ID (so Fader with ID 1 is set to 10 etc.)

```
For i = 1 to 10 {
      Label1.Text = "Let's move Fader" + i
      WDFaderValue(i,i*10)
      WDWait(1)
}
```

You can also use variables instead of a defined value, which is especially interesting for "end" values. The next example shows a for loop that runs as many times as a list variable has elements.

```
var list = ["Apple","Bee","Coke"] //defines a local list variable with 3 elements =>
for loop will run 3 times
for i=1 to list.Count {
      WDLabelText(i,list[i-1])
}
```

The result of this script is, that the first Label displays "Apple", the second Label "Bee" and the third one "Coke". The "Count" member of the list variable returns how many elements the list has (here: 3). The iterating variable i starts with 1 continues to 2 and is lastly 3. So in the first cycle the Label with ID 1 is assigned with "list[1-1]" which is list[0] which is the first element of the list because indexing starts with 0.

In all examples so far, "i" started with 1 and is increased by 1 in each cycle until it reached the end value. It is increased by "1" because that is the default value for the step count. If you like to increment differently, simply define it via: `Step Value`
The following script has the cycles: 1,3,5,7,9.

```
For i = 1 to 10 Step 2 {
      Label1.Text = "Current i is: " + i
      WDWait(1)
}
```

Other examples which combine the for-loop with an if-statement [1886] and Project members can be found in the chapter "Project and Context Member [1906]".

Exiting the For Loop

With the statement "`break`" it is possible to exit a for loop before it finishes all its cycles, which is especially of interest the longer and complex your for loops get.
The usual case is to define a break condition with an if-statement. As soon as the "break" expression is called, everything following this expression is skipped and the script resumes after the brackets indicating the end of the for loop.

```
For i = 1 to 10 {
      if varLimit = 42 {
            break
      }
      Label1.Text=i
      WDWait(1)
}
Label1.Text = "end"
```

In this example, the for loop is only executed as long as the if-statement is not true. As soon as the variable "varLimit" equals 42, the for loop is canceled, i.e. no more cycles are executed. The script continues immediately with the line that follows the for loop which is in this example the one telling the Label to display "end".

In short: Everything following "break" **within** the for loop will not be executed!

You might think of another solution with a nested if but without a break statement. Based on the above example, the following could also be an option if you like to run the for loop only under the circumstance that varLimit does not equal 42 (the syntax for "equals not" is: !=).

```
For i = 1 to 10 {
      if varLimit != 42 {
            Label1.Text=i
            WDWait(1)
```

```
	}
}
Label1.Text = "end"
```

In difference to the first example, here, ALL cycles are initialized which means that the script checks varLimit 10 times. Depending on the true/false answer, the Label displays a new number or does not, but the for loop itself continues running and is not canceled as above. With a short example like this, there is not a big difference which work flow you attempt. However, the longer or complex the for loops get, the more it is of interest to use a solution with a better performance which means canceling for loops when they are not needed any more by using the break statement.

The following script is a good example for that. Imagine you have a ListView with 1000 rows showing a name in one cell and associated data in other cells. A Textbox allows to enter a name and with the following "Search" script you could highlight the row that contains that name and then quit the search process.

```
var SearchWord = TextBox1.Text
for i=1 to 1000 {
	if ListView1.GetCell(1,i) = SearchWord {
		ListView1.SelectedRow = i
		break
	}
}
```

The next topic explains the "exit [1892]" statement which provides the same functionality, but stops the complete script following the statement and **does not resume after the for loop**.

7.8.1.3.4 Exit

The "`exit`" statement is a tool that can be used in every script and in combination with every other programming statement, or even without.

It enables you to abort the currently running script in the moment where it runs into the "exit" statement. Especially useful is the combination with an if-statement [1886], to check variables or logical expressions for a certain condition and initiate an abort if necessary.

When the script runs into an "exit", everything after this line will not be executed.
If it is used for example in a macro [1897] or function, only the execution of the script context where it occurs is stopped. Other scripts from which this script might have been called are not influenced and will continue running.

Exiting a script can be very helpful with very long and complex scripts to avoid an execution of certain parts if a specific condition is met. Note that the "exit" statement is used to stop the current script. To cancel other scripts, or other instances from the current script, use the command `ScriptCancel` [1600].

Examples

Example 1:

For this example, please create first a global Boolean variable [1900] with the name "stopLoop". The expression "`if stopLoop`" is short for "`if stopLoop = true`".

```
stopLoop = false
for i = 0 to 1000 {
	if stopLoop {
		exit
	}
	Label1.Text = "Counting up: " + i
	WDWait(1)
}
Label1.Text = "Count completed"
```

This script sets the variable "stopLoop" to false at the beginning and enters a for-loop [1890], which counts up from zero to one thousand.

In every iteration, the current count value is written into Label1 and the script pauses for one second until the next iteration is started.

As long as the variable "stopLoop" remains in false state the cycle continues until Label1 would finally display "Count completed". To stop the loop before that, simply set the variable to true, e.g. with a CustomScript button and the script:

```
stopLoop = true
```

In contrary to the for-loop's "break [1891]" statement, which only exits the for-statements but resumes the script following it, the "exit" statement does not execute the last line of this example script and Label1 will show the last count.

Example 2:

Imagine a setup using several different sensor values (pressure, temperature, buttons,...) to trigger a series of events within a highly sophisticated script.
There might be a case that this series needs to be stopped if sensor number 1 and sensor number 4 show a temperature level above 42°C to prevent the setup from overheating
If this is the case, all you would need to do is add the following line at a few significant places in your script:

```
if (sensorValue1 > 42) AND (sensorValue4 > 42) {exit}
```

Whenever this state is detected, every script currently running into this condition is stopped immediately.

7.8.1.4 Mathematical Expressions

When working with variables [1900], you sometimes need to assign its value directly, but other times you might also want to recalculate your variable and use it for other mathematical calculations. You can achieve this with special commands, or with a very easy to understand mathematical syntax.

Obviously, the variable must be of the type integer or double to allow mathematical operations. Integers are rounded to the according whole number. However there is one exception to this rule. Adding letters is actually possible and equals appending letters.
Array variables are supported too. You call an index with square brackets: Variable[Index].

First, the below examples for a relative value assignment work in direct and in common commands. They offer five basic arithmetical operations: addition, subtraction, multiplication, division and exponentiation. The value itself can be a number you choose, another Variable, or a Member Value.

direct command	Result varRes	corresponding common command
Var VarName = Value (creates a LOCAL variable) Var varRes = 123	123	VCreate(VarName,Value) (creates a GLOBAL variable) VCreate("varRes",123)
Variable += Value varRes += 3 Variable = Variable + Value varRes = varRes +3	126	VAdd(VarName,Var1,Var2) VAdd("varRes",varRes,3)
Variable -= Value varRes -= 1 (or: varRes += -1) Variable = Variable - Value varRes = varRes -1	125	VSubtract(VarName,Var1,Var2) VSubtract("varRes",varRes,1)
Variable *= Value varRes *= 2 Variable = Variable * Value varRes = varRes *2	250	VMultiply(VarName,Var1,Var2) VMultiply("varRes",varRes,2)
Variable /= Value varRes /= 5 Variable = Variable / Value varRes = varRes /5	50	VDivide(VarName,Var1,Var2) VDivide("varRes",varRes,5)
Variable = Variable ^ Value varRes = varRes ^4	6.250.000	VPow(VarName,Var1,Var2) VPow("varRes",varRes,4)
Variable = Variable ^ 0.5 varRes = varRes ^0.5 varRes = varRes ^(2/2)	11.0905...	VSqrt(VarName,Var1) VSqrt("varRes",varRes)
Var VarName = Value (creates a LOCAL variable) Var varRes = "he"	he	VCreate(VarName,Value) (creates a GLOBAL variable) VCreate("varText","he")
Variable += Value varRes += "llo"	hello	VAdd(VarName,Var1,Var2) VAdd("varText",varText,"llo")

For more complex operations, you can use mathematical syntax as you would write it, see some examples below, for var1 = 5 and var2 = 11. Remember to assign a suiting type to the variable for the result, e.g. Integer.

Syntax	Result
varRes = calculation	
varRes = var1 + var2	16
varRes = var1 - 10	5
varRes = var1 * var2 * 2	110
varRes = 45 / var1	9
varRes = var1 ^4	625

For more sophisticated mathematical functions, like sinus, logarithms or modulo, please also refer to the chapter Math Object [1924]. This object provides several members for further mathematical operations.

Conditions

If you use those conditions outside of an if-statement, it will return a Boolean value.
Normal condition statements like "is bigger as?" or "equals?" can be combined with AND or OR statements.
Please note that those two expressions are not case sensitive.

Description and Syntax	Result
Checks if value 1 is bigger than value 2 varRes = (3 > 4)	False
Checks if value 1 is bigger than or equals value 2 varRes = (3 >= 3)	True
Checks if value 1 is smaller than value 2 varRes = (3 < 4)	True
Checks if value 1 is smaller than or equals value 2 varRes = (4 <= 4)	True
Checks if value 1 equals value 2 varRes = (3 = 4)	False
Checks if value one does not equal value 2 varRes = (3 != 4)	True
Checks if ALL conditions are true varRes = ((3 < 4) AND (3*2 = 6)) varRes = ((3 < 4) AND (3*5 = 7)) varRes = ((3 > 4) AND (3*5 = 7))	True False False
Checks if AT LEAST ONE condition is true varRes = ((3 < 4) OR (3*5 = 15)) varRes = ((3 < 4) OR (3*5 = 7)) varRes = ((3 > 4) OR (3*5 = 7))	True True False

7.8.1.5 Scripting Cheat Sheet V6

The Scripting Cheat is primarily meant for all programmers that are familiar with the Widget Designer scripting language.

Local variables
Using the "var" keyword variables can be defined that only exist within the scope (e.g. script or block) where they were defined. The variable's type is automatically determined by evaluating the assigned default value.

```
var x = "This is a string."

var y = 23
```

Literals
Anything that declares a literal expression (i.e. is not a number, command, function, variable…) must be enclosed in double quotation marks.

```
var myText = "Hello, world!"

var myNumber = 1.23
```

Parameters
Command parameters must be enclosed in round brackets. Advantage: parameters can be nested, i.e. a parameter can contain a function that has its own set of parameters.

```
WDLabelText(1, "Hello, again!")

WDLabelText(1,
Round(Fader1.Value, 2))
```

Object Notation
Commands that apply to a widget can also be called using "object notation", i.e. by specifying the object and method.

Not just widgets, but practically all items have methods or properties (depending on their type) that can be accessed using the object notation.

```
WDFaderGoDown(1, 10)
Fader1.GoDown(10)

label1.Text="abCDEfgh"
var z = Label1.text
var x = z.SubString(2,3)

DebugMessage(x, x.Length)
```

Expressions
Mathematical expressions (formerly requiring the "Math" keyword) are now accessible as global functions.

```
DebugMessage(Acos(0.5))
```

Conditions
Conditions can be combined (using "and"/"or"), nested and are not "space sensitive" anymore.

```
If x = 1 and (a="A" or b="B") {
  DebugMessage("Ok")
  }
```

If, ElseIf, Else
All code blocks are now enclosed in curly brackets. These can be at the end of a line or the beginning of the next one.

"If" can now be followed by any number of "ElseIf" statements and one "Else".

```
If x=1 {
  DebugMessage("x is 1") }
ElseIf x=2 {
  DebugMessage("x is 2") }
Else {
  DebugMessage("x is " + x)
}
```

Select is now Switch
The former "Select" command was renamed to "Switch".

```
Switch x {
  Case 1
    DebugMessage("x is 1")
  Case 2
    DebugMessage("x is 2")
  Case Else
    DebugMessage("x is " + x)
}
```

For Loops
Loops implemented with "For" now define the loop-variable and can optionally contain a "Step" parameter.

```
For i = 10 To 2 Step -1 {
  DebugMessage(i)
}
```

7.8.2 Functions and Macros

The Function and Macro Editor lets you store a series of commands as a script. It can include variable parameters that can be changed easily and any time when calling the script to be executed. In other words, a script is a sub routine, with or without input variables, that allows dynamic programming. The difference between functions and macros is explained below.

Using scripts in form of functions or macros is helpful when the same series of commands should be called from different places especially when the same command structure should be executed but its effect should be adjusted flexibly. Those scripts save time entering and editing commands.

Setting up a Function or Macro

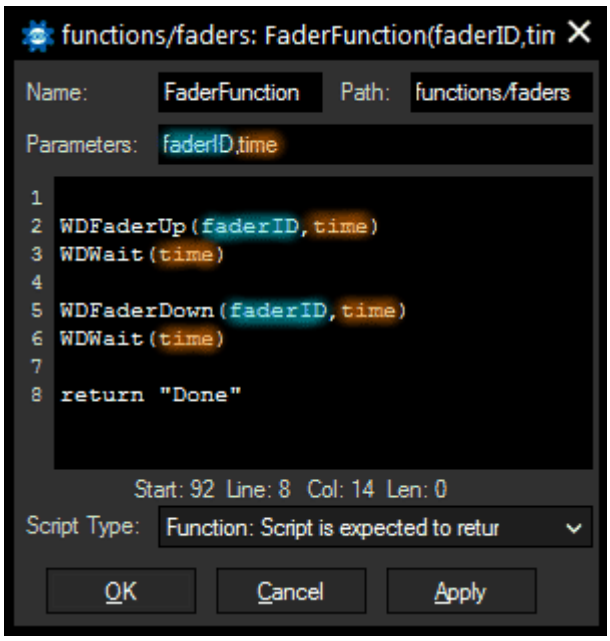
To open the editor go to "Scripting" and "Scripts (Functions & Macros)". An exemplary function is depicted below. Enter a function or macro name and optional arguments (the colored placeholders) if needed, and then the scripting text itself referring to these arguments.
Click "Apply" to save the script or changes you made in it.

If you have many macros and functions in your project, it might come in handy to have them sorted in folders. The "**Path**" field enables you to enter a path that creates folders within the Scripting menu and files your script there. In the example below, the function can be found in the folder "functions" in the sub folder "faders".

In the scripting field you may enter commands to be executed, just type in the command directly in the text field. The topic Script Language [1511] explains this in more detail.
See here a list of all commands [1511].
You can also use other functions and macros, Variables [1900] and other programming statements [1886] e.g. an If-clause [1886]. It is not recommended to call the same function or macro recursively as you might create a deadlock!

A function can end with an optional return statement, a line that says "return" and "something". You can return all kind of values, e.g. variables, strings or Boolean values. This value can be used from outside the function for further usage or maybe simply as a "function completed" message. Macros do not have return expressions.

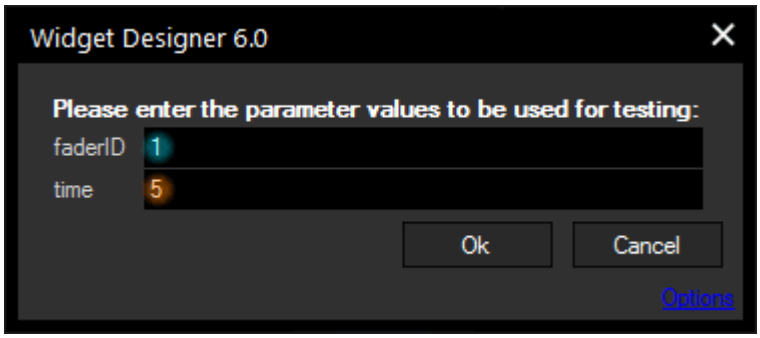
If you decide you would rather use a macro than a function, you can change the type anytime with the drop-down below the scripting section.



Testing a Function or Macro

To test a script, you can right-click in the script field and press "Test". A new window will open where you can enter values to substitute your optional arguments. When you use this way of testing, you don't have a display for your

function return value. You could for example enter a "DebugMessage" command right before your return statement to check if your return value was computed correctly.



Calling a Macro

You can call a macro within scripts, functions or other macros and you have two possible ways to do so. The first option is to call it by its name and values for your arguments in round brackets, like any normal Widget Designer command.

e.g.: `FaderMacro(1,5)`

Alternatively, the command "WDMacro" can be used. The optional macro arguments are added as arguments to WDMacro:

```
WDMacro("FaderMacro",1,5)
```

Macros are always called **asynchronously**, i.e. they run simultaneously and do not "block" other scripts or macros. The following example will execute the macro "FaderMacro" three times at once, which means that all three Faders fade up and down together within 4 seconds:

```
FaderMacro(1,2)
FaderMacro(2,2)
FaderMacro(3,2)
```

Calling a Function

Functions are called similar to macros. If you want to make use of the return value, you need an additional object receiving this value:

e.g.: `Label1.Text = FaderFunction(1,5)`

The fading script is being executed and the text of Label1 is set to "Done".
The receiving object has to match the data type [1914] of the return value. In this case, we return a string value, so we could for example also set a string variable [1900] "varString", or other Widgets' properties like "InputBox1.Text", "Playlist2.Title" and "Fader3.Note" instead of "Label1.Text".

Equivalent to this syntax, the command "VGetFunctionResult [1633]" can be used to execute the function and write its return value in a predefined variable:

```
var result = ""
VGetFunctionResult("result","FaderFunction",1,5)
```

Calling the script with the function name only will also execute the fading script, if a return value was set, it is discarded:

```
FaderFunction(1,5)
```

Alternatively, the command "WDFunction" can be used. The optional function arguments are added as arguments to WDFunction:

```
WDFunction("FaderFunction",1,5)
```

It is also possible to integrate the return value in another command argument. This script would have the same result as the example above:

```
WDLabelText(1,FaderFunction(1,5))
```

You can also write the return value into the DebugLogger [812] and add the system Variable "Now" which returns the date and time from the moment it is accessed, e.g. `DebugMessage(Now + " - " + FaderFunction(1,5))` results in the message: `2017-10-10 17:06:28.143 - Done`

In this example, we use the FaderFunction from above and generate an entry in the DebugLogger as log for a successful execution:

```
DebugMessage(Now + " - " + FaderFunction(1,2))
DebugMessage(Now + " - " + FaderFunction(2,2))
DebugMessage(Now + " - " + FaderFunction(3,2))
```

Functions in Widget Designer are always called in blocking mode, i.e. **synchronously**. This means that it is not possible to run several functions simultaneously within one script, they will always be executed one after another, i.e. after the previous functions has finished.

The example above shows a synchronous call, it would fade Fader1 up and down first and write the log message, then Fader2, then Fader3. Each fade begins only after the previous one has finished. The complete script would be finished after 12 seconds (4 seconds per call), the DebugLogger would show something similar to this:

```
2017-10-10 17:05:52.266 - Done
2017-10-10 17:05:56.289 - Done
2017-10-10 17:06:00.308 - Done
```

Important Note

Scripts in WD are processed from top to bottom in the order you wrote them in. Some methods in WD are synchronous, others asynchronous. Examples are the command "WDFaderUp" which can fade several faders simultaneously, whereas the command "WDWait" blocks the script until the wait time is elapsed. When you write a script, you can decide how it is supposed to be processed.

A macro that is supposed to run in blocking mode can either be used inside of a function, or you can simply write the macro script as a function without return statement.
A function can never be called asynchronously. If you do not need a return statement but want to have your functions processed simultaneously, use a macro instead.

Additional Example for a Function

This example shows how a function can be used to compare two numbers and return the higher number so that it can be used for displaying it in a label and fading an according fader.

Function Name	`max(num1,num2)`

Function Script	`If num1 > num2 {`
	`var result = num1`
	`}`
	`Else {`
	`var result = num2`
	`}`
	`return result`

Calling Script	`Label6.Text = max(5,3)` `WDFaderUp(max(5,3),2)`

7.8.3 Variables

What is a variable?

Variables are a powerful feature in Widget Designer. Any value can be stored within a variable and be called from another widget or node or command. Variables can hold one (or more) variable value(s) that update automatically, or they can hold one (or more) constant value(s).

Variables are subdivided into these types: String, Double, Integer, Boolean, Date, List, Color and JSON.

String	Ex. 1:	'hello'
= ASCII text in 'single' or "double" quotation marks (do not use accents like ` or ´)	Ex. 2:	'C:\folder1\subfolder2'
If you want to use quotation marks as part of your string, use the other kind for declaring the string.	Ex. 3:	'Hello, "World"!'
Double	Ex. 1:	-5.567
= number with decimal places	Ex. 2:	12.123456789
(a floating-point number)	Ex. 3:	99999.9
Integer	Ex. 1:	-50
= number without decimal places	Ex. 2:	0
	Ex. 3:	51347
Boolean	Ex. 1:	True
= Boolean value "True" or "False" (the letters are case insensitive)	Ex. 2:	False
Date	Ex. 1:	11/30/2016 09:30:38
= a special data type containing time and date information		
List	Ex. 1:	["Sample", "John", 31, false] (two strings, an integer, a Boolean)
= formerly known as arrays; holds several objects of the same or different data type even other variables and other lists can be part of a list	Ex. 2:	[3, "abc", v_double, ["i", "j", "k"]]
Color	Ex. 1	#808080 (dark gray)
= an object containing 8bit information (value ranges from 0 to 255) about the red, green, blue and alpha channel of a color	Ex. 2:	#8000FF (intense violet)
The RGB value in the variable list is displayed as hexadecimal code, this page might come in useful when translating colors to and from hex code: https://www.w3schools.com/colors/colors_hexadecimal.asp	Ex. 3:	#FFFFFF (white)
JSON	Ex. 1	{"name": "Tom", "age": 29}
= a format for easily storing, editing and exchanging sets of data.	Ex. 2	{"string":"Hello!", "bool": true, "pi": 3.14}
For more information concerning syntax and usage, please refer to the chapter Using JSON 1926		{"arr": [12,34,56], "map": {"a":1.2, "b": 2.3, "c": 3.4}}

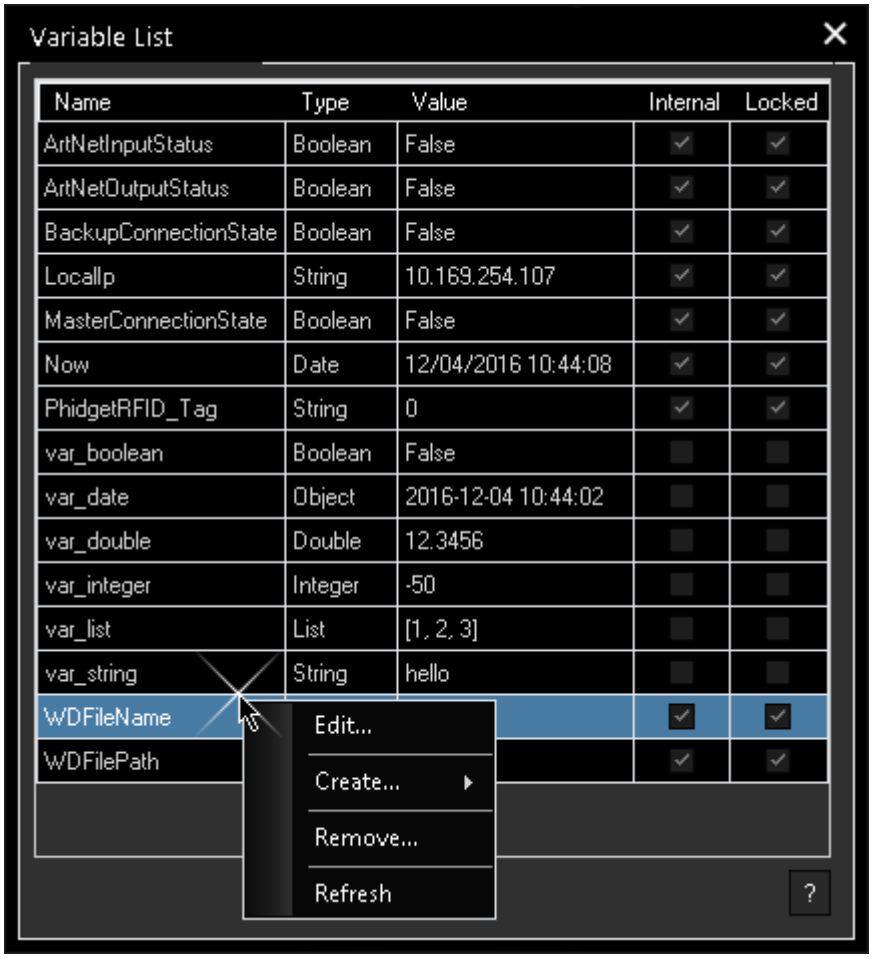
Variables can be defined either globally or locally:
Global variables can be called and edited within the complete project, they are saved in the Variable List.
Local variables are initialized inside a script and will be deleted as soon as this script is finished, their values are also only accessible from inside this single script.

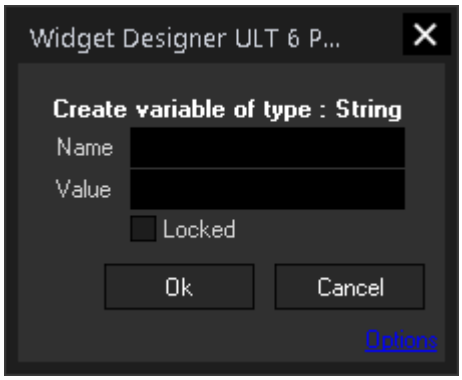
How to create a global variable?

Widget Designer includes a (below depicted) tool called "Variable List". Go to the Scripting menu and choose "Variables" to open it. The list shows all existing global variables, their type and value.
The free version does not include the Variable List though it supports variables.

If you open the Variable List the first time, you will see that there are already some variables. These system variables are called internal variables and are created per default, update automatically and can not be deleted nor edited. The system Variable "Now" for example returns time and date at the moment it is accessed and can be useful for debug messages, protocols or timed events.

The additional variables starting with "var_" were declared by the user as follows:





First of all, a variable needs to be declared. The first way to do this is via the right-click menu within the Variable List. Simply decide what type, name and (starting) value it should have and click "Ok".
Please note that the name may consist of letters (lowercase or capital) and numbers, as well as underscore "_", the first character must be a letter. A valid variable name would be e.g. "Var_String123".

For most variable types you simply pick the value you like to assign and click "Ok". With the **Locked** option the variable stays constant, you cannot edit the value until it is unlocked.
A new Boolean variable is initiated with the value "False". Tick the check box "True / Active / Yes" to turn it to "True".
For a list variable enter the elements and separate them with the symbol you choose from the drop-down, e.g. "1,2,3". The square brackets you see later, are added automatically. Building nested lists is only possible from within scripts, adding a list in a list is not possible by directly editing a global variable.

The second way to declare a variable is via the command `VCreate` [1623], which is described in the Command List [1520]. Commands are also interesting in the further workflow e.g. calling and editing a variable, particularly a list variable.
The best practice is, to declare global variables first and use them then in separate scripts.

Every Variable type has its own group of available members. To learn more about object members, please refer to the chapter Object and Member Notation [1904].

A complete list of all members related to special types can be found in the chapter Data Type Specific Member [1914].

How to assign another value to a global variable?

After having declared a variable it can adopt a different value at any time. Obviously the new value must be of the same type. For assigning a new value, there are three ways: use the right-click menu in the Variable List, a node or commands.

The Variable Output Node can be found under Nodes > Output > Generic and can be linked to a filter or input node in order to use their value as the new variable value. The Nodes chapter [936] explains the usage of nodes. The output node updates the variable constantly with a new value. An example is depicted below.

There are very fast commands like `Variable = Value` to assign a new (member) value. This is also called a direct command, in difference to common commands, they are more direct, fast and flexible. Please see the below table for possibilities to assign new values to variables. Commands are listed in the command list [1620], and more information about the way to write commands is described in the chapter "Script Language" [1511].

direct command	corresponding common command
`Variable = Value`	`VValue,VarName,Value`
`varNumber = 123`	`VValue("varNumber",123)`
`varString = "Hello"`	`VValue("varString","Hello")`
`ListVariable[Index] = Value`	`VSetArrayValue,VarName,Index,Value`
`varList[5] = 123`	`VSetArrayValue("varList",5,123)`
`Variable = Variable`	`VValue,VarName,Variable`
`varNumber1 = varNumber2`	`VValue("varNumber",varNumber2)`
`varString1 = varString2`	`VValue("varString",varString2)`
`varList[5] = varNumber`	`VSetArrayValue("varList",5,varNumber)`
`varList[varNumber] = 5`	`VSetArrayValue("varList",varNumber,5)`
`Variable = MemberID.MemberValue`	`VValue,VarName,MemberID.MemberValue`
`varNumber = fader2.value`	`VValue("varNumber",fader2.value)`
`varString = label1.text`	`VValue("varString",label1.text)`
`varList[varNumber] = fader2.value`	`VSetArrayValue("varList",varNumber,fader2.value)`
`varList[fader2.value] = varNumber`	`VSetArrayValue("varList",fader2.value,varNumber)`
`varList[label1.text] = varNumber`	`VSetArrayValue("varList",label1.text,varNumber)`
`(if label holds numeric value!)`	

As you see above you can assign values in many ways. In most cases, you can only assign a correct variable type, e.g. a string to a string variable. If you like to assign another type, you need to convert it first, e.g. like this `varInteger = varBoolean.ToInteger` The chapter Object and Member Notation (dot syntax) [1904] explains this syntax in more detail and gives more examples.

In case you work with list variables, you might wonder how to write or read them if they get more complex. As an example the variable varList holds different value types, first the integer 3, then the string "abc" and a (so called nested) list with the strings "i" and "j". So the list should look like this: [3,"abc",["i","j"]]

```
VCreate("varList",[3,"abc",["i","j"]])
```

Elements of the list can be accessed via an index in square brackets.
varList[0] -> returns 3
varList[1] -> returns "abc"
varList[2] -> returns ["i", "j"]
varList[2][1] -> returns "j"

There are also commands like `Variable += 1`, that adds 1 to the current value of the variable. `Variable = (1+2*3)` even does a complex mathematical calculation for you. More of these advanced variable assignments can be found in the chapter Math Expression [1894].

There are many ways to draw a value from another control like a Label, ListView element, Fader etc.: Start typing the command "VGet" and you will see the full list of possibilities. Alternatives are described in the topic Object and Member Notation (dot syntax) [1904].

If you like to lock a variable, use the command `VLock(Name)`. The variable can not be changed until it is unlocked. In the Variable List there is a check box called **Lock**.
Fading a variable is possible with these commands `VFadeTo(VarName,Value,Time)` and `VFade(VarName,StartValue,EndValue,Time)`.

How to use variables?

Once a variable is declared it can be used by its alias name. There are several ways to do so; one is creating nodes.
The Variable Input node can be found under Nodes > Input > Generic and can be linked to a filter or output node in order to use the variable's value there. The Variable Output node can be found under Nodes > Output > Generic and can be linked to a filter or input node in order to use their value as the new variable value. In the depicted example the Fader Input node writes the fader's value into a variable. Then the Variable Input node writes the value into a label.

Some controls, e.g. a label, allows accessing a variable directly. In the example we could delete the Variable Input Node (ID 1) and the Label Output Node (ID 3) and instead tick the check box "Variable Source" in the label's Item Properties.

The last way is using a variable in commands. There are several commands that allow writing a value into a variable, in the command list they start with VGet... [1636]. As well you may use commands to assign a specific value or result of a math operation to a variable.
To use a variable's value instead of an argument within a command simply use the variable name. For example, `TCPSend(1,var_String)` will send "hello" via the TCP Connection if you have declared this variable before.
The chapter Object and Member Notation (dot syntax) [1904] gives more examples of commands with variables.

How to create and use a local variable?

You can define a local variable inside of a script by using the indicator "var " (var and space) before a variable name. The variable name must be new and valid, i.e. as for global variables, it may consist of letters only (lowercase or capital) and numbers, as well as underscore "_" and the first character must be a letter. A valid variable name would be e.g. "Var_String123".
A value can, but does not have to be assigned in the same step.

The type of the local variable will be determined automatically by its first value as seen in the following examples:

Create a local string variable and set Label1 to "example" and Label2 to "another example":

```
var x = "an example"
Label1.text = x
var y = "another example"
WDLabelText(2,y)
```

Create a local double variable and set the Fader1 to 50 and Fader2 to 150:

```
var z = 50
Fader1.Value = z
WDFaderValue(2,z+100)
```

Create a local list variable and fill a DropDown List with the content, then set Label1 to "Berlin" and Fader1 to value 3:

```
var x = ["Berlin","Cherry",3]
DropDownList1.SetItemsFromArray("x")
Label1.Text = x[0]
Fader1.Value=x[2]
```

You can also use "data type specific members [1914]" to specify the type of the variable. In this example "x Date" has a date type whilst "xString" is a string. With the "Type" member you can verify the type, e.g. if you are debugging a script that is not working:

```
var xDate = Now
var xString = Now.ToString
DebugMessage(xString.Type) //returns "String"
```

For more examples of use and practical hints concerning everything around the Widget Designer's scripting language, please also have a look at the Scripting Guide [1511].
Note that the two above examples only work with local variables. When working with global variables, please create them first (e.g. via the command `VCreate`) and use their values in separate scripts. The reason is, that scripts are compiled before they are executed and global variables cannot be declared and used in one compilation. This is when local variables should be used.

What is the difference between a variable and a member value ?

Member values, e.g. "Fader.Value", could be explained as internal variables. Widget Designer declares them automatically with each item you add to your project. Their name and value relate to the item they belong to. Otherwise you can use member values and variables in the Script language in the same way. See the next chapter Object and Member Notation (dot syntax) [1904] for more details.

7.8.4 Object and Member Notation (dot syntax)

What are objects and members?

The term **"object"** in this context refers to windows [803], pages [805], widgets [818], nodes [936] and (global and local) variables [1900].
From version 6.1.1 on, external devices connected to WD via the Configuration dialog [1305] such as Christie Terra [1332], Lightware [1378] matrices, Phidgets [1383] etc. can be addressed as objects. Their members are described in the respective chapters from the Configuration dialog [1305].

Each object has a unique name, e.g. CustomScript3, Node14 or var_String, by which it can be accessed in scripts [1511]. You can change the automatically generated name of every object in the object's properties dialog. Variable names can not be edited once they are declared, however, you can replace them.

The term **"properties"** refers to specific values of those objects that can be requested or set. The term **"methods"** describes actions that can be performed by this object.
Properties and Methods are so called **"members"** of the object. Each object, e.g. Fader, CustomScript, string variable, etc. has its specific members. A list of all possible members of an object is visualized in the Script Assistant [1511] when placing a dot "." after an object name, this is why it is also called the "dot syntax".

Example:
"Fader1.Value" returns the current value of Fader1 as a double value (Property)
"Fader1.FadeUp(2)" fades Fader1 to 100% within 2 seconds (Method)

The Script Assistant also gives information whether the member is a property or a method, and which data type [1900] the property has (i.e. whether it is a string, Boolean, integer, ...), or which arguments the method expects.

Every command in Widget Designer that handles a window, page, widget or node has an equivalent method.

There are two special objects besides the already mentioned, the are called "Project" and "Context". Those provide ancillary members, for example for session related values. Please refer to the chapter "Project and Context Member" [1906] for further information.

Data type specific members are explained in a separate sub chapter [1914] in detail. Each data type (String, Integer, etc.) has own members that can be used for conversion, editing and information. The data type specific members can be used for variables as well as for object properties. Simply type in another dot after the variable's or object's name and the Script Assistant will list all available members. Example: varString and Label1.Text are both strings. When writing "varString." or "Label1.Text." the string members are displayed, including the member "contains". Hence you can ask whether varString or Label1 contain a specific expression.
Please note that these methods only return values, they do not change the object they refer to themselves.

How to use members?

Property members can be used exactly the same way as variables are being used. You can perform mathematical actions [1894] or use them for conditions [1895]. You can assign them to variables or other object properties. You can even use further methods respective to the type.

Some examples:

`var_Fader = Fader3.Value + 10`	This assigns the current value of Fader3 plus 10 to the double type variable var_Fader.
`Wheel1.Max = 314`	This assigns a max value of 314 to the widget Wheel1.
`Fader6.Value = var_bool.ToInteger`	This assigns either 0 or 1 to the widget Fader6, depending on the value of the Boolean type variable var_bool. The additional method "ToInteger" is necessary to convert the Boolean terms "True" and "False" into the values "0" and "1".
`var_int = Fader6.Value.ToInteger`	This assigns the value of Fader6 to the variable var_int, rounded to an integer value.
`if CustomScript5.Fix = true {Label1.Text = "Button fixed"}`	Only in case the button CustomScript5 is fixed, Label 1 will say "Button fixed".
`If CustomScript2.Notes.Contains("A1") {CustomScript2.Click}`	Only in case the Notes field in CustomScript2 contains the string "A1", the button is clicked. The next chapter shows how to combine this example with a for loop to address many widgets.

Method members can be used like the normal WD-commands, if they require parameters, the Script Assistant points them out. As the object is already defined by the object name in front of the method, you do not have to enter an ID or name as a parameter. The parameters, too, can be substituted by variables or other object properties, even functions [1897] are possible if they return the correct value type.

Some examples:

`Fader1.FadeToValue(5,var_Fader)`	This fades Fader1 to the value of variable var_Fader within 5 seconds. This method equals the command `WDFadeToValue(1,5,var_Fader)`.
`Label5.SetFlashInterval(500)`	This sets the flash interval of Label5 to 500ms.
`CustomScript1.Click`	This clicks CustomScript1, no parameter is needed to perform the action.
`TextBox2.AddTextFromLabel(14)`	This adds Label14's text to TextBox2.

The next chapters include more information regarding object and member notation.
The chapter Project and Context Member [1906] describes two special, superior objects, Project and Context.
The chapter Data Type Specific Members [1914] explains how the members of each data type (String, Integer, etc.) can be used for conversion, editing and information.
The chapter Using JSON [1926] gives a quick overview of the JavaScript Object Notation, a format for easily storing, editing and exchanging data.

Remark for advanced users: Dynamic scripting with objects

Scripts are always compiled before they are executed. This procedure can cause problems if you are deleting and re-creating objects in a script.
The following example shows a situation causing an error and how to resolve this:

```
WDControlDelete("Playlist2")                                    -> deletes "Playlist2"
WDControlCopy("Playlist1",800,100,"Page1","Playlist2",2)        -> copies "Playlist1" and calls it "Playlist2"

Playlist2.Title = "test1"
                                                                -> renames the Title of "Playlist2"
```

This script will not work as expected, because the object "Playlist2" in the last line still refers to the instance that existed at the moment the script compiled. As this instance is deleted when the script actually is being executed, the action of changing the title is performed on the deleted object that is still preserved in the memory, and not on the new instance (with the same name "Playlist2") that was created with the WDControlCopy command.

In order to avoid this misbehavior, you can use the Project object [1906] for assigning the title. The Project object is a dynamic reference and refers to the instance existing at the moment in which the script is executed:

```
Project.Playlist("Playlist2").Title = "test2"
```

Another approach would be to source the last line out and write it in an external Macro [1897]:

```
WDControlDelete("Playlist2")
WDControlCopy("Playlist1",800,100,"Page1","Playlist2",2)

Macro1()
```

Macro1:

```
Playlist2.Title = "test3"
```

This way, the macro is being compiled after the Playlist object was already replaced and "Playlist2" thus refers to the new instance.

7.8.4.1 Project and Context Member

The previous topic describes members referring to objects like widgets or variables [1904]. Besides those, there are two special, superior objects, **Project** and **Context**. They provide information and functionality more abstracted than the normal objects.

Those two powerful tools enable you to program highly sophisticated and automated interfaces, they can turn scripting more complex on one hand, but much more flexible and effective on the other hand.

Project member

The Project is the main parent element of the whole Widget Designer. Using this object permits you dynamic access to all child objects like widgets, nodes and global variables.
The main purpose of using the Project object is automating actions that should be performed on many child elements at once. This makes it for example easy to substitute a row of widget IDs with iterating variables to access their members, or to write a script that searches for all widgets with a special type or value.

This is possible because you can use variables instead of explicitly naming an object. One way is to write:

```
Label1.StartFlash
Label2.StartFlash
etc.
Label10.StartFlash
```

For this example it would also be possible to use common commands and shorten the script by using a for-loop [1890]:

```
For i = 1 to 10 {
      WDLabelStartFlash(i)
}
```

But you have to keep in mind that the actions or possibilities with commands are limited, whilst member notation permits full access. E.g if you wanted to change the background color of Label 1 to 10 from black as the default value to red. The Label member "BackColor" offers this function, but there is no common command equivalent that can be used here for a for-loop. The Script also disables the transparency option for labels, allowing the background color to be visible.

This is where the Project member comes in handy: It allows you to have flexible access on every object member:

```
For i = 1 to 10 {
      Project.Label(i).BackColor.SetRGB(255,0,0)
      Project.Label(i).Transparent=False
}
```

It is also possible to combine this way of scripting with variables. The following script is an alternative for the last one:

```
for i = 1 to 10{
      var f = Project.Label(i)
      f.BackColor.SetRGB(255,0,0)
      f.Transparent=False
}
```

Lastly, it is possible to combine this with an if-statement as shown in the examples below the table.

The Project member already offers you all available widget types when you set the dot. The widget name (e.g. "Label1") or the ID are used to specify a certain object of which all members are available afterward.

Object	Members	Further Members	Description	Example

Project	.WidgetType
	.Variables.Type
			The Script Assistant offers you a list of all available widget types, nodes and variables that exist in your project.
			If you want to access variables, you also have to specify the data type.
				`Project.Fader`
				`Project.CustomScript`
				`Project.Variables.String`
				`Project.Variables.List`

	.WidgetType(ID or name)
	.Variables.Type(name)
			To access the members of a specific object, you have to enter either the ID (for widgets and nodes) or the name of the object.	`Project.Fader("Fader15")`
				`Project.CustomScript(3)`
				`Project.Variables.String("var_string")`
				`Project.Variables.List("var_list")`

		.WidgetMember
		.NodeMember
			Every object specific member is accessible from the Project object too. All properties can be read or set, all methods can be executed.
			If you access a property, you can even use the data type specific members on them.
				`Project.Fader("Fader15").Value = 128`
				`Project.CustomScript(3).ExecuteClick`
				`Project.Label("Label22").ForeColor.SetRGB(50,0,90)`
				`Project.Node(15).ConnectTarget(12)`

		.VariableDataTypeMember	All variables accessed by the Project object have their respective data type member.	`Project.Variables.Integer("var_int") = 10` `Project.Variables.Boolean("var_bool").ToInteger`

Please note that every member returning a value, like string, double, Boolean or list, can have further members respective to their data type. This is explained in this chapter 1914.

Example 1:
Imagine a WD project that is supposed to monitor position values of a theatrical rigging system. Everything is already set up correctly and the height of twenty different flown battens is displayed in twenty different labels, Label1 to Label20. The names of the respective battens are displayed next to height values with Label21 to Label40. The pairing would then look like this:
Batten 1: 20 [Label21] [Label1]
Batten 2: 15 [Label22] [Label2]
Batten 3: 3.5 [Label23] [Label3]
...

Maybe you want to check quickly which ones of them are still unused and thus at their maximum height, 20m. With one click, all of those batten name labels should flash and turn green.

You could then use a normal CustomScript button to trigger this action by entering the following script:

```
for i = 1 to 20 {
	if Project.Label(i).Text = 20 {
		Project.Label(i + 20).ForeColor.SetRGB(0,255,0)
		Project.Label(i + 20).StartFlash
	}
}
```

This script searches through all labels with ID 1 to 20 for a text value "20" and performs the respective action (set fore color and start flashing) on the label next to the ones where the condition applies (label ID + 20).

Of course this is a very small and simple example, but as the Widget Designer offers you a maximum GUI of 8K x 8K pixels, huge monitoring displays can be arranged, also with BarGraphs [852], Gauges [856] and other kinds of optical display options.
Finding something special there quickly can become quite difficult, this example provides a simple solution for this problem. The search algorithms can of course be a lot more complex and also include user input, searches can also be automatically timed.

Example 2:
Imagine a WD project made for a venue. The venue has several rooms with multiple displays and for each room there is a page in Widget Designer. Each page has widgets, e.g Faders that fade images and videos on those screens. Whenever you enter another room and switch to another page, you like to "reset" i.e. fade down the faders, but not all of them as some Faders influence sound which should stay on until the end of the day. To distinguish those two groups of faders, you could use for example the "Notes" field. Let's say, all faders that should be faded down when you switch a page say "room1", "room2" etc. in the Notes field whilst the Faders for the sound say "audio". With this script, you fade down all "room1" Faders:

```
for i=1 to 50 {
	if Project.Fader(i).Notes.Contains("room1"){
		Project.Fader(i).FadeDown(2)
	}
}
```

As the Notes field is available for many widgets, it is a very easy and versatile tool to group, address or identify them. Another common application is to "reset" toggle buttons, in the way that all toggle button that are currently in the "Pressed" mode should be clicked (but those in the "Released" mode stay). To achive this , you could replace or add the following command in the above if-statement: `WDCustomScriptForceReleased(i)`

If you name your pages "room1", "room2" etc. you could create a string Variable [1900] and use this command in a PageEnter script [805] to write the current page name into the variable, e.g. `v_currentpage = Window1.PageName`
Now, you can substitute the room string in the above example with the variable and use this script anywhere you like without the need to edit it: in a Custom Script Button (even if it is set to "Fix [822]") or Macro or even the PageLeave script.

```
for i=1 to 50 {
	if Project.Fader(i).Notes.Contains(v_currentpage){
		Project.Fader(i).FadeDown(2)
```

```
        }
}
```

As shown with all these examples, the Project object enables you to use a complete new dimension of dynamic and flexible scripting within the application.

Context members

The object "Context" always refers to the context of the script wherein it is run.
For example, if you execute a script by clicking a CustomScript button, the context includes the page and window where the widget is located. In addition, it holds the information with which client it was executed. Imagine your project is displayed in your GUI and at the same time via the Web Server in an external browser. It makes a difference whether you click the button in your GUI, or in the browser, they have two different contexts. If you add faders to the project, each client can have different values for the same fader, because the widget is used in different contexts.
If you are interested in the Web Server and the control of those client differing values, please have a look at the topic Group Values [1933].

However, as said above, the context does not only refer to multiple clients. It is also useful for addressing windows, pages and widgets automatically. Try the following:
Create a project and set up one CustomScript button in Page1. Create another window with Page2 and set up a label.
Open the property dialog of the button and type the following into the "On Press Script" section:
`DebugMessage(Context.Name)`
Click on the button to execute the script, the Debug Logger will show: Window1/Page1/CustomScript1/ClickScript. This is the whole context of the executed script, including all parent elements.
Repeat this with the label's On Click Script, and you will get this result: Window2/Page2/Label1/ClickScript.

This should give you a good impression of what a script's context is, it always matters from where it is executed.

Please be aware that **scripts must be executed "for real"**, i.e. for example, a button must be clicked. If you right-click in the script field and choose **"Test" or "Test Selected Lines", there is no context!** Always execute the script the way it will be by the user afterward.

Here is a list of all available Context members and what they are intended to do, it is recommended to take some time and try them all out. The Debug Logger is a useful tool for this purpose:

Object	Members	Further Members		Description
Context	.Name			Example: `Debug Message(Context.Name)` Returns the name of the widget, macro, etc. from which the script is executed, including the parent objects. A CustomScript button executing the example script opens the Debug Logger which shows: Window1/Page1/CustomScript1/ClickScript
	.Stack			Example: `Debug Message(Context.Stack)` Returns a list of all objects (different widgets or macros that call other objects/ macros/ functions) taking part at the chain of executing scripts. If a CustomScript button executes another Macro which includes the example command, the Debug Logger will open and show: [["Window1/Page1/CustomScript1/ClickScript","WDMacro Command"]]
	.Page	*.PageMember* e.g. Color1		Example: `Context.Page.Name = "new"`

				Accesses all available page members for the page the object is located on. A CustomScript button with the example script changes the name of that page it is located on itself to "new".

	.WidgetID			Example:
				`WDFaderUp(Context.WidgetId,1)`
				`DebugMessage(Context.WidgetId)`
				Returns the ID of the widget whereof the script is executed as an integer.
				If the example script is executed from the CustomScript button with ID 2, the Fader with ID 2 will fade and the Debug Logger will open and display:
				2

	.WidgetName			Example: `DebugMessage(Context.WidgetName)` Returns the name of the widget whereof the script is executed as a string. If the example script is executed from the CustomScript button with ID 2, the Debug Logger will open and display: CustomScript2
	.WidgetType			Example: `DebugMessage(Context.WidgetType)` Returns the type of the widget whereof the script is executed as a string. If the example script is executed from any CustomScript button, the Debug Logger will open and display: CustomScript
	.Script			Example: `DebugMessage(Context.Script)` Returns the type of script field from which it is executed as a string. The example script would open the Debug Logger and display "ClickScript", "EnterScript" or "LeaveScript" etc.

	.Client
	(Client Key - optional)
		.Groups	*.Group Name*
			e.g. Session
				Example:
				`DebugMessage(Context.Client.Groups.Client)`
				`DebugMessage(Context.Client.Groups.InternalExternal)`
				`DebugMessage(Context.Client.Groups.IpAddress)`
				`DebugMessage(Context.Client.Groups.Session)`
				This member refers to Group Values [1933], it offers a list of available groups to request or edit the respective key. The optional Client Key parameter can be used to address other clients in the system, apart from the particular client executing the script. The Script Assistant offers you available Client Keys.
				A CustomScript button executing the example commands will open the Debug Logger showing the according Client key, Internal or External information, the IP address (or localhost) and the Session Key from

				the computer / Session it is executed from. It could look as follows:
				c21e4700-11aa-22bb-33cc-44dd55ee66ff
				Internal
				localhost
				5e551040-f00f-e99e-d88d-c7b6a55a6b7c
				
				Example 2:
				`DebugMessage(Context.Client("c21e4700-11aa-22bb-33cc-44dd55ee66ff").Groups.InternalExternal)`
				A CustomScript button executing this command will open the Debug Logger showing the according Internal or External key from the computer with the defined Client Key.
				
				Example 3:
				`If Context.Client.Groups.Session = "5e551040-f00f-e99e-d88d-c7b6a55a6b7c" {WDMacro("Adminmacro")}`
				A CustomScript button executing this command will execute a macro only if called from the context of the defined Session, not if clicked in other Sessions.
				
				Example 4:
				`Context.Client.Groups.Team = "team_red"`
				This example shows how to assign a Key to a Custom Group called "Team".

		.Key		Example: `DebugMessage(Context.Client.Key)` Returns the Client Key of the indicated client context. A CustomScript button executing the example command will open the Debug Logger showing the according Client Key, e.g. "c21e4700-11aa-22bb-33cc-44dd55ee66ff".

		.IP		Example:
				`DebugMessage(Context.Client.Ip)`
				`DebugMessage(Context.Client("c21e4700-11aa-22bb-33cc-44dd55ee66ff").Ip)`
				This member returns the IP address of the indicated client context. With this, you can also retrieve the IP address of every connected client with the Client Key. E.g. the member "Context.ClientKeys" returns a list with the Keys of all connected clients.
				A CustomScript button executing the example script will open the Debug Logger which shows for example "localhost" (if it was executed from the local Widget GUI) or "192.168.1.100" for an external client, as well as "192.168.1.102", the IP address corresponding to the entered Client Key.

		.PageName		Example: `Context.Client.PageName = "Page2"` This expression can be used to navigate between different pages of the same window. When a Group Value [1933] is applied to the window, different clients can

				access pages independently from each other. Otherwise, the pages are synchronized. A CustomScript button executing the example script will send the client where it is executed to Page2.
	.ClientExists (Client Key)			Example: `DebugMessage(Context.ClientExists("c21e4700-11aa-22bb-33cc-44dd55ee66ff"))` Returns a Boolean value (False or True) that indicates if a client with the respective Client key is currently connected. A CustomScript button executing the example script will open the Debug Logger which shows "True" if the session with this Client Key is connected, or "False" if disconnected.
	.ClientKeys			Example: `DebugMessage(Context.ClientKeys)` Returns a list containing all Client Keys of currently connected clients. A CustomScript button executing the example script will open the Debug Logger showing the Client Keys from all open Sessions, for example: [["c21e4700-11aa-22bb-33cc-44dd55ee66ff","a381490e-9a00-4bd1-ab52-0d835705edb3"]]

	.Session	.Value(specifier)		Example:
				`Context.Session.Value("name") = "Jonathan"  // stores a new value "name" and sets it to "Jonathan"`
				`Context.Session.Value("country") = "Switzerland"  // stores a new value "country" and sets it to "Switzerland"`
				`DebugMessage(Context.Session.Value("name"))`
				`DebugMessage(Context.Session.Value("country"))`
				This command can can set or retrieve your own custom value, stored within the WD and responding to the Key in the Session cookie. Please note that you can only store **string values**!
				A CustomScript button executing the example script will open the Debug Logger showing "Jonathan" and "Switzerland", both strings retrieved from the Session where the script was called from.
				Please refer to the topic Session and Session Value [1913] for further information.

		.Contains (search expression)		Example: `DebugMessage(Context.Session.Contains("Jonathan"))` Returns a Boolean value (False or True) if the indicated expression is stored in the Session. A CustomScript button executing the example script will open the Debug Logger showing "true" if the Session value was set like in the example before.

	.Window	.CustomHeader		Example:
				`Context.Window.CustomHeader = "<!DOCTYPE html>`
				`<html>`

				<style>
				.WdWindow{
				position: relative;
				width: 1200px;
				margin: 0 auto;
				}
				</style>
				</html>"
				The Custom Header enables you to implement your own HTML5 based design in the Widget Designer interface. The here defined header affects the objects in the specified window.
				This example script causes the pages to be located not at the left side of your window but in the middle, assuming that the pages' width is 1200px.

		.PageName		Example: Context.Window.PageName = "Page2" This expression can be used to navigate between different pages of the same window. A CustomScript button executing the example script will send the window where it is executed to Page2.

Each member that returns a value can have additional members referring to their data type [1914]. The next chapter describes the terms "Session" and "Session Values" [1913].

7.8.4.2 Session and Session Value

The Session mentioned in the chapters Project and Context Member [1906] as well as Group Values [1933] enables you to distinguish between different instances accessing your Widget Designer project. While it originally belongs to the section "Group Values", it is not only available in the Unlimited Webclients version, you can even use it in the Free Version.

The Session Value here is a valuable tool for generating Session- or user-based data sets.

What is a Session?

A Session is generated every time a new browser connects to the Widget Designer project. As the main GUI is also displayed in an internal browser, this is always the first Session. The Free Version offers only one additional connection, the normal and Unlimited Version allow as many connected browsers as physically possible.

A new browser tab belongs to the same Session as the previous one, but if you open a second, different browser, like Chrome and Internet Explorer or Mozilla Firefox and Opera, a second Session is created. Each Session can be identified by its Key, a unique combination of letters and numbers. The Keys can be accessed from the Web Server menu, just go to Edit > Web Server Settings and open the tab "Sessions". There you will find a table containing all currently connected Sessions and the amount of Clients connected to this Session (note: two browser tabs of the same browser equal one Session and two Clients).

When a new Session is generated, a temporary cookie is stored with the browser and enables the Widget Designer to retrieve the stored information for several days, even if the browser was disconnected in between. As a part of this conjunction between WD and specific clients, data can be saved corresponding to each Session: the Session Value.

How can I use the Session Value?

A Session Value can be set with the Context object [1909]. It always consists of the specifier and a string value, other data formats are not allowed. Note that you can define an unlimited number of Session Values, you are not limited to only one data set per Session.
The value can be initialized, changed and retrieved with the expression:
`Context.Session.Value(specifier)`

Imagine a user log in for every person accessing the WD interface via a remote browser.

The user, let us call him "Jonathan" can enter his name with an InputBox [886] and submit this with a CustomScript button next to it:

```
Context.Session.Value("Username") = InputBox1.Text
```

A new Session Value called "Username" with the value "Jonathan" was generated and is accessible for this Session.

Now you want to use this information, e.g. log who accessed the interface and when. A ListView [890] might be an option here:

```
ListView1.SetCell(1,1,Context.Session.Value("Username"))
ListView1.SetCell(2,1,Now)
```

Of course you can have more sophisticated logic behind this application and check if the respective cell is empty before writing something in it. If it is not, you would proceed to the next one or start from the beginning if the list is already full.

The true power of the Session Value comes into view when you combine this tool with JSON expressions [1926]. You can create entire large sets of data for each Session, store this with the Session Value and have complex, user input defined scripts and actions according to that.

Please bear in mind that a script containing the Context object, especially in combination with the Session Value, has to be executed from a real script, e.g. a CustomScript click. The "Test" function will not show the correct behavior!

7.8.4.3 Data Type Specific Members

Each data type in Widget Designer, String, Integer, Double, Boolean, Date, List, Color and JSON, has its own, specific members. Those Members can be used for conversion, editing and information. Those of the members returning a result can even be used with further members.

Please note that these methods only return values, they do not change the object they refer to themselves. One exception here are some members of the color and JSON data types.
The data type of the return value is highlighted in the table below.

Example:

```
var x = "Hello World!"
var y = x.Contains("llo").ToInteger
DebugMessage(y)
```

The Debug Logger shows "1", as the Boolean value (true) returned by the Contains-member was additionally converted to an integer value (1). The variable "x" keeps its value "Hello World!".

Even object properties can be used like this:
"var x = Fader1.Value.Round(2)" assigns the value of Fader1, rounded to two digits, to the local variable "x".

.Contains(search expression)	Returns a **Boolean** value, indicating if the search expression is included at the object string	`Var x = "Hello World"`
		`Label1.Text = x.Contains("ello")`
		-> Label1 shows "True"
		-> Label1 shows "matriarchal"

Member	Description	Example
Data type: String		

.Contains(search expression)	Returns a **Boolean** value, indicating if the search expression is included in the object string	`Var x = "Hello World"`
		`Label1.Text = x.Contains("ello")`
		-> Label1 shows "True"

.EncodeBytes	Returns a **list** containing the UTF-8 byte value of each character as integer	`Var x = "Hello World"`
		`Var y = x.EncodeBytes`
		-> y = [72,101,108,108,111,32,87,111,114,108,100]

.EndWith(search expression)	Returns a **Boolean** value, indicating if the object string ends with the search expression	`Var x = "Hello World"`
		`Label1.Text = x.EndWith("orld")`
		-> Label1 shows "True"

.Format(string representation, ...)	Uses the C# method "String.Format" on the object string and returns the formatted **string**. For further information on String.Format, please visit https://msdn.microsoft.com/en-us/library/system.string.format(v=vs.110).aspx#Format_Custom or http://timtrott.co.uk/string-formatting-examples/	`var x = "I like {0} and {1}."`
		`Label1.Text = x.Format("trains","turtles")`
		-> Label1 shows "I like trains and turtles."

.IsEmpty	Returns a **Boolean** value, indicating if the object string contains a value or is empty.	`var x = ""`
		`var y = "Hello!"`
		`Label1.Text = x.IsEmpty + " and " + y.IsEmpty`
		-> Label1 shows "True and False"

.IsMatch(search expression)	Returns a **Boolean** value, indicating if the search expression is included in the object string. This method is compatible to RegEx (regular expressions). For further information on RegEx, please visit: https://msdn.microsoft.com/en-us/library/az24scfc%28v=vs.110%29.aspx or http://timtrott.co.uk/string-formatting-examples/	`Var x = "Hail the Banana King"`
		`Label1.Text = x.IsMatch("(na){2,3}"))`
		-> searches if x contains the expression "na" two or three times in a row
		-> Lable1 shows "True"

.IsNumeric	Returns a **Boolean** value, indicating if the object string contains a single numeric value. This member is particularly useful for checking a string value before converting it to an integer or double with ".ToInteger" or "ToDouble".	`var x = "42"`
		`Label1.Text = x.IsNumeric`
		-> Lable1 shows "True"

.Left(characters)	Separates the object string at the indicated number of characters (starting at the beginning) and returns all characters of the left split string as a **string** value	`Var x = "mathematical"`
		`Label1.Text = x.Left(6)`
		-> Label1 shows "mathem"

.Length	Returns an **integer** with the length of the string object	`Var x = "mathematical"`
		`Label1.Text = x.Length`
		-> Label1 shows "12"

.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = LocalIP.Locked` -> Label1 shows "True"

.PadLeft(total length, padding character (optional))	Pads the object string with white spaces (or the optional padding character) in front of the expression, until the total length of characters is achieved and returns the respective **string**	`Var x = "BOX"`
		`Label1.Text = x.PadLeft(7,"+")`
		-> Label1 shows "++++BOX"

.Padright(total length, padding character (optional))	Pads the object string with white spaces (or the optional padding character) behind the expression, until the total length of	`Var x = "BOX"`
		`Label1.Text = x.PadRight(7,"+")`
		-> Label1 shows "BOX++++"

	characters is achieved and returns the respective **string**	

.Replace(search string, replacement)	Searches the object string for the indicated string and replaces all found strings with the secondly indicated one, the result is also returned as **string** value	`Var x = "mathematical"`
		`Label1.Text = x.Replace("hematic","riarch")`
		-> Label1 shows "matriarchal"

.Right(characters)	Separates the object string at the indicated number of characters (starting at the end) and returns all characters of the right split string as a **string** value	`Var x = "mathematical"`
		`Label1.Text = x.Right(6)`
		-> Label1 shows "atical"

.Split(separator)	Splits the object string indicated by the separator and returns a **list** of all sub strings	`var x = "name|date|address"`
		`var y = x.Split("|")`
		-> y = ["name","date","address"]

.StartsWith(search expression)	Returns a **Boolean** value, indicating if the object string starts with the search expression	`Var x = "Hello World"`
		`Label1.Text = x.EndWith("Hell")`
		-> Label1 shows "True"

.SubString(start, length)	Returns a sub **string** of the object string that starts at the indicated index (first character has index 0) and has the indicated length	`Var x = "chemically"`
		`Label1.text = x.SubString(3,5)`
		-> Label1 shows "mical"

.ToCamelCase	Returns the object **string** formatted to a CamelCase expression	`Var x = "WD custom script click"`
		`Label1.Text = x.ToCamelCase`
		-> Label1 shows "WDCustomScriptClick"

.ToCharArray	Returns a list where each character of the source string is an individual list element	`var x = "Great job"`
		`var y = x.ToCharArray`
		`-> y = ["G","r","e","a","t","`
		`" ","j","o","b"]`

.ToColor	Returns the object string formatted to a **color** type object, the object string has to be a six-digit hexadecimal value	`Var x = "#ABC123"`
		`Label1.text = x.ToColor.G`
		-> Label1 shows "193" (value of the green component of this color)

.ToDate	Returns the object string formatted to a **date** type object	`Var x = "2016-12-16 15:52:48.593"`
		`Label1.text = x.ToDate.Day`
		-> Label1 shows "16"

.ToDouble	Returns the value of the object string in **double** format if the characters form a real number	`Var x = "3.14156"`
		`Var y = x.ToDouble`
		-> y = 3.14156

.ToInteger	Returns the value of the object string in **integer** format if the characters form a real whole number	`Var x = "703"`
		`Var y = x.ToInteger`
		-> y = 703

.ToJson	Returns the value of the object string in **JSON** format if the syntax applies to the respective rules for JSON objects [1926]	

.ToLower	Returns the the object **string,** formatted with only lower case letters.	`var x = "WD Custom Script Click"`
		`Label1.Text = x.ToLower`
		-> Label1 shows "wd custom script click"

.ToSnakeCase	Returns the object **string** formatted to a Snake Case expression	`Var x = "WD Custom Script Click"`
		`Label1.Text = x.ToSnakeCase`
		-> Label1 shows "wd_custom_script_click"

.ToUpper	Returns the the object **string,** formatted with only upper case letters.	`var x = "WD Custom Script Click"` `Label1.Text = x.ToLower`

		-> Label1 shows "WD CUSTOM SCRIPT CLICK"

.Trim	Returns the object **string** without any white spaces at the end or the beginning	`Var x = "  BOX  "`
		`Label1.text = x.Trim`
		-> Label1 shows "BOX"

.TrimEnd	Returns the object **string** without any white spaces at the end	`Var x = "  BOX  "`
		`Label1.text = x.TrimEnd`
		-> Label1 shows " BOX"

.TrimStart	Returns the object **string** without any white spaces at the beginning	`Var x = "  BOX  "`
		`Label1.text = x.TrimStart`
		-> Label1 shows "BOX "

.Type	Returns the type of the respective object as **string**	`Var x = "hypnotized"`
		`Label1.Text = x.Type`
		-> Label1 shows "String"

.Unescape	Returns the object **string** with the correct formatting concerning "\n" (new line), "\r" (carriage return) and "\t" (tab).
	Without unescaping these expressions, they are handled as simple characters.
		`Var x = "Hello World,\n\r\tHow are you?"`
		`Label1.Text = x.Unescape`
		-> Label1 shows:
		"Hello World,
		How are you?"

Data type: Integer		
.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = LocalIP.Locked` -> Label1 shows "True"

.ToString	Returns the value of the object integer as a **string**	`Var x = 703`
		`Var y = x.ToString`
		-> y = "703"

.Type	Returns the type of the respective object as a **string**	`Var x = 703`
		`Label1.Text = x.Type`
		-> Label1 shows "Integer"

Data type: Double		

.Ceiling	Returns the rounded up value (to a whole number) of the object double as **integer** value	`Var x = 3.14156`
		`Var y = x.Ceiling`
		-> y = 4

.Floor	Returns the rounded down value (to a whole number) of the object double as **integer** value	`Var x = 3.14156`
		`Var y = x.Floor`
		-> y = 3

.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = LocalIP.Locked` -> Label1 shows "True"

.Round(digits)	Returns the value of the **double** object, rounded to the indicated number of decimals.	`Var x = 3.14156`
		`Var y = x.Round(4)`
		-> y = 3.1416

.ToInteger	Returns the value of the object double as an **integer** (and rounds the value to a whole number)	`Var x = 3.14156`
		`Var y = x.ToInteger`
		-> y = 3

.ToString	Returns the value of the object double as a **string**	`Var x = 3.14156`
		`Var y = x.ToString`
		-> y = "3.14156"

.Type	Returns the type of the respective object as a **string**	`Var x = 3.14156`
		`Label1.Text = x.Type`
		-> Label1 shows "Double"

Data type: Boolean		
.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = MasterConnectionStatus.Locked` -> Label1 shows "True"

.Not	Returns the negated **Boolean** value of the object Bool	`Var x = True`
		`Var y = x.Not`
		-> y = False

.ToInteger	Returns an **integer** for the object value. "False" is converted to "0", "True" to "1"	`Var x = True`
		`Var y = x.ToInteger`
		-> y = 1

.ToString	Returns the value of the object Bool as a **string**	`var x = True`
		`Var y = x.ToString`
		-> y = "True"

.Type	Returns the type of the respective object as a **string**	`Var x = True`
		`Label1.Text = x.Type`
		-> Label1 shows "Boolean"

Data type: Date		
.AddDays(days)	Returns the object **date** plus the indicated amount of days	
.AddHours(hours)	Returns the object **date** plus the indicated amount of hours	
.AddMilliseconds(milliseconds)	Returns the object **date** plus the indicated amount of milliseconds	
.AddMinutes(minutes)	Returns the object **date** plus the indicated amount of minutes	
.AddMonths(months)	Returns the object **date** plus the indicated amount of months	
.AddSeconds(seconds)	Returns the object **date** plus the indicated amount of seconds	
.AddYears(years)	Returns the object **date** plus the indicated amount of years	
.Day	Returns the object date's days as an **integer**	

.DayOfWeek	Returns the weekday of the object's date information as an **integer**	`Var x = Now`
		`Label1.Text = x.DayOfWeek`
		-> Label1 shows "2" if the script was called on a Tuesday

.DayOfYear	Returns the days of the year of the object's date information as an **integer**	`Var x = Now`
		`Label1.Text = x.DayOfYear`
		-> Label1 shows "241" if the script was called on August 29th 2017

.DiffDays(date)	Returns the amount of days of difference between the object date and the indicated date as **integer**	`Var x = Now`
		`Var y = x.AddYears(1)`
		`Label1.Text = x.DiffDays(y)`

		-> Label1 shows "-365"

.DiffHours(date)	Returns the amount of hours of difference between the object date and the indicated date as an **integer**	`Var x = Now`
		`Var y = x.AddDays(5)`
		`Label1.Text = x.DiffHours(y)`
		-> Label1 shows "-120"

.DiffMilliseconds(date)	Returns the amount of milliseconds of difference between the object date and the indicated date as an **integer**	`Var x = Now`
		`Var y = x.AddSeconds(5)`
		`Label1.Text = x.DiffMilliseconds(y)`
		-> Label1 shows "-11000"

.DiffMinutes(date)	Returns the amount of minutes of difference between the object date and the indicated date as an **integer**	`Var x = Now`
		`Var y = x.AddDays(5)`
		`Label1.Text = x.DiffMinutes(y)`
		-> Label1 shows "-7200"

.DiffSeconds(date)	Returns the amount of seconds of difference between the object date and the indicated date as an **integer**	`Var x = Now`
		`Var y = x.AddHours(3)`
		`Label1.Text = x.DiffSeconds(y)`
		-> Label1 shows "-10800"

.Format(format string)	Uses the C# method "String.Format" on the object date and returns the formatted **string**. For further information on String.Format, please visit https://msdn.microsoft.com/en-us/library/system.string.format(v=vs.110).aspx#Format_Custom or http://timtrott.co.uk/string-formatting-examples/	`Var x = Now` (e.g. 12/01/2016 11:45:23) `Label1.Text = x.Format("t")` -> Label1 shows "11:45"
.Hour	Returns the object date's hours as an **integer**	
.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = Now.Locked` -> Label1 shows "True"
.Millisecond	Returns the object date's milliseconds as an **integer**	
.Minute	Returns the object date's minutes as an **integer**	
.Month	Returns the object date's months as an **integer**	
.Second	Returns the object date's seconds as an **integer**	
.ToString	Returns the value of the object date as a **string**	`Label1.Text = Now.ToString` -> Label1 shows the current date

.Type	Returns the type of the respective object as a **string**	`Var x = Now`
		`Label1.Text = x.Type`
		-> Label1 shows "Date"

.Year	Returns the object date's years as an **integer**	
Data type: List		

.Avg	Returns the average value of all list items as **double**.	`var x = [10, 20, 30]`
		`Label1.Text = x.Avg`
		-> Label1 shows "20"

.Copy	Returns a copy of the **list** object.
	List variables are reference type variables, meaning that if you set one list variable equal to another, you will reference to the same memory address.
	If you simply want to copy the values of one list, use the Copy member to create an independent copy.
		`var x = [10, 20, 30]`
		`var y = x`
		`var z = x.Copy`
		`x[0] = 0`
		-> x and y will show [0, 20, 30]
		-> z will show [10, 20, 30]

.Count	Returns an integer with the number of elements of the list object.
	This member is especially useful for running a for loop 1890 over each element in a list. Remember that a list index starts with 0!
		`var x = [10, 20, 30, 40]`
		`for i = 0 to (x.Count - 1) {`
		`  x[i] += 5`
		`}`
		-> x will now show [15, 25, 35, 45]

.DecodeBytes	Returns a **string** decoded from the UTF-8 byte values of each element of the list object. The integers in the list have to range from 0 to 255.
	Please also refer to the string member ".EncodeBytes"
		`var x = [72,101,108,108,111,32,87,111,114,108,100]`
		`Label1.Text = x.DecodeBytes`
		-> Label1 shows "Hello World"

.Distinct	Removes duplicate elements from the list and returns the resulting **list**.	`var x = [10, 20, 30, 10, 10, 20]`
		`Label1.Text = x.Distinct.Join(", ")`
		-> Label1 shows "10, 20, 30"

.IndexOf(search expression)	Searches for the indicated search expression and returns its index as an **integer** value. If the expression is contained more than once, the index if the first one is returned, if the expression is not included at all, "-1" is returned	`var x = [a, "sdf", 130, 132]`
		`Label1.Text = x.IndexOf(130)`
		-> Label1 shows "2"

.Join(separator)	Concatenates all items of the list object, separated by the specified separator string, and returns this value as a single **string**	`var x = [10, 20, 30]`
		`Label1.Text = x.Join("/")`
		-> Label1 shows "10/20/30"

.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = LocalIP.Locked` -> Label1 shows "True"

.Max	Searches the list object for the item with the highest value and returns this as an **integer** value. Only applicable for lists containing solely integer values	`Var x = [10, 20, 30]`
		`Label1.Text = x.Max`
		-> Label1 shows "30"

.Min	Searches the list object for the item with the lowest value and returns this as an **integer** value. Only applicable for lists containing solely integer values	`Var x = [10, 20, 30]`
		`Label1.Text = x.Min`
		-> Label1 shows "10"

.Remove(index)	Removes the element with the specified index from the original list and returns a **Boolean** value indicating if the removal was successful.	`Var x = [10, 20, 30]`
		`x.Remove(1)`
		`Label1.Text = x.Join("/")`
		-> Label1 shows "10/30"

.Sort(Boolean expression)	Sorts the list object according to increasing (true) or decreasing (false) values, where the Boolean expression differs between the two methods and returns the sorted **list**	`Var x = [5, 203, 144, 42]`
		`DebugMessage(x.Sort(false))`
		-> DebugLogger shows "[203,144,42,5]"

.Type	Returns the type of the respective object as a **string**	`Var x = [10, 20, 30]`
		`Label1.Text = x.Type`
		-> Label1 shows "List"

.WhereRegex(search expression)	Returns a **list** of all items of the list object where the search expression is included. This method is compatible to RegEx (regular expressions). For further information on RegEx, please visit: https://msdn.microsoft.com/en-us/library/az24scfc%28v=vs.110%29.aspx	`Var x = ["banana","handkerchief","Ananas","that's German for 'pineapple'"]`
		`DebugMessage(x.WhereRegEx("(na){2,3}"))`
		-> searches which elements of x contain the expression "na" two or three times in a row
		-> DebugLogger shows "["banana","Ananas"]"

Data type: Color		

.A	Returns the alpha value of the object color as an **integer** (0-255)	`Var x = ColorPicker1.SelectedColor`
		`Label1.Text = x.A`
		-> Label1 shows "255"

.B	Returns the blue value of the object color as an **integer** (0-255)	`Var x = ColorPicker1.SelectedColor`
		`Label1.Text = x.B`
		-> Label1 shows "255"

.G	Returns the green value of the object color as an **integer** (0-255)	`Var x = ColorPicker1.SelectedColor`
		`Label1.Text = x.G`
		-> Label1 shows "0"

.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = LocalIP.Locked` -> Label1 shows "True"

.R	Returns the red value of the object color as an **integer** (0-255)	`Var x = ColorPicker1.SelectedColor`
		`Label1.Text = x.R`
		-> Label1 shows "128"

.SetRGB(red,green,blue)	Sets the RGB values of the object color	`Var x = ColorPicker1.SelectedColor` (e.g. R=128 G=0 B=255)
		`x.SetRGB(100,200,30)`
		`Label1.Text = x.B`
		-> Label1 shows "30"

.SetRGBA(red,green,blue,alpha)	Sets the RGB values of the object color	`Var x = ColorPicker1.SelectedColor` (e.g. R=128 G=0 B=255)
		`x.SetRGB(100,200,30,0)`
		`Label1.Text = x.A`
		-> Label1 shows "0"

.ToHex	Returns the hexadecimal color value of the object color as a **string**	`Var x = ColorPicker1.SelectedColor` (e.g. R=128 G=0 B=255) `Label1.Text = x.ToHex` -> Label1 shows "#8000FF"
.Type	Returns the type of the respective object as a **string**	`Var x = ColorPicker1.SelectedColor`

		`Label1.Text = x.Type` -> Label1 shows "Color"

One special data type is JSON, it enables you to store large sets of data and edit and retrieve this information for other purposes. The "location" of a JSON element is the key (e.g. "arr"), or in case of nested maps the path expressed with a dot syntax (e.g."map.b"). If you are interested in the use of JSON objects, please refer to the chapter Using JSON 1926.

Data type: JSON		

.Add(array location,value)	Adds a new item to an array at the specified location and returns the changed **JSON** array (optional)	`var x = '{"arr":[12,34,56], "map": {"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.Add("arr", 78).ToInteger`
		-> Label1 shows "[12,34,56,78]"

.Clear(map location)	Removes all child items at the specified location and returns the **JSON** element containing the changed item (optional)	`var x = '{"arr":[12,34,56], "map": {"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.Clear("map")`
		-> Label1 shows "{"arr":[12,34,56],"map":{}}"

.Copy	Returns a copy of the **JSON** object. JSON, like list variables, are reference type variables, meaning that if you set one JSON variable equal to another, you will reference to the same memory address. If you simply want to copy the values of one JSON, use the Copy member to create an independent copy.	`var x = '{"name":"Arthur Dent"}'.ToJson`
		`var y = x`
		`var z = x.Copy`
		`x.Set("name","Zaphod Beeblbrox")`
		`Label1.Text = x.GetString("name")`
		`Label2.Text = y.GetString("name")`
		`Label3.Text = z.GetString("name")`
		-> Label1 and 2 show "Zaphod Beeblbrox", Label3 shows "Arthur Dent"

.Count(array or map location)	Returns the number of child items at the specified location as an **integer** value. If the location is not a map or an array, an exception is thrown	`var x = '{"arr":[12,34,56], "map": {"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.Count("map")`
		-> Label1 shows "3"

.Get(location)	Returns the value at specified location as a **JSON** expression	`var x = '{"arr":[12,34,56], "map": {"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.Get("map.b").ToString`
		-> Label1 shows "2.3"

.GetDate(location)	Returns the value at the specified location as a **date** value. If the value cannot be converted, a default value is returned	`var x = '{"start":"2016-02-03"}'.ToJson`
		`Label1.text = x.GetDate("start").Month`
		-> Label1 shows "2"

.GetDouble(location)	Returns the value at the specified location as a **double** value. If the value cannot be converted, a default value is returned	`var x = '{"arr":[12,34,56], "map": {"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.GetDouble("map.a")`
		-> Label1 shows "1.2"

.GetInteger(location)	Returns the value at the specified location as an **integer** value. If the value cannot be converted, a default value is returned	`var x = '{"arr":[12,34,56], "map": {"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.GetInteger("map.c")`
		-> Label1 shows "3"

.GetString(location)	Returns the value at the specified location as a **string** value.	`var x = '{"arr":[12,34,56], "map":{"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.GetString("arr")`
		-> Label1 shows "[12,34,56]"

.Keys(optional array or map location)	Returns a **list** of all keys/indices at the specified location	`var x = '{"arr":[12,34,56], "map":{"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.Keys("map")`
		-> Label1 shows "["a", "b", "c"]"

.Locked	Returns a **Boolean** value, indicating if the object is locked. Only useful for global variables, as they are the only lockable objects. Local variables are always unlocked.	`Label1.Text = LocalIP.Locked` -> Label1 shows "True"

.Remove(location)	Removes the selected location as well as all of its child elements.	`var x = '{"arr":[12,34,56], "map":{"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`x.Remove("map")`
		-> x does now only contain "arr":[12,34,56]

.Set(location, value)	Changes the value or defines a new one at the specified location and returns the **JSON** element containing the changed item (optional)	`var x = '{"arr":[12,34,56], "map":{"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.Set("map.a",123).ToString`
		-> Label1 shows "{"a":123,"b":2.3,"c":3.4}"

.ToFile(filePath)	Writes the JSON data, formatted as pretty string, into a text file located at the specified path. If the file does not exist, a new file is automatically created.	`var x = '{"arr":[12,34,56], "map":{"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`x.ToFile("C:\Christie\content\projects\test.txt")`
		-> Writes the JSON data into the text file "test.txt"

.ToPrettyString	Returns a **string** with the JSON data formatted with line breaks and indented for better readability.	`var x = '{"arr":[12,34,56], "map":{"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`TextBox1.text = x.ToPrettyString`
		-> TextBox1 now shows the JSON data in separate lines indented with tabs where applicable

.ToString(location)	Returns the JSON object as a **string**	`var x = '{"name":"Arthur Dent"}'.ToJson`
		`Label1.text = x.GetString("name").Left(6)`
		-> Label1 shows "Arthur"

.Type	Returns the type of the respective object as a **string**	`var x = '{"start":"2016-02-03"}'.ToJson`
		`Label1.Text = x.Type`
		-> Label1 shows "Json"

.Values(array or map location)	Returns a **list** of all values at the specified location	`var x = '{"arr":[12,34,56], "map":{"a":1.2, "b":2.3, "c":3.4}}'.ToJson`
		`Label1.text = x.Values("map").Join(", ")`
		-> Label1 shows "1.2, 2.3, 3.4"

The object data type is always applied when it is not possible to determine the specific type of an object beforehand, e.g. for list elements, local variables without a value assigned or for the return value of a function.

Data type: Object		

.ToBoolean	Returns the object content formatted to a **Boolean** type object if the characters form: 0,1,true,false	`Var x = ["a","b",123,0,"#8000FF"]`
		`vcolor=x[4].ToColor`
		-> vcolor shows "#8000FF"

.ToColor	Returns the object string formatted to a **color** type object, the object string has to be a six-digit hexadecimal value	`Var x = "#ABC123"`
		`Label1.text = x.ToColor.G`
		-> Label1 shows "193" (value of the green component of this color)

.ToDate	Returns the object content formatted to a **date** type object	`Var x = ["a","2017-03-17",123.456,true,[3,6,9]]`
		`Label1.Text = x[2].ToDate.Year`
		-> Label1 shows "2017"

.ToDouble	Returns the value of the object content in **double** format if the characters form a real number	`Var x = ["a",b,123.456,true,[3,6,9]]`
		`Fader1.Value = x[2].ToDouble`
		-> Fader1 will take the value 123.456

.ToInteger	Returns the value of the object content in **integer** format if the characters form a real whole number	`Var x = ["a",b,123,true,[3,6,9]]`
		`Fader1.Value = x[2].ToInteger`
		-> Fader1 will take the value 123

.ToJson	Returns the value of the object content in JSON format if the syntax applies to the respective rules for JSON objects 1926	

.ToList	Returns the object content formatted to a **list** type object	`Var x = ["a",b,123,true,[3,6,9]]`
		`Label1.Text = x[4].ToList.Avg`
		-> Label1 shows "6"

.ToString	Returns the object content formatted to a **string** type object	`Var x = ColorPicker1.SelectedColor` (e.g. R=128 G=0 B=255) `Label1.Text = x[0].ToString` -> Label1 shows "a"

.Type	Returns the type of the respective object as a **string**	`Var x`
		`Label1.Text = x.Type`
		-> Label 1 shows "Object"

7.8.4.4 Math Object

The Math object offers a wide variety of mathematical functions. It always returns the computed result as a numeric value without changing the argument itself.

Find here a list of all available methods. Remember to assign a suiting type to the variable for the result, e.g. "Double". In the interests of clarity, the result is rounded to two digits.

Description and Syntax	Result
Abs returns the absolute value of a specified number. `varRes = Math.Abs(-1.23)`	1.23
Acos returns the angle whose cosine is the specified number (in radians!). `varRes = Math.Acos(0.99)`	0.14
Asin returns the angle whose sine is the specified number (in radians!). `varRes = Math.Asin(0.99)`	1.43
Atan returns the angle whose tangent is the specified number (in radians!). `varRes = Math.Atan(0.99)`	0.78
Ceiling returns the smallest integer greater than or equal to the specified number. `varRes = Math.Ceiling(1.5)`	2
Cos returns the cosine of the specified angle (in radians!). `varRes = Math.Cos(30)`	0.15
Deg2Rad returns the radiant of the angle specified in degrees. The formula is: 1° × π/180 = 0,01745rad	3.14

Description and Syntax	Result
`varRes = Math.Deg2Rad(180)`	
E returns the mathematical constant "Eulers number" `varRes = Math.E`	2.72
Exp returns e (Eulers number) raised to the specified power. `varRes = Math.Exp(1)`	2.72
Floor returns the largest integer less than or equal to the specified number. `varRes = Math.Floor(1.5)`	1
Log returns the natural logarithm of a specified number. `varRes = Math.Log(2.718)`	0.99
Log10 returns the base 10 logarithm of a specified number. `varRes = Math.Log10(5)`	0.7
Max returns the larger of two specified numbers. `varRes = Math.Max(1, 2)`	2
Min returns the smaller of two numbers. `varRes = Math.Min(3, 4)`	3
Mod returns the remainder after the division of one number by another. `varRes = Math.Mod(10, 3)`	1
Pi returns the mathematical constant π ("Archimedes' constant") `varRes = Math.Pi`	3.14x
Pow returns a specified number raised to the specified power. `varRes = Math.Pow(5, 2)`	25
Rad2Deg returns the degree of the angle specified in radians. The formula is: 1° × π/180 = 0,01745rad `varRes = Math.Rad2Deg(3.142)`	180
Returns a random integer value between the indicated min and max value. As common practice in most programming languages, the lower bound is included, the upper bound is excluded from the returned numbers. `varRes = Math.Random(0, 100)`	42
Round rounds a value to the nearest integer or specified number of decimal places. `varRes = Math.Round(1.234, 2)`	1.23
Sign returns a value indicating the sign of a number. `varRes = Math.Sign(-10)`	-1
Sin returns the sine of the specified angle (in radians!). `varRes = Math.Sin(30)`	-0.99
Sqrt returns the square root of a specified number. `varRes = Math.Sqrt(25)`	5
Tan returns the tangent of the specified angle (in radians!). `varRes = Math.Tan(30)`	-6.41
Truncate calculates the integral part of a number. `varRes = Math.Truncate(10.73)`	10

7.8.4.5 Using JSON

The previous chapters explained the Object and Member Notation [1904] in Widget Designer, this chapter will give you a quick overview of the JavaScript Object Notation, a format for easily storing, editing and exchanging data.

Additionally, you will learn how to make use of this format in your Widget Designer project.

What is JSON?

JSON, or JavaScript Object Notation, is a format designed for data exchange. It derived from JavaScript, but is by now also included in many other programming languages.
One of its advantages is that the stored data consists of human readable text and is based on a very simple and straight forward structure.

Each set of data consists of a key and a value, separated by a colon ":". Several sets can be separated by a comma "," and a collection of sets of data is called a JSON object, enclosed in curly brackets "{ }".
A key is always a string, enclosed in double quotation marks, a value can be one of the following types:

Number: Can be either an integer or a decimal number, exponential E notation is possible, too.
String: a sequence of characters, delimited with double quotation marks
Boolean: either one of the expressions `true` or `false`
Array: an ordered list of zero or more values of any type, delimited with square brackets "[ ]"
Map (Object): JSON objects are nestable, so an object can always contain another object, called map.
null: an empty value indicated with the expression `null`

White spaces, line feeds and tabs don't affect the functionality of the object, but can be used to arrange code.

Example

```
{
	"company": "Christie",
	"year_founded": 1929,
	"over_1000_employees": true,
	"products": {
		"media_server":"Pandoras Box",
		"projector":"D4K40",
		"video_wall":"MicroTiles LED"
	}
}
```

How to use JSON in Widget Designer?

JSON is available as a global variable: simply right-click on the variable list [1900], choose "Create > Json" and enter a name and a value. As a JSON object can be quite large and hard to overview, it might be easier to write the complete object down in a script window and then either copy and paste the whole text into the "Value" field, or directly assign the value via a script to an empty JSON variable.

Other than other variable types, the automatic type recognition for local variables is not available for the JSON type. You would have to type in a string with the correct format (pay attention to use the single quotation marks ' ' to indicate the string!) and convert it via the .ToJson member:

```
var participant1 = '{
      "name":"Jonathan",
      "age":27,
      "nationality":"UK",
      "languages": {
            "English":"expert",
            "French":"basic",
            "Spanish":"advanced",
            "other":["Russian-little","German-advanced"]
            }
}'.ToJson
```

Now you can make use of the various JSON members [1922] to update, retrieve and use the stored data. When accessing nested maps and arrays, the location is defined by a dot-separated path: "participant1.Count("languages")" returns "4", "participant1.Count("languages.other")" returns "2".

It is even possible to store and readout data as a Session value by using the Context object [1909]:

```
Context.Session.Value("JSON_data") = participant1
Label1.Text = Context.Session.Value("JSON_data").ToJson.GetString("name")
```

-> Label1 shows "Jonathan"

For more detailed information on JSON and its syntax, please refer to: http://json.org/

What to Bear in Mind When Using JSON in WD

It is important to keep the JSON in the correct format if you want to make use of its members.
JSON has a very open and flexible syntax, there are many variations you can apply and still get valid JSON code without an error message.
If you want to make use of the many helpful JSON members [1922], you need to pay attention to initializing the variable correctly and, if you add elements manually, the above explained format.
Using JSON members to set elements is not only comfortable, but also eliminates sources for syntax errors

They enable you to readout and edit elements in the JSON comfortably without editing the text directly, which eliminates sources for syntax errors.

```
var jsonLocal = '{}'.ToJson
```

If you have a global JSON variable without value, you still need to add the brackets. This can happen either with entering them in the Create / Edit dialog of the variable, or with the same script:

```
jsonGlobal = '{}'.ToJson
```

Not using the brackets will result in an invalid format of the variable and can cause issues.

When you fill your JSON with values, you need to pay attention to the correct format as well, any structures not confirming the JSON specifications can cause errors. Contents like the one shown below are not compatible as the necessary key/value assignment is not given:

```
var jsonLocal = '{0}'.ToJson
```

7.8.4.6 Node Commands

Please note that this topic will be updated soon with changes introduced with version 6. In the meantime, please refer to the support team if you have any questions.

Widget Designer version 4.5 introduces the feature to remote control nodes [936] with so called "Node Commands". Up until now, commands were "only" for items like a Fader etc. but with the introduction of more advanced nodes, the need arose to control their features with commands too. These more complex nodes allow to enter advanced settings directly in the node and are able to execute functions themselves. Nodes that have remote controllable functions are for example:
- Projector Control Input node [1024]
- OptiTrack ID Tag Input node [985]
- Watchfolder to PB Playlist Interaction node [1252], etc.

Node commands are executed as any other command. It can be written, for example, in a Custom Script Button to be executed as soon as the button is pressed. Please refer to the chapter "Script Language [1511]" for more details about commands.

There are two ways to write a node command. One form is a more direct form whilst the other reminds of the structure of other common commands. As with all commands, the original form includes placeholders. By replacing them with according information the command addresses one particular setting from one particular node. So by typing the command, it assumes a definite shape.

Node Commands that Access a Function of a Node

As an example please see the table below. It shows the two basic command forms and examples how they could be filled out.

	Direct node command	Common node command
Basic command to access an function	`NodeID.Nodecommand`	`WDNodeCommand,NodeID,Args0,ArgsN`

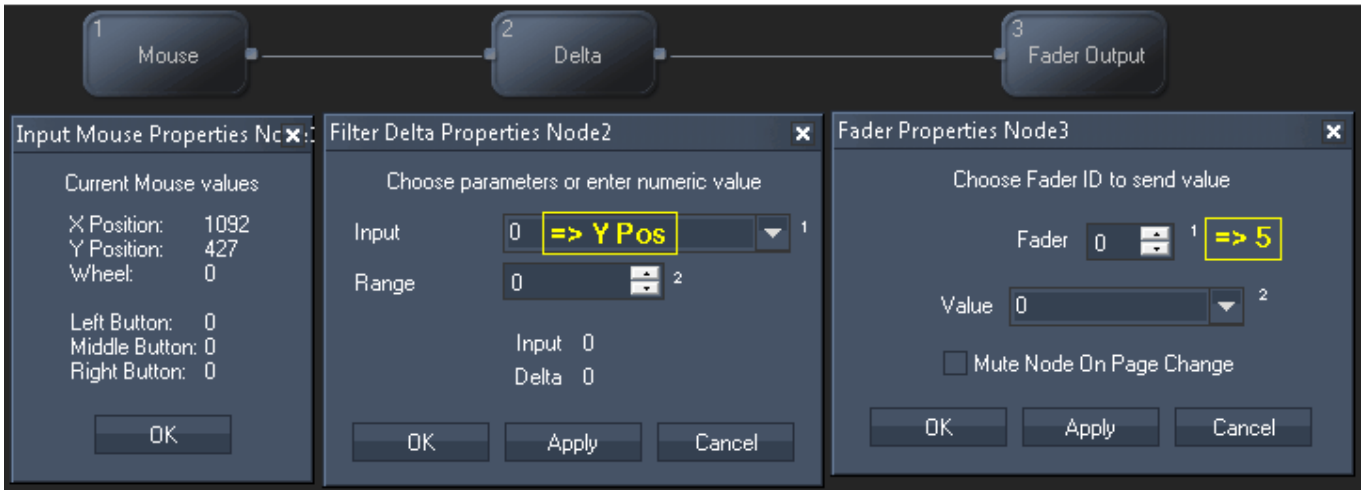
Examples for: Projector Control	`node1.PowerOn`
	`node1.PowerOff`
	`node1.LampOn`
	`...`
	`node1.Input4`
		`WDNodeCommand,1,PowerOn`
		`WDNodeCommand,1,PowerOff`
		`WDNodeCommand,1,LampOn`
		`...`
		`WDNodeCommand,1,Input4`

Examples for: Watchfolder to PB Playlist	`node2.ResetPlaylist` `node2.ClearWatchFolder`	`WDNodeCommand,2,ResetPlaylist` `WDNodeCommand,2,ClearWatchFolder`

The Script Assist feature, offers the available functions for the according node:
- For the direct form: As soon as the dot behind the ID is typed...
- For the common form: As soon as the comma behind the ID is typed...
... , Script Assist looks up what this node is (e.g. a Projector Control) and displays the according commands in a list box. Use the arrow keys [Up] or [Down] to select an entry and press [Enter]. The [Esc] key hides the list box.

Node Commands that Access a Parameter of a Node

In addition to the purpose of controlling functions, node commands can also be used to assign a value to a parameter within a node. The examples refer to the depicted node chain.



Example for entering the Fader ID 5 into the Fader Output node:
The Fader Output node has the NodeID "3" as depicted in the upper left corner in the node. Its parameter for "Fader" is identified with "1" as depicted by the small superscript number next to the parameter field in the Node Properties dialog.
There are 3 possible command forms: (please note that the two direct forms do not support variables yet)
1) A direct node command
2) A 2nd direct node command
3) Common node command

	Node Commands	Examples
1	`NodeID.ParamID@Value`	`node3.1@5`
2	`NodeID.SetParamValue,ParamID,Value`	`node3.SetParamValue,1,5`
3	`WDNodeSetParam,NodeID,ParamID,Value`	`WDNodeSetParam,3,1,5` `WDNodeSetParam,3,1,Variable`

Example for setting the "Input" source in the Delta Filter node to "Y Pos" from the Mouse Input node:
The Delta Filter node has the NodeID "2" and the Mouse Input node "1". Their IDs are depicted in the upper left corner in the node. The parameter for "Input" (that can accept input values from other nodes) is identified with "1" as depicted by the small superscript number next to the parameter field in the Node Properties dialog. The Mouse Input node provides several values, "Y Pos" is the second of them, thus it gets the ID "2".

	Node Commands	Examples
1	`NodeID.ParamID&NodeID,ParamID`	`node2.1&1,2`
2	`NodeID.SetParamValue,ParamID,Value`	`node2.SetParamSource,1,1,2`
3	`WDNodeSetParamSource,NodeID,ParamID,SourceNodeID,SourceNodeParamID`	`WDNodeSetParamSource,2,1,1,2` `WDNodeSetParamSource,2,1,1,Variable`

7.9 Web Server

Since version 4 Widget Designer (WD) is able to publish its pages as html files and operates as a Web Server. Simply open your web browser and connect to the IP address and port number where the Widget Designer application runs on. Now you can view the Widget Designer page with its controls and use them.
Since version 6, all controls are supported as a web control too. All widgets like Custom Script Buttons and Faders can be used within the web browser in the same way as in Widget Designer. This includes of course the possibility to execute one of the 1500 commands of WD's build-in script language [1511].
Please make sure to use the latest updates for your web browser. Currently Google Chrome, Mozilla Firefox, Opera, Apple Safari and Windows Internet Explorer are recommended.

Session Synchronization and Multi-Session

As soon as a user connects to the Web Server, a so called session is created, together with a unique session key. A special session cookie stores this data within the browser and can last several days. A second user with another

connection would create a second session and gets another key. If your page, for example, contains a fader, there are two possible scenarios how the two users may interact with the fader:

- the page(s) and the faders are published and displayed synchronized.
  All web faders are copies or instances of the main, local WD faders; their values are always linked to each other. If a web fader is moved, it reports its value to the local fader and this one in turn to the other web instances. In other words, no matter whether the programmer moves the fader in the Widget Designer interface or a user does so in the web browser interface, everybody sees the same fader and value.
  If your project consists of several pages, every session would switch to the same page as soon as one user (or operator) calls another page.

- the page(s) and controls are published and displayed in multi-session mode (Unlimited Web Clients Version only, see below)
  Each session and user can have an individualized view and control setting. If one user moves the fader now, only his fader will adopt to its individual session value. Another user might set up the fader with another session value, he is not influenced by the first user or rather the WD operator. The main Widget Designer application is able to receive, collect and evaluate all session values. For example, if the fader is used for a voting event, WD could evaluate the highest / lowest value or the average one.
  Regarding the page behavior, each user can view and switch the page of choice and is not linked to the other users.
  The Unlimited version supports even a combination of these two scenarios. If desired you may choose to have a synchronized page behavior but multi-session tool behavior, it is even possible to arrange the grouping of values for different clients.

Web Server features in different Widget Designer Versions

The Web Server is a very powerful feature in Widget Designer. Its three editions have different implementation levels.

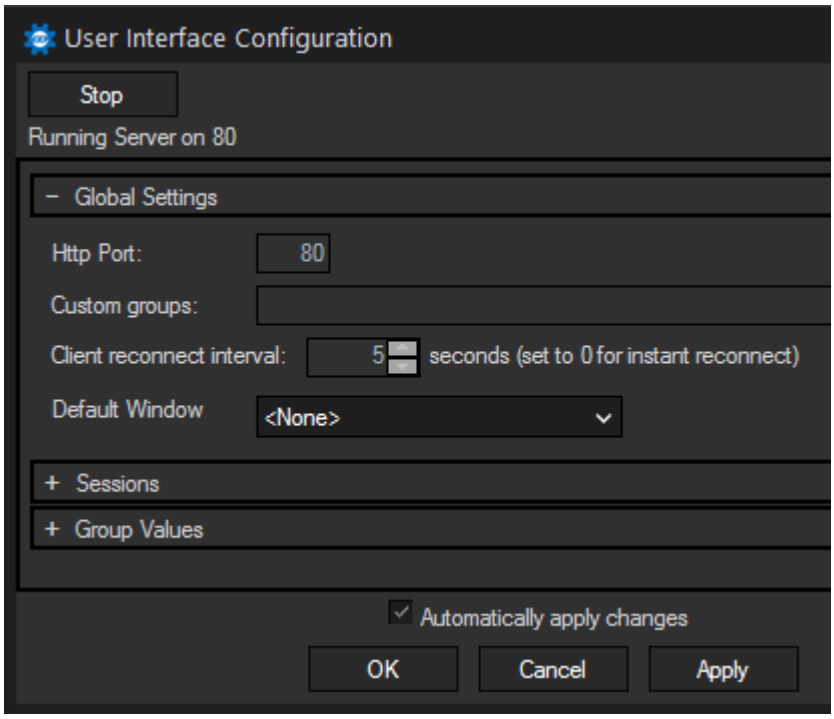
WD Free Version may publish one page to be accessed by one user. It is not possible to create a second session. If the project consists of several pages, you may decide which one should be published, it does not necessarily need to be "Page1".

WD may publish an unlimited number of pages to be accessed by an unlimited number of users at the same time. If the project consists of several pages, you may decide which one should be published as the index page - the one you will be automatically redirected to, when connecting to a web server. It is not possible to create a multi-session, all sessions are synchronized. Thus, all users will have access to the same page and its controls simultaneously.

WD Unlimited Web Clients may publish an unlimited number of pages to be accessed by an unlimited number of users at the same time. If the project consists of several pages, again, you may decide which one should be published as the index page. In addition you may decide whether all sessions should be synchronized or whether a multi-session should be supported which will allow simultaneous as well as individual setups. In individual setups each user can see a different page and in addition multi-session controls like Faders and Custom Script Buttons may have deviating values.

How to use the Web Sever?

In the main menu bar: Click on Edit > Web Server Settings



The web server is automatically enabled, you do not need to start it manually to use this feature. The default ports being used are: 80 for HTTP, 8080 for WebSockets and 81 for uploads. If those ports are already in use, for example by another WD instance, free ports in the range of 30300 and higher will be searched and applied automatically. The chapter Ports Used by PB and WD [781] lists all occupied ports.

If you use the web server on your local PC, open a browser and type in "localhost". You will then be redirected to your index page or a directory of all available pages. For other PCs in the network enter the IP address of the WD machine, e.g.:
http://10.169.10.31
As long as you are using the default ports, no additional value has to be entered. However, if you have to use another port than 80 for HTTP, you will have to specify so in the address in order to get to the correct WD instance interface.
Example for HTTP port 30300: http://10.169.10.31:30300

A specific window can be addressed by entering only the window name behind the IP address, the page set up as default page in the window's properties [803] will be displayed:
http://10.169.10.31/Window1

Certain pages can be accessed by typing in the respective window and page name and separate them with a hash tag (#):
http://10.169.10.31/Window1#Page1

If you like to use the feature not only within a local area network (LAN), you may do so as well. First of all your router must be set up to allow access on the chosen port. Then it must be configured to forward this wide area network (WAN) request to the specific Widget Designer IP address within your local area network (LAN). Now you may enter your IP address and again the port number.
If you like to use the feature for a longer period, keep in mind that you might not have a static IP address. Most providers assign a deviating IP address every 24h, thus you have a dynamic IP address. In that case, a ddns service (dynamic domain name system, e.g. dyndns.org) is very useful. The system provides a persistent domain name and points to a changing, dynamic IP address on the internet. Your router must support ddns in order to update the system's database.

Global Settings

If you want to change anything in this section, you need to stop the web server first by clicking the "Stop" button at the top left corner. Press the "Start" button to enable it again.
You can set the ports to custom values if needed and also adjust the amount of listeners. The last may come in handy, if you have a very large amount of clients accessing your project.

When you use group values, you can define your custom groups here. Just enter a valid name for a group, multiple groups can be separated either by a white space or a comma.

Sessions

The log text field lists for example whether a new session has connected to the web server as well as the respective session keys and the count of all clients connected via the session. The first session is always the internal GUI.

Group Values

If you work with group values, you will find a table with all values of all widgets according to their groups. For more information, please refer to the chapter Group Values [1933].

If you like to have an index page that is displayed instead of the page directory when you did not specify a certain page, switch to this page and edit the settings. Either right-click somewhere in the empty main background and choose Pages > Edit Page or open the Pages menu in the main menu bar and choose the command there. Tick the check box "Redirect to this page (when invalid URL is requested)".
If you like to exclude pages from the web access, uncheck the box "Make this page accessible for external clients".

Visual and technical differences between the interfaces in Widget Designer and web browser

As WD V6 is natively based on HTML5, the main GUI and every client interface should look exactly the same. Depending on the browser version, there might occur slight differences, especially if exotic fonts are used.

Technically, it is very important to understand that all nodes, tools and all widgets are "live" in the Widget Designer. The web browser interface is a second interface, it represents what is set up in the main WD interface, it can be seen as an instance or copy of it.

Only the WD itself gives you the possibility to create new elements and to change their Item Properties.
For example, if you create a fader in WD, the browser adopts and displays a fader too. If you like to change its look or value range, do so in Widget Designer and it will send the new properties to the web instances. So, the look of every interface will always be the same and the widget's Item Properties always match.

A widget can be actually used in both interfaces. Some of the most important widgets: Window, DropDown List, Fader, ColorPicker, Encoder, Wheel, InputBox, Label and TextBox are able to respond to Group Values [1933]. All others are always synched.
For example, you may change the fader's value by dragging its handle either in WD or in the web browser. Depending on the setting to have a synchronized session or to use group values, the other web copies of the fader will update too and move automatically to the new value; or in case of a group value setup, they can be only moved in the local interface, corresponding to the group. That means that a fader can not only hold one value but several ones and each web instance shows the same fader but with a different value, its own group value.

In a nutshell, the widgets' properties are identical, but their values can be customized separately.

A command only works in the Widget Designer application and is executed there. You may send a command, either from within the WD or a browser but it only affects the control in WD.
For example, if you send "WDFaderUp(ID,Seconds)" only the local WD Fader will actually do the fade. Of course, if you have a synchronized session, all web faders will fade too. This is not the case when using group values, as those faders are now detached.
Importantly, if you send a command that alters the appearance of a control, e.g. the color of a Custom Script Button (WDCustomScriptTint(ID,Red,Green,Blue)) again, this alters the local button and as the Item Properties always match, all other web buttons will be tinted in that color too.

Some of the property members of the group value compatible widgets can be alternated separately for each client using the widget's group items (e.g.: Label1.GroupItems.ForeColor). Please refer to the point Group Items [1936] for an overview of available widgets and properties.

A node can be used in WD only and similarly to commands, it works up local values. Also, no group values may be assigned to a widget that is used in combination with a node.

As mentioned above, Widgets [818] can be multi-session controls and have different values per session (or other group participant). Widgets, pages and variables may have group values. It is possible to read these out and

evaluate them or to process them in commands or variables. Please read the topic Group Values [1933] if you are interested in that feature.

7.9.1 Group Values

What are Group Values?

Group Values are only of interest for those who use the Web Server feature with the optional feature "Unlimited Web Clients" in Widget Designer. The Web Server allows to view your Widget Designer interface with an external web browser. As you can "load" the programmed interface in several browsers, you can see and use it in various places.
Sometimes, you might not want all those interfaces to show exactly the same thing, e.g. when one participant moves a fader, the faders in the other interfaces should not move. You might want to use each fader in each browser, or even in each browser tab, independently from each other, and you might want to be able to read out every single fader value.

This is what Group Values are for. If a Widget is assigned to a Group, it creates instances of itself. Instances can be synchronized or differ from other instances. Two fader instances for example can have the same or a different value. Label instances can be even more multifaceted: two label instances can display the same text but have different text colors. In other words, if a Label is assigned to a group, its text and its color property become Group Values that can differ within the Group. The Fader value and the Label color depend on the place where you see and use the interface.

Before we have a deeper look at the meaning of "Groups", just bear in mind that every Widget supports different Group Values. A Fader supports only a multifaceted value (handle position) whilst a Label supports also some multifaceted properties but a CustomScript button cannot be assigned to a group i.e. has no Group Values at all. A CustomScript button can only be used globally. As nodes are also operating globally only, the usage of those in a Group context is not possible.

Working with Group Values requires being familiar with the concept of Context object notation [1906]. In this chapter, you will also find a list and description of context based members dealing with clients and Group Values.

What is a Group and a Key?

Each widget can belong to exactly one Group. Within the WD project however, you can define multiple Groups that you use for different widgets and windows. Per default, a widget is not assigned to a Group at all, which means that it is synchronized in all interfaces (the main Widget Designer interface and all browsers that connect to it).

Widget Designer offers four default Groups, and on top you can create custom ones.
This default Group consists of two Group sections, the first is called "Internal" and the other "External". The Internal Group section refers to the main Widget Designer user interface and cannot be subdivided further. The external section however refers to all interfaces outside of WD. Every browser that connects to the Web Server to render our widgets is part of the external Group section. Per definition, the widget's Group Values are synchronized when they belong to the same section, which means they are synchronized in all browsers. The main WD is the internal section and can differ from the external one.
The explanation of this behavior introduces the term "Key". By assigning a widget to a Group, instances are created. Each instance is identified by a so called Key. The Keys in the InternalExternal Group are simply called "internal" and "external". So there is a set of widget instances that have the "internal" Key and there is a set that has the "external" Key. The main Widget Designer displays the widgets instance with the internal Key, all other browsers get a copy of the widget instances with the external Key. For other Groups it is possible that every single widget instance gets a different Key when it is displayed in another interface, those Keys are less readable as they are a strings with letters and numbers.

Default Groups

There are four default Groups which generate Keys automatically for each object. The Keys are always created for any client accessing the interface, no matter which version of Widget Designer is running.
However, only the **Unlimited Webclients** version enables you to make use of Groups and Group Values, including differing instances of the same object.

- Client:
Each tab inside a single browser equals one Client. Even if you refresh a browser tab, a new Client Key is generated for this tab. The Key consists of a string of hexadecimal characters, e.g. "c21e4700-11aa-22bb-33cc-44dd55ee66ff"

- Session:
Each browser that connects to the Widget Designer generates a new Session Key. This Key looks similar to a Client Key, e.g. "5e551040-f00f-e99e-d88d-c7b6a55a6b7c".
This is probably the most used group as it covers the majority of applications and additionally offers you the possibility to store and retrieve data. Please also have a look at the chapter Session and Session Value 1913 for an introduction to this powerful tool.
This also means that a second, different browser creates a new Session and a new Client, while a new tab of the same browser only generates a new Client Key. It still belongs to the same Session as the first tab.

- InternalExternal:
"Internal" comprises all objects inside the main Widget Designer user interface, "external" describes every object accessed from an external browser (even on the same PC). The Keys here are simply "internal" and "external".

- IPAddress:
This Group distinguishes between different IP addresses for assigning Keys. The key here equals the participating IP address, or "localhost" if it is the Main GUI.

How to assign Groups

For a better understanding of the previous point, the workflow of assigning Groups will be explained with the help of a short exercise, it will only take you about ten minutes.

First, open a new project and create four faders next to each other. Open the the faders' property dialogs and assign each of them to one of the four default Groups. This can be done by opening the Group Values tab and selecting the respective Group in the drop-down. For having a better overview, you can add optional labels for each fader showing which Group it belongs to.

Next, open a browser on the same PC and access the project with "localhost\Window1#Page1". Open a second tab in the same browser and enter the same address. You have now created one Session with two clients. Open a second, different browser and access the project again, you have now created a second Session with one client. Regarding the Internal/External Group, you have now one client with the Key "internal", which is the Main GUI, and three clients with the Key "external".

Now, go back to the Widget Designer and set the faders to different values whilst looking at the faders in the different tabs and browsers to see which faders are linked. Of course, you can also move a "web" fader. You can also add more tabs to the browsers or refresh a tab and see what happens. If you open the Group Values List (Fader Item Properties > Group Values > Show Group Values or Edit > Web Server Settings > Group Values), you can see all values of all fader instances and their respective Keys. The list does not update automatically, so you will have to press "Refresh" once in a while. If you have the possibility to add a second computer to your network, you can also access the project with this one, to see that the fader with the "IP" Group now also has another instance.

Custom Groups

Widget Designer also offers you the possibility to create your own Groups and assign your own Keys. These are called "Custom Groups".

To create Custom Groups, simply go to Edit > Web Server Settings > Global Settings.
If you want to change anything there, you have to stop the Web Server first by clicking the "Stop" button in the top left corner.
Enter the names of your custom groups in the respective field, multiple Group names can be separated either by a comma or a white space. Use only letters (capital or lowercase), numbers and underscores for your group names, the name has to start with a letter.
Do not forget to start the Web Server again after doing your settings.

Now let us add this feature to our test project with the faders. Add a Custom Group "Sea" to your project and create a fifth fader. Assign this fader the new Group, which was automatically added to the Group drop-down.

What you have to consider when using Custom Groups, is that the Key here is not automatically generated, you have to assign your own Key manually to each client participating the project. We will now choose three different Keys, "Octopus", "Swordfish" and "Orca". To assign them, create three different CustomScript buttons and label them respective to the Key they generate. The script that will assign the Custom Key is:

```
Context.Client.Groups.Sea = "Key"
```

Put this script in each button's On Press Script field and replace "Key" with "Octopus", "Swordfish" and "Orca". If you press one of those buttons now, you assign this Key to this client. This specific browser tab (the script has to be executed for each new client) now has e.g. the Key "Swordfish" for all widgets belonging to the Group "Sea". Of course you can change this Key anytime by pressing one of the other buttons. The Context object makes sure that the script is executed ONLY in this single client and not globally for every client, although the button is indeed pressed simultaneously in all clients (the CustomScript button cannot have Group Values). For more information about context in general and the Context object, please refer to the chapter Project and Context Member 1906.

Now try out the buttons and see how the "Sea"-faders are linked when the clients have the same Key for this Group.

This is of course a training example without any use in a real project. But imagine a game show, where there are three different teams, "Red", "Green" and "Blue". All members of these teams have to solve different tasks and each participant has a tablet PC where he or she has to enter the results, adding up a score. You can then create a Custom Group "Team" and use the Keys "TeamRed", "TeamGreen" and "TeamBlue" to synchronize the score for each group, while every member still can operate his or her own interface. Exchanging data or information between the team members is also imaginable.

Summary: A Custom Group is initialized in the Web Server Settings. To assign a Widget to a Group, open its Item Properties in the Main WD Interface. To initialize a key and assign all Widget instances to it, execute a command in the remote interface or main WD.

Accessing Widget Properties with Group Values

If you want to access the values of different instances of one widget with a script, you need to comply with a special syntax. If you used the normal object property, like e.g. "Label1.Text", it would set or return only the one global value. The global value of every widget is still active, even when a Group is set, but Group Values will always override it.

As soon as you have assigned a Group to a widget, you will notice that the Script Assistant will offer you four more members for it:

Object	Group Member	Further Members	Description
WidgetID	.Group		Example: `Label1.Text = Fader1.Group` Returns a string with the Group name the widget belongs to. Pleas note that this is a read-only property, setting the Group is only possible manually in the widget's property dialog. In this example, the Label will display "Session" if the Fader was assigned to the Group "Session".

	.GroupItems(optional Key)	.Property	Example:
			`Label1.Text = Fader1.GroupItems.Value`
			`Label1.Text = Fader1.GroupItems("192.168.1.100").Value`
			`Label2.GroupItems.BackColor.SetRGB(255,0,255)`
			`Label2.GroupItems("5e551040-f00f-e99e-d88d-c7b6a55a6b7c").BackColor(255,0,255)`
			
			Returns the client specific value of the selected widget property, based on the context wherein the command is executed. Each widget has different available properties. A label offers e.g. "BackColor", "ForeColor", "Text" and "Transparent", a fader only offers "Value". If a specific Key is specified for the Group Items, the selected widget property for this specific Key will be returned, regardless of the context in which the command was executed. It is also possible to use further method or property members when entering an additional dot ".".
			The example could have the result that Label1 shows "123" when the first command is executed in the WD GUI but "10" when executed in an external browser.
			The second command could have the result that Label1 shows "42" no matter in which context it is executed.

			The third and fourth command would set the background color of Label2 to magenta, using the same logic.
	.GroupKeys		Example: `DebugMessage(Fader1.GroupKeys)` Returns a list of used Keys for the indicated widget. The example could have the result that the Debug Logger displays: [["192.168.1.100","192.168.1.200"]] or [["External","Internal"]].
	.GroupValues	.Property	Example: `DebugMessage(InputBox1.GroupValues.Text)` Returns a list containing all different values of the selected widget property at the group. The example could have the result that the Debug Logger displays: [["WD","web1","web2"]].

As all of these members return values like strings, doubles, list [1900], etc., the respective data type members [1914] can be used in addition to that.
E.g.: "`Fader5.GroupValues.Value.Avg`" returns the average fader value of all Group Values of this fader.

With those members, you can proceed scripting like before, you can set and retrieve any Group Value property like a normal property. You can store them in variables for evaluation or integrate them in your application. An extremely useful tool for a user login system is the Session Value [1913], of course it can also be used in combination with Group Values.

The Context object provides additional members for working with Keys and Session Values, you can e.g. find out which Client Key is used from which IP address or generate a list of Session Keys. Please refer to the chapter Project and Context Member [1906] for more information and examples.

Group Items

Each widget that is able to hold Group Values has a specific set of properties affected by the Group Values. They can be either read out or set like any other property.

Widget	Available Group Values
DropDown List [868]	Index
Fader [874]	Value
Encoder [877]	Value
ColorPicker [880]	SelectedColor
Wheel [883]	Value
InputBox [886]	Text
Label [888]	BackColor ForeColor Text Transparent
TextBox [923]	Text
Window [803]	PageName

Christie
Pandoras Box

Part 8

Hardware and Accessories

8 Hardware and Accessories

This topic focuses on the hardware description of Christie Pandoras Box products.

- Server Hardware 1939 including the LCD Menu 1942
- Compact Player 1944
- Input and Output Cards 1947 (graphics card, live inputs, etc.)

The following hardware devices are optional features. They can be purchased as stand-alone accessories.
- AirScan 1988
- Controller Boards 1991, i.e. Jog/Shuttle Controller 1991 and the Fader Extension 1993
- DMX Link 1999
- SMPTE Link 2000
- NET Link and Calibration Link 2000

The following devices are discontinued, but their description is still available:
- DMX Link 8 2009
- DVI and SDI Processor 2017
- EDID Link 2026
- SENSOR Link 2033
- SERIAL Link 2044

8.1 Server Hardware

The Pandoras Box Server hardware can be purchased with or without additional software, such as Pandoras Box [67] or Widget Designer [786]. Pandoras Box is a software and hardware based solution for show control and real-time video processing. The playback performance depends on the content format, its resolution and frame rate. The Download-Center includes a performance sheet listing many examples.

The Server hardware is 19" wide and 4 rack units high and includes server grade components built for 24/7 use. There are three Server hardware models, called performance kits PK1 - PK3, which define the amount of hard drives (SSD), hard drive space and CPU speed. The number of physical outputs is defined through the graphics card. Either three or four DisplayPort outputs are available and one additional USB-C output. A maximum of four outputs can be used per card simultaneously.

A LCD interface allows for basic setup of the server such as network, output configuration or to select a test pattern without the need of setting up a network before.

For more information regarding the Pandoras Box product structure, please see the chapter "Product Overview" [62] in the Pandoras Box manual. It also includes links to chapters that explain the different software solutions and possible hardware accessories. We offer specialized interfaces for SMPTE I/O, DMX, serial and sensor control.

Drivers

All drivers are preinstalled on the system. Also, software like the PB Menu is preinstalled. Please check the Download-Center for drivers, software updates and more documentation.

PB Menu

There is software pre-installed on Pandoras Box hardware called the PB Menu [2097]. It starts automatically when booting. It covers the Windows desktop and consists of a few buttons that give access to the most needed actions, e.g. starting the Pandoras Box Master or Client or opening the Windows Explorer.
The PB Menu includes another software called the Server Management Application [2109]. It allows to manage multiple hardware systems and connect via VNC to them.

Product Specifications

Operating system	Windows 10
Processor	XEON processor
SSD drives (for content)	PK1: 1 TB; PK2: 8 TB; PK3: 32 TB
RAID	RAID-0
Storage for operating system	480GB SSD
RAM	PK1: 24 GB; PK2: 48 GB; PK3: 48 GB
USB ports	8x USB ports (front: 2x USB 3.0, rear: 4x USB 3.0, 2x USB 3.1 (1x Type C)
LAN	1x 1GbE port + 1x 10GbE port + 1x IPMI port
Audio (onboard)	Optical S/PDIF output; 3x stereo playback output; 1x stereo playback input (Line-in) and 1x microphone input
Graphics card	NVIDIA Quadro P4000, RTX 4000 or RTX 6000
Product size (WxHxD)	422mm x 177mm x 596mm - without handles (19" wide and 4U high); 482mm x 177mm x 636mm - with handles
Weight	approx. 30kg (depends on performance kit and optional cards)
Power	100-240V AC, 12-6A, 47-63Hz with an integrated 850W power supply unit
Operating temperature	10° to 35°C / 50° to 95°F
Relative humidity	10% to 90% noncondensing

Optional cards (total number defined by space and PCIe slots):
- video input boards: HDMI 2.0, Display Port 1.4, 12G-SDI, 3G-SDI
- 8ch ADAT or 256ch Dante audio card
- Sync card (for Framelock and Genlock)
- second graphics card
- 10GbE Ethernet card

For more information see the chapter "Input and Output Cards [1947]" in the main Pandoras Box manual.

Hardware

Front view, connections from left to right

Power switch	The power switch allows powering on and off the system. Press the power button after connecting the supplied power cord.
2x LED Status Lights	The upper blue LED indicates the power status. The lower orange LED indicates access to hard drives.
2x USB 3.0 port	Connect USB devices such as external hard drives, SSDs and flash drives to this port. The USB 3.0 port supports SuperSpeed USB 5Gbps.
LCD interface + Rotary push button	The display allows for basic setup of the server such as network, output configuration or to select a test pattern without the need of setting up a network before. Use the rotary push button to navigate through the menu.

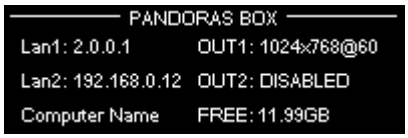
Rear view, connections from left to right, top to bottom

Power input 100-240V / 850W	Use the provided power cord to connect the server to a suitably rated grounded AC receptacle. Then press the power switch on the front panel to turn on the system.
1x 1GbE LAN port	Connect a network router, hub or switch or another computer to this port. The eight-pin RJ-45 LAN port supports standard Ethernet cables for connections to local area networks (LAN) with speeds of 100/1000 Mbps.
2x USB 3.1 port (1x Type C)	Connect USB devices such as external hard drives, SSDs and flash drives to this port. Both USB 3.1 ports support SuperSpeed USB 10Gbps.
1x COM port	Connect a serial device to this port.
1x VGA port	Onboard VGA port only for service, no Pandoras Box output!
1x IPMI port	Dedicated IPMI LAN Port
2x USB 3.0 ports	Connect USB devices such as external hard drives, SSDs and flash drives to this port. The USB 3.0 ports support SuperSpeed USB 5Gbps.
1x 10GbE LAN port	Connect a network router, hub or switch or another computer to this port. The eight-pin RJ-45 LAN port supports standard Ethernet cables for connections to local area networks (LAN) with speeds of 100/1000/10000 Mbps.
2x USB 3.0 ports	Connect USB devices such as external hard drives, SSDs and flash drives to this port. The USB 3.0 ports support SuperSpeed USB 5Gbps.
3x Audio Out jack 2x Audio In jack	Connect amplified speakers or headphones to the stereo output jack (3.5mm) and receive the system's audio output signal. There are three output connections: orange: Center / LFE, black: Rear / Surround, lime: Line Out / Front Connect a Line signal to the light blue stereo input jack (3.5mm). Connect a microphone signal to the pink stereo input jack (3.5mm).
1x S/PDIF Out	Connect digital audio receivers and speakers to the optical S/PDIF output.
3x / 4x DisplayPort 1.4	Connect a digital display device to the Display port (max. resolution: 3840x2160 @120Hz or 5120x2880 @60Hz). The number of available ports depends on the graphics card.
1x USB-C port	For the RTX4000 card, you can use this port with the shipped DisplayPort adapter. Alternatively, you can connect a monitor supporting the Vesa DisplayPort Over USB-C standard.

8.1.1 LCD Menu

The LCD Menu is part of any Pandoras Box Server ans cannot be purchased as an optional feature for other hardware.

Status Display



The top level menu of the front panel LCD display is the status display.

The status displays indicates:

The units IP address of the "Lan" adapter	Output1 Resolution
The units IP address of the "Lan2" adapter	Output2 Resolution
The units computer name	Free hard disc space

By pressing the encoder once you will enter the main menu.

Main Menu

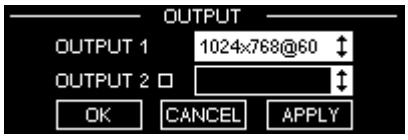
From the main menu you may access several sub menus: Use the encoder to navigate through the menu entries and press the encoder to enter each of the sub menus. Press STATUS to return to the top level status menu.

Lan1 & Lan2 - Network Menu

The network menu allows you to quickly change the unit's IP address and subnet mask settings as well as the DHCP enabled-setting of the unit.

Use the encoder to navigate through the network menu and turn the encoder to set the individual settings.
Please note that applying changes to the network setting may take several seconds until the unit is ready to proceed.
Press CANCEL to return to the main menu without changes.

Output Menu



The output menu gives you a direct access to the available default resolutions.

Use the encoder to navigate and choose from the list of display modes.
To enable Output2 please make sure that the check box next to Output2 is checked in order to setup both outputs.

Please note: The current version of Pandoras Box Server supports only matching resolutions on both outputs.

For more advanced resolution and refresh rate setups please use the onscreen menu to open the Output Nvidia Control Panel window.

Press CANCEL to return to the main menu without changes.

Pattern Menu

When working offline (i.e. neither Pandoras Box Master or Client is running) you may assign test-patterns to the outputs.

Press MENU to return to the main menu, the last selected pattern will stay active until it is changed or rebooted.

Playback Menu

The functions of the playback menu are enabled when Pandoras Box is in Master mode in order to control the timeline. When working in Master mode, the playback buttons are enabled as well.

- Press PLAY to run the timeline.
- Press STOP to reset the timeline to 00:00:00:00
- Press LAST to jump to the last cue available
- Press NEXT to jump to the next cue available
- Edit GOTO TIME lets you enter a specific timecode to jump to after you entered all value in hh:mm:ss:ff format
- Edit GOTO CUE lets you enter a specific cue number to jump to.
- Press MENU to return to the main menu.

Mode Menu

The mode menu defines which version is loaded for the onscreen menu buttons. It also determines which Master or Client mode to choose if multiple revisions are installed on the unit.

In the top scroll list you may choose one of the installed revisions.

Choose the mode you want to start by clicking on either Master or Client.

To start a chosen mode, select START.
To close a running Master or Client, choose CLOSE.

Press MENU to return to the main menu.

Display Menu

The display menu lets you setup both brightness and contrast of the LCD display.

Press MENU to return to the main menu

Shutdown Menu

The Shutdown Menu lets you either shutdown or reboot the unit.

Press MENU to return to the main menu.

8.2 Compact Player

The Pandoras Box Server hardware can be purchased with or without additional software, such as Pandoras Box [67] or Widget Designer [786]. Pandoras Box is a software and hardware based solution for show control and real-time video processing. The playback performance depends on the content format, its resolution and frame rate. The Download-Center includes a performance sheet listing many examples.

Its distinctive hardware feature is its size: the Compact Player is only one rack unit high and roughly half 19" wide.

For more information regarding the Pandoras Box product structure, please see the chapter "Product Overview" [62] in the Pandoras Box manual. It also includes links to chapters that explain the different software solutions and possible hardware accessories. We offer specialized interfaces for SMPTE I/O, DMX, serial and sensor control.

Drivers

All drivers are pre-installed on the system. Also, software like the PB Menu is pre-installed. Please check the Download-Center for drivers, software updates and more documentation.

PB Menu

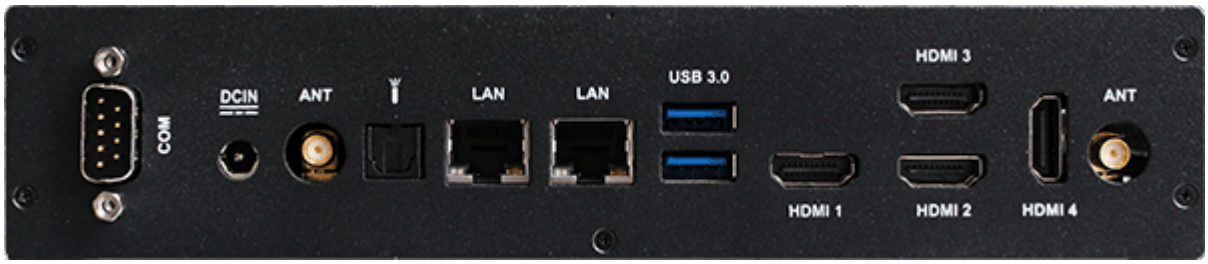
There is software pre-installed on Pandoras Box hardware called the PB Menu [2097]. It starts automatically when booting. It covers the Windows desktop and consists of a few buttons that give access to the most needed actions, e.g. starting the Master or Client software or opening the Windows Explorer (File Browser).
The PB Menu includes another software called the Server Management Application [2109]. It allows to manage multiple hardware systems and connect via VNC to them.

Product Specifications

Operating system: Windows 10
Microprocessor: Intel i5 processor
RAM: 8GB
SSD drives: depends on hardware edition, i.e. PK1 - 480GB, PK2 - 960GB, PK3 - 1920GB
USB ports: 3x USB 3.0 ports (1x front and 2x rear panel)
LAN: 2x 1 Gigabit Ethernet ports
WiFi: IEEE 802.11ac/a/b/g/n
Bluetooth: Bluetooth 4.0
COM: 1x D-Sub male DE-9 connector (RS-232)
Audio*: Optical S/PDIF output; stereo analog audio output and microphone input
Graphics card: NVIDIA Quadro P1000 with 4x HDMI 2.0
Product Size (WxHxD): 210mm x 44.4mm x 176.6mm
Power: DC 19.5V/120W with an external power supply unit

Front view, connections and LEDs from left to right

Power switch	The power switch allows powering on and off the system. Press the power button after connecting the supplied PSU to the power jack on the system rear panel. Note that the button's backlight LED lights up in red when the PSU is connected and in green when the system is switched on.
HDD LED	This LED indicates that the hard drive is accessed.
WiFi LED	This LED indicates that data is transmitted via WiFi.
USB 3.0 port	Connect USB devices such as external hard drives, SSDs and flash drives to this port. The USB 3.0 port supports SuperSpeed USB 3.0 devices and is backwards compatible with USB 2.0/1.1 devices.
Memory card slot	The built-in memory card reader reads SD/SDHC/SDXC cards.
Microphone jack	Connect a microphone to the microphone jack (3.5mm).
Headphone/ Audio Out jack	Connect amplified speakers or headphones to the stereo headphone jack (3.5mm) and receive the system's audio output signal.



Rear view, connections from left to right

COM port	Connect your serial device via a RS-232 cable.
Power input DC 19.5V/120W	Connect the supplied power adapter to this jack. Then press the power switch on the front panel to turn on the system. Please note: To prevent damage to the PC, always use the supplied power adapter. The power adapter may become warm to hot when in use. Do not cover the adapter and keep it away from your body.
2x Antenna	Connect the supplied SMA RF antennas for WiFi and Bluetooth to this and to the connector to the right of the HDMI ports.
Optical (S/PDIF out) jack	Connect digital audio receivers and speakers to the optical S/PDIF output.
2x Ethernet ports	Connect a network router, hub or switch or another computer to this port. The eight-pin RJ-45 LAN port supports standard Ethernet cable for connections to local area networks (LAN) with speeds of 10/100/1000Mbps.
2x USB 3.0 ports	Connect USB devices such as external hard drives, SSDs and flash drives to this port. The USB 3.0 port supports SuperSpeed USB 3.0 devices and is backwards compatible with USB 2.0/1.1 devices.
4x HDMI port	Connect a digital display device to the HDMI 2.0 port (max. resolution: 4096x2160 @60Hz).

8.3 Input and Output Cards

This chapter includes information about input and output boards for the Server hardware. The Server hardware itself is described in the chapter "Server Hardware" [1939]. If you are interested in other Christie Pandoras Box hardware, please see the introductory chapter Hardware and Accessories [1938].

Cards are either included by default or optional. Since version 8, the total number for the boards is not limited by the software anymore but by available space and PCIe ports.

For all cards it is recommended to check the Download-Center from time to time in order to always use the up-to-date driver.

Graphics Card

The graphics card is the only board included be default in every hardware. For the Server R5 hardware you can choose between three NVIDIA Quadro graphics card: P4000, RTX4000 or RTX 6000. Please see the chapter Graphics Card Settings [1948].

Optional Boards

There are various input and output boards available for our Server hardware.

Sync Card
The Sync Card [1981] allows for a frame lock option for frame and pixel line accurate synchronization. With the card you can either generate an internal sync or lock to an incoming signal.

Audio In- and Output Cards
Audio Cards allow for more audio input and output channels to be used in Pandoras Box. They also include a MIDI option. The cards offer various audio protocols and connections.
ADAT Audio Card [1982] (8 channels; 32ch is discontinued)
Dante Audio Card [1986] (256 channels)

Video Input Cards
3G-SDI Input Cards [1969] (Single, Dual and Quad)
12G-SDI Input Cards [1972] (12G/Quad 3G-SDI, HDMI/DP board)
Flex Input Cards [1975]
HDMI Input Card [1978] (Dual)

Ethernet Output Card
10GbE Ethernet Card [1987] (Dual)

The Single and Dual DVI Input Cards [2014] and MADI Audio Card [2031] are discontinued.

Custom Boards

If you prefer to install a custom board please bear in mind that hardware changes void the warranty. In addition, using Christie Pandoras Box Input and Output Cards guarantees the best software support especially in terms of synchronization and latency. Regarding other live inputs, please note that Pandoras Box supports only input devices that are conform and support DirectShow.

Using Live Inputs in Pandoras Box

Audio and Video Inputs can be used as audio and video sources in the PB project. Please see the chapter about the Assets tab [131] to learn how to import them to the project in order to play them on a Layer. In addition, Audio and Video Export [135] and Recording [137] is offered depending on your license [62].

8.3.1 Graphics Card Settings

This chapter is the introductory chapter for graphics card settings and includes:
- Changing the resolution [1949]
- Setting up multiple displays [1952] (Mosaic)
- Setting up frame lock [1961]
- Installing and setting up the graphics driver [1967]

These topics refer to the NVIDIA Control Panel as all Pandoras Box hardware is equipped with NVIDIA graphics cards.

The **PB Menu** allows to setup display output settings in a simple way, e.g. setting the resolution for each display. Simply click the Settings button and choose "Display Setup [2106]".

Another possibility to set the resolution is the **LCD Menu**. In case you are working with Server hardware that is equipped with a LCD Menu [1942], you may access basic resolution settings through the front panel i.e. without the NVIDIA Control Panel.

The **NVIDIA Control Panel** itself offers all settings including the more advanced ones, e.g. the way multiple displays should be addressed, frame lock options, etc. The available output settings depend on your graphics card model but in general they match those explained in the following chapters referring to the model "Quadro M4000". If you are not using the PB Menu which can open the control panel, open the right-click menu from the Windows desktop or the Start menu and choose the link from there.

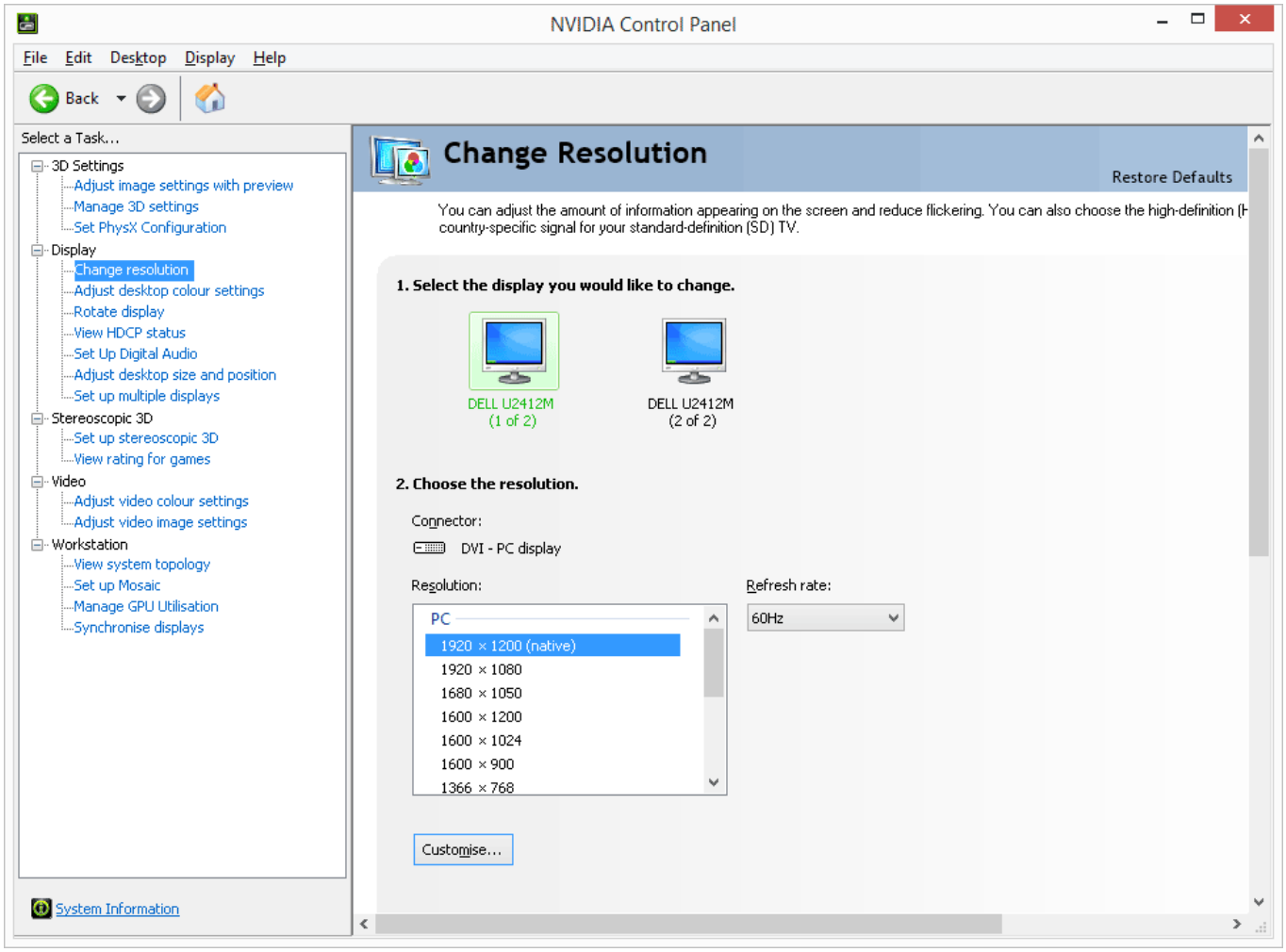
Do not disconnect any display cable from the graphics card during runtime!
When changing display cables directly on the graphics card, we recommend to always shut down the system first!

In addition it is required to close the Pandoras Box Client or Master software when modifying the display setting.
Restart Pandoras Box software after applying changes regarding the resolution or display mode.

8.3.1.1 Changing the Resolution

This topic explains how to change a resolution with the NVIDIA Control Panel [1948]. The next topic describes the setup of multiple displays [1952], synchronization [1961] and driver update [1967].
As with all display settings, we recommend to close Pandoras Box before applying any changes.

In the NVIDIA Control Panel you may choose from any of the reported display resolutions by selecting the "Change resolution" menu in the task tree on the left of the control panel. To change the resolution of a specific display, click on the monitor icon and then select one of the available resolutions below to set the desired resolution for the selected output.

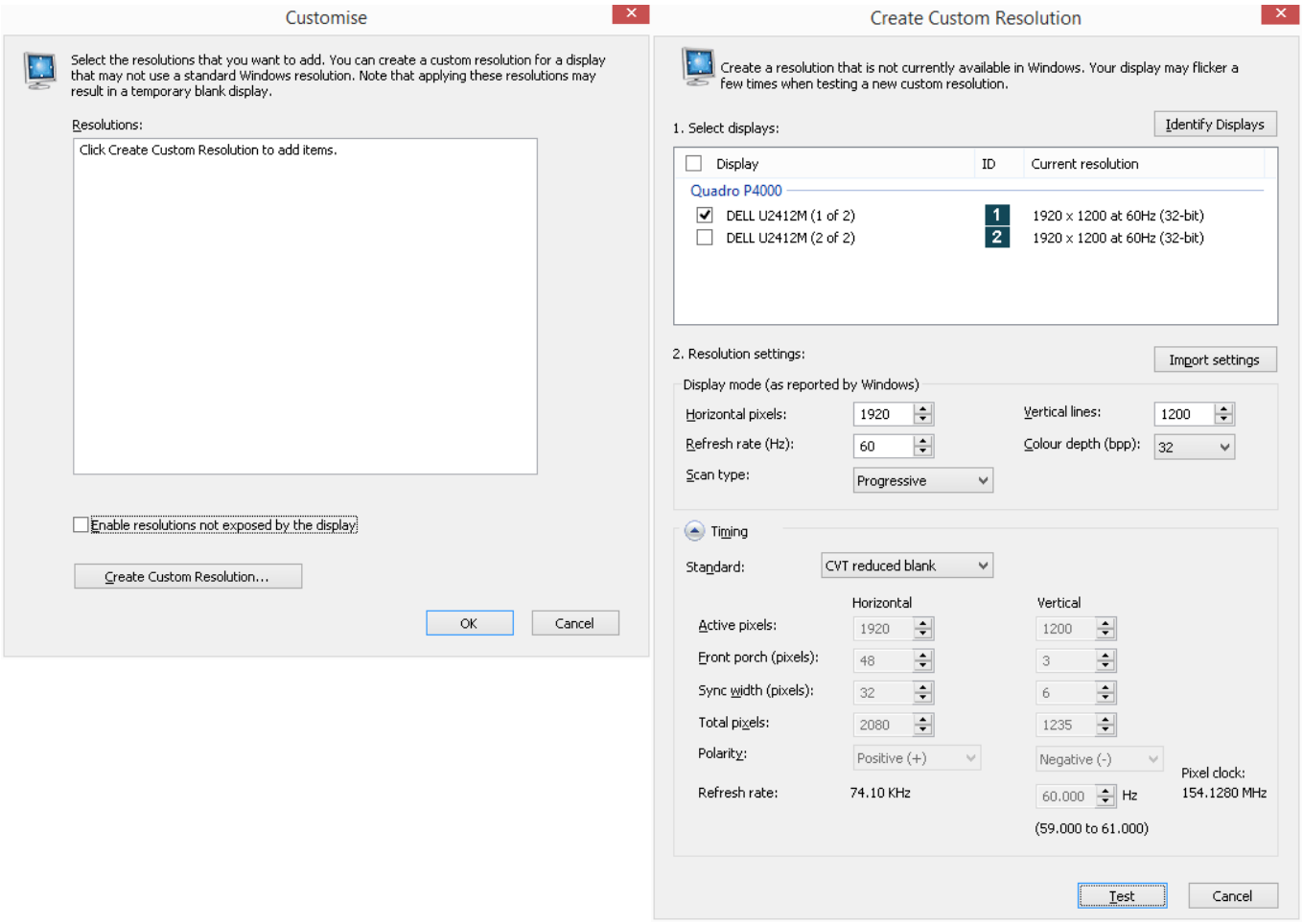


If the desired resolution is not listed in the change resolution menu, we recommend to use EDID managers or to load a supported edid file you saved before. A less recommended yet possible solution for some timings is to set up a so called "Custom Timing" if your display supports it.

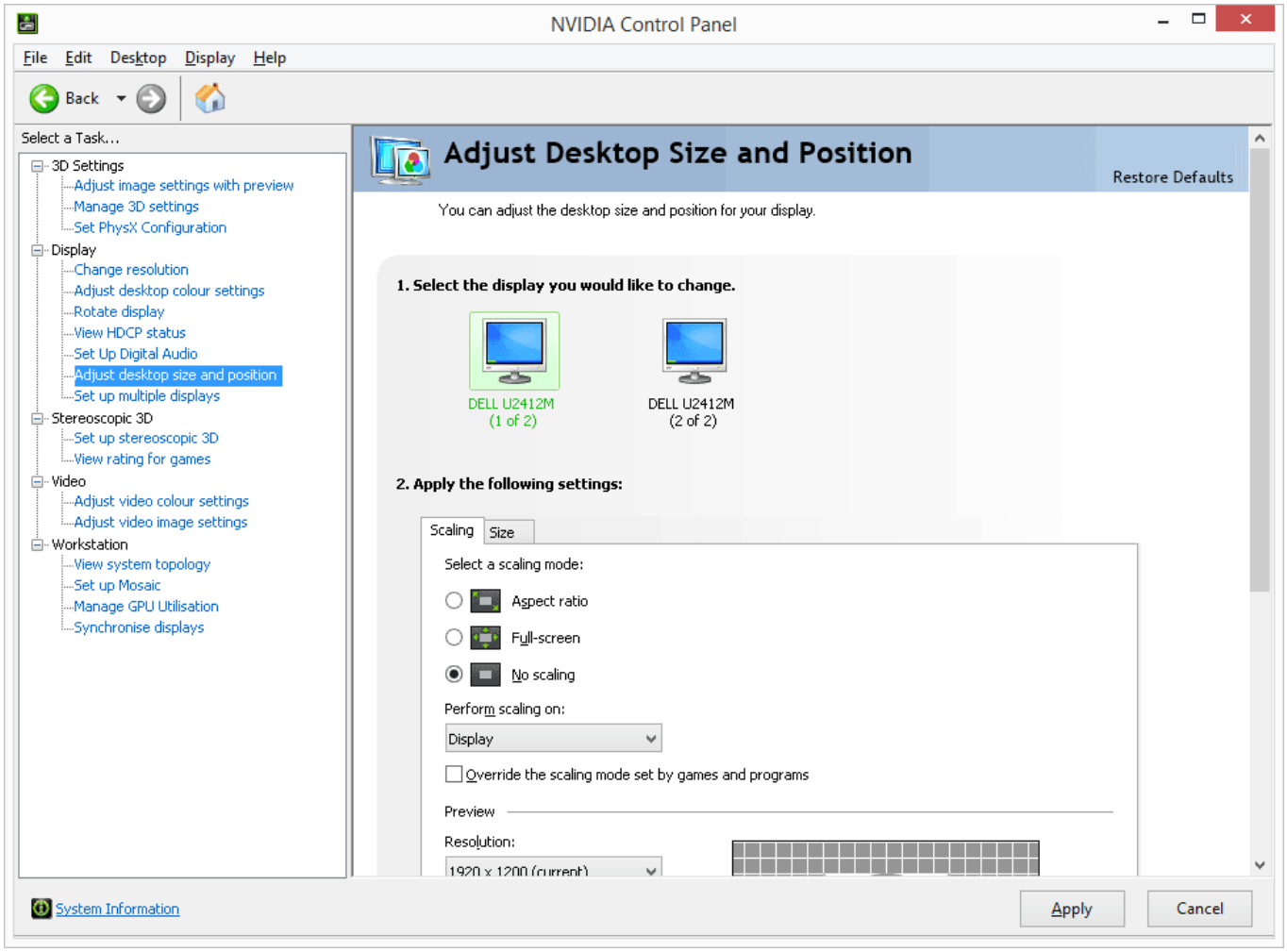
Please be aware that changing your graphics card to unsupported resolutions and frequencies might damage your display and graphics card. Christie can not be held liable for any damages that may occur from these custom setups. Instead of creating custom timings, we **strongly recommend** to use EDID managers to force the graphics card into the resolution you need. This is also due to the fact, that those custom timings are not supported by Mosaic and the Pandoras Box render engine needs more time to initialize. Eventhough the EDID Link hardware from coolux is discontinued, the chapter "EDID Link [2026]" explains EDIDs and EDID managers.

If you still like to setup a custom timing, click the "Customize" button below the "Resolution" list. Choose a resolution from the list or click the button "Create Custom Resolution".

In the first section, select the output you like to modify. Further down, in the "Resolution settings" section, you may setup the resolution and refresh rate. The timing details change automatically depending on the "Standard". If you changed it to "Manual" you could adopt these settings as well but this is only recommended for very advanced users.



For the menu "Adjust desktop size and position" it is recommended to set the scaling to "No Scaling" and disable any desktop resizing (in the Size tab).



8.3.1.2 Multiple Displays

This topic explains how to setup multiple displays with the NVIDIA Control Panel [1948]. The previous topic describes how to change a resolution [1949], the next ones how to frame lock displays [1961] and update the graphics driver [1967].
As with all display settings, we recommend to close Pandoras Box before applying any changes.

Separate Displays versus Mosaic

With Windows XP it was possible to set multiple displays to either "Horizontal span" or "DualView". With Windows 8.1, these terms have changed: "Horizontal span" is replaced by "Mosaic" whilst "DualView" has no name in the NVIDIA control panel so this manual describes the mode as "separate" displays. The meaning and recommendation for Pandoras Box however have not changed fundamentally.

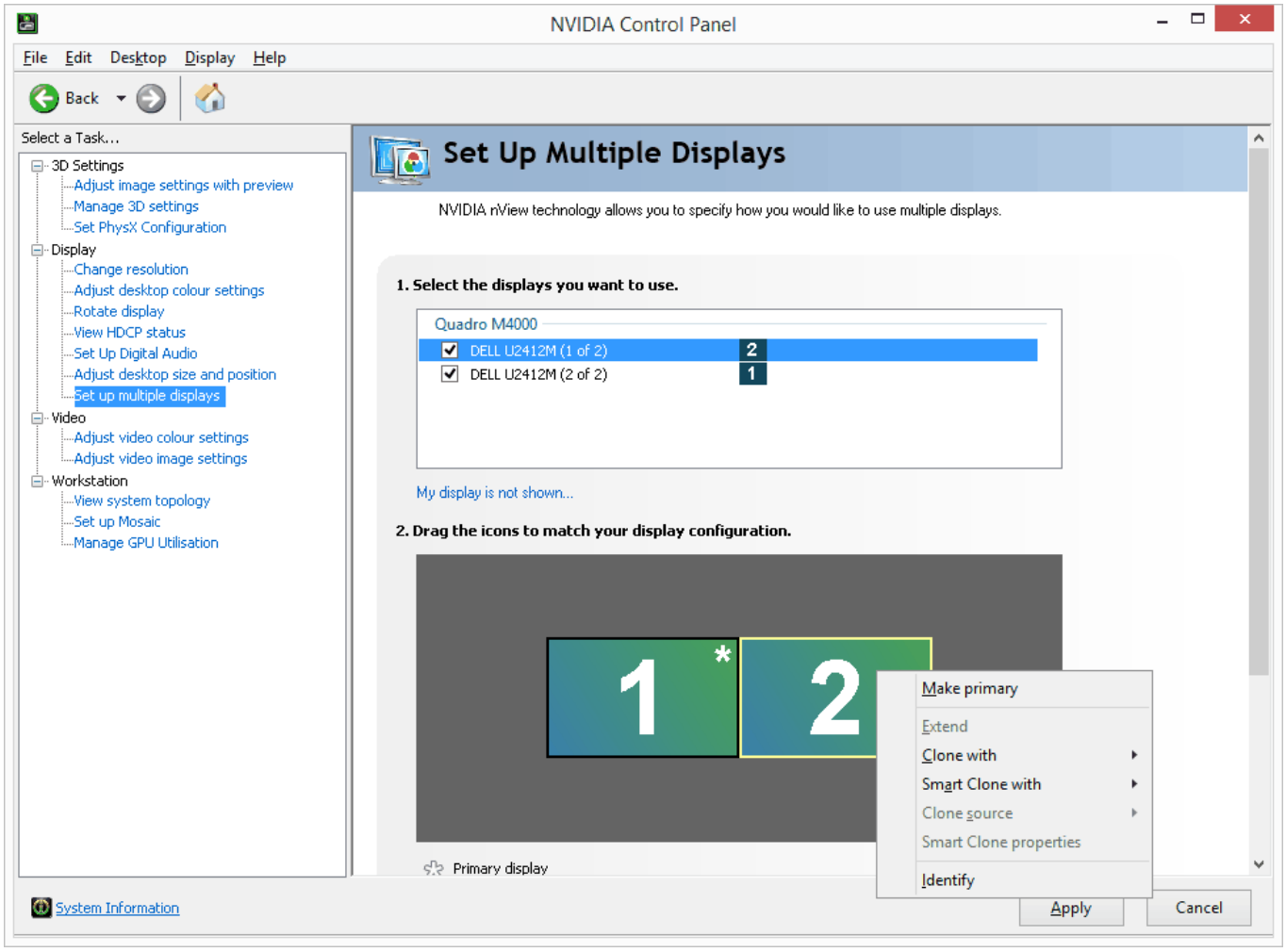
Mosaic is NVIDIA's software technology that abstracts multiple physical displays into a larger virtual display. In other words, it spans one desktop image across multiple displays. NVIDIA calls the result the Mosaic display and reports only one display with an accordingly large resolution to Windows and other application like Pandoras Box. In the tab Configuration > Render Engine [162] you will see only one output adapter. If you like you can subdivide it here again to have individual access. In regards to the image build-up, it might be of interest, that the (virtual) large display is also output like this. For example, if you combine outputs horizontally, the result is, that the graphics card writes each line across all outputs before switching to the next one. This is of advantage for some applications including (soft-edge) projections. Additionally, this means that the outputs are frame locked. The downside of this mode is, that Mosaic sets the same resolution and frame rate to all connected outputs, hence it is mandatory that there is at least one timing shared by all monitors and / or projectors. Lastly, please note that the Pandoras Box render engine can only render across multiple graphics cards (GPUs) if they are set up in Mosaic mode. And using Mosaic with multiple GPUs is only possible with a sync board [1981]. Hence, the OCTO Server, that is equipped with a sync card and two graphics cards, needs to be set up in Mosaic mode if you like to use all eight outputs!

In short: If you like to render fullscreen on multiple displays which are set up with the same resolution and refresh rate, it is recommended to use the Mosaic mode for them. This ensures synchronous playback and best video rendering quality.
For the OCTO Server (with two graphics cards) rendering fullscreen on both GPUs is only possible when they are part of the same Mosaic display.

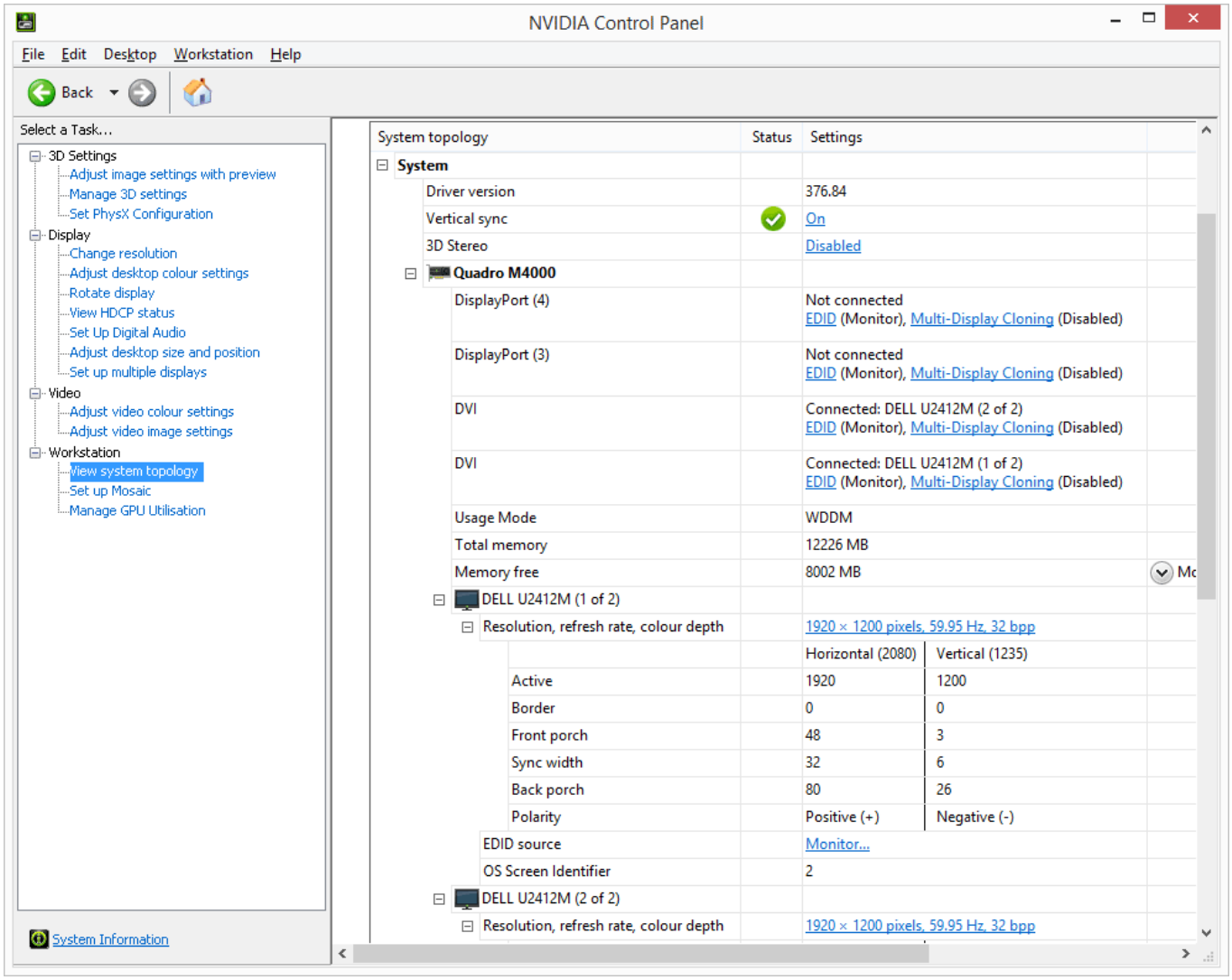
Without the Mosaic mode, the outputs count as separate displays that can be set to individual timings. For Compact Players [1944] and some custom hardware Mosaic is not available and this output setup is the only possibility.

Setting up Multiple Displays as Separate Displays

In the menu "Set up multiple displays" you can influence the layout of your screens or projectors. The rectangles depict an output and show its position according to the entire desktop size. You can drag the icon to a new position. Note that it makes sense to set up the resolution [1949] first, as the rectangle size depend on it.
You can also right-click an icon to choose for example to make it to the primary screen. This is the screen that displays the Windows desktop with its icons and the taskbar.

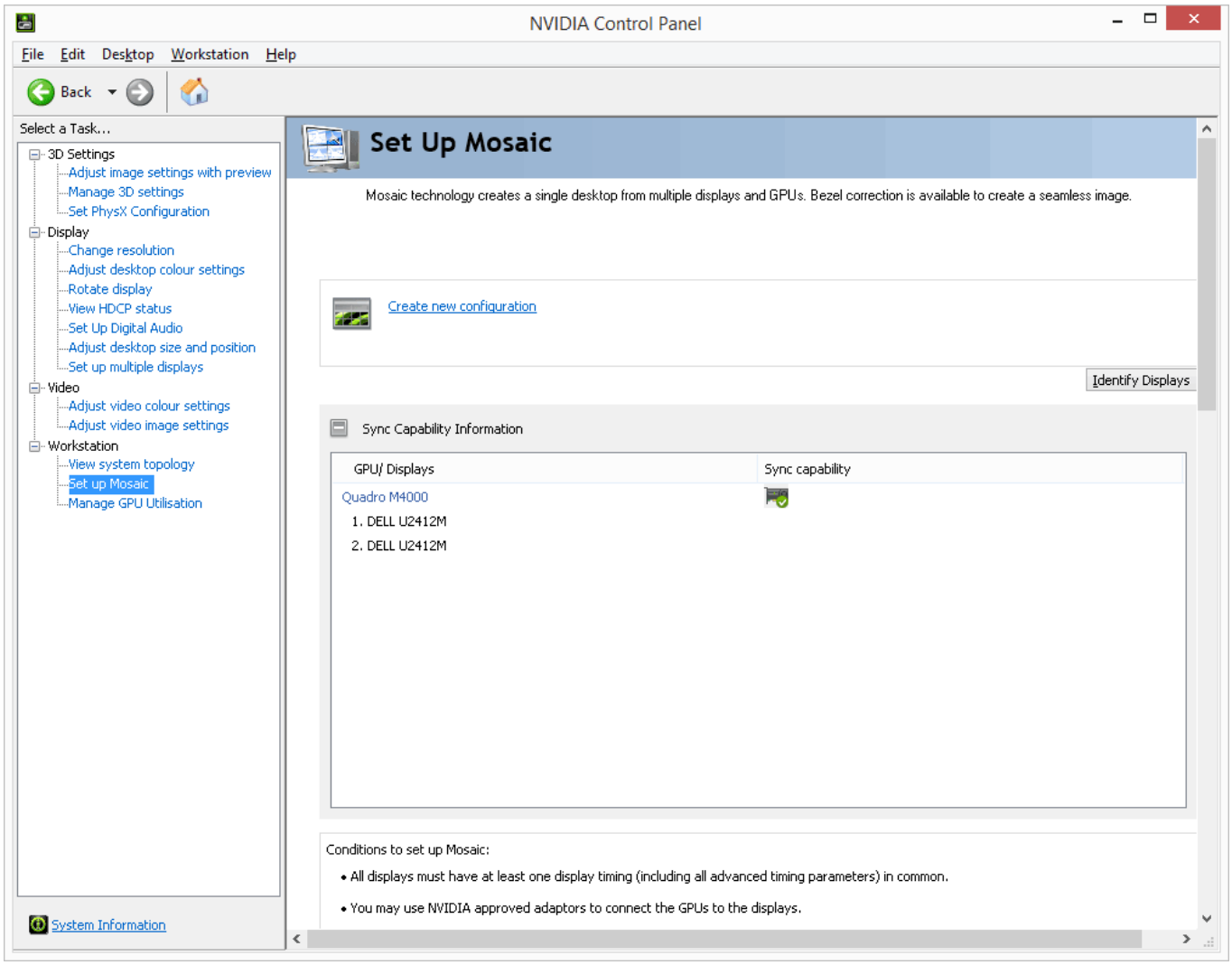

NVIDIA Control Panel
File Edit Desktop Display Help
Back
Select a Task...
3D Settings
Adjust image settings with preview
Manage 3D settings
Set PhysX Configuration
Display
Change resolution
Adjust desktop colour settings
Rotate display
View HDCP status
Set Up Digital Audio
Adjust desktop size and position
Set up multiple displays
Video
Adjust video colour settings
Adjust video image settings
Workstation
View system topology
Set up Mosaic
Manage GPU Utilisation
Set Up Multiple Displays
NVIDIA nView technology allows you to specify how you would like to use multiple displays.
1. Select the displays you want to use.
Quadro M4000
DELL U2412M (1 of 2)
DELL U2412M (2 of 2)
My display is not shown...
2. Drag the icons to match your display configuration.
1 *
2
Make primary
Extend
Clone with
Smart Clone with
Clone source
Smart Clone properties
Identify
Primary display
System Information
Apply
Cancel

In the menu "View system topology" you can check your resolution, frame rate, and output layout. With the above setup, it looks similar too this:

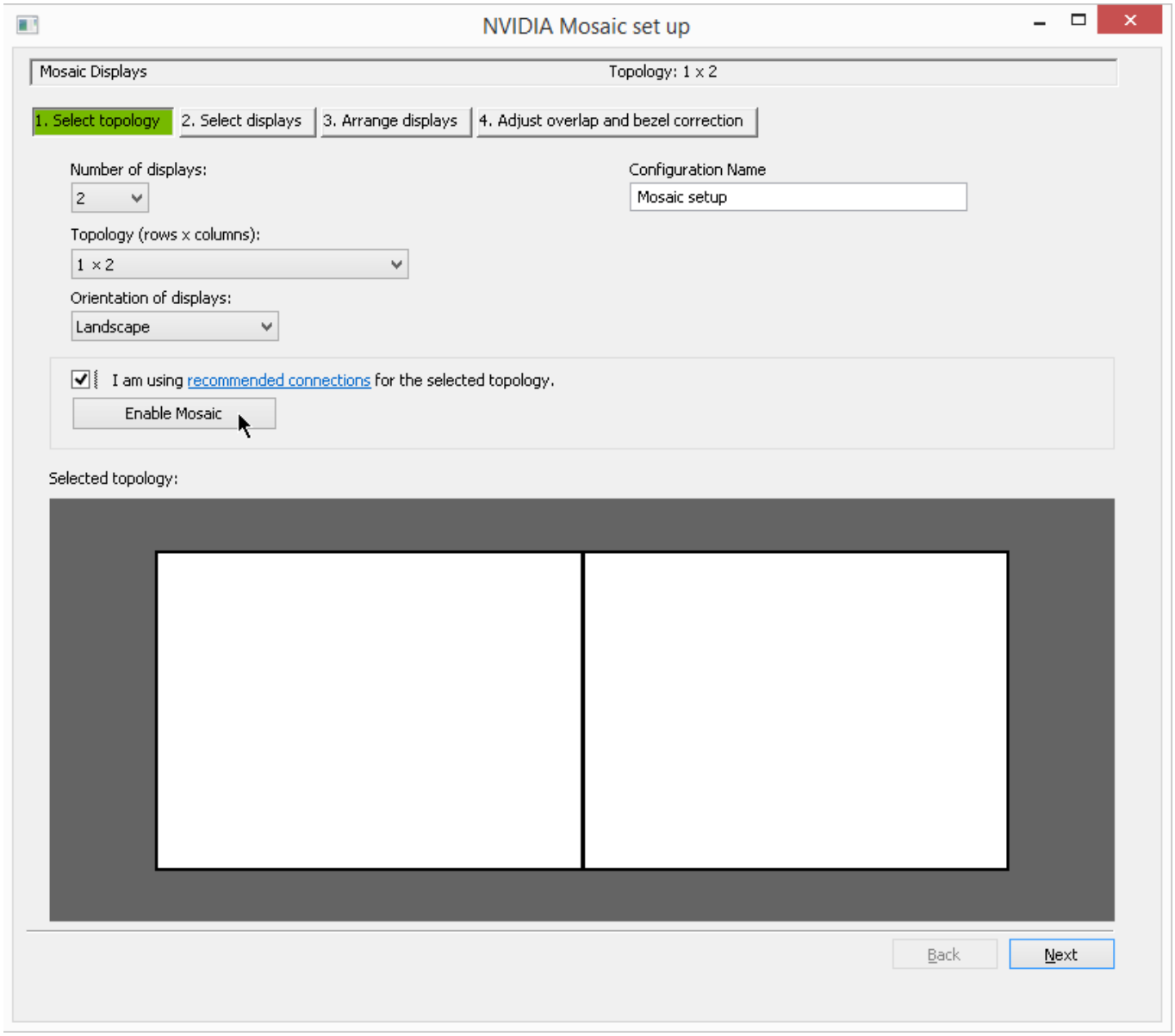


Setting up Multiple Displays in Mosaic Mode

Please select the menu "Set up Mosaic" and click on "Create new configuration".



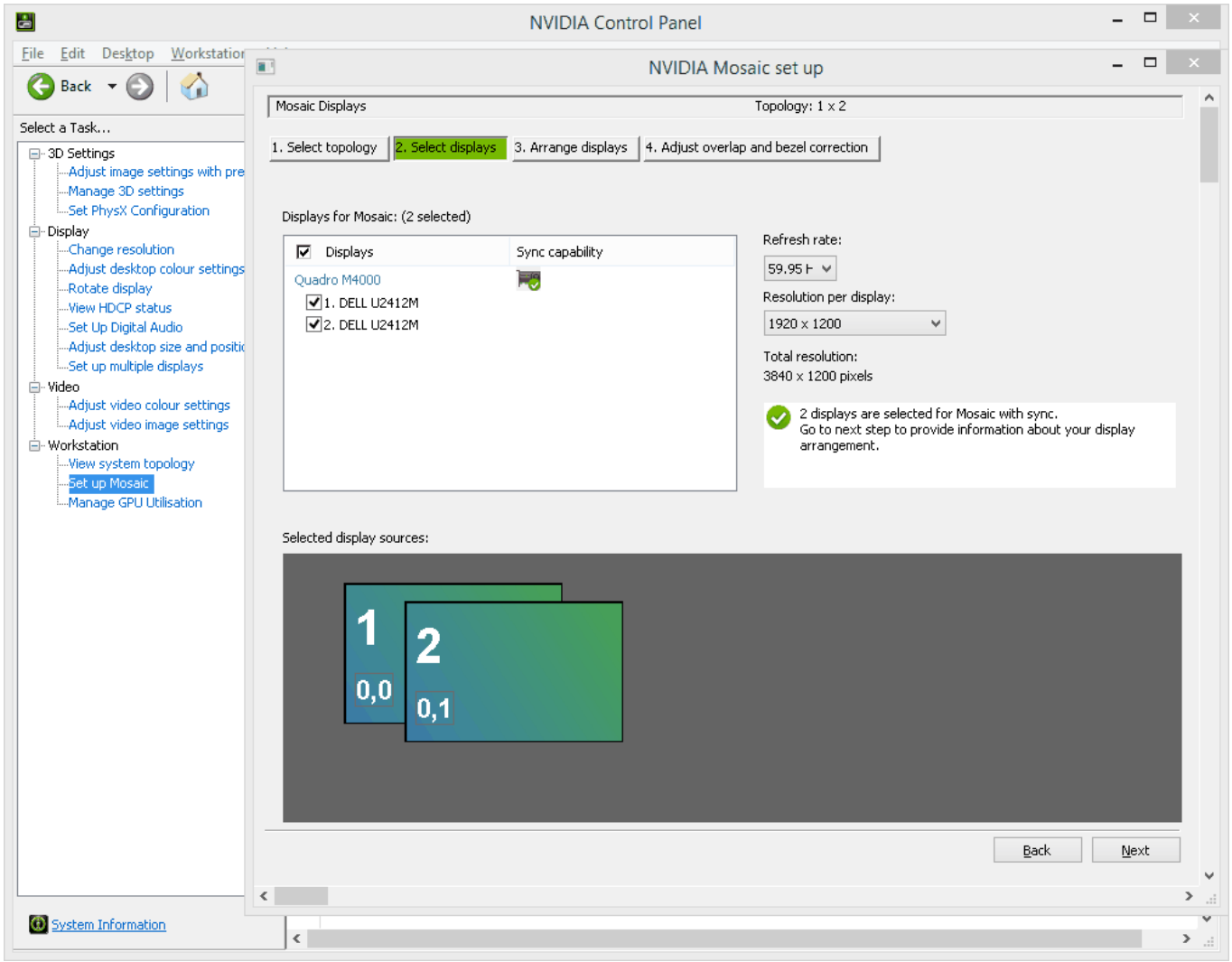
The next steps will define how the Mosaic mode combines the displays; or in other words how the virtual (Mosaic) display spans across the physical displays. First, select the number of displays and their layout to define the display topology, then confirm the note with the check box and click "Enable Mosaic".

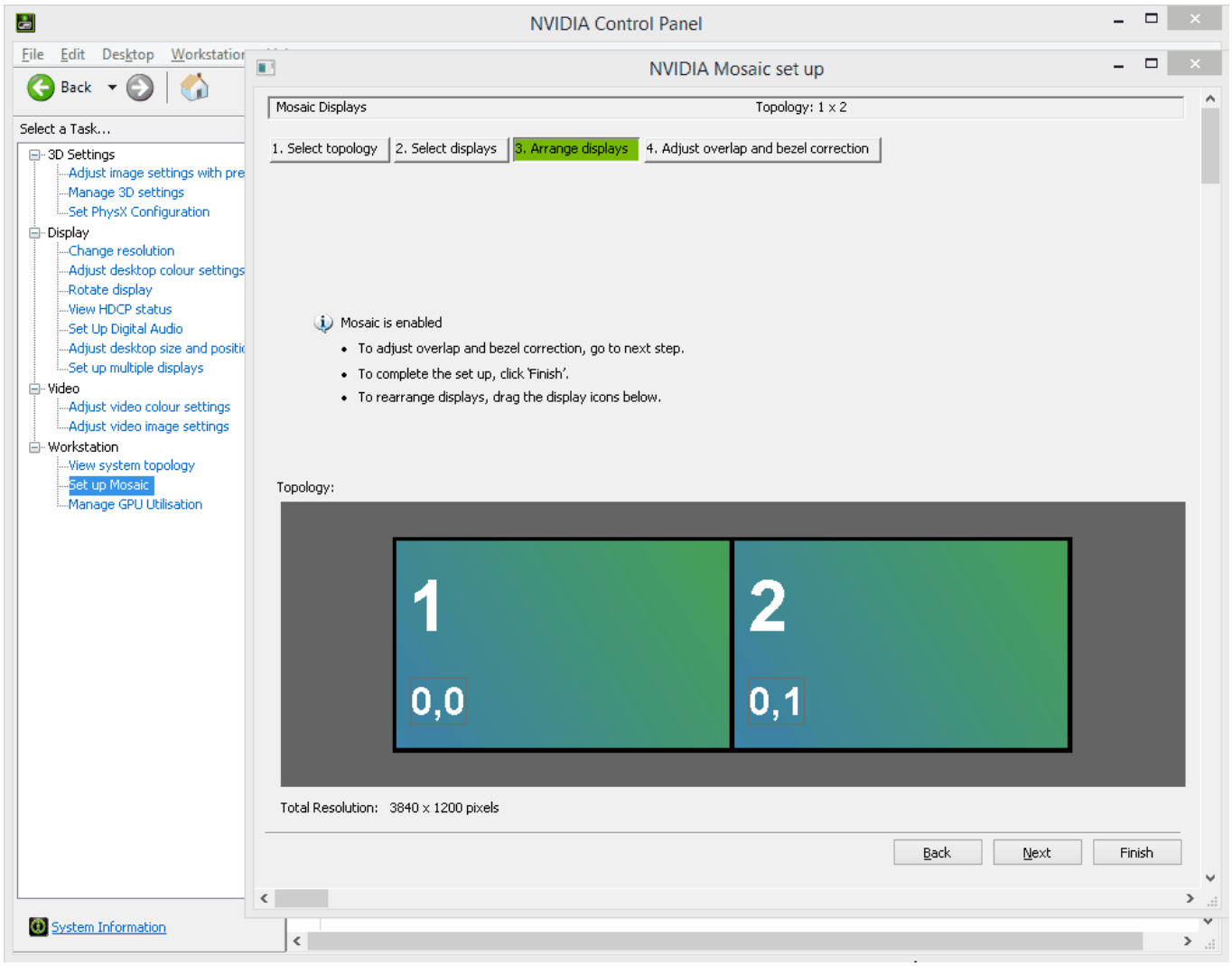


Now, you could already click the "Finish" button, or adjust the Mosaic mode with the tabs:
- 2. Select displays
- 3. Arrange displays
- 4. Adjust Overlap and bezel correction

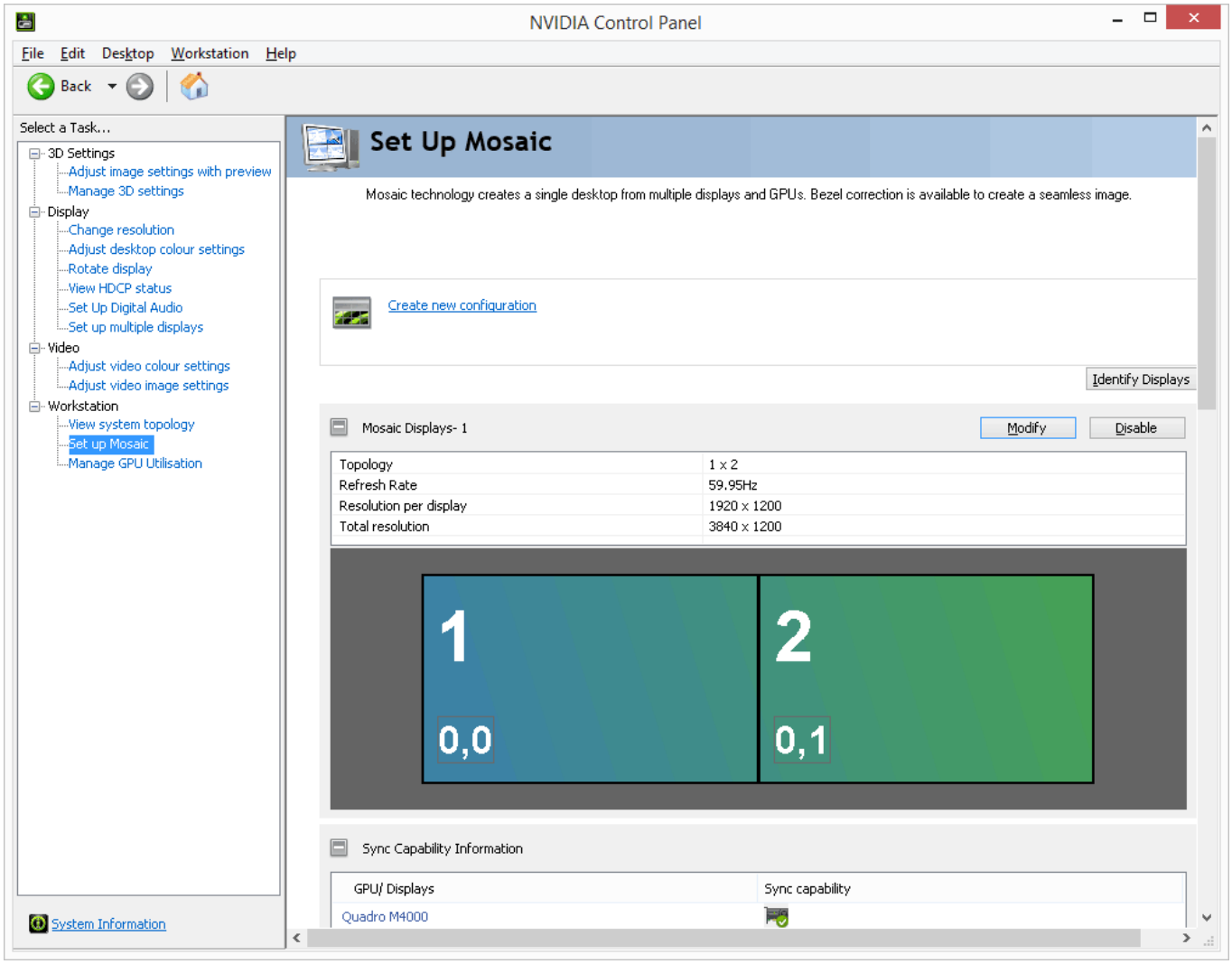
Switch the tabs using the "Next" and "Back" buttons.

Regarding the second tab, you might encounter difficulties selecting the displays, and unfortunately the information text is not always very accurate. If this is the case, the following steps might help solving the problem.
First bear in mind, that Mosaic sets the same resolution and frame rate to all displays, hence it is mandatory that there is at least one timing shared by all monitors and / or projectors. It is not necessary to set the resolution first (with the "Change resolution [1949]" menu) before activating the Mosaic mode. However, this could ensure, that the desired timing can be chosen for all displays. The "System topology" lists all displays with their current timing and proves to be a good overview. As with all EDID issues, EDID managers (like the discontinued EDID Link [2026]) help communicating EDIDs correctly.
If you cannot select all displays that are connected to one graphics card using different connector types (e.g. DVI and DisplayPort), try to adopt them so that you use one connector type only. And in case you are working with multiple displays connected to multiple graphics cards (e.g. the OCTO Server) try to connect them to the same ports. For example, if you use the first DVI port on the first GPU, make sure, that you are also using the first one on the second GPU instead of using another (free) port.

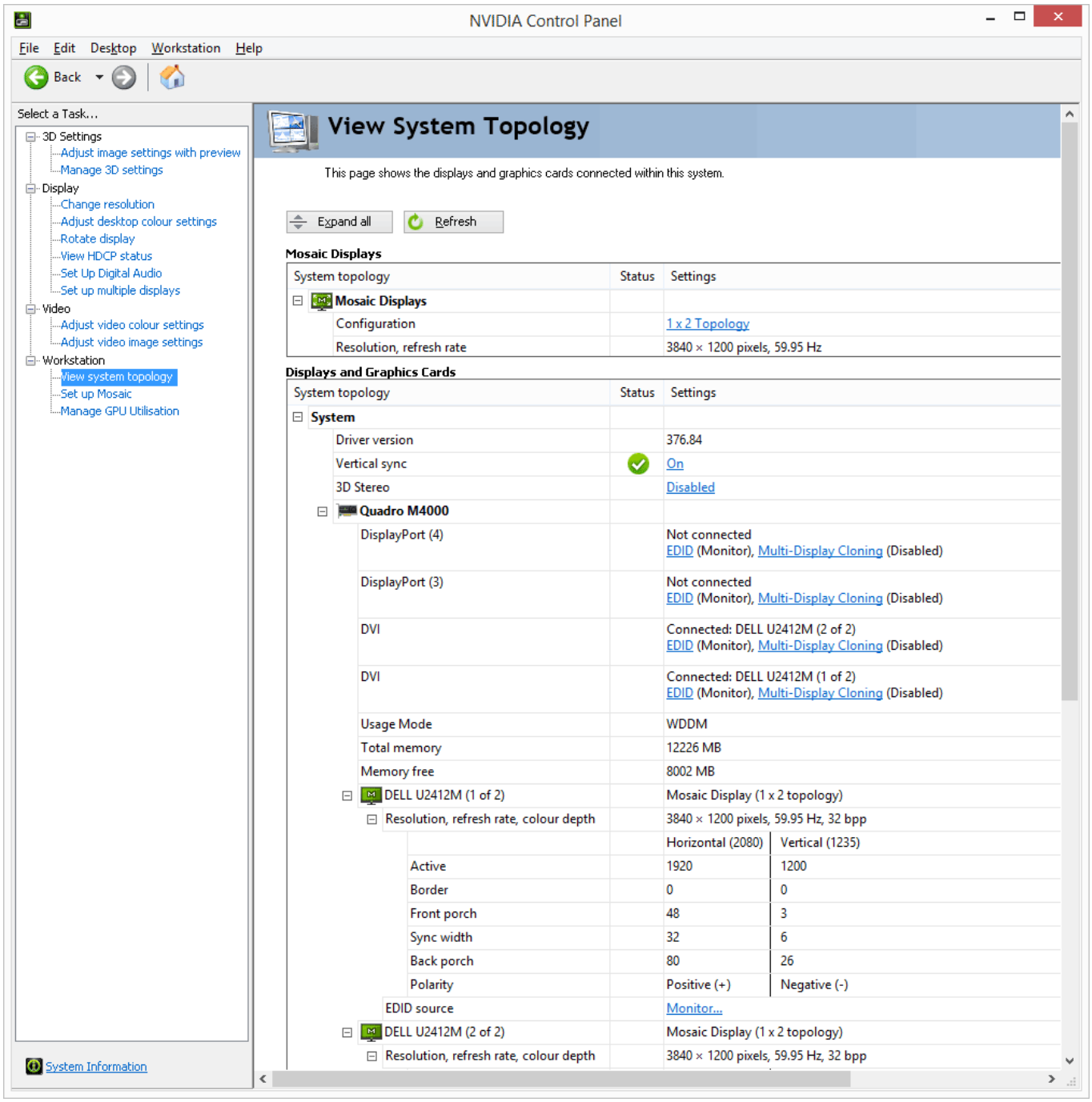



NVIDIA Control Panel
File Edit Desktop Workstation
Back
Select a Task...
3D Settings
Adjust image settings with pre
Manage 3D settings
Set PhysX Configuration
Display
Change resolution
Adjust desktop colour settings
Rotate display
View HDCP status
Set Up Digital Audio
Adjust desktop size and positic
Set up multiple displays
Video
Adjust video colour settings
Adjust video image settings
Workstation
View system topology
Set up Mosaic
Manage GPU Utilisation
System Information
NVIDIA Mosaic set up
Mosaic Displays
Topology: 1 x 2
1. Select topology
2. Select displays
3. Arrange displays
4. Adjust overlap and bezel correction
Mosaic is enabled
To adjust overlap and bezel correction, go to next step.
To complete the set up, click 'Finish'.
To rearrange displays, drag the display icons below.
Topology:
1
0,0
2
0,1
Total Resolution: 3840 x 1200 pixels
Back
Next
Finish

If the Mosaic mode is set up successfully, the menu "Set up Mosaic" will look like the below image. Note that the "Modify" button lets you change the current Mosaic mode and "Disable" brings you back to the state where each display is handled as a separate one.



With an activated Mosaic mode, the menu "View system topology" looks like this.



8.3.1.3 Setting up Frame Lock

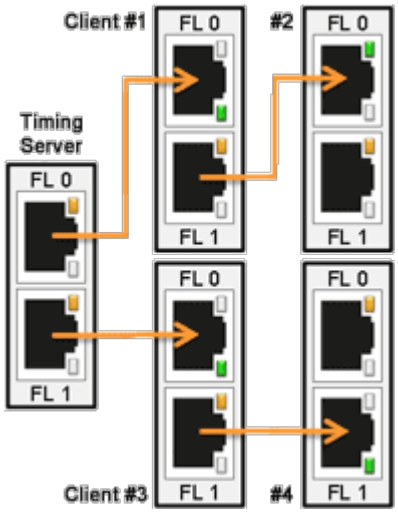
This topic explains how to setup the frame lock option for frame and pixel line accurate synchronization. Please refer to the main chapter if you are interested in other NVIDIA settings [1948] e.g. settings for resolution, multiple displays and the driver.
As with all display settings, we recommend to close Pandoras Box before applying any changes.

General Information Regarding Frame Lock

The frame lock option is for NVIDIA cards with a sync board [1981], e.g. the K5000 or M4000 and is available for Server hardware [1939]. These graphics cards have the possibility to synchronize to an external gen lock signal or to the sync signal generated by the sync board itself.

These are possible and recommended scenarios how to use the sync board:

- Multiple systems (i.e. two or more) without an external gen lock



In a network with multiple systems, set up one system to be the timing master [1961] and all others to be timing slaves [1964]. Both frame lock connectors (RJ-45) on the timing master, also called timing server, will output the sync signal. On the timing slaves, the top connector is the frame lock input and the second can be used as an output to daisy-chain the signal to other timing clients. The left image from the sync card manual [1981] shows the cabling.

In case you have one system only and want to be sure, that all outputs are synced, please use the Mosaic mode [1955] as described in the previous chapter.

- One or multiple system(s) listening to an external gen lock signal generated by another third-party device
In this case, connect the sync signal via the BNC connector on the sync board to each system and set up each system to receive an external sync signal [1966].

In any case, the sync signal must fit to the frame rate that is set up in the graphics card [1949], otherwise the timings can not be synced.
Then the below described driver settings are necessary to set up the graphics card. It will then clock to the incoming signal. This procedure might take up to 10 minutes as the card's clock needs to shift step by step towards the external clock.

For your information, if you need frame locked inputs, you need to feed the sync signal to the device that produced the live feeds and make sure that the live inputs are fed frame locked into Pandoras Box.

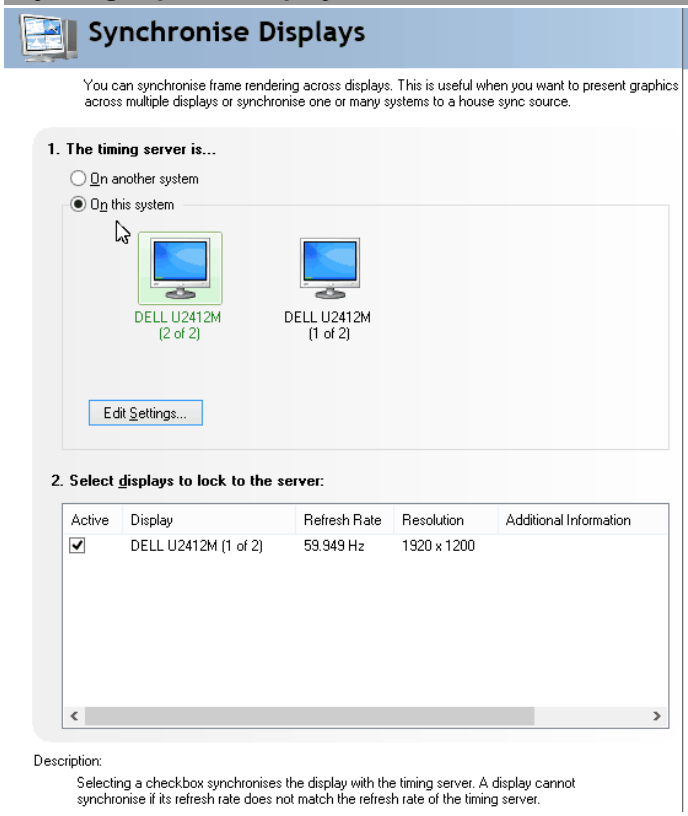
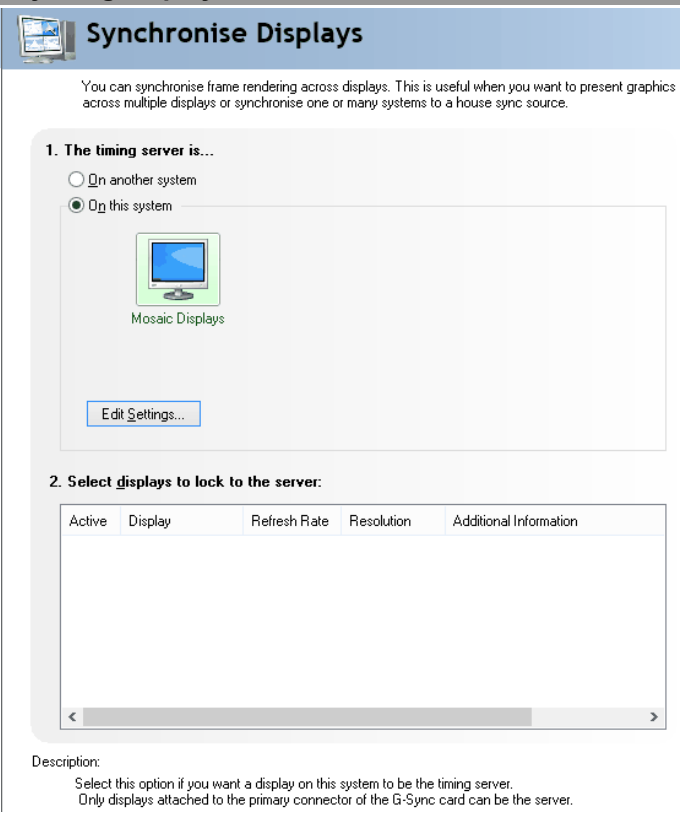
Setting up the Timing Master

After setting up a timing master, the sync card can output the sync signal which can then be used in other systems.

The master provides an external sync signal via the RJ-45 plug on its sync board. Connect the board's output with an ethernet cable to the input of the next Server, which is then the slave. You may daisy-chain the sync signal to all slaves as depicted above.

Open the NVIDIA Control Panel as described in the first chapter [1948] and then the menu "Synchronize displays".

If you have multiple displays attached to one graphics card, they can be set up as separate outputs [1952] or can be combined using Mosaic mode [1955]. In both cases, choose in the "Synchronize displays" menu that the timing server is "On this system". Without the mosaic mode, select in the second section that all displays should be locked. With the Mosaic mode, you cannot choose other displays as they are already combined.

Syncing separate displays	Syncing displays in Mosaic mode
	

This is how the menu "View system topology" looks before and after setting up the timing master, first for the setup with two separate displays, and below the setup using Mosaic mode.

No Sync activated for separate displays

System topology	Status	Settings	
System			
Driver version		385.69	
Vertical sync	✓	3D Application controlled	
3D Stereo		Disabled	
Quadro Sync II			
Framelock 0		Not used	
Framelock 1		Not used	
External sync signal		Not present	
Framelock sync pulse	⚠	Not present	
Sync settings		Synchronise Displays	
Quadro P4000			
DisplayPort (4)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (3)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (1 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (2 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
Usage Mode		WDDM	
Total memory		24550 MB	
Memory free		7880 MB	
DELL U2412M (2 of 2)			
Display state		Not synced	
Resolution, refresh rate, colour de...		1920 × 1200 pixels, 59.949 Hz, 32 bpp	
		Horizontal (2080)	Vertical (1235)
Active		1920	1200
Border		0	0
Front porch		48	3
Sync width		32	6
Back porch		80	26
Polarity		Positive (+)	Negative (-)
Timing		The display is using internal timing.	
EDID source		Monitor...	
OS Screen Identifier		2	
DELL U2412M (1 of 2)			

Sync activated for separate displays

System topology	Status	Settings	
System			
Driver version		385.69	
Vertical sync	✓	3D Application controlled	
3D Stereo		Disabled	
Quadro Sync II (server)			
Framelock 0		Out	
Framelock 1		Out	
External sync signal		Not present	
Framelock sync pulse	✓	Present	
Sync settings		Synchronise Displays	
Quadro P4000			
DisplayPort (4)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (3)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (1 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (2 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
Usage Mode		WDDM	
Total memory		24550 MB	
Memory free		7880 MB	
DELL U2412M (2 of 2)			
Display state		Client	
Resolution, refresh rate, colour de...		1920 × 1200 pixels, 59.949 Hz, 32 bpp	
		Horizontal (2080)	Vertical (1235)
Active		1920	1200
Border		0	0
Front porch		48	3
Sync width		32	6
Back porch		80	26
Polarity		Positive (+)	Negative (-)
Timing	✓	The display is locked to the frame lock sync pulse.	
EDID source		Monitor...	
OS Screen Identifier		2	
Stereo sync	✓	Client Stereo is in phase with the server.	
DELL U2412M (1 of 2)			

No Sync activated for displays in Mosaic mode

Mosaic Displays

System topology	Status	Settings
Mosaic Displays		
Configuration		1 x 2 Topology
Resolution, refresh rate		3840 × 1200 pixels, 59.95 Hz
Display Sync State		Quadro Sync II Not synced
Timing		The display is using internal timing.
OS Screen Identifier		1

Sync activated for displays in Mosaic mode

Mosaic Displays

System topology	Status	Settings
Mosaic Displays		
Configuration		1 x 2 Topology
Resolution, refresh rate		3840 × 1200 pixels, 59.95 Hz
Display Sync State		Quadro Sync II Server
Timing		The display is locked to an internal timing signal.
OS Screen Identifier		1

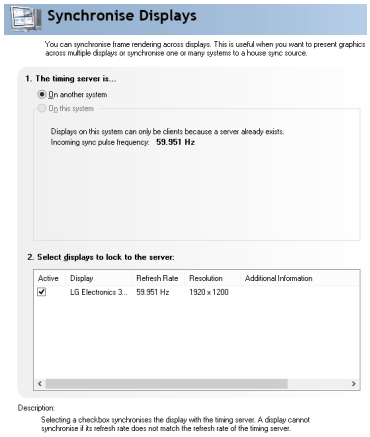
System topology	Status	Settings	
System			
Driver version		385.69	
Vertical sync	✓	3D Application controlled	
3D Stereo		Disabled	
Quadro Sync II			
Framelock 0		Not used	
Framelock 1		Not used	
External sync signal		Not present	
Framelock sync pulse	⚠	Not present	
Sync settings		Synchronise Displays	
Quadro P4000			
DisplayPort (4)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (3)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (1 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (2 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
Usage Mode		WDDM	
Total memory		24550 MB	
Memory free		7880 MB	
DELL U2412M (2 of 2)		Mosaic Display (1 x 2 topology)	
Display state		Not synced	
Resolution, refresh rate, colour de...		3840 × 1200 pixels, 59.949 Hz, 32 bpp	
		Horizontal (2080)	Vertical (1235)
Active		1920	1200
Border		0	0
Front porch		48	3
Sync width		32	6
Back porch		80	26
Polarity		Positive (+)	Negative (-)
Timing		The display is using internal timing.	
EDID source		Monitor...	
DELL U2412M (1 of 2)		Mosaic Display (1 x 2 topology)	

System topology	Status	Settings	
System			
Driver version		385.69	
Vertical sync	✓	3D Application controlled	
3D Stereo		Disabled	
Quadro Sync II (server)			
Framelock 0		Out	
Framelock 1		Out	
External sync signal		Not present	
Framelock sync pulse	✓	Present	
Sync settings		Synchronise Displays	
Quadro P4000			
DisplayPort (4)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (3)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (1 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
DVI		Connected: DELL U2412M (2 of 2) EDID (Monitor), Multi-Display Cloning (Disabled)	
Usage Mode		WDDM	
Total memory		24550 MB	
Memory free		7863 MB	
DELL U2412M (2 of 2)		Mosaic Display (1 x 2 topology)	
Display state		Server	
Resolution, refresh rate, colour de...		3840 × 1200 pixels, 59.949 Hz, 32 bpp	
		Horizontal (2080)	Vertical (1235)
Active		1920	1200
Border		0	0
Front porch		48	3
Sync width		32	6
Back porch		80	26
Polarity		Positive (+)	Negative (-)
Timing		The display is locked to an internal timing signal.	
EDID source		Monitor...	
DELL U2412M (1 of 2)		Mosaic Display (1 x 2 topology)	

Setting up the Timing Slave

After setting up a timing slave, the sync card locks to the sync signal received from another system.

Open the NVIDIA Control Panel as described in the first chapter [1948] and then the menu "Synchronize displays". Choose that the timing server is "On another system" and activate your outputs in the second section.



This is how the menu "View system topology" looks before and after setting up the timing master.

No Sync activated for one display

System topology	Status	Settings	
System			
Driver version		376.84	
Vertical sync	✓	On	
3D Stereo		Disabled	
Quadro Sync (client)			
Framelock 0		In	
Framelock 1		Out	
External sync signal		Not present	
Framelock sync pulse	✓	Present	
Stereo sync	⚠	Stereo is not in phase with the server	
Sync settings		Synchronise Displays	
Quadro M4000 (1 of 2)			
DisplayPort (4)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (3)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (2)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (1)		Connected: LG Electronics 31MU97	
SLI Mode		Disabled	
Usage Mode		WDDM	
Total memory		24518 MB	
Memory free		7977 MB	
LG Electronics 31MU97			
Display state		Not synced	
Resolution, refresh rate, colour de...		1920 × 1200 pixels, 59.951 Hz, 32 bpp	
		Horizontal (2720)	Vertical (1481)
Active		2560	1440
Border		0	0
Front porch		48	3
Sync width		32	5
Back porch		80	33
Polarity		Positive (+)	Negative (-)
Timing		The display is using internal timing.	
EDID source		Monitor...	
OS Screen Identifier		1	

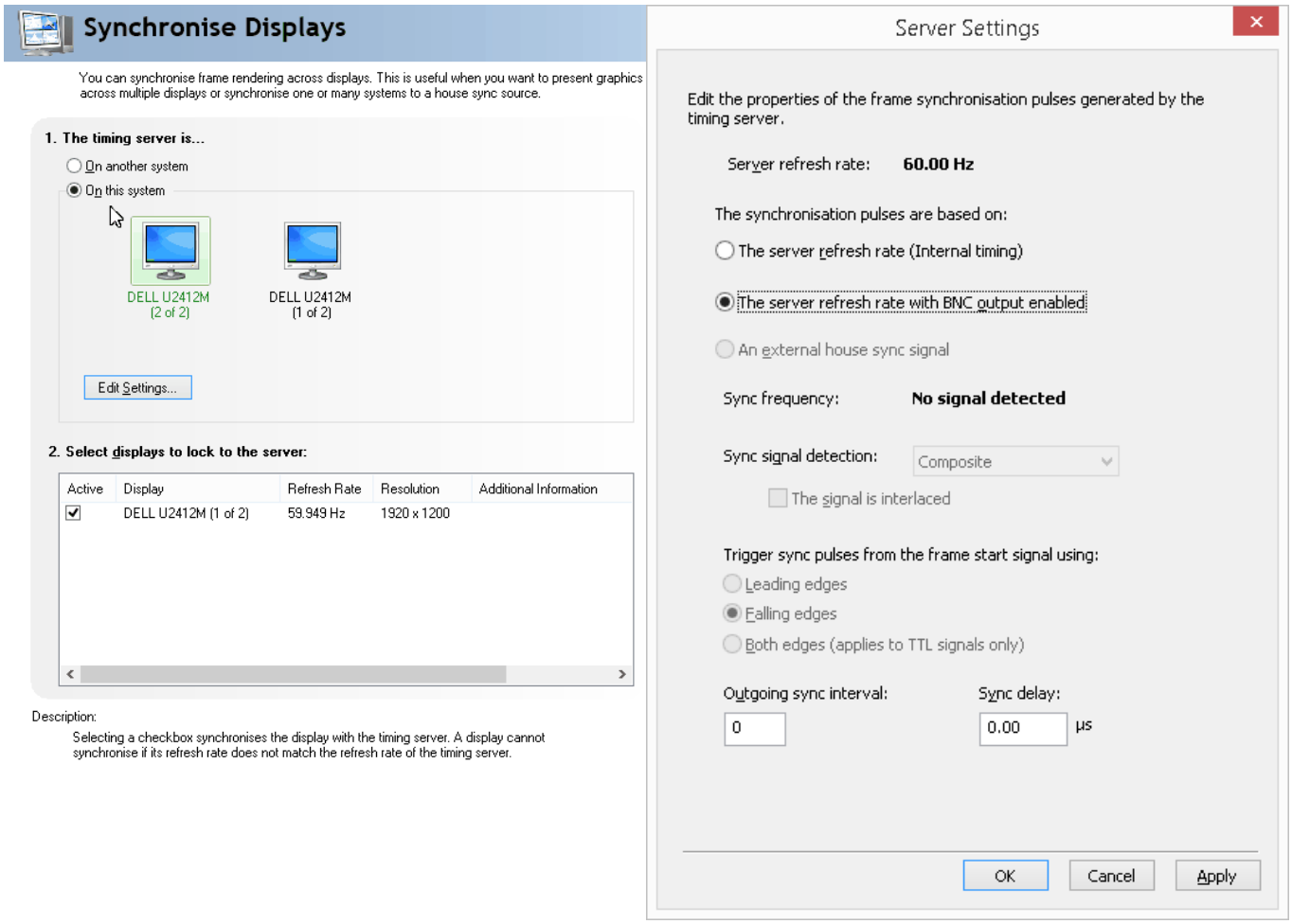
Sync activated for one display

System topology	Status	Settings	
System			
Driver version		376.84	
Vertical sync	✓	On	
3D Stereo		Disabled	
Quadro Sync (client)			
Framelock 0		In	
Framelock 1		Out	
External sync signal		Not present	
Framelock sync pulse	✓	Present	
Stereo sync	✓	Stereo is in phase with the server	
Sync settings		Synchronise Displays	
Quadro M4000 (1 of 2)			
DisplayPort (4)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (3)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (2)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)	
DisplayPort (1)		Connected: LG Electronics 31MU97	
SLI Mode		Disabled	
Usage Mode		WDDM	
Total memory		24518 MB	
Memory free		7977 MB	
LG Electronics 31MU97			
Display state		Client	
Resolution, refresh rate, colour de...		1920 × 1200 pixels, 59.951 Hz, 32 bpp	
		Horizontal (2720)	Vertical (1481)
Active		2560	1440
Border		0	0
Front porch		48	3
Sync width		32	5
Back porch		80	33
Polarity		Positive (+)	Negative (-)
Timing	✓	The display is locked to the frame lock sync pulse.	
EDID source		Monitor...	
OS Screen Identifier		1	
Stereo sync	✓	Client Stereo is in phase with the server.	

Receiving Gen Lock from another Device

After setting up a gen lock, the sync card locks to the sync signal received from another device.

Open the NVIDIA Control Panel as described in the first chapter [1948] and then the menu "Synchronize displays". Choose that the timing server is "On this system" and activate your outputs in the second section. Then click the button "Edit Settings" to set the properties of the frame synchronization pulses generated by the timing server.



- choose the setting "An external house sync signal" as the basis for the synchronization pulses
- make sure that the sync frequency matches with the frame rate
- only if needed, change the drop-down list for the sync signal detection and tick the check box whether the "signal is interlaced"
- choose how the sync pulses from the frame start signal should be triggered: leading -, falling -or both edges
- in most cases the outgoing sync interval is 0
- set the correct Sync delay, then "Apply" the settings to verify their correctness
- the sync delay depends on your setup and the current graphics card clock. It is likely you will have to try out different delays. As experienced, it is mostly 300-500µs, you may start with 300µs and increase in 50µs steps if it was not correct. Mind that it takes the system up to 10 minutes to reach the set clock.

Last step: You may check the "View system topology" menu whether all settings are correct:
- the external sync signal should be present and in use
- the frame lock sync pulse should be present
- the first display should be listed with "Output (1 of 2)", the second with "Output (2 of 2)"
- the first output's display state should be "Server" and its timing depends on the external house sync signal
- the second output's display state should be "Client" and its timing depends on the internal frame lock sync pulses

8.3.1.4 Graphics Driver

This topic explains how to install and setup a new graphics driver and how to check your current driver version. Please refer to the main chapter if you are interested in other NVIDIA settings [1948].

Installing a Graphics Driver

The graphics card driver on your Christie hardware product is tested and certified by us for best video rendering and playback performance. Even if the installed driver might not be the newest one available from the NVIDIA website it is NOT recommended to update this driver to their latest one.

If it is necessary or you were told by our support team to reinstall or update the graphics card driver please follow these steps:

1) Download the driver according to your hardware version. Please refer to the our Download-Center for recommended driver versions.

2) Run the driver setup in user custom (advanced) mode.

3) Deselect the options for NVIDIA update and nView but select the option for "Perform a clean installation"

4) After the installation is finished please reboot the computer, also if it does not do it automatically

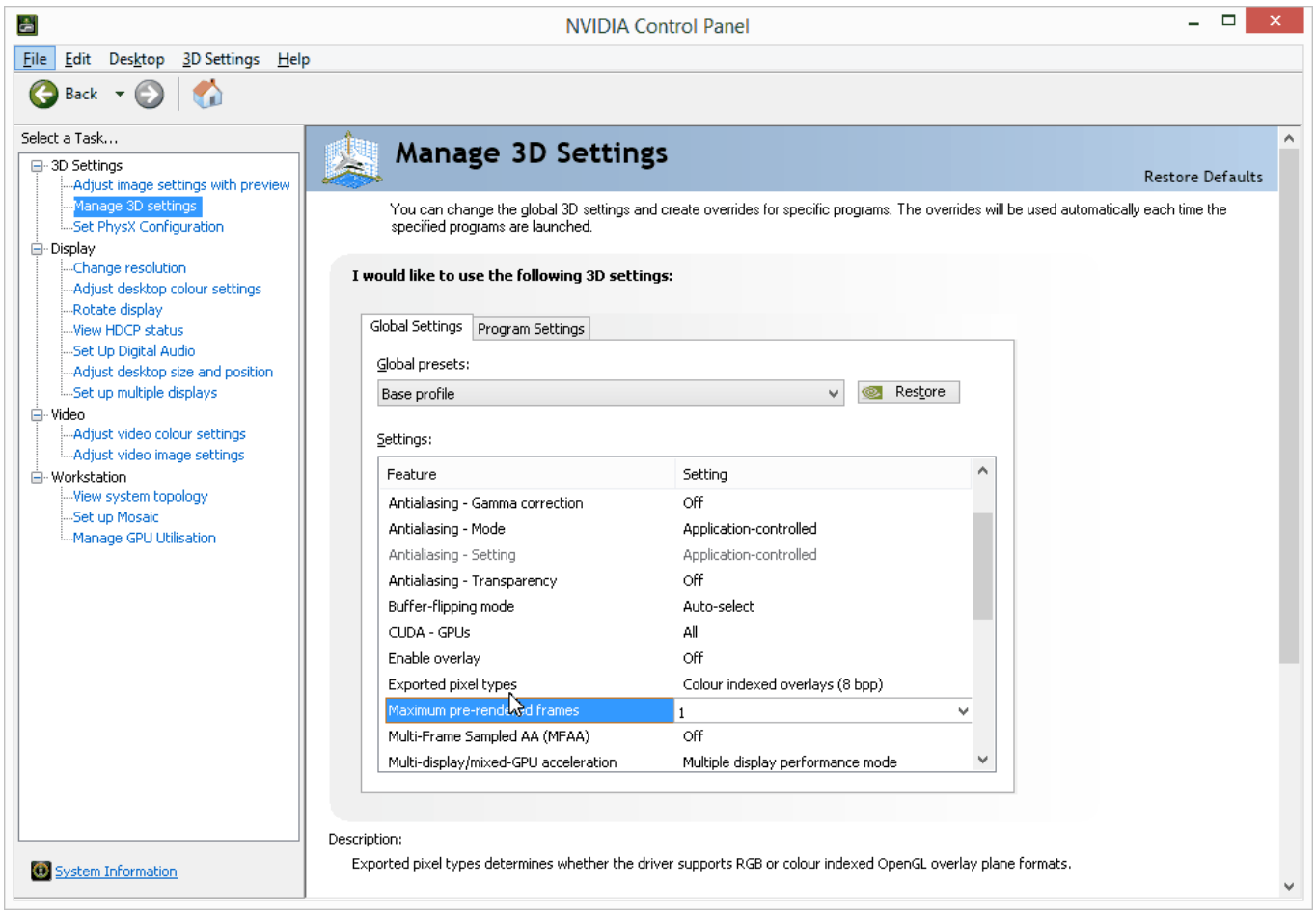
5) After the reboot go to the NVIDIA Control Panel, section "Manage 3D settings" and change the following options to match our defaults:

Antialiasing – Gamma correction = OFF

Maximum pre-rendered frames = 1

Texturing filtering = High Quality

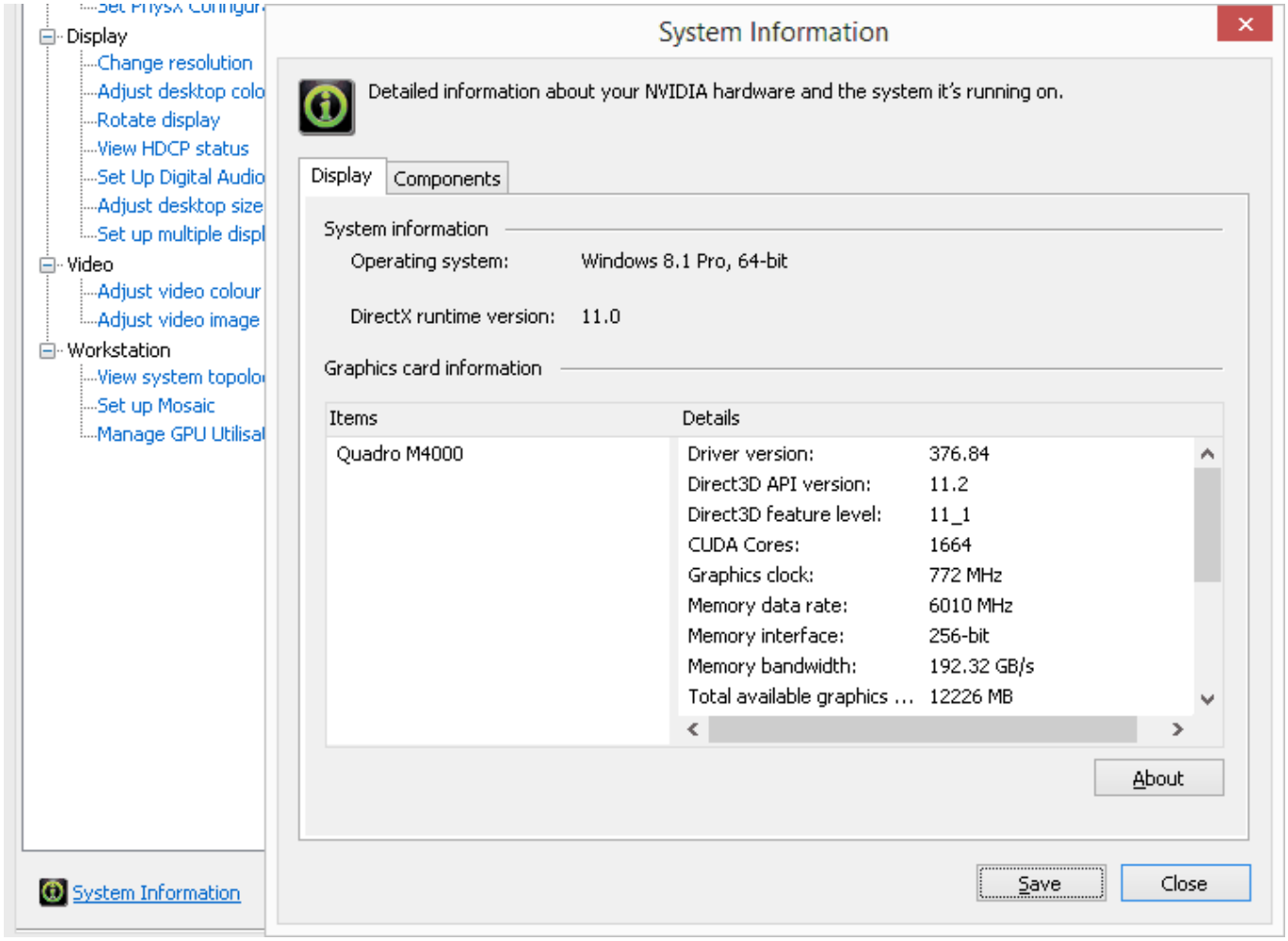
Vertical Sync = Force ON



Checking the Current Driver Version

You can always see which graphics card model and driver version is installed in your hardware by clicking on the "System Information" button on the bottom left corner of the NVIDIA Control Panel.

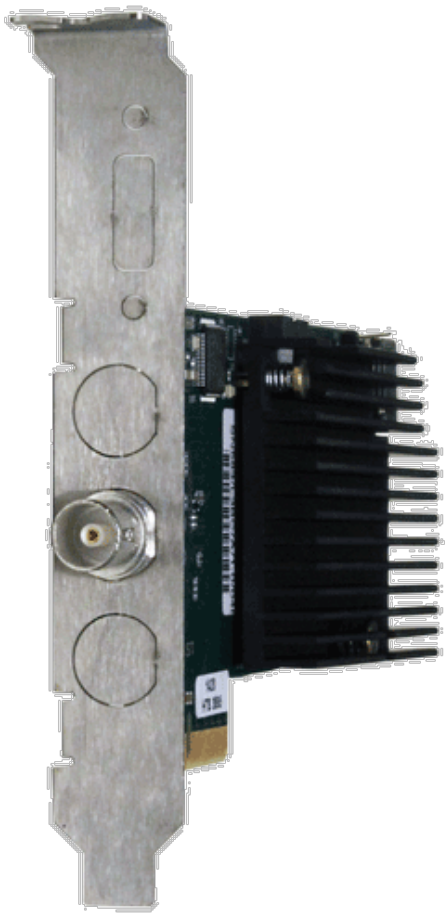
In the upcoming window you can see the graphics card model on the left and the driver version in the first line on the right part of the window.



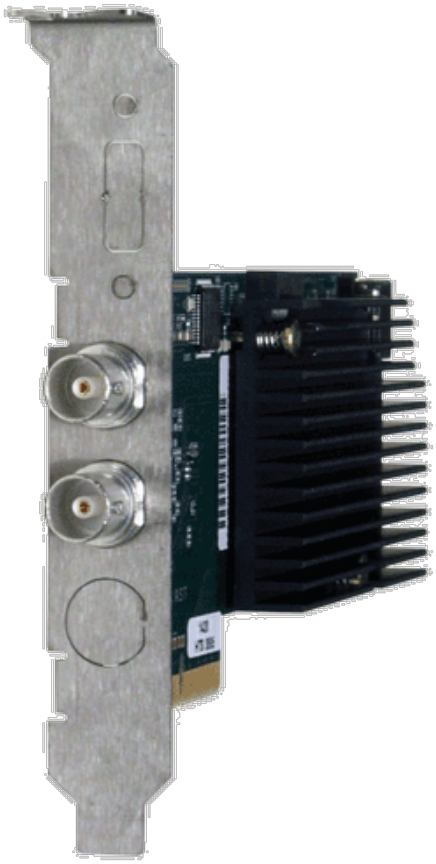
8.3.2 3G-SDI Input Cards

This chapter describes the optional 3G-SDI input cards. For other input or output boards, please see the introductory chapter [1947].

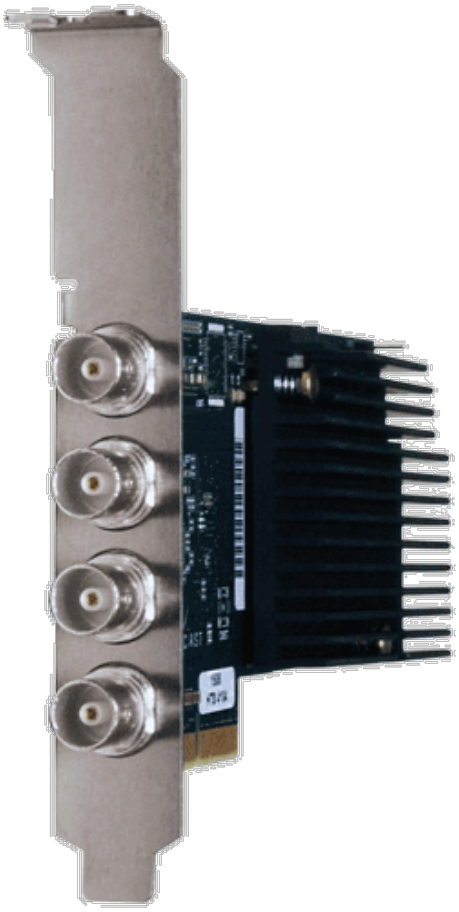
There are three 3G-SDI input cards available: Single, Dual and Quad.
They allow to input one, two or four simultaneous 3G-, HD- or SD-SDI video signals to be used within Pandoras Box.



Single 3G-SDI input card



Dual 3G-SDI input card



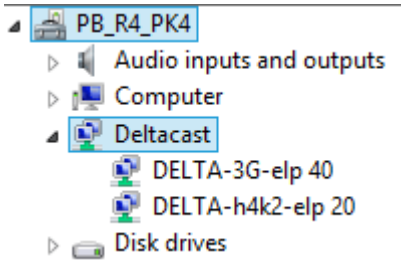
Quad 3G-SDI input card

Supported Video and Audio Formats

The following table lists the supported video formats. As for the audio formats, de-embedding of up to 16 digital audio channels per HD-SDI link or up to 32 digital audio channels per 3G-SDI link is possible.

Single, Dual and Quad 3G-SDI input card	
SD	625i50 (PAL), 525i59.94 (NTSC)
HD 1280x720	720p60, 720p59.94, 720p50
HD 1920x1080	1080i60, 1080i59.94, 1080i50, 1080p30, 1080p29.97, 1080p25, 1080p60, 1080p59.94, 1080p50
2K 2048x1080	1080p30, 1080p29.97, 1080p25, 1080p24, 1080p23.98, 1080p60, 1080p59.94, 1080p50, 1080p48, 1080p47.952
Quad 3G-SDI input card	
UHD 3840x2160	2160p60, 2160p59.94, 2160p50, 2160p30, 2160p29.97, 2160p25, 2160p24, 2160p23.98 (synchronized quadrants)
4K 4096x2160	2160p50, 2160p30, 2160p29.97, 2160p25, 2160p24, 2160p23.98 (synchronized quadrants)

Driver



The card drivers are preinstalled on your Server hardware. All drivers can be found locally under `C:\system` and in the Download-Center where you can also check for updates. The driver installer searches automatically for existing cards and installs the appropriate version.

Starting with Pandoras Box version 6.4.1 we are supporting the original Deltacast drivers (version 6.14) which we have added to our driver package in the Download-Center.
The 12G-SDI board [1972] can only be used with driver version 6.14 (or later).
The DVI board [2014] (and old HD-SDI board) can only be used with driver version up to 6.01.
The 3G-SDI board [1969] and HDMI board [1978] can be used with both versions.

When having two or more cards in a single computer they have to run with the same driver version. Therefore, a DVI board cannot be used along with the 12G board in a single computer. Using different driver version on separate machines is not a problem.

The picture shows the "Deltacast" folder in the Windows Device Manager after installing a 3G-SDI and HDMI card using the Deltacast driver.

Working with Video Inputs in Pandoras Box

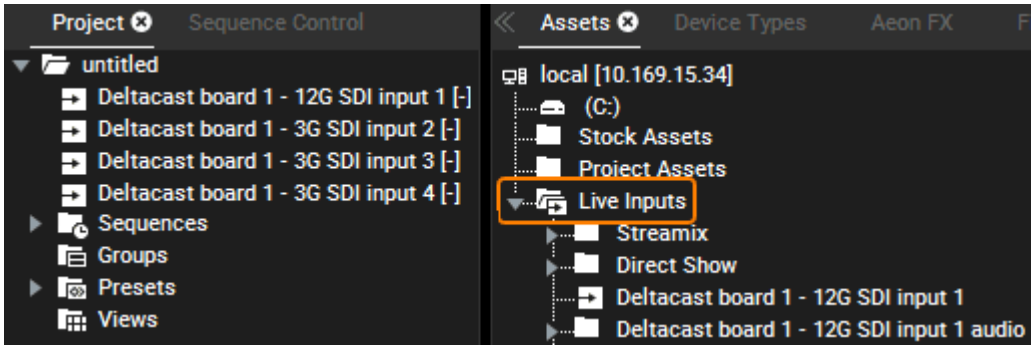
Video Inputs in Pandoras Box are managed by the Pandoras Box Master, that is responsible for the entire show control. Only if you use the Master in stand-alone mode, you would connect the video device to the Master's input board. In case you work in a Master-Client-setup and you like to use the stream on a Client, the Client must be equipped with an input board (not the Master). In other words, the physical signal input happens on the same machine which should render and output the image. The video stream only "exists" on the Client, it is not distributed to the Master nor to another Client. If you like to display the stream on several Clients, all of them must be connected to the video source.

Follow this step-by-step description after having connected the video source to the input card:

- Start the Master computer and Pandoras Box in Master mode. If the input card is in the Client, start there the Client mode from the same version.
- then go to the Master, Asset tab [131] > local or Client > folder "Live Inputs"
- drag the according card into the Project tab [278]; this initializes the card
- assign the live input to a layer in the Device Tree [173] the same way as using common image or video files

Bear in mind that it is recommended to leave the signal source connected and not change any settings regarding the signal once Pandoras Box is started. Otherwise you might need to re-configure the input in Pandoras Box which requires to toggle the fullscreen window on a Client.

It is also possible to record the incoming signal. However, the Client computer must then be started in stand-alone mode, that is with the Pandoras Box Master software. Please see the chapter "Audio and Video Recording [137]" for more information.



Synchronized and 4K SDI Live Input

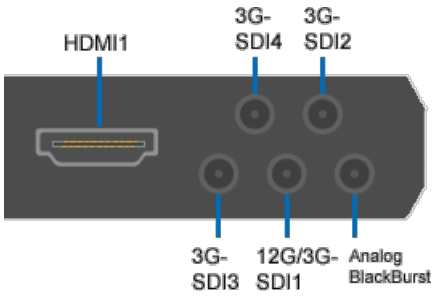
Since version 5.7 SDI input cards offer the possibility to synchronize genlocked input feeds with the same resolution and frame rate. This is of interest for a "mask-and-fill" setup or for creating a larger format like a 4K

texture. For a Dual SDI card, you can synchronize both feeds from the same card. For a QUAD SDI card, you can synchronize two pairs (the first two and the last two) and for the QUAD 3G-SDI card or the 12G-SDI card, you can in addition synchronize all four inputs.
To synchronize two or four inputs: Pandoras Box Master > Assets tab [131] > choose the Client with the card and open its "Live Input" folder. Drag all feeds into the Project tab [278], select them there and make a right-click to choose to synchronize them. To maintain the sync, make sure to always assign all synced inputs to Layers, even if only one should be used.
To create a 4K input texture: Pandoras Box Master > Assets tab [131] > choose the Client with the card and open its "Live Input" folder. Drag only the first feed into the Project tab [278], and there, right-click it to choose to toggle into 4K mode.

Note: The icon of the Live Input in the Project tab will change if the synchronized or 4K mode is activated.
Important: Our input card driver installer 5.16 or higher is needed to synchronize SDI feeds and / or create a 4K texture.

8.3.3 12G-SDI Input Cards

This chapter describes the optional 12G-SDI / HDMI input cards. For other input or output boards, please see the introductory chapter [1947].



The 12G-SDI is indeed a 12G/Quad 3G-SDI, HDMI/DP input board as it offers the following inputs:
- 1x 12G-/3G-SDI input (Micro BNC (HD-BNC)) *
- 3x 3G-SDI inputs (Micro BNC (HD-BNC))
- 1x HDMI 2.0 input (HDMI Type A), which can also be used for DisplayPort.

The card is shipped with four Micro BNC adapters and one DisplayPort-HDMI adapter.

* The 12G input can also be used as a 3G input so that you can either work with one 12G stream or four 3G-SDI streams. The input detects automatically whether it is connected to a 3G or 12G source. If a 12G-SDI source is detected, the three 3G-SDI's cannot be used. As described below [1974], the 3G-SDI inputs can be synchronized or combined to a 4K input.

The card is tailored to support two 4K60 streams, with one 4K60 feed being captured on the HDMI input, and a second 4K60 feed being handled either as 1x12G-SDI or 4x3G-SDI.

In total, the card allows to input one HDMI (or DisplayPort or DVI) signal and additionally one 12G-SDI signal or up to four simultaneous 3G-, HD- or SD-SDI video signals to be used within Pandoras Box.

12G/Quad 3G-SDI, HDMI/DP input card

Supported Video and Audio Formats

The following table lists the supported video formats.

The HDMI 2.0 input supports any HDMI format, including 4K @ 60Hz with a color depth of 24bit/px. Since Pandoras Box version 6.4.0 you can choose another chroma sub sampling when working with HDMI inputs, i.e. RGB 4:4.4, in addition to YUV 4:2:2. You can find this option in the configuration window from the input.

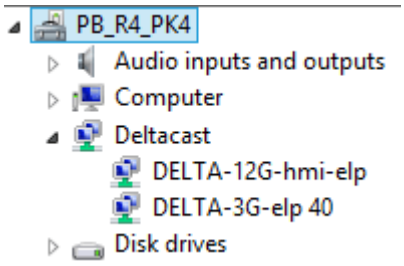
You can also use the HDMI input for a DisplayPort signal, an adapter is included. Note that HDMI is also backward compatible with DVI (single-link DVI-D, but not DVI-A). It is possible to use DVI to HDMI adapters and keeping the video quality as no signal conversion is required.

Please note that HDCP-protected signals can not be processed!

As for the audio formats, de-embedding of up to 16 digital audio channels per HD-SDI link or up to 32 digital audio channels per 3G-SDI link is possible. With Pandoras Box version 6.5 or higher you can also use up to 8 audio channels from the HDMI signal, which are displayed as four stereo channels in the Assets tab [131].
To playback the audio part on an Audio Track [661] you always have to play the video part on a Video Layer simultaneously. A sample rate of 48k is possible for SDI and HDMI audio and of 44.1k for HDMI audio. As Inputs are not resampled please make sure that the audio device matches this sample rate (Configuration tab > Sites > ASIO Audio) [166]. The sample rate of the audio channel is displayed in the Inspector [191] when selecting it in the Project tab.

HDMI and SDI video formats	
SD	625i50 (PAL), 525i59.94 (NTSC)
HD 1280x720	720p60, 720p59.94, 720p50, 720p30, 720p29.97, 720p25, 720p24 ,720p23.98
HD 1920x1080	1080p60, 1080p59.94, 1080p50, 1080p30, 1080p29.97, 1080p25, 1080p24, 1080p23.98, 1080i60, 1080i59.94, 1080i50,
2K 2048x1080	1080p60, 1080p59.94, 1080p50, 1080p48, 1080p47.952, 1080p30, 1080p29.97, 1080p25, 1080p24, 1080p23.98
UHD 3840x2160 4K 4096x2160	2160p60, 2160p59.94, 2160p50, 2160p30, 2160p29.97, 2160p25, 2160p24, 2160p23.98
HDMI graphic formats	
HDMI	This is a list of the maximum frame rate for some standard 16:9 formats: 1920x1080 @ 120Hz 2560x1440 @ 120Hz 3840x2160 @ 60Hz 4096x2160 @ 60Hz Frame rates from 23.98Hz to 120Hz depending on formats

Driver



The card drivers are preinstalled on your Server hardware. All drivers can be found locally under `C:\system` and in the Download-Center where you can also check for updates. The driver installer searches automatically for existing cards and installs the appropriate version.

Starting with Pandoras Box version 6.4.1 we are supporting the original Deltacast drivers (version 6.14) which we have added to our driver package in the Download-Center.
The 12G-SDI board [1972] can only be used with driver version 6.14 (or later).
The DVI board [2014] (and old HD-SDI board) can only be used with driver version up to 6.01.
The 3G-SDI board [1969] and HDMI board [1978] can be used with both versions.

When having two or more cards in a single computer they have to run with the same driver version. Therefore, a DVI board cannot be used along with the 12G board in a single computer. Using different driver version on separate machines is not a problem.

The picture shows the "Deltacast" folder in the Windows Device Manager after installing a 12G-SDI card using the Deltacast driver.

Working with Video Inputs in Pandoras Box

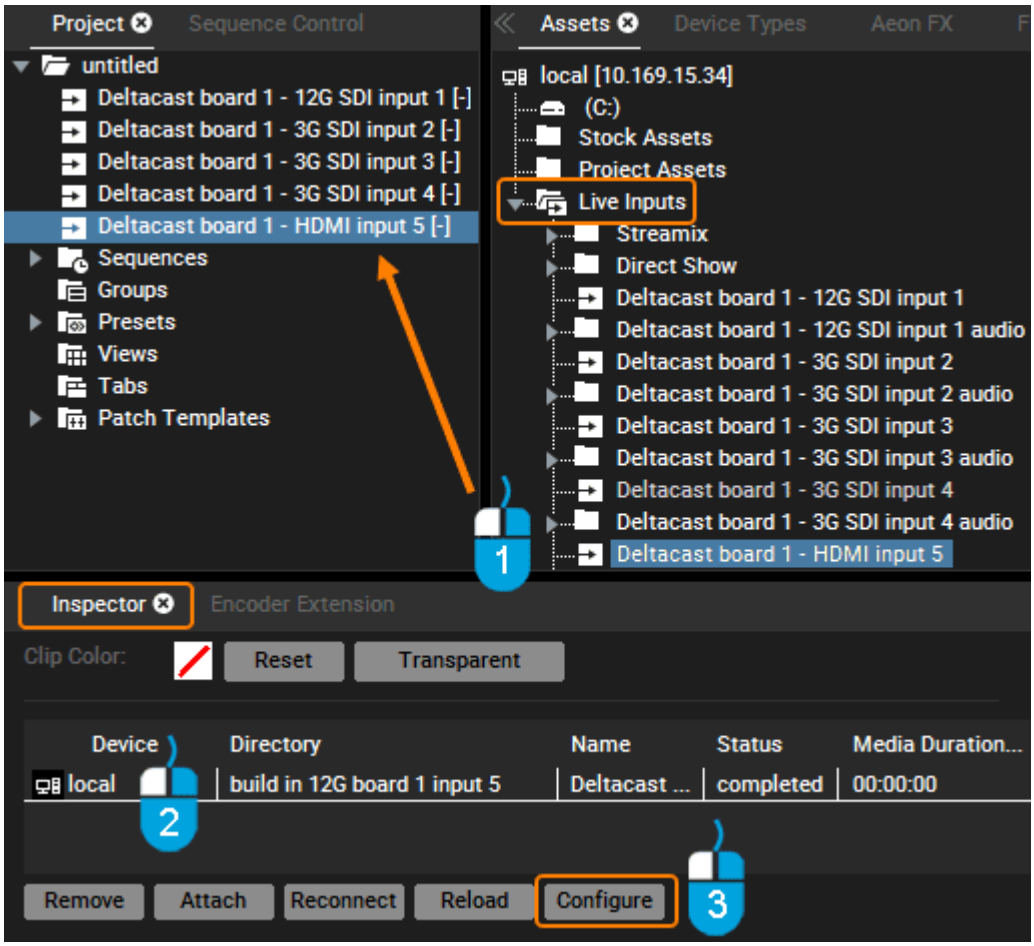
Video Inputs in Pandoras Box are managed by the Pandoras Box Master, that is responsible for the entire show control. Only if you use the Master in stand-alone mode, you would connect the video device to the Master's input board. In case you work in a Master-Client-setup and you like to use the stream on a Client, the Client must be equipped with an input board (not the Master). In other words, the physical signal input happens on the same machine which should render and output the image. The video stream only "exists" on the Client, it is not distributed to the Master nor to another Client. If you like to display the stream on several Clients, all of them must be connected to the video source.

Follow this step-by-step description after having connected the video source to the input card:

- Start the Master computer and Pandoras Box in Master mode. If the input card is in the Client, start there the Client mode from the same version.
- then go to the Master, Asset tab [131] > local or Client > folder "Live Inputs"
- drag the according card into the Project tab [278]; this initializes the card
- if you like to change the chroma subsampling for the HDMI input, select it in the Project tab to display its properties in the Inspector [190]. Scroll down in the Inspector to see the file table where you can select the live input and click the "Configure" button below (the below image shows the inputs from a 12G card)
- assign the live input to a layer in the Device Tree [173] the same way as using common image or video files

Bear in mind that it is recommended to leave the signal source connected and not change any settings regarding the signal once Pandoras Box is started. Otherwise you might need to re-configure the input in Pandoras Box which requires to toggle the fullscreen window on a Client.

It is also possible to record the incoming signal. However, the Client computer must then be started in stand-alone mode, that is with the Pandoras Box Master software. Please see the chapter "Audio and Video Recording [137]" for more information.



Synchronized and 4K SDI Live Input

Since version 5.7 SDI input cards offer the possibility to synchronize genlocked input feeds with the same resolution and frame rate. This is of interest for a "mask-and-fill" setup or for creating a larger format like a 4K texture. For a Dual SDI card, you can synchronize both feeds from the same card. For a QUAD SDI card, you can synchronize two pairs (the first two and the last two) and for the QUAD 3G-SDI card or the 12G-SDI card, you can in addition synchronize all four inputs.
To synchronize two or four inputs: Pandoras Box Master > Assets tab [131] > choose the Client with the card and open its "Live Input" folder. Drag all feeds into the Project tab [278], select them there and make a right-click to choose to synchronize them. To maintain the sync, make sure to always assign all synced inputs to Layers, even if only one should be used.
To create a 4K input texture: Pandoras Box Master > Assets tab [131] > choose the Client with the card and open its "Live Input" folder. Drag only the first feed into the Project tab [278], and there, right-click it to choose to toggle into 4K mode.

Note: The icon of the Live Input in the Project tab will change if the synchronized or 4K mode is activated.
Important: Our input card driver installer 5.16 or higher is needed to synchronize SDI feeds and / or create a 4K texture.

8.3.4 Flex Input Cards

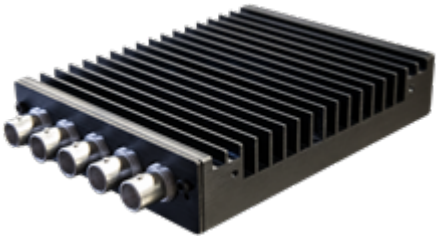
This chapter describes the optional Flex input card from Deltacast. For other input or output boards, please see the introductory chapter [1947].

The FLEX solution is a combination of a gateway card and external modules/boards which are connected through specific cables. To get more or different connector types, simply connect another boards that offers the needed connectors. This allows to mix various connection types just as your show requires it. Changing the available inputs is just a matter of seconds and there is not need to open the chassis. Thus the FLEX input card is especially of interest for Rental and Staging customers or whenever changing setups are needed. Another advantage is that there is less space needed in the Server itself as the gateway card just occupies one PCI Express slot. The external modules can be used as floating devices or can be integrated in a 19" rack offered by Deltacast.

The gateway card offers four inputs and each can be connected to one of the following input / capture modules. So in total, you could input up to 16 SDI channels (12x 3G or 4x 12G) or alternatively, four 4K60 streams via DisplayPort 1.2 or HDMI 2.0, or a mixture of them.

Important information:
Never hot plug the modules! Shut down the system first and then connect or disconnect them.

12G-SDI Module: Supported Video and Audio Formats



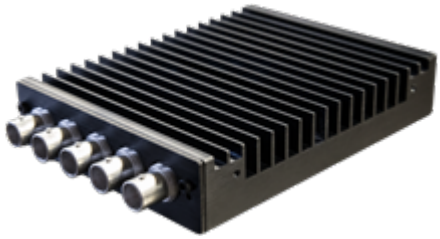
The 4-input 3G-SDI or single input 12G-SDI module is called FLEX-12G.

It hosts four independent BNC inputs ports. One of these ports supports line rates up to 12G-SDI while the three other ones go up to 3G-SDI. The below table lists the supported video formats. The module provides capture of 4K pictures at up to 60p without any compromise and over any type of interface (Quad Link 3G Square Division, Quad Link 3G 2-Sample Interleave, or Single Link 12G).

As for the audio formats, de-embedding of up to 16 digital audio channels per HD-SDI link or up to 32 digital audio channels per 3G-SDI link is possible. They are displayed in the Assets tab [131] in the "Live inputs" folder.
To playback the audio part on an Audio Track [661] you always have to play the video part on a Video Layer simultaneously. A sample rate of 48k is possible. As Inputs are not re-sampled please make sure that the audio device matches this sample rate (Configuration tab > Sites > ASIO Audio) [166]. The sample rate of the audio channel is displayed in the Inspector [191] when selecting it in the Project tab.

SDI video formats	
SD	625i50 (PAL), 525i59.94 (NTSC)
HD 1280x720	720p60, 720p59.94, 720p50, 720p30, 720p29.97, 720p25, 720p24 ,720p23.98
HD 1920x1080	1080p60, 1080p59.94, 1080p50, 1080p30, 1080p29.97, 1080p25, 1080p24, 1080p23.98, 1080i60, 1080i59.94, 1080i50,
2K 2048x1080	1080p60, 1080p59.94, 1080p50, 1080p48, 1080p47.952, 1080p30, 1080p29.97, 1080p25, 1080p24, 1080p23.98
UHD 3840x2160 4K 4096x2160	2160p60, 2160p59.94, 2160p50, 2160p30, 2160p29.97, 2160p25, 2160p24, 2160p23.98

3G-SDI Module: Supported Video and Audio Formats



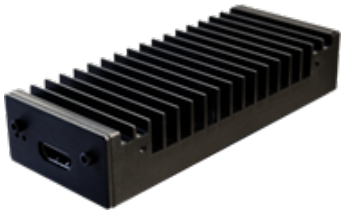
The 4-input 3G-SDI module is called FLEX-3G.

It hosts four independent four independent SDI input channels via BNC connectors. They support formats from SD to 3G-SDI. The below table lists the supported video formats. The channels can be combined together to capture 4K pictures at up to 60p without any compromise (10-bit, Square Division or 2-Sample Interleave).

As for the audio formats, de-embedding of up to 16 digital audio channels per HD-SDI link or up to 32 digital audio channels per 3G-SDI link is possible. They are displayed in the Assets tab [131] in the "Live inputs" folder.
To playback the audio part on an Audio Track [661] you always have to play the video part on a Video Layer simultaneously. A sample rate of 48k is possible. As Inputs are not re-sampled please make sure that the audio device matches this sample rate (Configuration tab > Sites > ASIO Audio) [1117]. The sample rate of the audio channel is displayed in the Inspector [191] when selecting it in the Project tab.

SDI video formats	
SD	625i50 (PAL), 525i59.94 (NTSC)
HD 1280x720	720p60, 720p59.94, 720p50, 720p30, 720p29.97, 720p25, 720p24 ,720p23.98
HD 1920x1080	1080p60, 1080p59.94, 1080p50, 1080p30, 1080p29.97, 1080p25, 1080p24, 1080p23.98, 1080i60, 1080i59.94, 1080i50,
2K 2048x1080	1080p60, 1080p59.94, 1080p50, 1080p48, 1080p47.952, 1080p30, 1080p29.97, 1080p25, 1080p24, 1080p23.98
UHD 3840x2160 4K 4096x2160	2 Sample Interleave or Square Division 2160p60, 2160p59.94, 2160p50, 2160p30, 2160p29.97, 2160p25, 2160p24, 2160p23.98

DisplayPort Module: Supported Video and Audio Formats



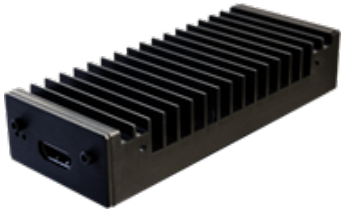
The single-input DisplayPort 1.2 module is called FLEX-DP 10.

It hosts one DisplayPort 1.2 input channel that supports video and graphics formats up to 4K60 without any compromise.

As for the audio formats, de-embedding of up to 8 digital audio channels is possible. They are displayed as four stereo channels in the Assets tab [131].
To playback the audio part on an Audio Track [661] you always have to play the video part on a Video Layer simultaneously. A sample rate of 44.1k is possible. As Inputs are not re-sampled please make sure that the audio device matches this sample rate (Configuration tab > Sites > ASIO Audio) [166]. The sample rate of the audio channel is displayed in the Inspector [191] when selecting it in the Project tab.

DP video formats	
Video Formats	PAL, NTSC, 720p,1080i,1080p, 2160p (4K DCI and UHDTV1)
Graphics Formats	up to 2560 x 1080
	Deep color and high frame rate formats supported

HDMI Module: Supported Video and Audio Formats



The single-input HDMI 2.0 module is called FLEX-HMI 10.

It hosts one HDMI 2.0b input channel that supports video and graphics formats up to 4K60 without any compromise.

The color depth is 24bit/px. Since Pandoras Box version 6.4.0 you can choose another chroma sub sampling when working with HDMI inputs, i.e. RGB 4:4.4, in addition to YUV 4:2:2. You can find this option in the configuration window from the input.

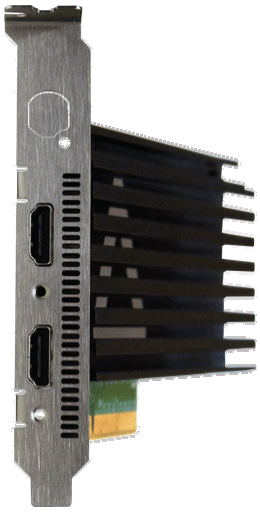
Please note that HDCP-protected signals can not be processed!

As for the audio formats, de-embedding of up to 8 digital audio channels is possible. They are displayed as four stereo channels in the Assets tab [131].
To playback the audio part on an Audio Track [661] you always have to play the video part on a Video Layer simultaneously. A sample rate of 44.1k and 48k is possible. As Inputs are not re-sampled please make sure that the audio device matches this sample rate (Configuration tab > Sites > ASIO Audio) [166]. The sample rate of the audio channel is displayed in the Inspector [191] when selecting it in the Project tab.

HDMI video formats	
Video Formats	PAL, NTSC, 720p,1080i,1080p, 2160p (4K DCI and UHDTV1)
Graphics Formats	up to 1920 x 1200
	Deep color and high frame rate formats supported

8.3.5 HDMI Input Card

This chapter describes the optional HDMI input card. For other input or output boards, please see the introductory chapter [1947].



There is one HDMI 2.0 card available with two inputs (HDMI Type A). The card is shipped with a DisplayPort-HDMI adapter.

The card allows to input HDMI (or DisplayPort or DVI) signals to be used within the Pandoras Box render engine.

Supported Audio and Video Formats

The HDMI 2.0 capture card supports on both inputs simultaneously any HDMI format, including 4K @ 60Hz with a color depth of 24bit/px. Since Pandoras Box version 6.4.0 you can choose another chroma sub sampling when working with HDMI inputs, i.e. RGB 4:4.4, in addition to YUV 4:2:2. You can find this option in the configuration window from the input.

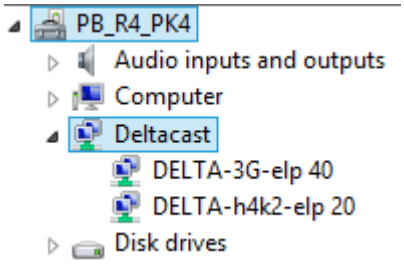
You can also use the HDMI input for a DisplayPort signal, an adapter is included. Note that HDMI is also backward compatible with DVI (single-link DVI-D, but not DVI-A). It is possible to use DVI to HDMI adapters and keeping the video quality as no signal conversion is required.

Please note that HDCP-protected signals can not be processed!

As for the audio formats, de-embedding of up to 8 channels (displayed as four stereo channels in the Assets tab [131]) is possible when using Pandoras Box version 6.5 or higher. To playback the audio part on an Audio Track [661] you always have to play the video part on a Video Layer simultaneously. Sample rates of 44.1k and 48k are supported. As Inputs are not resampled please make sure that the audio device matches this sample rate (Configuration tab > ASIO Audio) [166]. The sample rate of the HDMI audio channel is displayed in the Inspector [191] when selecting it in the Project tab.

Dual HDMI input card	
HDMI	This is a list of the maximum framerate for some standard 16:9 formats: 1920x1080 @ 120Hz 2560x1440 @ 120Hz 3840x2160 @ 60Hz 4096x2160 @ 60Hz Frame rates from 23.98Hz to 120Hz depending on formats

Driver



The card drivers are preinstalled on your Server hardware. All drivers can be found locally under `C:\system` and in the Download-Center where you can also check for updates. The driver installer searches automatically for existing cards and installs the appropriate version.

Starting with Pandoras Box version 6.4.1 we are supporting the original Deltacast drivers (version 6.14) which we have added to our driver package in the Download-Center.
The 12G-SDI board [1972] can only be used with driver version 6.14 (or later).
The DVI board [2014] (and old HD-SDI board) can only be used with driver version up to 6.01.
The 3G-SDI board [1969] and HDMI board [1978] can be used with both versions.

When having two or more cards in a single computer they have to run with the same driver version. Therefore, a DVI board cannot be used along with the 12G board in a single computer. Using different driver version on separate machines is not a problem.

The picture shows the "Deltacast" folder in the Windows Device Manager after installing a 3G-SDI and HDMI card using the Deltacast driver.

Working with Video Inputs in Pandoras Box

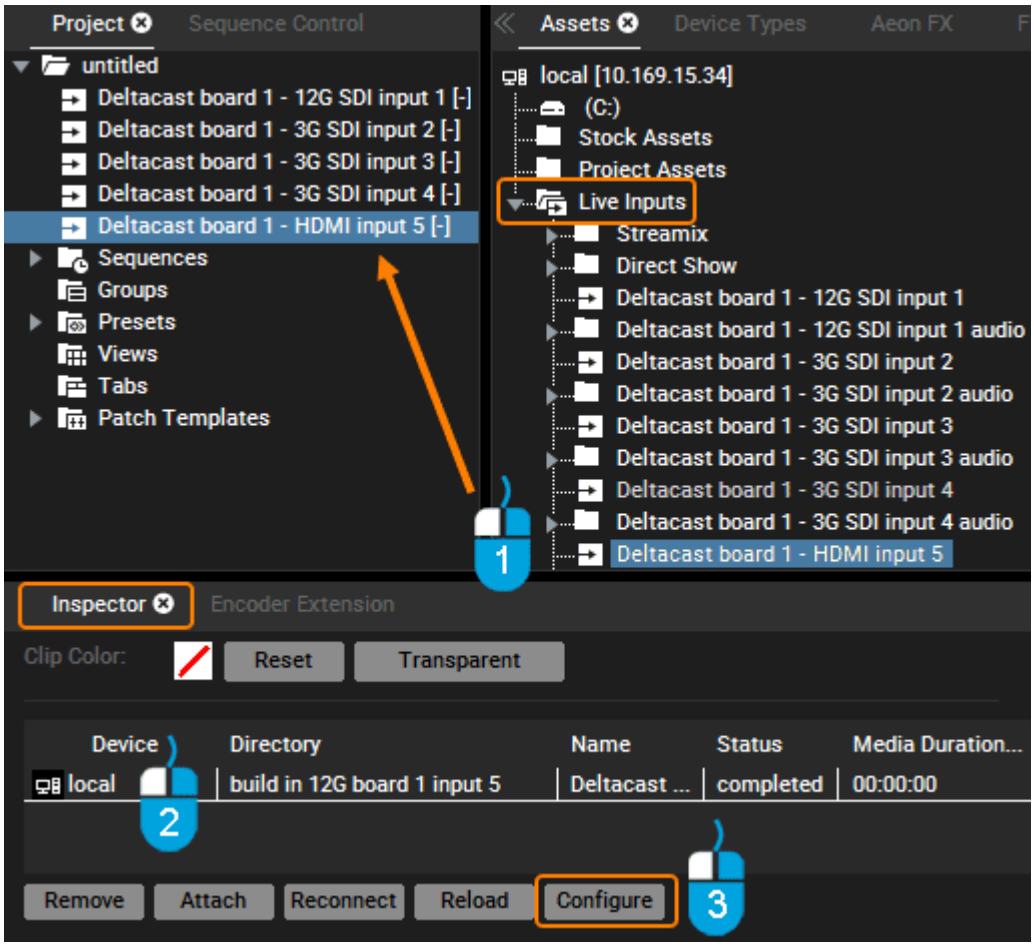
Video Inputs in Pandoras Box are managed by the Pandoras Box Master, that is responsible for the entire show control. Only if you use the Master in stand-alone mode, you would connect the video device to the Master's input board. In case you work in a Master-Client-setup and you like to use the stream on a Client, the Client must be equipped with an input board (not the Master). In other words, the physical signal input happens on the same machine which should render and output the image. The video stream only "exists" on the Client, it is not distributed to the Master nor to another Client. If you like to display the stream on several Clients, all of them must be connected to the video source.

Follow this step-by-step description after having connected the video source to the input card:

- Start the Master computer and Pandoras Box in Master mode. If the input card is in the Client, start there the Client mode from the same version.
- then go to the Master, Asset tab [131] > local or Client > folder "Live Inputs"
- drag the according card into the Project tab [278]; this initializes the card
- if you like to change the chroma subsampling for the HDMI input, select it in the Project tab to display its properties in the Inspector [190]. Scroll down in the Inspector to see the file table where you can select the live input and click the "Configure" button below (the below image shows the inputs from a 12G card)
- assign the live input to a layer in the Device Tree [173] the same way as using common image or video files

Bear in mind that it is recommended to leave the signal source connected and not change any settings regarding the signal once Pandoras Box is started. Otherwise you might need to re-configure the input in Pandoras Box which requires to toggle the fullscreen window on a Client.

It is also possible to record the incoming signal. However, the Client computer must then be started in stand-alone mode, that is with the Pandoras Box Master software. Please see the chapter "Audio and Video Recording [137]" for more information.


Project
Sequence Control
untitled
Deltacast board 1 - 12G SDI input 1 [-]
Deltacast board 1 - 3G SDI input 2 [-]
Deltacast board 1 - 3G SDI input 3 [-]
Deltacast board 1 - 3G SDI input 4 [-]
Deltacast board 1 - HDMI input 5 [-]
Sequences
Groups
Presets
Views
Tabs
Patch Templates
Assets
Device Types
Aeon FX
local [10.169.15.34]
(C:)
Stock Assets
Project Assets
Live Inputs
Streamix
Direct Show
Deltacast board 1 - 12G SDI input 1
Deltacast board 1 - 12G SDI input 1 audio
Deltacast board 1 - 3G SDI input 2
Deltacast board 1 - 3G SDI input 2 audio
Deltacast board 1 - 3G SDI input 3
Deltacast board 1 - 3G SDI input 3 audio
Deltacast board 1 - 3G SDI input 4
Deltacast board 1 - 3G SDI input 4 audio
Deltacast board 1 - HDMI input 5
1
Inspector
Encoder Extension
Clip Color:
Reset
Transparent
Device
Directory
Name
Status
Media Duration...
local
build in 12G board 1 input 5
Deltacast ...
completed
00:00:00
2
3
Remove
Attach
Reconnect
Reload
Configure

8.3.6 Sync Card

This chapter will be updated soon.

This chapter describes the optional sync card. For other input or output boards, please see the introductory chapter 1947.

The NVIDIA sync card is part of the OCTO Server. For other hardware with (a) Quadro M4000 graphics card(s) it is an optional accessory board. The sync card has mainly two purposes:

Multi-GPU Mosaic	Mosaic is NVIDIA's software technology that abstracts multiple physical displays into a larger virtual display. When the virtual display spans across multiple graphics cards, the sync board is needed to synchronize them. In other words, When working with multi-GPU Mosaic, the sync card automatically keeps all included displays synchronized. This is of special interest for OCTO Servers as they are equipped with two graphics cards which need to be combined for the Pandoras Box render engine.
Frame lock and gen lock	Frame lock synchronizes multiple displays (in one or multiple systems) to a timing master. Frame Lock can be used with individual physical displays and single or multi-GPU Mosaic displays. You can sync a display to an external timing generator, the gen lock. The selected display can be an individual physical display or Mosaic display and can be used as the frame lock master if desired.

The topic "Setting up Frame Lock 1961" in the graphics cards chapter describes the settings in the NVIDIA Control Panel.

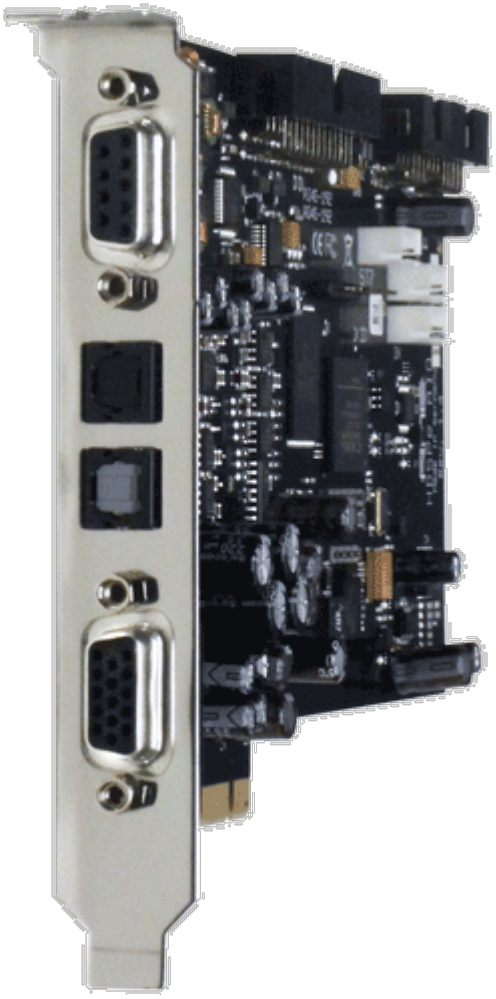
This chapter will be updated soon.

8.3.7 ADAT Sound Cards

This chapter describes two optional ADAT audio cards. Please note that the 32ch card is discontinued. For other input or output boards, please see the introductory chapter [1947].

Both ADAT audio cards allow to input and output audio (in accordance with the ASIO protocol) to and from Pandoras Box.

Sound Card 8ch ADAT (RME HDSPe AIO)



The 8ch ADAT card carries:
S/PDIF in / out (9-pin D-sub)
AES/EBU in / out (9-pin D-sub)
ADAT in (optical; black)
ADAT out (optical; gray)
Analog / MIDI in / out (15-pin D-sub)

Analog, ADAT and SPDIF I/Os can be used simultaneously as well as S/PDIF (phono) and AES/EBU (XLR).

The following table shows the channel count for recording and playback according to the sample rate:

SF/Port	AES	S/PDIF	ADAT	Analog
48 kHz	2	2	8	2
96 kHz	2	2	4	2
192 kHz	2	2	2	2

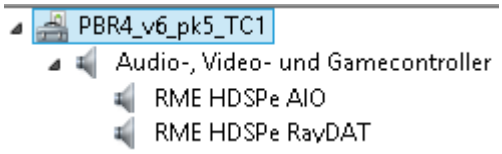
The ADAT I/O can also be used as optical SPDIF I/O, if set up accordingly in the Settings dialog.

The 9-pin digital breakout cable has two RCA connectors as coaxial SPDIF I/O (the red phono socket is the output), and an XLR AES/EBU input and output.

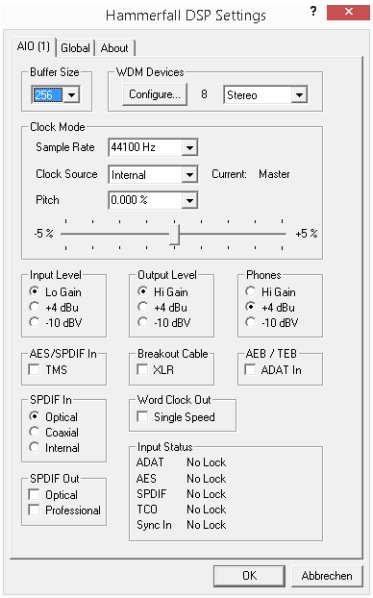
The 15-pin analog breakout cable has four RCA connectors (stereo analog I/O), a 1/4" TRS jack (headphones), and two 5-pin DIN connectors (MIDI I/O). Using this analog XLR breakout cable, the 8ch ADAT card offers balanced Line inputs and outputs via female and male XLR connectors. When using it, please make sure that "Breakout Cable / XLR" is selected in the Settings dialog, else the analog output level will be 6 dB too high!

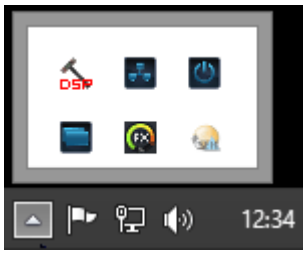
Driver

The card drivers are preinstalled on your Server hardware. All drivers can be found locally under `C:\system` and in the Download-Center where you can also check for updates.
The following picture shows the Windows Device Manager and a correctly installed 8ch and 32ch ADAT card in the folder "Audio-, Video and Gamecontroller".



Settings and TotalMix Dialog



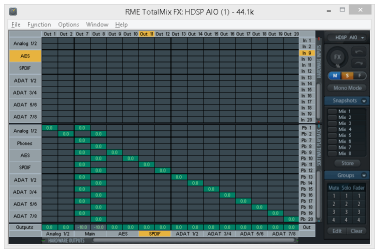


In the system tray of the taskbar you might find the hammer icon which opens the Settings dialog and the Mixer or FX icon which opens the TotalMix dialog. In case your taskbar is hidden due to the default behavior of the PB Menu [2097], open the "Configuration" menu and click "Taskbar On".

The Configuration of both, the 8ch and the 32ch ADAT card is done via its own settings dialog. It can be opened by clicking on the hammer symbol in the system tray.
The first image shows the dialog "Hammerfall DSP Settings" for the 8ch ADAT card, the one for the 32ch card is very similar. Both offer the following:
- Input selection
- Level of analog I/Os
- Configuration of digital I/Os
- Synchronization behavior
- State of input and output
- Current sample rate
- Latency

Any changes made in the Settings dialog are applied immediately - confirmation (e.g. by clicking on OK or exiting the dialog) is not required. However, settings should not be changed during playback or record if it can be avoided, as this can cause unwanted noises. Also, please note that even in 'Stop' mode, several programs keep the recording and playback devices open, which means that any new settings might not be applied immediately. The status displays at the bottom of the dialog box give the user precise information about the current status of the system, and the status of all digital signals.

The digital real-time mixer, the "Hammerfall DSP Mixer" (TotalMix) can be opened by clicking on the mixer icon in the system tray.
TotalMix allows for practically unlimited mixing and routing operations, with all inputs and playback channels simultaneously, to any hardware outputs. Every single input channel, playback channel and hardware output features a Peak and RMS level meter, calculated in hardware (hardware output is Peak only).



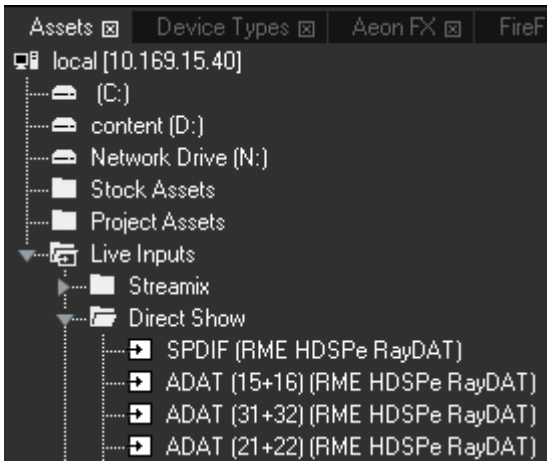
For more information about the Settings and Mixer dialog please see the audio card's manual: http://www.rme-audio.de/download/hdspeaio_e.pdf and http://www.rme-audio.de/download/raydat_e.pdf

Working with Audio In- and Outputs in Pandoras Box

To work with audio in- and outputs requires that the sound card is set up in the Configuration tab under ASIO Audio [166]. The linked chapter explains in detail how to do so and includes information regarding synchronization, timecode, sample rate setup and sample accurate audio and master volume.

To output ASIO audio from Pandoras Box, simply assign audio files from the Project tab [278] to the dedicated Audio tracks [661] in the Device Tree [173] or Sequence [292]. Audio tracks [661] allow to playback ASIO WAV files and inputs while being synchronized to a master clock. The tracks play mono and stereo PCM Wave files (16, 24 or 32bit) and they refer to the sample rate set in the Configuration tab. In order to assign a track to a specific audio channel of your ASIO sound card, please refer to the Track Inspector [217].

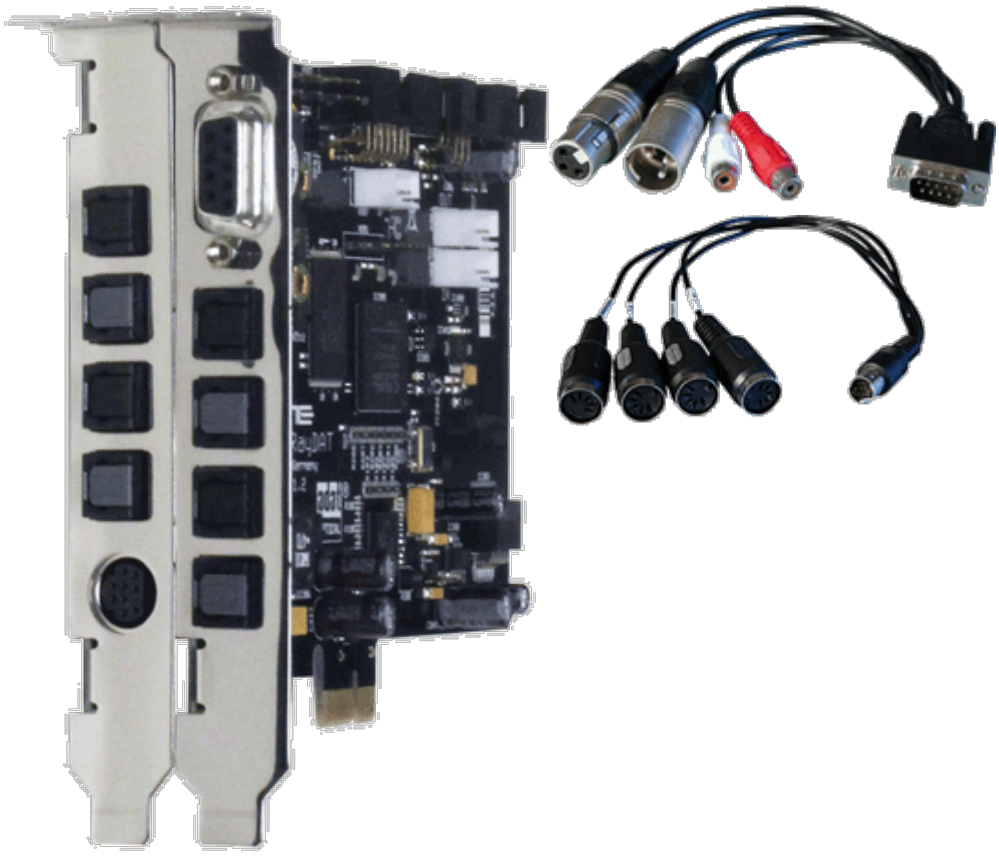
The Pandoras Box Software License offers unlimited Audio Tracks. Note that in a larger master-client-setup it is possible to use either a single audio card or multiple ones and it is possible to use the audio card from the Master or from the Client.



To input audio, you just need to go to the Asset tab [131] > local (or Client) > folder "Live Inputs" > folder "Direct Show" and drag the input to the Project tab. Then assign the live input to an Audio Track in the Device Tree the same way as using wave files.

It is also possible to record the incoming signal. However, the Client computer must then be started in stand-alone mode, that is with the Pandoras Box Master software. Please see the chapter "Video Recording [137]" for more information.

Sound Card 32ch ADAT (RME HDSPe RayDAT) --- Discontinued!



The 32ch ADAT card carries:
S/PDIF in / out (9-pin D-sub)
AES/EBU in / out (9-pin D-sub)
4 x ADAT in (optical; black)
4 x ADAT out (optical; gray)
2 x MIDI in / out (9-pin Mini-DIN)

AES and SPDIF I/Os can be used simultaneously.

The following table shows the channel count for recording and playback according to the sample rate:

SF/Port	AES	S/PDIF	ADAT
48 kHz	2	2	32
96 kHz	2	2	16
192 kHz	2	2	8

The forth ADAT I/O (on the image the left top one) can also be used as optical SPDIF I/O, if set up accordingly in the Settings dialog.

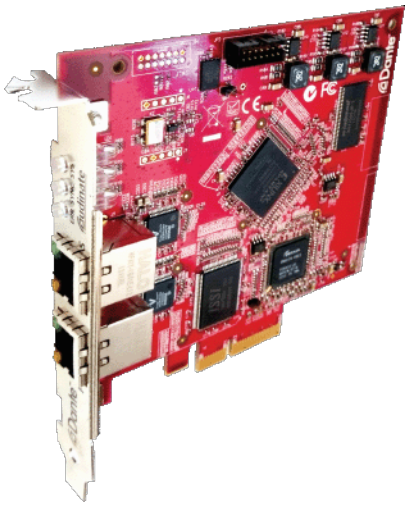
The 9-pin digital breakout cable has two RCA connectors as coaxial SPDIF I/O (the red phono socket is the output), and an XLR AES/EBU input and output.

The included MIDI breakout cable can be connected to the 9-pin Mini-DIN connector, providing two MIDI inputs and outputs.

8.3.8 Dante Audio Card

Under construction

This chapter describes the optional Dante audio card from Audinate which provides a maximum of 256 uncompressed, digital audio channels to be used in Pandoras Box. For other input or output boards, please see the introductory chapter [1947].



Dante distributes high-quality, low-latency media signals with perfect synchronization over existing IT data networks. Dante is a scalable, reliable and easy to use media network solution. Dante has become the de facto standard for media networking and digital audio over IP. Dante does not require dedicated network infrastructure and runs on off-the-shelf computer networking hardware. The card is compatible with all Dante-enabled audio equipment over standard IP/Ethernet network

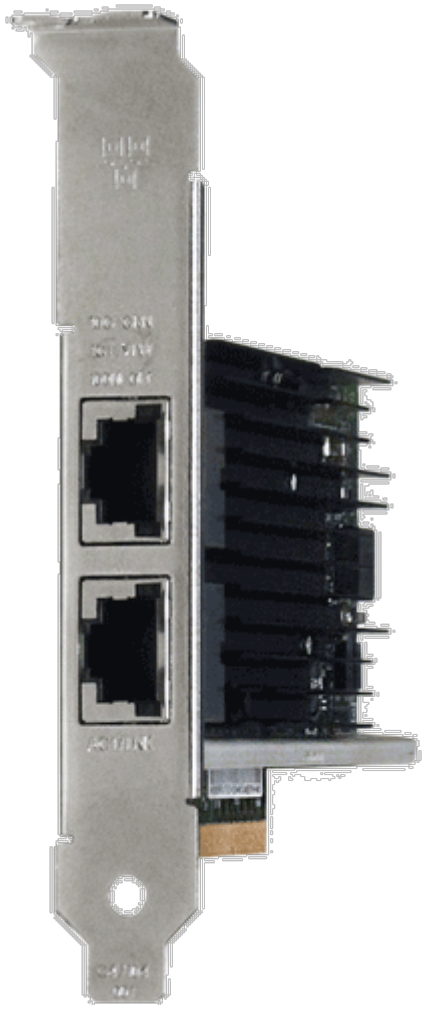
With the Dante audio card you can transmit a maximum of 256 redundant audio channels: 128 input and 128 output channels with a sample rate of 44.1, 48kHz, 88.2, or 96KHz. If you set up a sample rate of 176.4 or 192kHz, you can still use 64 redundant input and 64 output channels. The sample bit depth is 24 bit PCM.

The card can act as a Dante Master that is necessary to unlock the Dante functionality per se, generate a clock and sync the signals. The "Dante Controller" software can be used to assign the Master functionality, route signals, match sample rates for sources and outputs, etc.

The audio interface presents as standard ASIO sound card. In Pandoras Box, you can setup the card in the Configuration tab. Please see the chapter "ASIO Audio" [166] for more information.

The Dante audio card provides two distinct network connections, a "Primary" and a "Secondary" (each 1Gbps, Ethernet RJ45 connectors) which are Redundancy ports offering failover functionality. The card features Audinate's patented Glitch Free redundancy and ultra-low latency with sub-microsecond synchronization. Audio continues uninterrupted in the event of a primary network connectivity issue.

8.3.9 Ethernet Card



This chapter describes the optional 10GbE ethernet card. For other input or output boards, please see the introductory chapter [1947].

This optional ethernet card allows your hardware to offer more ethernet ports and with a higher data rate.

The card carries two RJ-45 ports, each supporting a data rate of 100Mb / 1GbE / 10GbE.

Driver

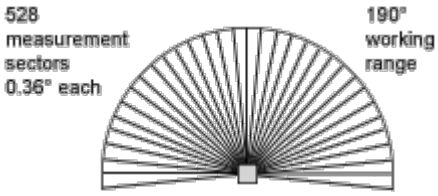
The card drivers are preinstalled on your Server hardware. All drivers can be found locally under `C:\system` and in the Download-Center where you can also check for updates.

8.4 AirScan Hardware

The AirScan is a laser scanner device that measures distances in two dimensions. Thus, it allows for optical and contactless detection of objects or persons or just their hands. It can be used for any touch-less interaction with a display or a projected touch surface.

In Widget Designer [786], our interactivity software, the AirScan is available as an optional input device and can be setup using the AirScan Tool [1277]. It is possible to read up to 24 input points and use them for various applications within Widget Designer or Pandoras Box.

Measurement Principle and Restrictions



The AirScan unit can be mounted above, below or on the side of a screen and its recommended distance to the input screen area should not be further away then 10m or 30ft. Longer distances (up to 65m or 200ft) can be set up, but will lead to a less accurate readout. The AirScan measures every 0.36 degrees if anything has crossed the scan area and transmits its data via TCP over network to Widget Designer for further data processing.

Please note the following restrictions to ensure correct usage and measurement readout:

- Due to its IR laser technology, the AirScan device itself should not be exposed to direct sunlight or direct tungsten lighting as this would distort the correct readout of the sensed data.

- Physical contact with the front cover of the sensor is to be avoided. Note that even a slight movement of the sensor affects the measurement values.

- If the AirScan is (partly) mounted into a wall or exhibition stand, make sure that the sensor (to be exact, the glass window) is not covered and the entire interactivity area is free.

- If a protective housing is provided for the sensor, the detection must not occur through additional window material (plastic, glass, etc.).

- Vapors, smoke, dust and all particles visible in the air could affect the measurement values.

- Glass and highly reflective materials such as mirrors could falsify the measurement value.

- Avoid large temperature fluctuations.

Drivers

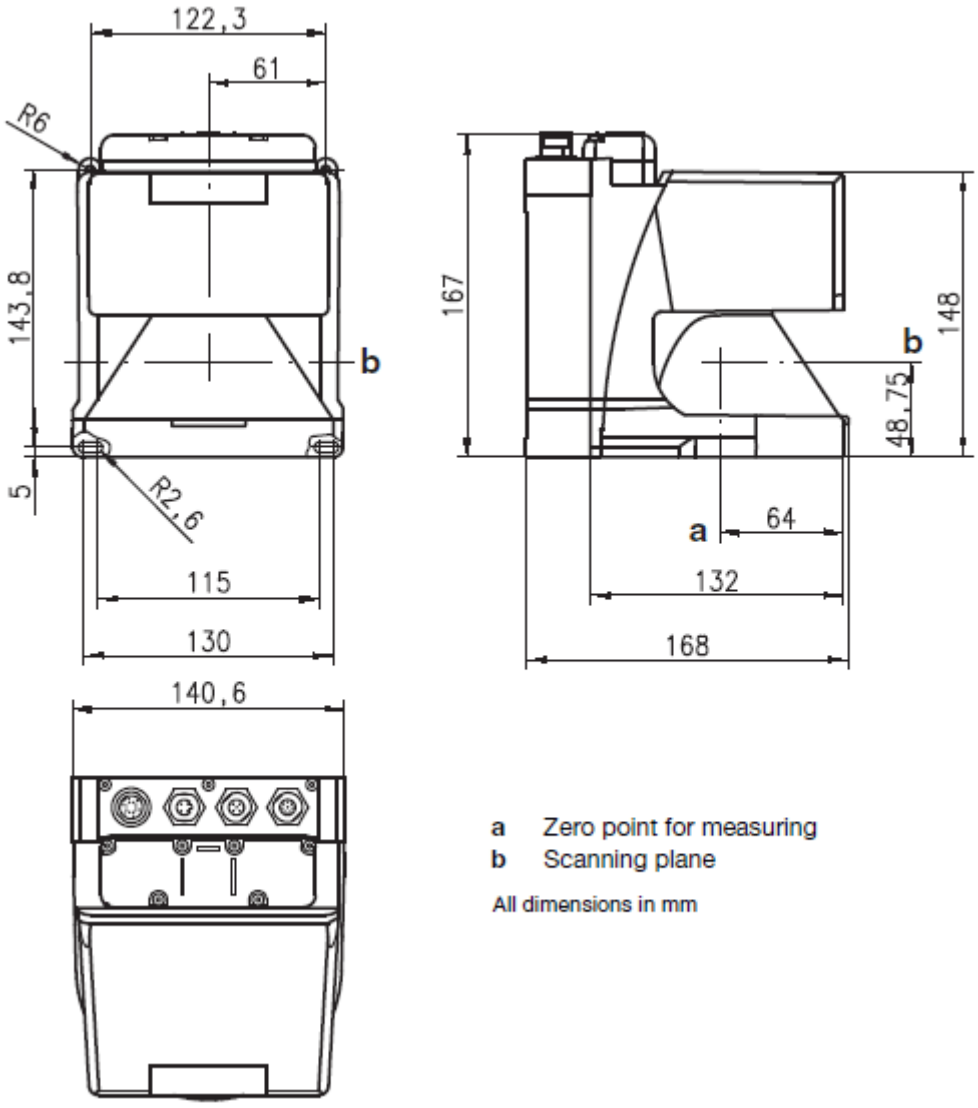
It is not necessary to install a driver. The communication between the AirScan and Widget Designer works over Ethernet and all necessary components come with the installation of Widget Designer.

Product Specifications

Ambient temp. (operation/storage) -0°C … +50°C/-20°C … +50°C
Weight: 2.3kg
Product Size (WxHxD): 141mm x 167mm x 168mm
Measurement data transmission: 100MBit/s

Optical data
Measurement range: 0-65m (recommended distance: 0-10m)
Angular range: max. 190°
Angular resolution: 0.36°
Scanning rate: 50 scans/s or 20ms/scan
Wavelength: 905nm
Pulse Duration: 3ns

Electrical data
Voltage supply: +24VDC
Overcurrent protection: fuse 2.5A
Current consumption: approx. 1A
Power consumption: < 75W



a Zero point for measuring
b Scanning plane

All dimensions in mm

AirScan Connectors and LEDs

Rear View with scanning direction going up: connections and LED indicators from left to right

Connectors	
M12 plug / Connector1	For special applications and not to be used.
M12 socket / Connector2	For special applications and not to be used.
M12 socket / Connector3	This is the 100MBit/s Ethernet socket to connect a network switch or a computer. Use the supplied Ethernet cable with the 4-pin D-coded connector for the AirScan and the RJ-45 connector for the switch or computer. The interface is configured with the IP address labeled on your device and the subnet mask is 255.255.255.0. Both can be changed with the "AirScan IP Configurator" which is available in the Download-Center on our web site.
M16 plug / Connector4	This is the power plug. Connect the supplied power cable with the 14-pin M16 connector to the AirScan and the XLR male connector to the XLR female connector of the low voltage power supply. Then connect its power cord. Note that the device powers on automatically and that there is no dedicated power switch.

Ethernet status indicator	
Left, red LED 1	Ethernet system ready
Middle, red LED 2	Ethernet connection present
Right, red LED 3	Ethernet data transmission active

General device status indicator	
Leftmost, yellow LED 1	- slowly (1Hz) flashing: internal warning message (e.g. due to soiled sensor cover) - faster (4Hz) flashing: internal error message - continuous light: for internal use
Green LED 2	Detection field up to 30m is free
Red LED 3	Detection field up to 30m is occupied
Yellow LED 4	Detection field over 30m is occupied
Rightmost, green LED 5	Sensor functions active

8.5 Controller Boards

The Jog/Shuttle Controller and the Fader Board Extension can be purchased as an optional accessory for Pandoras Box products.

The Jog/Shuttle Controller [1991] allows to control up to two sequences. Based on the selected sequence 16 cue buttons let you go to 16 cues directly.
The Fader Controller [1993] is the extension for the Jog/Shuttle Controller and allows to control up to 6 sequences.

See here information about the products Jog/Shuttle [1994] (discontinued) and Playback Extension [1995] (discontinued).

8.5.1 Jog/Shuttle Control

The Jog/Shuttle Controller is available to control up to two sequences.

Based on the selected sequence (use the select button for each sequence) the top 16 cue buttons let you go to 16 cues directly.
Based on the selection, use the Jog/Shuttle controller to navigate back and forth through the timeline.
The fader on the left will control one sequence; the fader on the right will control a second sequence.

In order to map the faders to sequences and buttons to cues in Pandoras Box, please open the Configuration tab > Controller Setup [154].

FADER CONTROLS

SELECT (Button above fader):

Selects the sequence. The blue led on the button will light as indicator for the selection.

As soon as a sequence is selected, all Cue Buttons as well as the Jog Dial and the Playback Buttons are linked to this sequence!

LED light status:
Blue LED is on: Sequence is selected.
Red LED is blinking: Sequence status is "Play"
Red LED is on: Sequence status is "Pause"

FADER:
Controls the sequence opacity.

Fader position on top = Opacity FULL
Fader position at bottom = Opacity 0%

PLAY/PAUSE (Button below fader):
Toggles the Sequence between Play and Pause. This Play/Pause Button can be used without having the sequence selected via the SELECT-Button.

CUE BUTTONS
For each sequence you may assign the 16 Cue Buttons via the Controller Setup Tab in Pandoras Box.
Depending on the selected sequence, the Cues Buttons are linked to this sequence.
The last Cue Button pressed will light up in blue.

JOG DIAL
The Jog Dial allows scrolling through the selected sequence:
The inner knob allows scrolling frame by frame, the outer know does a faster scrolling.

PLAYBACK BUTTONS
Depending on the selected sequence, the Playback Buttons are linked to this sequence.

LAST / NEXT CUE
These buttons allow to step backward / forward from cue to cue.

PLAY / PAUSE
This button toggles the selected sequence between the playmodes Play and Pause.

8.5.2 Fader Extension

The Fader Controller Board acts as extension for the Jog/Shuttle Controller [1991]. It allows controlling 6 more sequences.

In order to map the faders to sequences in Pandoras Box, please open the Configuration tab > Controller Setup [154].

FADER CONTROLS

SELECT (Button above fader):
Selects the sequence. The blue led on the button will light as indicator for the selection.
As soon as a sequence is selected, all Cue Buttons as well as the Jog Dial and the Playback Buttons are linked to this sequence!

LED light status:
Blue LED is on: Sequence is selected.
Red LED is blinking: Sequence status is "Play"
Red LED is on: Sequence status is "Pause"

FADER:
Controls the sequence opacity.
Fader position on top = Opacity FULL
Fader position at bottom = Opacity 0%

PLAY/PAUSE (Button below fader):
Toggles the Sequence between Play and Pause. This Play/Pause Button can be used without having the sequence selected via the SELECT-Button.

As soon as a sequence is selected on the Fader Extension, the Cue Buttons, the Jog Dial and the Playback Buttons on the Jog/Shuttle Controller are assigned to this selected sequence.
See the Jog/Shuttle Controller [1991] for more information about Cue Buttons, Jog Dial and Playback Buttons.

8.5.3 Jog/Shuttle Control (Discontinued)

Please note that this device is neither available nor supported since Widget Designer version 6 and Pandoras Box version 6.4.

The The Jog/Shuttle Controller is available to control up to two sequences.

Based on the selected sequence (use the select button for each sequence) the top 18 cue buttons let you go to cue 1-18 directly.

Based on the selection, use the Jog/Shuttle controller to navigate back and forth through the timeline.

The keyboard block on the left will control the sequence with the ID 1, the block on the right will control the sequence with the ID 2.

For each sequence choose:
SELECT: selects the sequence
ON/OFF: to hide all layers from the selected sequence
KEY LAST/NEXT
CUE LAST/NEXT
STOP PAUSE
PLAY

Use the "STORE" key to store all currently active value to the selected sequence.

Use the "CUE" key to store a new cue at the current time of the selected sequence.

Use "COPY" and "PASTE" for key frames in the selected sequence.

Use "RESET" to reset all devices to default.

Use "DELETE" to delete chosen keys in the selected sequence.

8.5.4 Playback Extension (Discontinued)

Please note that this device is neither available nor supported since Widget Designer version 6 and Pandoras Box version 6.4.

The Playback Extension enables a Pandoras Box Manager PRO to control up to 8 individual sequences directly, plus giving access to the command line architecture via the numeric keypad and dedicated command buttons.

Depending on the desired operation, any command line input must be started with a mode button:

"VIEW", "GROUP", "DEVICE", "SEQUENCE", "PRESET" and ends with pressing the "ENTER" key.

DEVICE CONTROL

Within the device tree, all items are organized in nodes that the devices belong to. In the properties you can assign a unique ID for both nodes and devices. For example if you have a Pandoras Box Server in the device tree, its node ID might be set to 1 and the individual layer ID count from 1 to 10.

DEV 1.5 Enter
to select layer 5 of node 1.

Once a device is selected, you may use the encoders to adjust the parameters values.
To select multiple devices you can keep adding devices to you current selection by simply selecting another one.
To clear your current selection press "CLEAR" once.
You may also use the "+" or "THRU" key to select multiple devices at once. For example: DEV 1.1 thru 1.10 Enter or DEV 1.1 + 1.3 Enter.

Once you start changing values with the encoders, wiper or joystick, all changed values will turn red and get indicated as active values. Active values will be processed during a store-operation. If a parameter is inactive it will not be stored at any time.
To de-activate and reset individual values to default, hold down the "RELEASE" key and turn the desired parameter encoder. In the device tree, the same operation can be achieved by right-clicking on the individual parameter and choosing "CLEAR ACTIVE". If you want to clear all active values you may also double click the "CLEAR" button. To reset all parameters back to default you may click the "CLEAR" button three times.

The same commands are also available from the right-click menus in the user-interface in the device tree or by right-clicking in the time-bar of a sequence.

Device Command Overview

[Preview DEV 1 Enter]
switches the Preview window to display the selected device

[Dev 1.1 Enter]
selects device 1.1 and allows access to encoders and parameters

[Dev1.1 + 1.2 Enter]
selects device 1.1 and 1.2

[Dev1.1 thru 1.11 Enter]
selects all devices from ID 1.1 to 1.11

[Align]
Based on the align setting <, >, <> or >< when multiple devices are selected you may align the parameters according to the align setting.

[Clear 1x]
Clear Selection

[Clear 2x]
Clear Active

[Clear 3x]
Reset All

[Release + Encoder]
de-activates and resets parameters to default of current selection

[Dev 1.1 Enter Enter]
sets all parameters of selected devices to active

[@@]
sets opacity value to 255 (100%)

GROUPS

If you need to control a single layer on multiple machines or to add the same values to multiple layers at once, the device tree control tab allows you to select multiple devices at the same time by clicking and holding the [Ctrl] key. All selected device icons will turn white, showing that they are selection members.

To release the current selection press [Esc] or [Clear] once.

Please note:
When multiple devices are selected, all changes of parameter values will apply to all members of the selection.

Creating & Editing Groups

Once a single or multiple devices is/are selected, you can store this specific selection as a group in the group bin of the project tab.

To create a group, please select the desired devices, then right-click on the group bin in the project folder and choose ‘create group' or type "Store GRP 5 Enter" to create a new group of the current selection with the ID 5.

To delete a group, right-click on the group element in the project and choose "REMOVE" or type "Delete GRP 5 Enter".

HINT: When working in device tree editing, you may also use the right-click menu on groups to determine if the members of the group should be activated, de-activated or reset.

PRESETS

Individual preset banks can be created by right-clicking on the preset folder in the project tab.
For each folder you may open a designated browser as a tab to be stored in views.
The presets are designed to hold a snap shot of active values, to be reused for later programming.

Presets apply to all selected devices. As source, the first stored values for each individual type of parameter are taken. When creating a preset with two layers that have different opacity values only the opacity value for the first

selected layer will be applied when assigning the preset on a selected device. Applying a preset always causes setting relevant parameters to active.

The preset folder contains default preset banks for active value filtering during store operation.

This means that by right-clicking on any of these categories only the parameters that match the category group will be stored there as a new preset.

Global	All Parameters
INT	Opacity, Transition FX
PRS	Position, Rotation, Scale
MEDIA	File Selection, Playback Controls
OBJ	Object Selection
COL	Colour FX
AUDIO	TBA
FX	Video FX
CTRL	Camera Control Settings

Preset Commands

[Store|ID.ID|Enter]
stores a new preset with specified ID.

[Delete|ID.ID|Enter]
deletes selected preset.

[ID.ID|Enter]
if no device is selected, the preset will recall the value to the stored selection
If devices are selected, the selected preset will be assigned to the entire selection.

SEQUENCE CONTROL

To create a new timeline, enter the following on the console:
"Store SEQ 1 Enter" or simply right-click on the sequences folder in the project tab.

Once the timeline is created, you will have access to it in the sequence tab by selecting it in the project bin.

Similar to the device control, all sequences can have a unique ID within the project. The ID can be set during the store operation from the console or via the properties of the sequence itself.

To load a SEQ into the Jog/Shuttle controller type "SEQ 1 Enter"

To load the SEQ into the sequence tab type "Edit SEQ 1 Enter"

To go to a specific timecode in the selected sequence type "Goto Time hhmmssff Enter"

Time-format is interpreted like this: Hours minutes seconds frames.

You can also type in "1000" => timecode 00:00:10:00.

To generate these new keys in the timeline, you will then have to right-click on the time-bar and choose "Store active" or type "Store Enter" to store all active values at the current time in the selected sequence.

SEQUENCE COMMAND OVERVIEW

Store|Enter – stores all active values at current time of selected sequence.

Store|Seq|X|Enter – stores new sequence with ID X.

Store|Cue – stores new cue at current time of selected sequence.

Delete|Seq|X|Enter – deletes specific sequence.

Seq|X|Enter – selects Seq X and loads to Jog/Shuttle Controller.

Goto|Time|XX|Enter – jumps to Timecode XX. Use syntax HMSF or e.g. "105" for Timecode 00:00:01:05

Goto|Cue|XX|Enter or Goto|XX|Enter – jumps to Cue XX

Goto|Time|+/-|HMSF|Enter – relative movement in the current sequence with the time value HMSF.

Store|Time|HMSF|Playback button – stores active values to the sequence controlled by playback button.

Store|Cue|XX|Enter – stores active values at time of cue XX in selected sequence.

Store|Cue|XX|Time|HMSF|Enter – creates cue with ID xx at time HMSF in selected sequence.

Store|@|Cue|XX|Enter – stores active values and creates cue XX in selected sequence.

Store|@|Cue|XX|Time|HMSF|Enter – stores active values and creates cue with ID XX at time HMSF in selected sequence.

Delete|Cue|ID|Enter – deletes selected cue.

Delete|Cue|ID|Thru|ID|Enter – deletes cues within selected ID range.

Delete|Time|HMSF|Enter – deletes all keys at selected time.

Delete|Time|HMSF|Thru|HMSF|Enter – deletes all keys in selected time range, independent of activation of selection of devices.

Delete|@|Cue|ID|Enter – deletes all keys at time of the selected cue ID.

Delete|@|Cue|ID|Thru|ID|Enter – deletes all keys in selected sequence which are between the entered Cue IDs.

Delete|TYP|...|Enter, with TYP @, Cue, or Time – can now be applied to another sequence than the selected sequence, if directly after "DELETE" follows the number of the sequence.

View Commands

Store|ID|Enter – stores the current view-layout as a new view with specified ID.

Delete|ID|Enter – deletes selected view ID.

8.6 DMX Link

The DMX Link is a DMX - USB interface offering an easy setup for DMX 512 control.

Only the Pandoras Box Master can send or receive DMX data. There are two versions available, the DMX Link In and the DMX Link Out. Please note that it is not possible to use gender changer adapters to turn the signal flow!

The DMX Link In allows to remote control Pandoras Box via a DMX device. In other words, it provides direct sequence and layer control from any common lighting controller.
Connect the DMX Link to your Master system. In case of remote controlling Clients, drag them from the Asset tab [131] or Device Types tab [183] into the Device Tree tab [173]. Patch the layers in the Patch tab [228] to the according channel, subnet and universe. There is no need to have another DMX-Link connected to a Client, it is controlled by the Master. Activate the DMX input in the Configuration tab [139] and as soon as data is sent to the Master system, all patched layers or sequence parameters are remote controlled.

The DMX Link Out allows to remote control DMX devices with Pandoras Box. It is mostly used to control moving lights, spots etc. with the Pandoras Box timeline for a synchronized video and light show.
Connect the DMX Link to your Master system. Drag the built-in or custom designed DMX devices from the Device Types tab [183] into the Device Tree tab [173]. Patch the layers in the Patch tab [228] to the according channel, subnet and universe. Activate the DMX output in the Configuration tab [139] and as soon as a key is stored in the timeline [292], the DMX data will be sent constantly to the devices. Only changes to DMX values will be sent. Resetting a DMX node will send all values at once.

Please be aware that a USB DMX interface will allow you to control a maximum of 512 DMX channels on subnet 0 and universe 0.

LED Status:

magenta	no driver found; please install the current Pandoras Box driver for USB devices
blue / magenta pulse	the driver is in idle state; please start a Pandoras Box Master software and setup to receive or send data as described above if using a DMX Link In, this status means that the connection between the DMX Link and PB software is not setup
blue / red pulse	only for DMX Link In; in contrast to the above, this status means that the connection between the DMX Link and the sending device is not alright
blue blinking	ok, the DMX Link is sending / receiving data

8.7 SMPTE Link

The Pandoras Box Master can send and receive LTC SMTPE via the USB SMPTE Link interface.

To connect the SMPTE Link please refer to the Configuration tab [139], section SMPTE Time Code [152]. Afterwards, the Sequence Inspector [204] allows to set up the "Mode", "Offset" and "Stop Action".

The SMPTE Link Interface provides balanced signal transmission:

Pin1: ground
Pin2: signal
Pin3: signal

It does not matter which signal is +/ -.

8.8 NET Link

The NET Link is a modifiable interface device build to provide sensor information as input signals and / or to control relays by sending output signals.
Those signals are processed by the software Widget Designer [786] that is connected to the NET Link via network. Widget Designer is an interaction software and can be for example programmed to control layer and sequence parameters in Pandoras Box [67]. In return, Pandoras Box, or any other device connected to Widget Designer, can trigger the output relays.

The hardware features [2002] are explained in the following chapter in detail. In short, the NET Link is a customizable device and comprises:
- a housing, also called base unit: the small chassis depicted above
- min. one processor unit
- changeable boards: Christie delivers the NET Link ready-to-use as ordered by you. However, you may modify the NET Link by changing the boards at any time. Currently, eight boards are available covering analog and digital boards as well as input and output ones. Show examples... [2004]

The "Calibration Link" is a NET Link device equipped with two analog fibre input boards for the input of 16 fibre cables (single-core, diameter of 0.98mm²). It is specially designed for an automated re-calibration of a projection setup, e.g. in a fixed installation where a projector or a screen have been moved. Show example... [2004]

The NET Link supports a sensors range from standard industrial encoders, distometers as well as analog potentiometers and contact closures to specially designed Phidget sensors or self-build solutions. As long as a sensor can be connected to one of the offered boards, meaning that it meets the technical specifications, its data can be send to Widget Designer. More details regarding the software [2008] are included on the following pages.

Discontinued Housing and Distinguishing Version 1 and 2

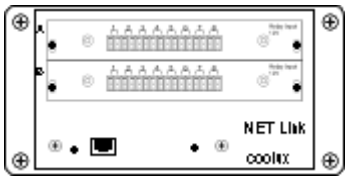
Currently, the smaller 4 slot high base unit is the only housing available. The 19" rack module is discontinued.

Furthermore, the current version for the housing, processor and boards is version 2. The front panel is labeled with "Pandoras Box" and the power input is 24V. The former version 1 sold via coolux was labeled with "coolux" on the front plate and had a 12V input. Bear in mind that **it is not possible to mix parts from version 1 and 2**!

8.8.1 NET Link Hardware

This chapter explains the current hardware of the NET Link. Please see the introductory chapter [2000] for other information and discontinued versions.

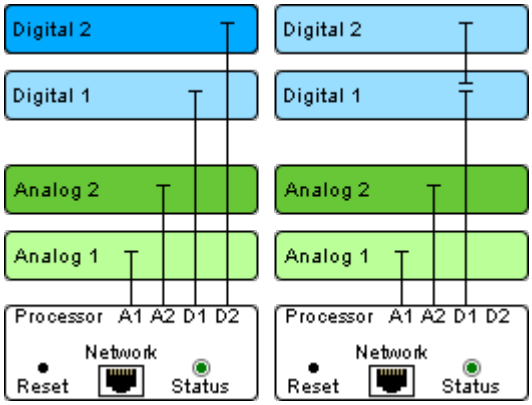
The Housing and Power Supply



As mentioned in the previous overview chapter, the NET Link is a customizable device.
The dimensions for the base unit are: H: 85mm, W: 172.1mm, D: 121mm. It is 4 slots high. The modules are mounted horizontally.
Inherently, the housing comes with one processor unit (2 slots high) that cannot be removed. Hence, there is space for two custom boards that you may choose from eight available board types, each 1 slot: analog / digital and input / output ones.

The NET Link is powered with 24V (note that the first version [2000] was powered with 12V!) . The AC adapter has an input of 100-240V, max. 2A and an output of 24V, 3.75A via a 4pin-XLR plug.
The allowed voltages and ampere for sensors depend on the board. Some boards do not provide power for the connections, thus an external power supply is needed!

The Processor Unit and Cabling



The processor is the part of the NET Link responsible for the communication to connected software [2008], i.e. Widget Designer [786].
It takes up the space of two slots and consists of:
- a network plug with a network adapter that can be set to any IP address and port number
- a "Reset" button to apply the default IP address (192.168.178.222), subnet mask (255.0.0.0 and 5000). To reset the device, hold the reset button down whilst recycling the NET Link, i.e. re-connecting the XLR power plug at the back.
- a "Run" LED indicating the status of the unit. The green LED blinks once a second if everything is alright and net work packages are send correctly. If the flashing is slower, the processor unit does not perform properly because network packages can not be send in the expected way. This could be caused by a faulty network environment like cables, switches or IP settings. Please check the status after approx. 100 seconds after establishing the communication between the NET Link and Widget Designer.

In addition it manages the connected input / output boards. The processor has four connectors: A1, A2, D1 and D2, available for the connection to the boards.

A1 and A2 can be connected to one analog board each. Currently there are only analog input boards available. In other words one processor can be connected to max. two analog boards.

D1 and D2 can be connected to digital boards. There are input and output boards available. A digital board has a connector itself to be linked to another board. The daisy chained boards must be of the same type (input or output) and in theory they must not exceed the maximum number of 16 but only two boards fit into the base unit. The advantage of daisy chaining the boards is that switching relays is absolutely in sync whilst switching relays on D1 and on D2 could have a very small delay. So, if you have one input and one output board, you have to connect them to D1 and D2. If you have two input boards or two output boards you can either connect them in the same way (to D1 and D2) or to only one connector and daisy chain them.

Board modules

Each board is depicted and described in detail in the next chapter. As well there are examples in which application they can be used.

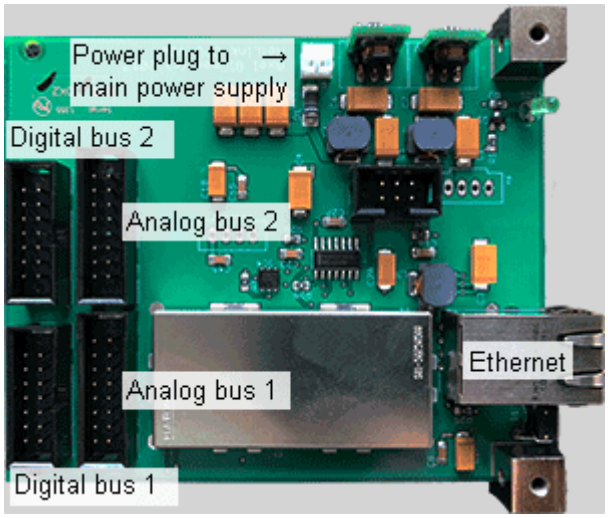
Analog boards - Input modules
- Calibration Fiber Input
- 0 - 5V Input
- 0 - 10V Input
- 0 - 5V Sensor Input

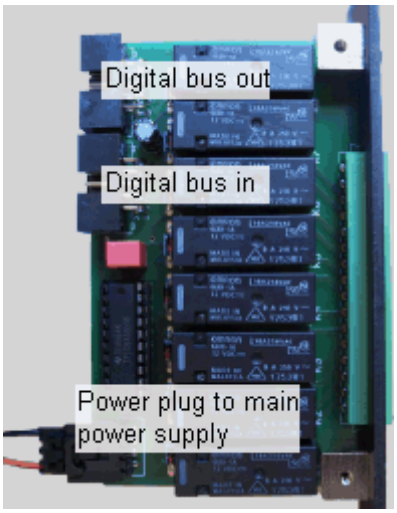
Digital boards - Input modules
- 12V Relay Input

- 24V Relay Input
- Encoder Input

Digital boards - Output modules
- 48V AC / 30V DC Relay Output

Every board is 1-slot-wide. As explained above they are connected either to the processor unit itself or to a preceding board. In addition it has a power connection plug.





8.8.2 NET Link Boards and Examples

This chapter explains the available boards for the NET Link and shows examples how to use them. Please see the introductory chapter [2000] for other information.

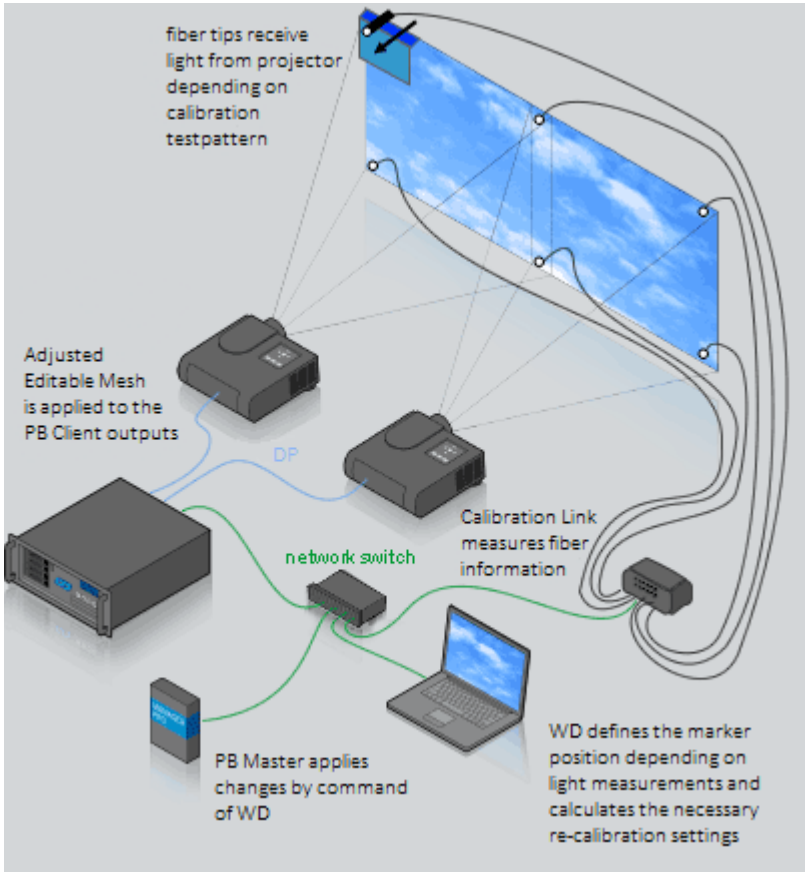
Analog board - Input modules

Currently there are only analog input boards available, no output ones. Each board has 8 galvanically isolated connectors. They detect the incoming voltage in specific, gradual steps providing detailed data information from the connected sensor. Software-wise this is expressed through a value range from 0 to 1023. An application would be a light sensor controlling automatically the brightness of the projection. Principally it is like in the below depicted example with the buzzer, only that the buzzer's place is taken by the sensor and that the switch closes gradually instead of open / close.

Please find below a list with all available input modules:

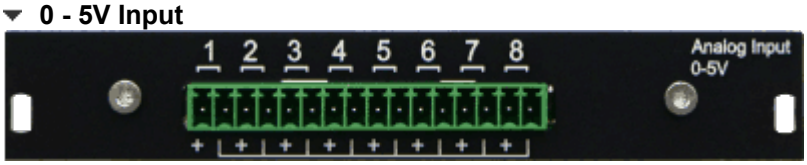
Calibration Fiber Input

8 fiber connectors for single-core (!) fiber cables with a diameter of 0.98mm²
cables obtainable at your local distributor
DIP switch on the printed circuit board that sets the sensitivity: coarse / fine; fine is default and should be only changed when measured values exceed the value range; coarse mode has the same value range but roughly halves the measured value



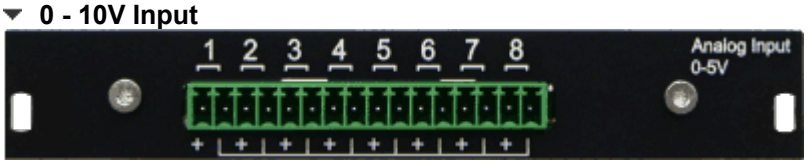
The depicted example is explained in detail in the chapter "Projector Calibration [1496]".

0 - 5V Input



8 connectors, galvanically isolated
phoenix plug (for two laces per connector ≤ 2.5mm²)
maximum voltage of 5V
30kΩ input impedance
value resolution (in software): 5V / 1024 = 4.88mV per step
external power source for sensors is needed
could be used for GPI In if 5V are used; an If-node could filter all values > 0 for "On"

0 - 10V Input



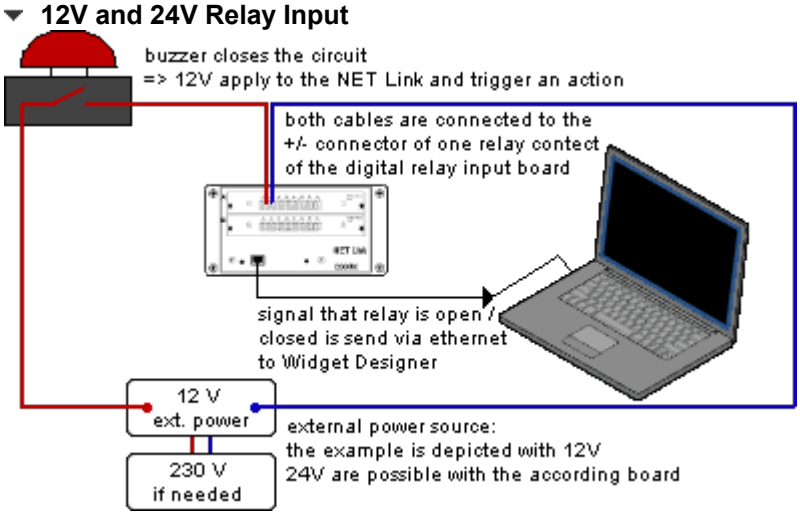
8 connectors, galvanically isolated
phoenix plug (for two laces per connector ≤ 2.5mm²)
maximum voltage of 10V
30kΩ input impedance
value resolution (in software): 10V / 1024 = 9.77mV
external power source for sensors is needed

0 - 5V Sensor Input

8 connectors, galvanically isolated
dedicated separate Molex plugs (for 3 laces per connector ≤ 0.5mm²; Black: Ground 0V, Red: Power +5V, White: Signal 0-5V)
maximum voltage of 5V
30kΩ input impedance
value resolution (in software): 5V / 1024 = 4.88mV per step
provides internal power source of 5V for each connected sensors

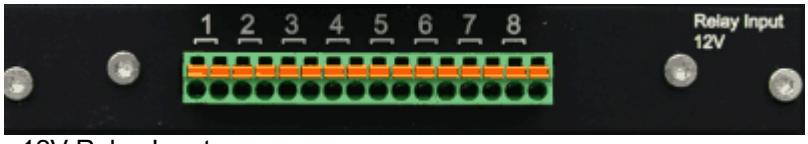
Digital boards - Input modules

12V and 24V Relay Input



There are two digital input boards: one for 12V and one for 24V. Both have 8 galvanically isolated connectors. They detect the incoming voltage in two steps and provide the information, whether the connected sensor is in On- or Off-mode. Software-wise this is expressed through the two values 0 and 1. If the voltage exceeds 80%, the

status switches to "On". If the voltage falls below 20%, it switches to "Off". An exemplary application would be a buzzer that triggers a cue.
If you need to work with 230V, a power circuit breaker is needed for 12V / 230V.



- 12V Relay Input
  8 connectors, galvanically isolated
  phoenix plug (for two laces per connector ≤ 2.5mm²)
  maximum voltage of 12V
  30kΩ input impedance
  "On" or "1": > 9,6V
  "Off" or "0": < 1,2V
  external power source for sensors is needed
  operating lifetime: 100 000 switchings
  could be used for GPI In if 12V are used; for the commonly used 5V, the analog input would apply

- 24V Relay Input
  8 connectors, galvanically isolated
  phoenix plug (for two laces per connector ≤ 2.5mm²)
  maximum voltage of 24V
  30kΩ input impedance
  "On" or "1": > 19,2V
  "Off" or "0": < 2,4V
  external power source for sensors is needed
  operating lifetime: 100 000 switchings

Encoder Input

4 x 5-pin M12 connectors, with following pin assignment for female connector 

Pin 1 = positive power supply, +10V to 30V
Pin 2 = Pulse signal B
Pin 3 = negative power supply, Ground
Pin 4 = Pulse signal A
Pin 5 = Pulse signal N, Reset

The Encoder Input Board replaces the discontinued Sensor Link device [2033] and is suitable to be connected to rotary sensors, e.g. encoders from Wachendorff (www.wachendorff.com) or another company. Whilst the Sensor Link could handle 7.000 steps per second, the new board can handle roughly 100.000 steps and thus can be connected to encoders that are 15 times faster or more precise.

If you for example choose an encoder with 4096 steps per rotation the board input could process 1480 rotations per minute, which equals nearly 25 rotations per second. If the encoder turns faster (or another encoder with a higher resolution is used) the board will lose track of the absolute step count of the encoder. This leads to drifting and wrong values.
101.000 steps/sec : 4096 steps/rotation = 24.7 rotations/sec => 1480 rotations/min

Due to the output circuit data of the encoder (which is "HTL" in Wachendorff sensors) there are four shoulders for one circuit. That means for our example: An encoder with 1024 cycles (or "PPR" = Pulses per Revolution, or "LPR" = Lines per Revolution) generates 4096 steps per second at one full rotation per second. This can also be called "CPR" = Counts per Revolution.

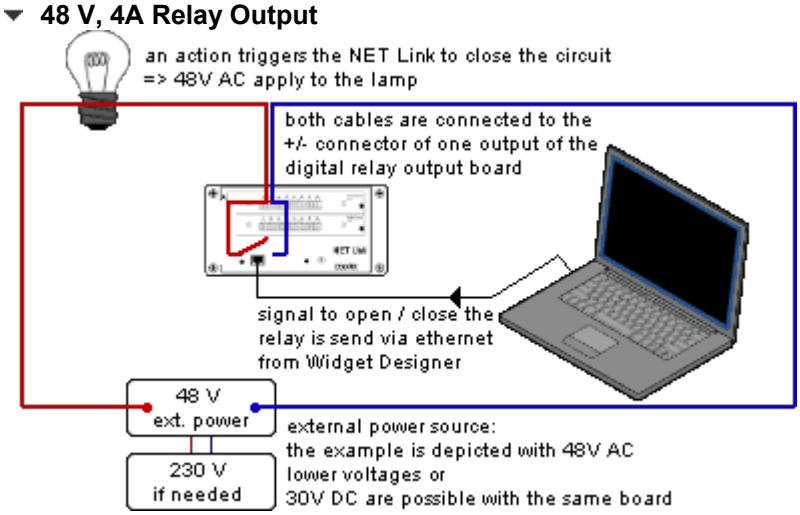
To calculate the precision of the encoder, we need to divide 360° by 1024 PPR which results in a 0.35° rotation angle.

To summarize the result of the two calculations above: the higher the resolution (steps) of the encoder, the less rotations per second will be possible to be processed in the Encoder board flawlessly but a finer resolution can be reached. Higher frequencies are more likely do lead to issues and shorter cable length. A cable length of 20m is roughly the maximum when operating at full frequency of 100.000 steps/s, while longer distances can easily be handled by reducing the data. Hence, it does not make sense to choose an encoder with the highest resolution available but one with the resolution you really need and benefit from optimized data processing and cabling.
On the other hand it is absolutely okay to use an encoder that you already have even though it offers more resolution than needed as long as you are not operating at full frequency of 100.000 steps/s and the cable length and quality suits your needs. In that scenario it is recommended to reduce the data processing in Widget Designer so that it outputs only the data to other soft- or hardware that is really needed. You could do this for example with the Filter nodes "Round [1132]" or "Divide [1122]".

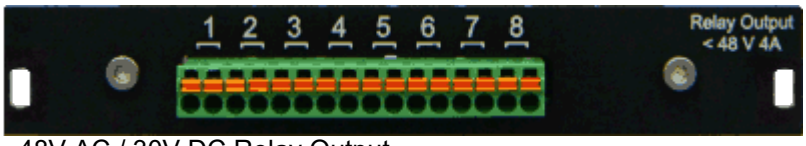
Please see the chapter Encoder Inputs E1 and E2 [2035] from the Sensor Link chapter to see how to order a suitable Wachendorff encoder. Bear in mind, that the resolution from the board is 100.000 steps. If you like to use encoders from another company, please contact our support team. In general a sensor must meet these requirements:
- Encoder input voltage: 10-30V via 2 pins as depicted above
- Encoder signal output: - must be via 3 pins as depicted above (e.g. ABC, ABN, ABM)
  - "on" state must have more than 6V
  - HTL signal (TTL only with above requirements regarding voltage)

Digital boards - Output modules

48 V, 4A Relay Output



Currently there is one digital output boards. It has 8 galvanically isolated connectors / contact closures. They close the circuit at the software's command.
All boards connected to the same digital bus are synchronized . If the maximum number of 16 daisy-chained output relays are connected to D1 (or D2), 16 x 8 = 128 contacts can be closed in sync. D1 and D2 can have a delay of maximum one frame.



- 48V AC / 30V DC Relay Output
  8 connectors, galvanically isolated, polarity can be reversed
  phoenix plug (for two laces per connector ≤ 2.5mm²)
  maximum voltage of 48V AC or 30V DC and maximum ampere of 4A
  30kΩ input impedance
  contacts are normally open (NO)
  external power source is needed
  could be used for GPI Out if external voltage is used as commonly done in broadcast environment; for other application, an external power source must be considered somehow

8.8.3 NET Link Software

This chapter explains the software settings for controlling the NET Link and getting information from it. Please see the introductory chapter [2000] for other information.

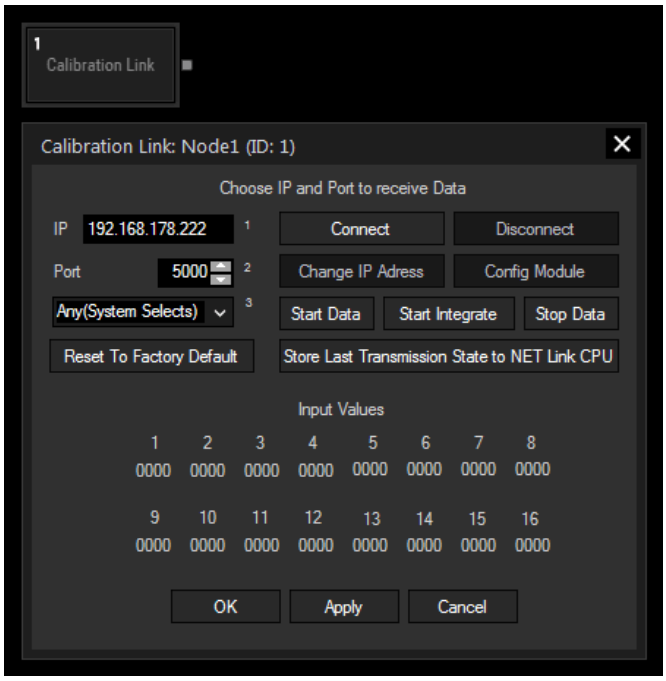
Software-wise the NET Link is included in the Widget Designer. The communication between WD and the device is based on an ethernet connection, not USB.

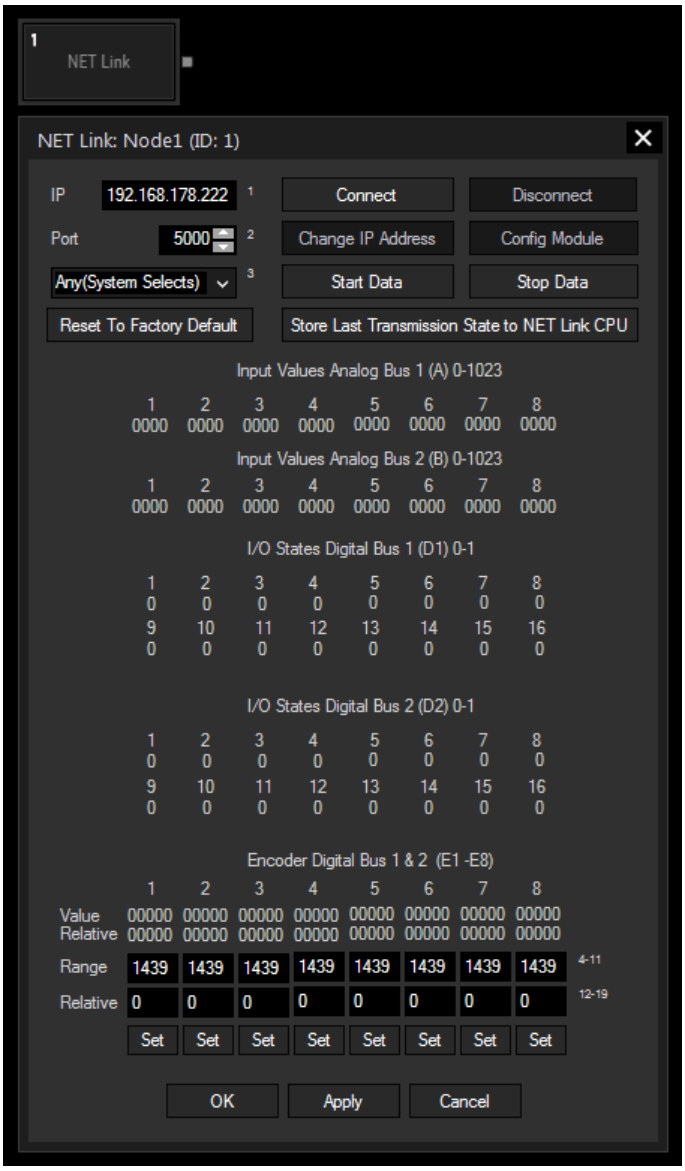
Nodes > Input > Devices > ... or Nodes > Output > Devices > ...
Relay Output node [1202]: receiving information from other nodes and controlling the connected NET Link, if it is mounted with (digital) relay output boards [2007]

Calibration Link Input node [981]: receiving information from the connected Calibration Link or NET Link with analog input board for fibers [2004]; values are transferred to other nodes; the Calibration Input works with the tool "Projector Calibration Manager [1496]"

NET Link Input node [1016]: receiving information from the connected NET Link (or Calibration Link) according to the input boards and transferring this information to other nodes

All nodes are controllable with node commands [1928]. Simply start a script with "Node1." for example, and Widget Designer will offer available commands. Using them, the NET Link input node turns into an output node as well. For instance, this command closes a relay. `Node2.SetRelayDigitalBus1(ID,State)`





8.9 Discontinued Accessories

8.9.1 DMX Link 8

Please note, that Christie Digital Systems has discontinued the DMX Link 8. Please refer to ELC and their product dmXLAN Node 8GB.

The DMX link 8 converts up to 10 DMX universes to Ethernet (Art-Net protocol).
Backwards, 8 DMX universes can be converted from Art-Net to DMX.

Furthermore it can be used stand alone as a universal programming digital merger (ex. 2 in 8 out) or splitter (ex. 2 splitters of 1 in to 4 out). Saving the lastly received signal DMX signal in the unit, it also acts as a perfect backup (ex. 4 in 4 out).

FEATURES

- Remote programming and monitoring
- DMX input options like Normal or Backup
- DMX output options like Normal, Merging in HTP/LTP or custom /Priority merging, Merging with channel by channel softpatch
- Remote takeover of fixture parameters
- Visualizer input support like WYSIWYG, ESP, Capture and many others

HARDWARE DESCRIPTION

Front Panel

The front panel has a graphic LC-display, 3 keys and a jog-encoder with push function and 3 LEDs.

Connections at the back

The dmXLAN node8 has many connectivity options. Viewed at the back from left to right:

RS-232 (top SUB-D 9pin)	Pin	Description
To connect the RS-232 port to a PC you'll need a 1 to 1 male/female sub-d 9pin cable. (NO null-modem / cross cable)	1	-
	2	TX
	3	RX
	4	-
	5	Ground
	6-9	-

Contact Closure Inputs (bottom SUB-D 9pin)	Pin	Description
These inputs have an internal pull-up resistor. A simple switch can be connected between the input pin and a ground pin. The contact inputs can be used to temporarily recall a patch setting.	1	Ground / Shield
	2	Input 1
	3	Input 2
	4	Input 3
	5	Input 4
	6-9	Ground

Network 10/100 MBit Ethernet (on Neutrik Ethercon)
The network connection uses a standard UTP network connection.

USB 2.0 Full-speed
The USB connection uses standard USB pinout.

DMX port 1 – 8	Pin	Description
The DMX ports 1 to 8 are bi-directional(via programmable setup), have an internal terminator resistor and are RDM prepared.	1	Ground / Screen
	2	Data -
	3	Data +
	4	-
	5	-

DMX port 9 and 10	Pin	Description
DMX ports 9 and 10 are fixed DMX inputs and have a fixed internal terminator.	1	Ground / Screen
	2	Data -
	3	Data +
	4	-
	5	-

There are 2 versions of node8 available, the normal and the Full Isolated (FI). The full isolated version has an optical/galvanic isolation barrier on all DMX ports (up to 1500V).

CONTROL THROUGH FRONT PANEL

Main Display

A Select config menu, to recall 1 of 4 previously saved configurations (patches)
B View DMX values
< (Setup) Enter setup menu
Encoder +/- Select information displays

INFORMATION DISPLAYS

DMX port status

x > port does not receive data (or DMX when input)
v > port receives data (or DMX when input)

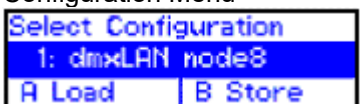
ID setting

IP address

Version and serial number

This product may only be used for controlling dimmers and moving lights. Using the product out of these specifications will remove all responsibility from the supplier

Configuration Menu



From the main display press A (config). Now you can either recall a previous setup or store the current one in one of the 4 options. Use the encoder to select a setup:

Press A to load the selected configuration
Press B to store the current configuration
Press _ to cancel the operation

The 4 contact closures on the back can be used to temporarily recall any of the 4 setups. The configuration is loaded on closing the contact and the normal configuration is loaded when the contact is opened. Contact closure 4 has the highest priority. Any change to the configuration, either thru the menu or via remote software (dmXLAN) is stored in the temporarily selected configuration.

DMX View Menu

small bars (120ch)

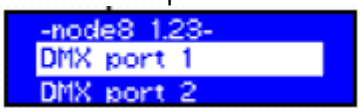
large bars (40ch)

decimal (6ch)

From the main display press B (View DMX). Now you can view the DMX values of each port (input or output). Use the encoder to change the channel:

Press A to select a different port
Press B to change the view mode
Press < to exit

Setup menu
Enter the setup menu from the main screen by pressing <



By using the encoder select the item you would like to adjust.

Options:
· Edit DMX port 1 to 8 or DMX input 9 and 10
· Set the ID
· Rename the current configuration
· Change network settings (IP address and netmask)
Edit the item by pressing A or the encoder.
All settings will be stored AFTER exiting the setup menu via the < key, a message will appear.

Edit DMX ports 1 to 8 or DMX input 9 and 10
Use the encoder to change the current selected option and press the encoder to select the next option.
Press A to store the configuration
Press B to set the DMX port to it's default value
Press < to cancel any changes

DMX port as output
When a port is set up as a DMX output, then you can select the working mode of that port.
Modes are:
· disabled > the DMX output is disabled
· outzero > the DMX sends out a DMX test signal with all channels at 0%
· single > the output sends out the DMX values of the selected (primary) Art-Net universe. The DMX output is enabled the first time when it detects the DMX universe on the network (or internal DMX inputs). If the universe on the network fails, the DMX output will go into DMX hold.
· dual HTP > like single, but merge two DMX universes (primary and secondary) in highest take precedence.
· dual LTP > like dual LTP, like dual HTP but in latest takes precedence (on a channel by channel basis)

DMX port as input
When a port is set up as a DMX input, it can be operated in the following modes.
· disabled > the DMX input does nothing
· normal > any valid DMX is sent onto the network (and internal use) using the

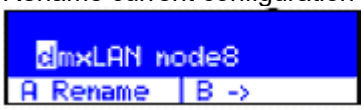
selected universe (Art-Net)
· backup > the DMX input will not send data if the selected universe is present on the network. If the selected universe is not present on the network for several seconds and the input has valid DMX, then it will send DMX data. If another sender starts sending the same universe, the input will go back input backup mode. Please note. DMX input 9 and 10 are always fixed to work as input.

Set ID

Used the encoder to change the current selected digit, and press the encoder to select the next digit. The digits are in hexadecimal (16 possibilities 0-9, A-F). This means you can have up to 4095 different IDs in the system (ID 000 means that the ID is not used).

Press A to store
Press B to disable (---)
Press < to cancel the operation

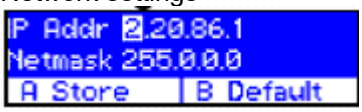
Rename current configuration



The name of the current configuration is presented in the main display and in the dmXLAN software. The name is also used to indicate the configuration setting when it is stored in one of the 4 built in configurations.
Use the encoder to change the current character, and press the encoder to select the next.

Press A to store the name
Press B to select the next character
Press < to cancel the operation

Network settings



Use the encoder to change the current parameter, and press to select the next.

Press A to store the configuration
Press B to select the factory default IP address and Netmask
Press < to cancel the operation

GENERAL INFORMATION

CE – Product
The DMX Link 8 permits to the CE requirements set up by the European Community.
This can be recognized by this label on the outside of the product.

CE

Technical Specifications:

Power: 85-264 VAC 47-440 Hz 10VA max
Dimensions 19" 1HE rack unit 483 x 44 x 150 mm
Weight 2kg

8.9.2 DVI Input Cards (Discontinued)

Please note, that Christie Digital Systems has discontinued the DVI Input Card. For other options, please see the chapter Input and Output Cards [1947].

This chapter describes the optional DVI input cards For other input or output boards, please see the introductory chapter [1947].

There are two DVI input cards available: Single and Dual.
The Single DVI card has one input whilst the Dual DVI card has two.

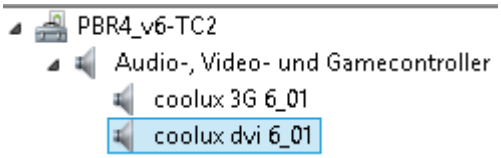
Both cards allow to input DVI signals (Single- and Dual-Link) to be used within the Pandoras Box render engine. However, please note that using a Dual-Link signal with the Dual (channel) DVI card deactivates the second input!

Supported Video Formats

The DVI capture cards support DVI-D (DVI Digital) as well as DVI-A (DVI Analog). Furthermore, Single-Link and Dual-Link DVI signals are supported which allows all common signals like FullHD 1920x1080p60 but also custom resolutions, e.g. 3840x1080p60. The following table lists the maximum supported video formats (based on a 16:9 or 4:3 image ratio). Please note that HDCP-protected signals can not be processed! Audio formats are also not supported.

Single and Dual DVI input card	
Analog RGB	up to 1600x1200 @ 60Hz, 1920x1080 @ 60Hz
DVI/HDMI	up to 1600x1200 @ 60Hz, 1920x1080 @ 60Hz (Single-Link DVI), up to 2560x1600 @ 60Hz, 1920x1080 @120Hz (Dual-link DVI)

Driver



The card drivers are preinstalled on your Server or Player hardware. All drivers can be found locally under `C:\system` and in the Download-Center where you can also check for updates. The driver installer searches automatically for existing cards and installs the appropriate version.

The picture shows the folder "Audio-, Video and Gamecontroller" in the Windows Device Manager after installing a 3G-SDI and Dual-DVI card card using the driver 6.01.

Starting with Pandoras Box version 6.4.1 we are supporting the original Deltacast drivers (version 6.14) which we have added to our driver package in the Download-Center.
The 12G-SDI board [1972] can only be used with driver version 6.14 (or later).
The DVI board [2014] (and old HD-SDI board) can only be used with driver version up to 6.01.
The 3G-SDI board [1969] and HDMI board [1978] can be used with both versions.

When having two or more cards in a single computer they have to run with the same driver version. Therefore, a DVI board cannot be used along with the 12G board in a single computer. Using different driver version on separate machines is not a problem.

Working with Video Inputs in Pandoras Box

Video Inputs in Pandoras Box are managed by the Pandoras Box Master, that is responsible for the entire show control. Only if you use the Master in stand-alone mode, you would connect the video device to the Master's input

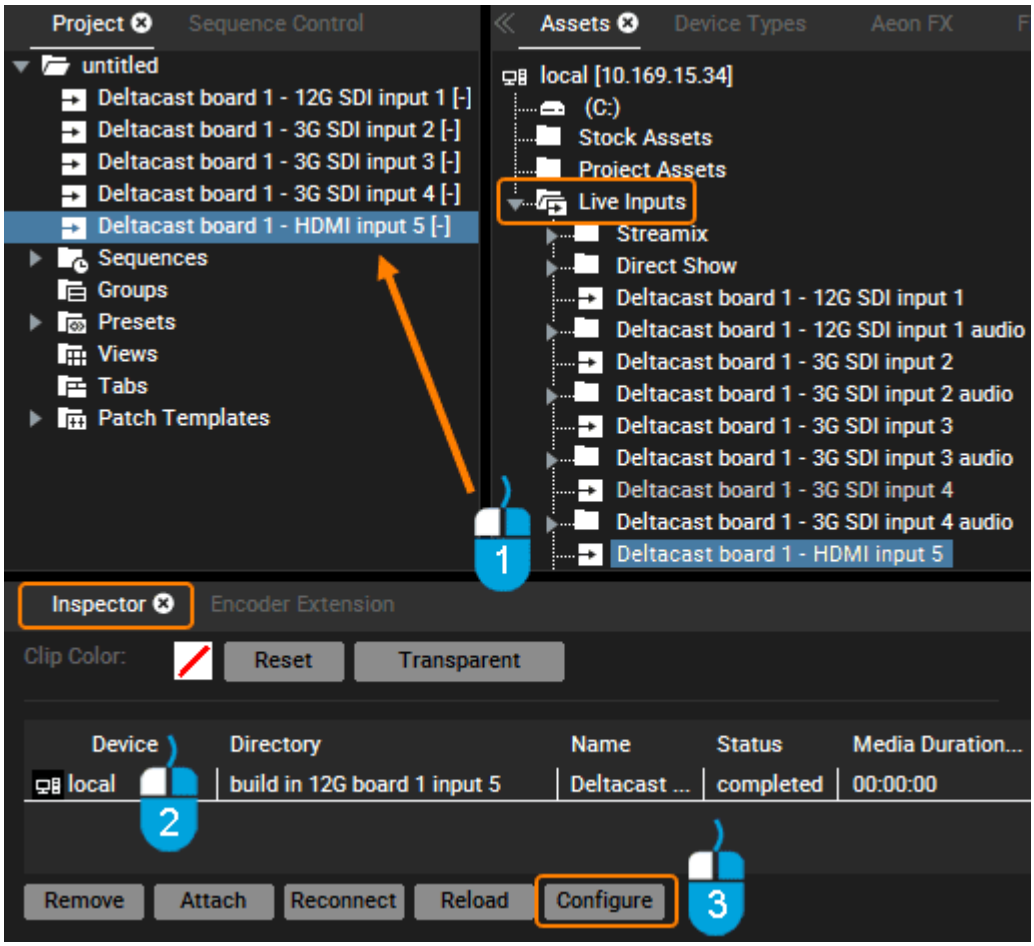
board. In case you work in a Master-Client-setup and you like to use the stream on a Client, the Client must be equipped with an input board (not the Master). In other words, the physical signal input happens on the same machine which should render and output the image. The video stream only "exists" on the Client, it is not distributed to the Master nor to another Client. If you like to display the stream on several Clients, all of them must be connected to the video source.

Follow this step-by-step description after having connected the video source to the input card.

Master-Client-setup:
- start the Master and Client [316] computer and make sure that Pandoras Box runs on both in the correct Master or Client mode and the same version
- then go to the Master, Asset tab [131] > your Client > folder "Live Inputs"
- drag the according card into the Project tab [278]; this initializes the card
- if needed, you may configure the card: make sure, that the PB Client is not in fullscreen, click the button "Configure Live Inputs" and choose the input that you want to configure from the live input list
- assign the live input to a layer in the Device Tree [173] the same way as using common image or video files

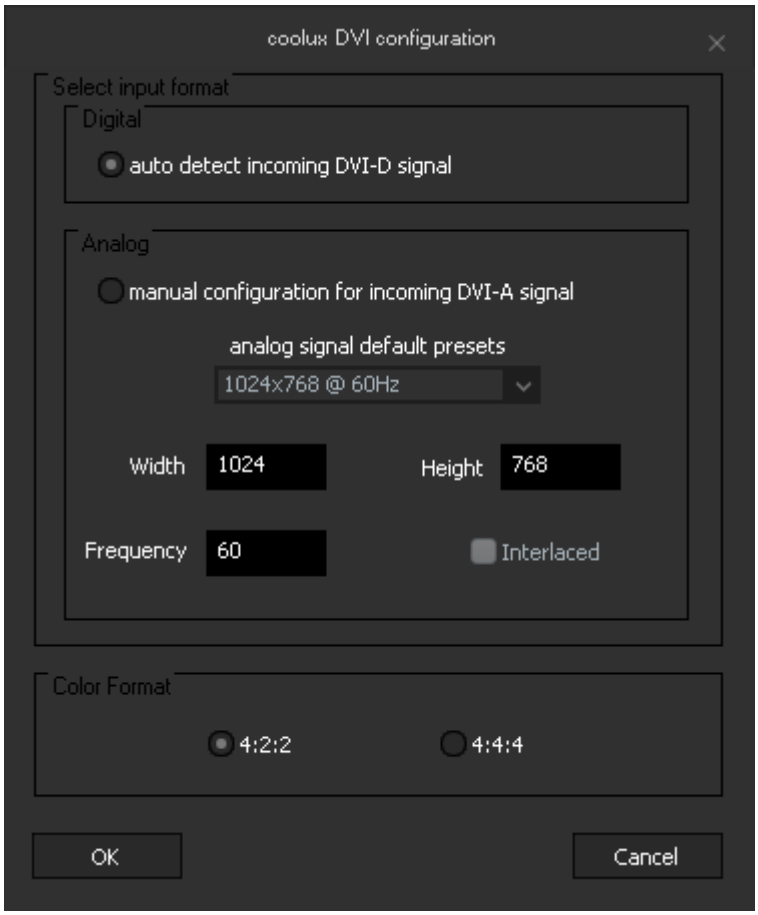
It is also possible to record the incoming signal. However, the Client computer must then be started in stand-alone mode, that is with the Pandoras Box Master software. Please see the chapter "Audio and Video Recording [137]" for more information.

Stand-alone setup:
- start the Master computer and Pandoras Box in Master mode
- then go to the Master, Asset tab [131] > your Client > folder "Live Inputs"
- drag the according card into the Project tab [278]; this initializes the card
- if needed, you may configure the card: select the input in the Project tab to display its properties in the Inspector [190]. Scroll down in the Inspector to see the file table where you can select the live input and click the "Configure" button below (the below image shows the inputs from a 12G card)
- assign the live input to a layer in the Device Tree [173] the same way as using common image or video files



Configuring the Video Input

As described above:
To configure the input on the Client side, please click on [Configure Live Inputs] and choose your input from the list.
To configure the input on the Master side, please select the input in the Project tab and Inspector and click on [Configure].



Select "auto detect incoming DVI-D signal" when capturing a digital DVI signal. Here you do not have to define the signals resolution.
When capturing an analog DVI signal, please check the option "manual configuration for incoming DVI-A signal" and choose the correct signal type from the drop-down list or type in the resolution and frequency in the textboxes below.

8.9.3 DVI and SDI Processor

Please note, that Christie Digital Systems has discontinued the DVI and SDI Processor.

The DVI Processor and SDI Processor can be purchased as separate products. In the past they were shipped with the QUAD Players, QUAD Servers, Broadcast DUAL Servers, or Broadcast QUAD Servers.

DVI Processor

The DVI Processor has two functionalities: on the one hand it splits the DVI signals coming from Pandoras Box hardware thus, it is possible to connect up to 4 monitors, projectors, other display devices via DVI. Please note that the DVI Processor can handle DVI-D signals only, any adaptation and conversion to analog signals like VGA or RGBHV is not supported! Many supported resolutions and timings accord to the international VESA standards. In addition, we included more 50Hz timings. Note that 75Hz and 85 Hz timings are not included but supported with custom EDIDs.
It is also possible to import custom EDIDs, defining not included resolutions and timings. DisplayIDs are supported as well; for the sake of simplicity this manual will use the term "EDID" meaning EDID as well as DisplayID.

All connected devices must support the same EDID. It is not possible to set up different resolutions per output.

On the other hand the DVI Processor converts the DVI signals coming from Pandoras Box hardware to SDI signals, wherewith SD-SDI, HD-SDI and 3G-SDI signals are supported. All supported SDI Timing EDIDs are standardized (EIA/CEA-861-D). The SDI signals accord to the specifications of SMPTE (274M, 296M, 125M, ITU-R BT.656).

The DVI Processor settings can be called via the PB Menu [2097], please find their description in the software chapter [2019] whilst the hardware [2018] is explained in detailed on the following page.

SDI Processor

The SDI Processor converts the DVI signals coming from Pandoras Box hardware to SDI signals, wherewith SD-SDI, HD-SDI and 3G-SDI signals are supported. All supported SDI Timing EDIDs are standardized (EIA/CEA-861-D). The SDI signals accord to the specifications of SMPTE (274M, 296M, 125M, ITU-R BT.656).

Optionally, the SDI Processor splits each signal into two slices. Each output signal is duplicated, thus it possible to connect up to 8 monitors, projectors, other display devices via SDI.

All connected devices must support the same EDID. It is not possible to set up different resolutions per output.

The SDI Processor settings can be called via the PB Menu [2097], please find their description in the software chapter [2019] whilst the hardware [2018] is explained in detailed on the following page.

8.9.3.1 DVI and SDI Processor Hardware

After dismantling please connect the DVI or SDI Processor to the power supply unit and plug the USB cable. The blue LEDs confirm the connected status.

We recommend establishing a network access via VNC. Depending on your display device it may scale the signal if it cannot display it properly, but it is possible as well that the display devices will stay black completely if you choose a resolution they do not support at all. Thus you will not see the desktop and the menu anymore.

The required driver for the DVI and SDI Processor is pre-installed. All drivers and their updates are free to download on the homepage's Download-Center after login.

Shut down the Player / Server. Do never hotplug the inputs into the DVI / SDI Processor as the graphics card driver might get damaged and needs to be reinstalled. Connect the DVI / SDI Processor's two inputs with the graphics card's outputs via the short DVI (dual link) cables provided. Please be sure, not to use other ones, especially longer cables in order to avoid signal noises or even breakdowns.

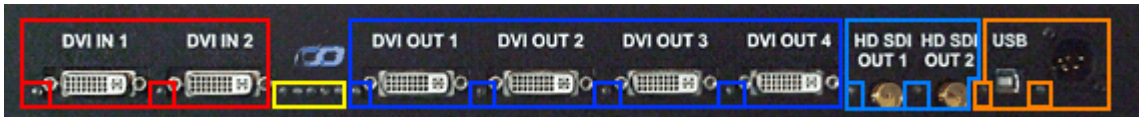
Connect the DVI Processor's outputs (either DVI or SDI) to your devices. Contrary to the inputs, the DVI Processor's outputs are hotplug-able.
Connect the SDI Processor's SDI outputs to your devices. Contrary to the inputs, the SDI Processor's outputs are hotplug-able.

Please make sure that the DVI / SDI Processor is powered before switching the Player / Server on.

All connected inputs and outputs are indicated by a blue LED. Red LEDs indicate that the signal is not passing through.

When choosing a different EDID all LEDs except the power and USB LED turn off, then according to your setting the input and output LEDs will change to magenta, red or blue once or twice. Please wait 5 seconds until the desktop background images are displayed properly and do not change anymore. If you choose different settings too fast, the graphics card driver might come into conflict. Then the two input LEDs will stay red no matter what EDID you choose. Please reboot the hardware.

LED Status of DVI Processor:



DVI input	red	inactive, no valid signal detected or input not connected
	blue	active, DVI single link signal detected
	magenta	active, DVI dual link signal detected
Sync status	off / blue	currently not used
DVI output	off	deactivated
	red	inactive, no valid signal
	blue	active, valid signal output
SDI output	off	deactivated
	red	inactive, no valid signal
	blue	active, valid signal output
USB / Power	off	deactivated, input not connected
	magenta	device is booting
	blue	active

LED Status of SDI Processor:

DVI input	red	inactive, no valid signal detected or input not connected
	blue	active, DVI single link signal detected
	magenta	active, DVI dual link signal detected
Sync status	off / blue	currently not used
SDI output	off	deactivated
	red	inactive, no valid signal
	blue	active, valid signal output
USB / Power	off	deactivated, input not connected
	magenta	device is booting
	blue	active

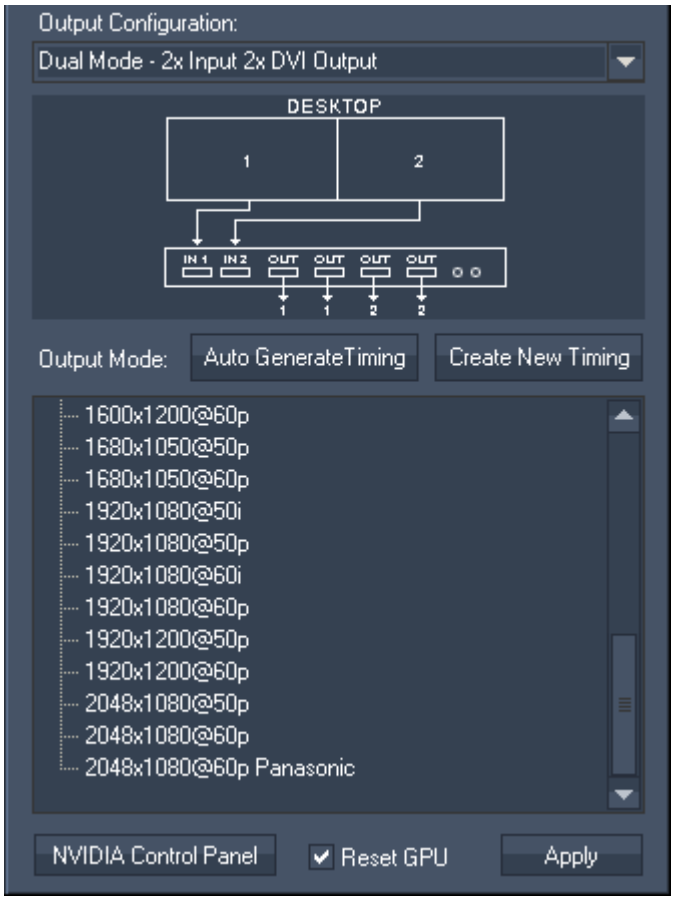
8.9.3.2 Software - Quad Setup Menu

Before accessing the DVI / SDI Processor's [2017] software settings, please connect it as explained in the previous hardware chapter [2018]. Now, you may access the menu by clicking "Setup" > "Display Setup". If the DVI / SDI Processor is not connected properly the button opens the NVIDIA control panel. In that case please check that the USB cable and power supply are connected and that the USB-driver is installed properly. Then press "Shutdown" > "Reset Menu".

With choosing the resolution and frame rate in the Quad Setup Menu, most of the changes and settings within the NVIDIA Control Panel have become obsolete. The only setting not influenced is the order of the primary and secondary screen. If you want to change it you may do so within the NVIDIA Control Panel as usual, as this does not influence the setting of the Quad Setup Menu.

In case the chosen timing does not seem right to you, we advise to have a look in the NVIDIA Control Panel, if the resolution is set to the native one for sure. The native resolution is now reported by the DVI / SDI Processor, not the display devices. In general, the driver is told to switch to the native resolution when the DVI / SDI Processor changes its timings. Please do not change any other resolution settings in the NVIDIA Control Panel as they will influence the DVI / SDI Processor's incoming signal but as the setting for the output ports are not changed, the displayed image will look as if it were cut off.

Thus the workflow is to take the resolution changes in the Quad Setup Menu only and to check in the NVIDIA Control Panel if the driver has switched to "native". Below, you find the available options for the DVI Processor's output configuration and at the end of this chapter the SDI Processor's output options [2023]. The next chapter explains how to create a new timing [2024] in case you like to complement the below output presets.



- Dual Mode – 2x Input 2x DVI Output

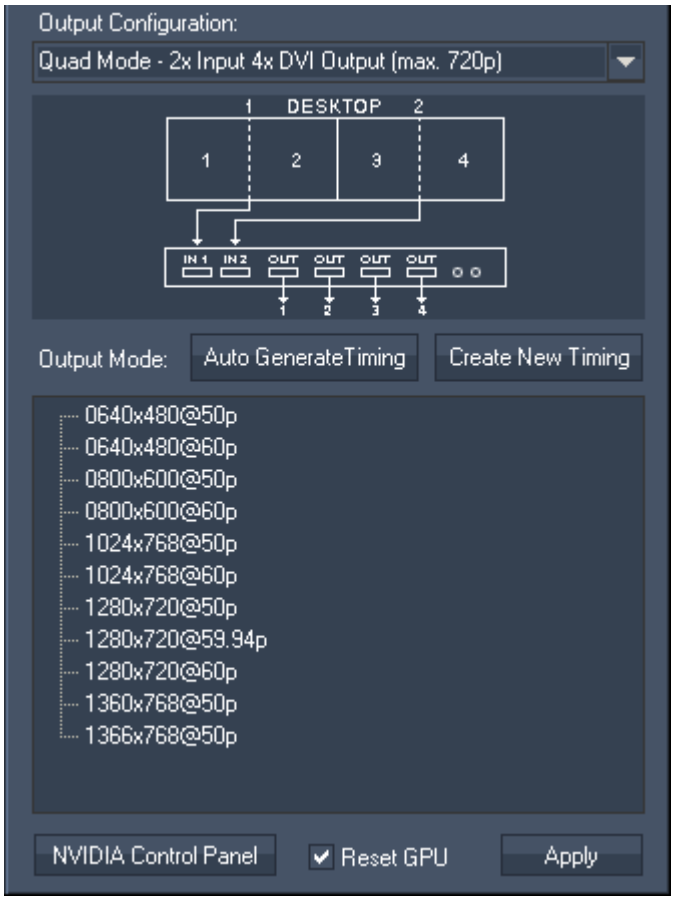
The primary Windows screen is distributed to DVI output 1 and 2. The secondary Windows screen is distributed to DVI output 3 and 4.

(The SDI outputs are inactive if the chosen EDID does not apply to one of the supported SDI specifications. If it does, the primary screen is passed to SDI output 1 additionally, as well as the secondary Windows screen is passed to SDI output 2.)

The button "Auto Generate Timing" generates a timing based on the information from the device plugged into DVI output 1. The timing is applied to DVI output 1-4 and the incoming signal is set accordingly.
The new timing is saved as an *.edid file in the according sub folder under `C:\system\LCD_Menu\Quad DVI SDI Modes`. It also appears in the according menu with a distinct name e.g. `1920x1080@50p custom AutoGenerated`, so that you can select it later again.

The button "Create New timing" opens a dialog to define any custom resolution. The dialog is explained in the next chapter 2024.

With both options, "Auto Generate Timing" and "Create New Timing", you add a new EDID to the output presets. As you are in a "Dual Mode", the EDID will be assigned to the graphics card outputs one-to-one. This behavior differs to the one from the "Quad Modes".



- Quad Mode – 2x Input 4x DVI Output (max 720p)

The left half of the primary Windows screen is shown on DVI output 1, the right half on DVI output 2. The left half of the secondary Windows screen is shown on DVI output 3, the right half on DVI output 4.

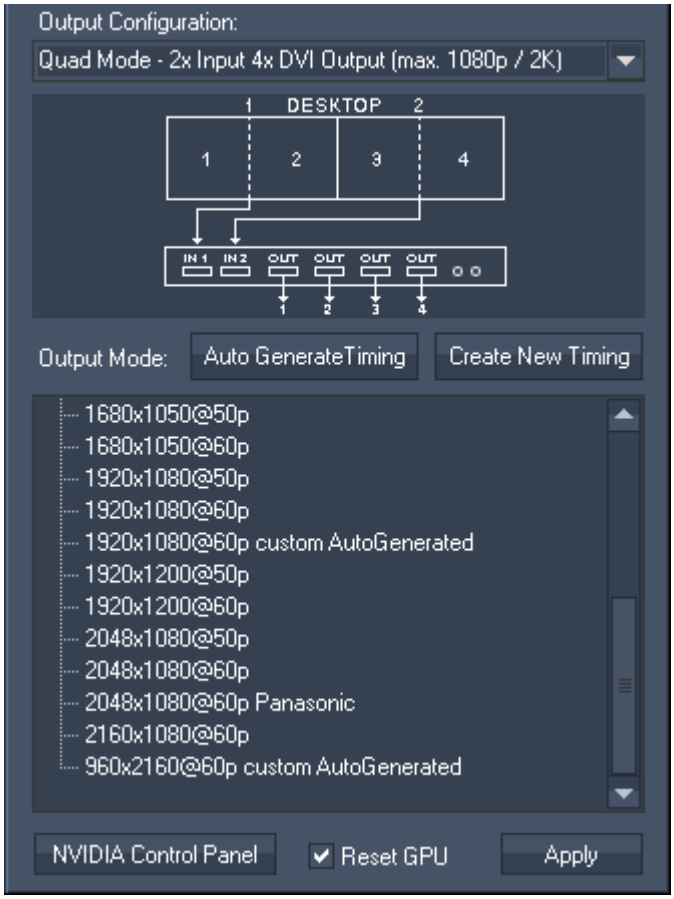
All resolutions and frame rates available are specified as single link DVI standard.

(The SDI outputs are inactive if the chosen EDID does not apply to one of the supported SDI specifications. If it does, the primary screen is passed to SDI output 1 additionally, as well as the secondary Windows screen is passed to SDI output 2.)

The button "Auto Generate Timing" generates a timing based on the information from the device plugged into DVI output 1. The timing is applied to DVI output 1-4 and the incoming signal is set accordingly.
The new timing is saved as an *.edid file in the according sub folder under `C:\system\LCD_Menu\Quad DVI SDI Modes`. It also appears in the according menu with a distinct name e.g. `1920x1080@50p custom AutoGenerated`, so that you can select it later again.

The button "Create New timing" opens a dialog to define any custom resolution. The dialog is explained in the next chapter 2024.

With both options, "Auto Generate Timing" and "Create New Timing", you add a new EDID to the output presets. As you are in a "Quad Mode", the EDID will automatically be doubled to replace the resolution of the graphics card outputs. If this timing does not exceed a DVI Single Link timing, you will find it in the menu "Quad Mode – 2x Input 4x DVI Output (max. 720p), otherwise it will be added to the "...max.1080p / 2k).



- Quad Mode – 2x Input 4x DVI Output (max 1080p / 2K)

The left half of the primary Windows screen is shown on DVI output 1, the right half on DVI output 2. The left half of the secondary Windows screen is shown on DVI output 3, the right half on DVI output 4.

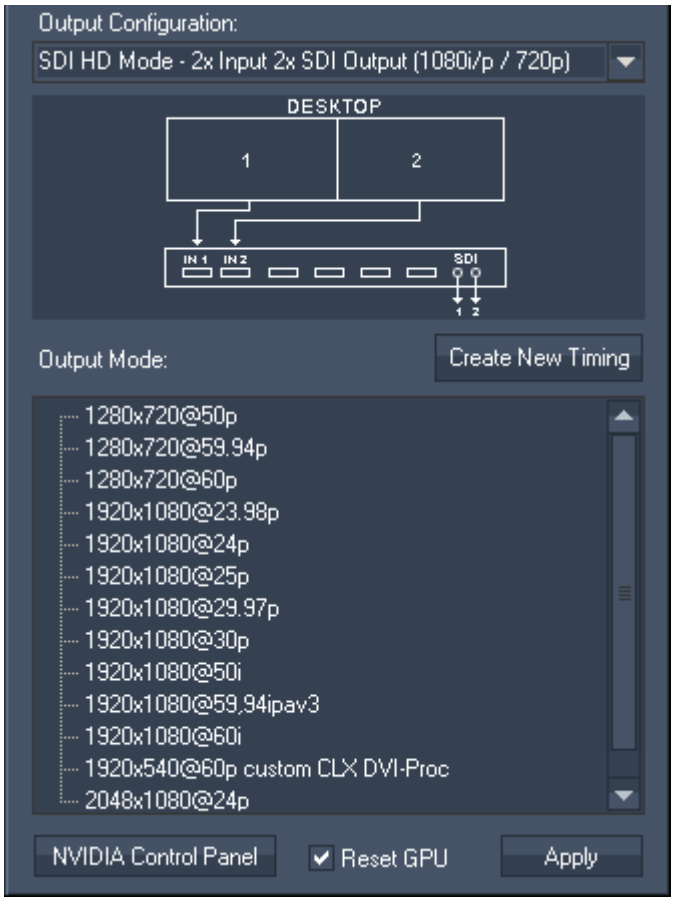
All resolutions and frame rates available are specified as dual link DVI standard.

The SDI outputs are inactive.

The button "Auto Generate Timing" generates a timing based on the information from the device plugged into DVI output 1.The timing is applied to DVI output 1-4 and the incoming signal is set accordingly.
The new timing is saved as an *.edid file in the according sub folder under `C:\system\LCD_Menu\Quad DVI SDI Modes`. It also appears in the according menu with a distinct name e.g. `1920x1080@50p custom AutoGenerated`, so that you can select it later again.

The button "Create New timing" opens a dialog to define any custom resolution. The dialog is explained in the next chapter 2024.

With both options, "Auto Generate Timing" and "Create New Timing", you add a new EDID to the output presets. As you are in a "Quad Mode", the EDID will automatically be doubled to replace the resolution of the graphics card outputs. If this timing does not exceed a DVI Single Link timing, you will find it in the menu "Quad Mode – 2x Input 4x DVI Output (max. 720p), otherwise it will be added to the "...max.1080p / 2k).



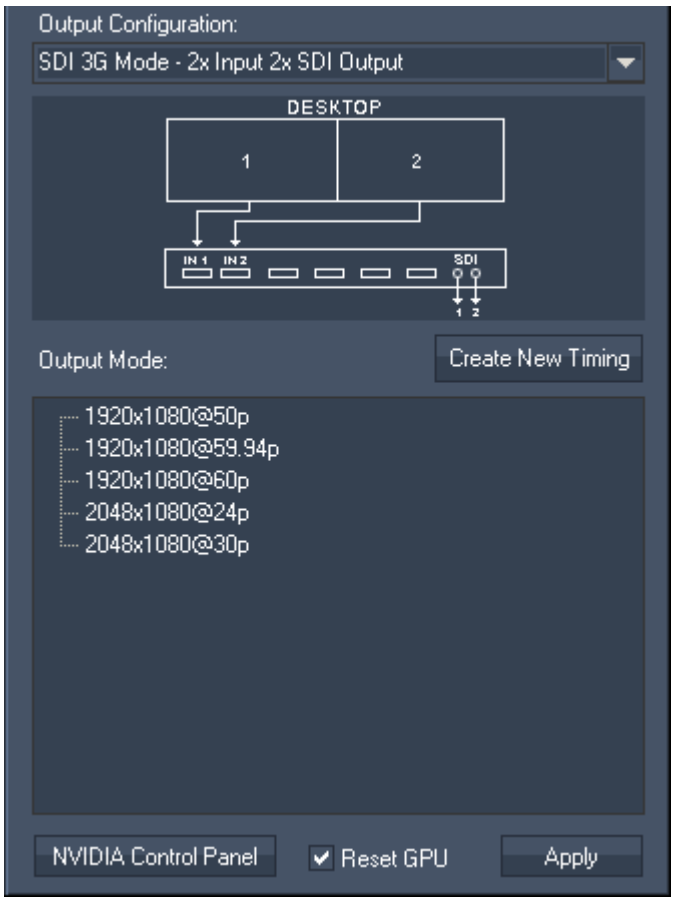
- SDI HD Mode – 2x Input 2x SDI Output (1080i/p / 720p)

The primary Windows screen is passed to SDI output 1, the secondary Windows screen to SDI output 2.

All resolutions and frame rates available are specified as HD-SDI standard.

The DVI outputs are inactive.

The button "Create New timing" opens a dialog to define any custom resolution. The dialog is explained in the next chapter 2024.



- SDI 3G Mode – 2x Input 2x SDI Output

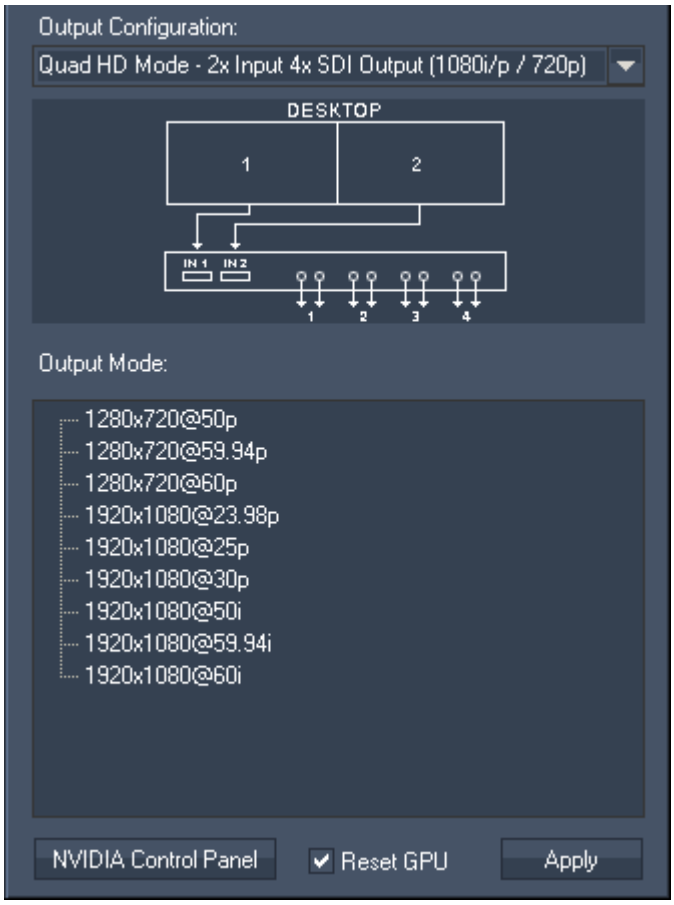
The primary Windows screen is passed to SDI output 1, the secondary Windows screen to SDI output 2.

All resolutions and frame rates available are specified as 3G-SDI standard.

The DVI outputs are inactive.

The button "Create New timing" opens a dialog to define any custom resolution. The dialog is explained in the next chapter 2024.

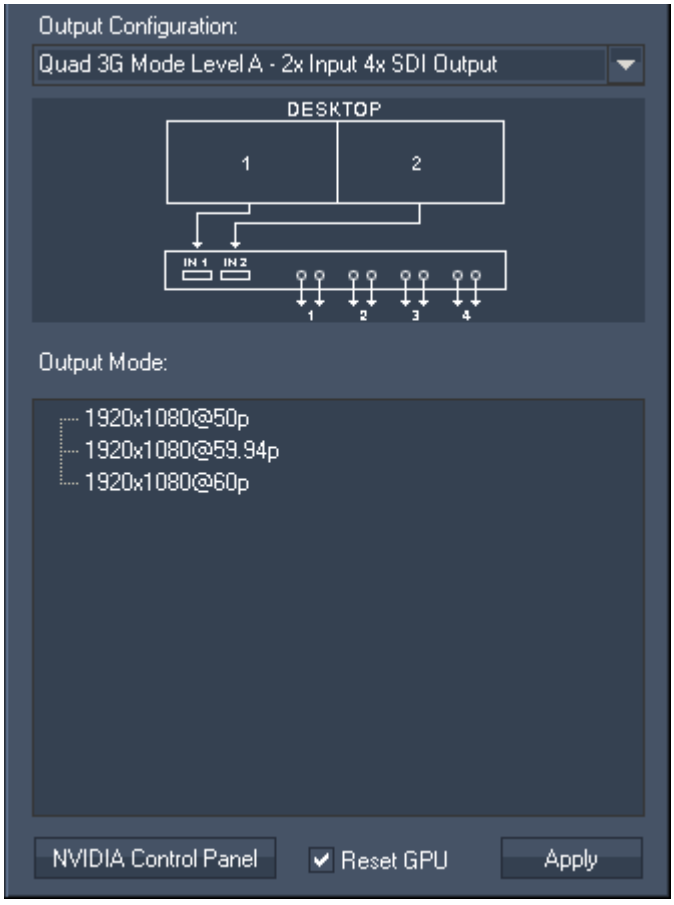
Output Configuration for the SDI Processor



- Quad HD Mode – 2x Input 4x SDI Output (max 1080i/p / 720p)

The left half of the primary Windows screen is shown on SDI output 1 and 2, the right half on SDI output 3 and 4. The left half of the secondary Windows screen is shown on SDI output 5 and 6, the right half on SDI output 7 and 8.

All resolutions and frame rates available are specified as HD-SDI standard.



- Quad 3G Mode Level A – 2x Input 4x SDI Output

The left half of the primary Windows screen is shown on SDI output 1 and 2, the right half on SDI output 3 and 4. The left half of the secondary Windows screen is shown on SDI output 5 and 6, the right half on SDI output 7 and 8.

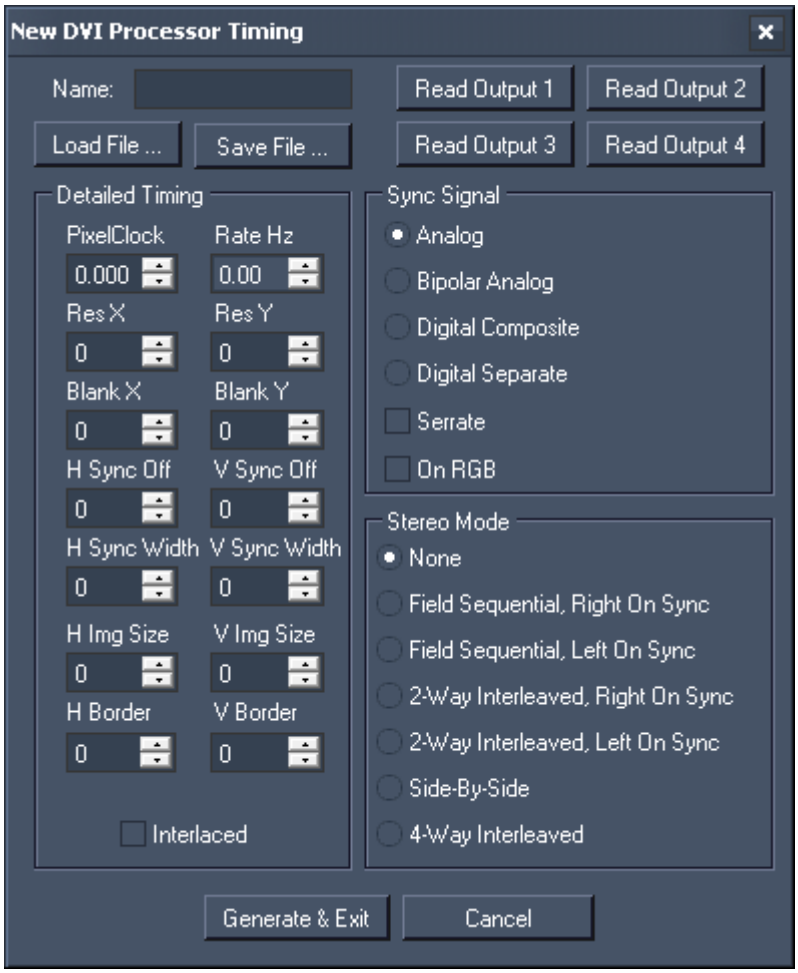
All resolutions and frame rates available are specified as 3G-SDI standard.

8.9.3.3 Software - Create New Timing

This chapter explains the dialog "Create New timing" that can be called from within the configuration menus [2019] from the DVI Processor [2017]. Please note that the term "EDID" means EDID as well as DisplayID.

The dialog offers settings to define a custom timing, which includes the settings to...:

- create an entirely new EDID
- read out a display's EDID or load it from a file
- modify EDIDs
- save new EDIDs to a file and generate a preset for the DVI Processor's output configuration menus.



Name
You can name your timing optionally. In case the timing is displayed as an preset in the output configuration menus, the name appears at the end of the entry, e.g. `1920x1080@50p custom "name"`. In addition, the name is also mentioned in the NVIDIA control panel as it is part of the EDID Descriptor Block.

Load File
Use "Load File ..." to load a stored EDID into the dialog. You can edit the timing, generate an output preset or save it as an *.edid file.

Save File
Use "Save File ..." to export the currently displayed EDID timing into a *.edid file.

Read Output 1...4
These buttons generate a timing based on the information from the device plugged into the according DVI output. Note that SDI outputs can not be read-out. The timing is displayed in the dialog. Note that all values will equal zero if no valid EDID can be read. You can edit the timing, generate an output preset or save it as an *.edid file.

Generate & Exit
This option generates an output preset based on the currently displayed timing and closes the dialog.

Please note, that it is of importance from which menu you opened the dialog. If you were in the DVI "Dual Mode" (2in 2out), your EDID is assigned to the graphics card outputs one-to-one. In case you were in a DVI "Quad Modes" (2in 4out, either Single Link with max. 720p or Dual Link with max.2k), your EDID is automatically doubled as the graphics card outputs are split in the Quad Modes. In both cases the displayed timing in this dialog and the entry in the menus always refer to the Processor's outputs.

The new EDID is saved as an *.edid file in the according sub folder under `C:\system\LCD_Menu\Quad DVI SDI Modes`. It also appears in the according menu with a distinct name e.g. `1920x1080@50p custom (optional name)`. Go to the DVI Processor's output configuration menu and select it to apply it to DVI output 1-4 whilst the incoming signal is set automatically.

Cancel
Use "Cancel" to close the dialog and discard all settings.

EDID Description

The EDID Description shows the content of the "First Detailed Timing Descriptor Block" along with the "Monitor Name" out of the EDIDs "Descriptor Block 2".
The displayed information changes automatically when "Load File" or "Read Output 1...4" is used. You can edit the timing, generate an output preset or save it as an *.edid file as described above.

Detailed Timing

Name	Read out or change the monitor name with a maximum of 13 characters.
PixelClock	This value shows the signal's PixelClock in MHz. The maximum value is 330.
Rate Hz	This value shows the signal's Refresh Rate in Hz when an EDID is loaded. It results of all other definable settings regarding pixel count and PixelClock.
Res X	This value shows the signal's active pixel per horizontal line. Active pixels are the shown ones within a display device and equal the horizontal resolution. The maximum value is 4095.
Res Y	This value shows the signal's active lines per image. Active lines are the shown ones within a display device and equal the vertical resolution. The maximum value is 4095.
Blank X	This value shows the signal's total horizontal blanking pixels including (definable) front porch, (definable) sync width and (resulting) back porch. The combination of the horizontal active, blanking and border pixels equals the picture's horizontal total pixels. The maximum value is 4095.
Blank Y	This value shows the signal's total vertical blanking lines including (definable) front porch, (definable) sync width and (resulting) back porch. The combination of the vertical active, blanking and border lines equals the picture's vertical total lines. The maximum value is 4095.
H Sync Off	This value shows the signal's horizontal sync offset (front porch) in pixels. The maximum value is 1023.
V Sync Off	This value shows the signal's vertical sync offset (front porch) in lines. The maximum value is 1023.
H Sync Width	This value shows the signal's horizontal sync width in pixels. The maximum value is 1023.
V Sync Width	This value shows the signal's vertical sync width in lines. The maximum value is 1023.
H Img Size	This value shows the monitor's horizontal image size in mm. This information value is optional and has no influence on the signal processing. The maximum value is 4095.
V Img Size	This value shows the monitor's vertical image size in mm. This information value is optional and has no influence on the signal processing. The maximum value is 4095.
H Border	This value shows the signal's horizontal border in pixels. The maximum value is 255.

V Border	This value shows the signal's vertical border in lines. The maximum value is 255.
Interlaced	This check box shows if the signal is interlaced (checked) or progressive (unchecked).

Sync Signal
Readout or select the desired sync option required by the signal.
Depending on the chosen sync option, there are additional check boxes to be de-/activated.

Stereo Mode
Readout or select the desired Stereo Mode required by the signal.

8.9.4 EDID Link

Please note, that Christie Digital Systems has discontinued the EDID Link.

Overview

The EDID Link is a device to manage DVI-EDIDs and communicate these to the graphics card. EDIDs (Extended Display Identification Data) are used to describe the (preferred) timing of a display device. The timing is describing the display's resolution and its sync pixels. It is communicated to the graphic source via the DVI-cable.

The EDID Link can be used to generate and constantly hold an EDID whenever a display device is not able to send a correct or the exact needed EDID to the source. In addition the EDID Link will ensure, that a monitor is simulated even when none is connected to it, or powered down. The EDID Link does not need any separate power as it receives the voltage from the graphics card.

The EDID Link handles DualLink-DVI-I signals and creates / emulates DVI-EDIDs. Analog signals will be routed through. The device is not HDCP compliant.

Features

Per default, the EDID Link comes with preinstalled EDIDs and can be used without additional software. The hardware features [2027] are explained in the following chapter. In short, it allows to:
- select an EDID via a preset
- work in transparent mode to route the display device's EDID information to the graphics card

For more advanced users, the EDID Link offers an additional software [2028]. In short, it allows to:
- read out a display's EDID or load it from a file
- create an entirely new EDID
- modify EDIDs
- write this EDID to one of the user presets or to a file

8.9.4.1 EDID Link Hardware

This chapter explains the features of the EDID Link's hardware. The previous chapter offers general information about the EDID Link [2026] whilst the next covers its software features [2028].

Presets

With its two selection knobs, you can select from up to 99 different EDIDs directly. Presets 01-49 are holding read-only default EDIDs while presets 50-99 can be configured by the user. All EDIDs can be exported to a file to be used on a different system, such as the Pandoras Box Quad Server with its DVI Processor [2017].

Preset 00 acts like a "transparent mode" which will route through the EDID coming from the connected display device. More than that, the last seen, valid EDID is stored to preset 00. Therefore you can disconnect the display while the graphics card will still see the display's information. Whenever a different EDID is transferred into the device, this information will be written into preset 00, even while emulating another preset at that time.

In order to read out a new EDID, the EDID Link needs to receive 5V either coming from the graphics card, or the USB-plug. With any 5V input connected, you may read out and store the display's information to preset 00 and later on edit it or download it to your computer using the software "EDID Link Editor" [2028].

Usage

Your EDID Link is directly ready to use. You do not need any installation in order to use the default EDIDs. A list is printed onto the device and is part of the delivered documentation.

- Power down your signal source (e.g. a computer) first.
- Connect the EDID Link's DVI "in" connector with the included short DVI cable to the graphics card's output of your signal source.
- Connect the EDID Link's DVI "out" connector to your display device (e.g. a monitor).
- Select the desired EDID by rotating the selection knobs.
- Power up your signal source.

The signal source will now receive the EDID adjusted on the EDID Link. You may need to select and confirm to change to this resolution using the graphics card driver. From now on, there is no direct connection from the graphic source to the display any longer. Therefore, you may power down the display without loosing your graphics card settings as the EDID Link is still emulating the device. Even a loose connection of the DVI cable will not harm the settings any more. When using presets other than 00, you will be able to switch display devices without transferring this information to the source.

It is recommended to power down the complete system before disconnecting a DVI cable.

Rotating the selection knobs while running the system will lead to a disconnect of the attached EDID Link and the graphics card. It may switch to a not connected state. As long as you turn the knobs, the EDID Link remains in the disconnected state. After two seconds of not changing the preset, the device will output the selected EDID. You may need to set the resolution again within your graphics cards settings or source device settings.

LED Codes

The LED next to the USB plug features different LED codes:

Red	blinking	the device is in firmware programming mode
Blue	pulsing to magenta every 5s	the device is idle and ready to use
Blue	switching once off for 0,5s	the device is used by the software to read from or write to a preset

Important information about internal EDID handling

EDIDs (Extended Display Identification Data) are used to describe the (preferred) timing of a display device. The timing is describing the display's resolution and its sync pixels. It is communicated to the graphic source via the DVI-cable.

A single EDID may feature a list of different timings which can be processed by the monitor. This may be a combination of "Established Timings", "Standard Timings" and "Detailed Timings" at the same time. This gives the user the flexibility to choose from more than one resolution, whenever the native one is not about to be used.

As this flexibility often leads to not having the desired resolution set within the signal source, the EDID Link limits all its EDIDs to only one single timing.

Using this fact, the graphics card will be forced to output the exact desired resolution and timing.

Note: All default EDIDs as well as imported or read out ones from any device are changed and limited to the "First Detailed Timing Descriptor". Extensions are being cut off as well as soon as they are stored into the EDID Link.

As preset 00 acts as a transparent mode, you will find the original EDID including all timings and extensions. Storing this data into a preset will reduce the timing to a single one. How to do this is explained in the next chapter covering the EDID Link's software [2028].

8.9.4.2 Software - EDID Link Editor

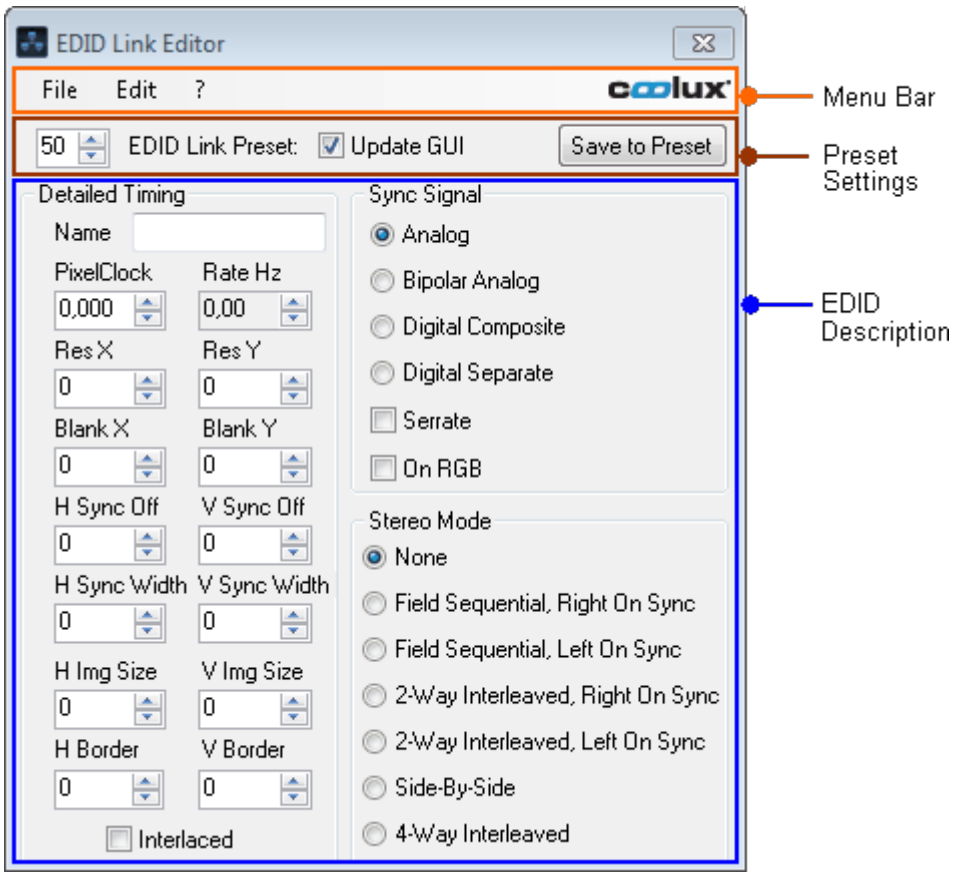
This chapter explains the features of the EDID Link's software. The previous chapters offer general information about the EDID Link [2026] and cover its hardware features [2027].

Installation and Software

Run the "EDID Link Editor Installer" on a Windows operating system and proceed through the installation. The installer is available on the delivered CD or in our Download-Center. After the installation you will find the EDID Link Editor software within the Windows start menu and a shortcut on your Desktop.

The EDID Link Editor software can be used with or without a connected EDID Link to create, edit or organize EDIDs.
If you like to read an EDID out or write presets onto it, connect your EDID Link via the included USB-cable to your computer. Use the USB-driver "coousb-driver v6.0.13.0" (or a newer one) to finish the hardware installation of the device. The EDID Link is now communicating with your system and ready to be used.

The software is divided into three sections as pictured below.



Menu Bar

File Menu
Use "Load EDID File ..." to load a stored EDID into the Editor. You can then edit and store this EDID to an EDID Link preset or export it into a file again.
Use "Save EDID File ..." to export the currently displayed EDID timing into a file.

Edit Menu
Use "Read Monitor EDID" to force the EDID Link to read the currently connected EDID (again) and display it within the Graphical User Interface (GUI). If no valid EDID can be read, all values will equal zero.
Note: If you want to read out what has already been stored automatically to preset 00 when the monitor has been attached, you can recall preset 00 without using this menu item.

Use "Initialize Default EDID List" in order to re-initialize the default EDIDs on all presets.
Note: This action will overwrite all user data.

Preset Settings

Use the "EDID Link Preset" numeric box in order to recall the particular preset 00-99 by either clicking up/down or typing in the desired preset number followed by the return key.
Choosing a preset will read out its content stored within the EDID Link and display it in the "EDID Description" section.

Check the "Update GUI" check box in order to directly recall content from a preset while changing the preset number. You may uncheck the check box in order to select a preset without loading its content (and overriding your changes done in the EDID settings).

Use the "Save to Preset" button in order to store the currently displayed EDID timing into the selected preset. You may want to uncheck the "Update GUI" check box in order to select a preset you want to overwrite without loosing your settings entered in the "Detailed Timing" settings below.

EDID Description

The EDID Description shows the content of the "First Detailed Timing Descriptor Block" along with the "Monitor Name" out of the EDIDs "Descriptor Block 2".
The displayed information can be edited and stored to the EDID Link or to a file as described above.

Detailed Timing

Name	Read out or change the monitor name with a maximum of 13 characters.
PixelClock	This value shows the signal's PixelClock in MHz. The maximum value is 330.
Rate Hz	This value shows the signal's Refresh Rate in Hz when an EDID is loaded. It results of all other definable settings regarding pixel count and PixelClock.
Res X	This value shows the signal's active pixel per horizontal line. Active pixels are the shown ones within a display device and equal the horizontal resolution. The maximum value is 4095.
Res Y	This value shows the signal's active lines per image. Active lines are the shown ones within a display device and equal the vertical resolution. The maximum value is 4095.
Blank X	This value shows the signal's total horizontal blanking pixels including (definable) front porch, (definable) sync width and (resulting) back porch. The combination of the horizontal active, blanking and border pixels equals the picture's horizontal total pixels. The maximum value is 4095.
Blank Y	This value shows the signal's total vertical blanking lines including (definable) front porch, (definable) sync width and (resulting) back porch. The combination of the vertical active, blanking and border lines equals the picture's vertical total lines. The maximum value is 4095.
H Sync Off	This value shows the signal's horizontal sync offset (front porch) in pixels. The maximum value is 1023.

V Sync Off	This value shows the signal's vertical sync offset (front porch) in lines. The maximum value is 1023.
H Sync Width	This value shows the signal's horizontal sync width in pixels. The maximum value is 1023.
V Sync Width	This value shows the signal's vertical sync width in lines. The maximum value is 1023.
H Img Size	This value shows the monitor's horizontal image size in mm. This information value is optional and has no influence on the signal processing. The maximum value is 4095.
V Img Size	This value shows the monitor's vertical image size in mm. This information value is optional and has no influence on the signal processing. The maximum value is 4095.
H Border	This value shows the signal's horizontal border in pixels. The maximum value is 255.
V Border	This value shows the signal's vertical border in lines. The maximum value is 255.
Interlaced	This check box shows if the signal is interlaced (checked) or progressive (unchecked).

Sync Signal
Readout or select the desired sync option required by the signal.
Depending on the chosen sync option, there are additional check boxes to be de-/activated.

Stereo Mode
Readout or select the desired Stereo Mode required by the signal.

8.9.5 MADI Sound Card (Discontinued)

Please note, that Christie Digital Systems has discontinued the MADI Sound Card. For other options, please see the chapter Input and Output Cards [1947].

This chapter describes the optional MADI audio card.

The MADI audio card allows to input and output audio (in accordance with the ASIO protocol) to and from Pandoras Box.

Soundcard 64ch MADI (RME HDSPe MADI)

The MADI board carries:
MADI in / out (optical and coaxial)
Stereo Analog Out (Phones)

The included expansion board carries:
Word Clock in / out (BNC)
MIDI in / out (via four 5-pin DIN jacks)

The MADI interface offers to record and playback 24-bit audio with
- 64 channels at a sample rate of up to 48 kHz
- 32 channels at a sample rate of up to 96 kHz or
- 16 channels at a sample rate of up to 192 kHz.

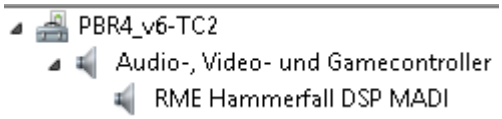
Transmission is done via a single line, either coaxial with BNC plugs or by fiber cable. In both cases more than 100 m cable length can be achieved.

The word clock input is galvanically isolated via transformer and operates on both single and double speed signals automatically.

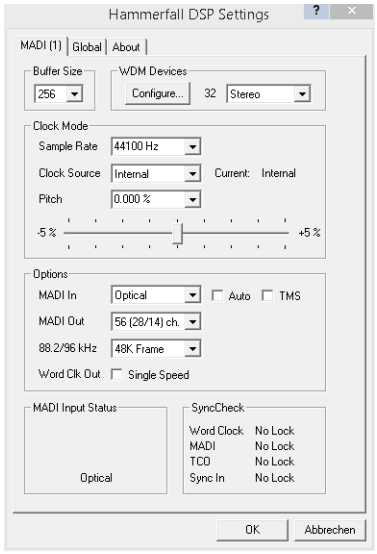
The MIDI solution offers two completely independent MIDI inputs and outputs via breakout cable (4 DIN sockets). A virtual MIDI port allows for a direct transmission of MIDI data over MADI.

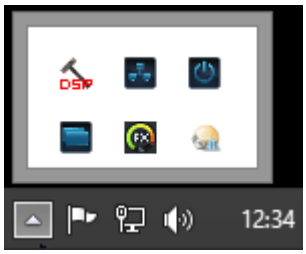
Driver

The card drivers are preinstalled on your Server or Player hardware. All drivers can be found locally under `C:\system` and in the Download-Center where you can also check for updates.
The following picture shows the Windows Device Manager and a correctly installed 64ch MADI card in the folder "Audio-, Video and Gamecontroller".



Settings and TotalMix Dialog

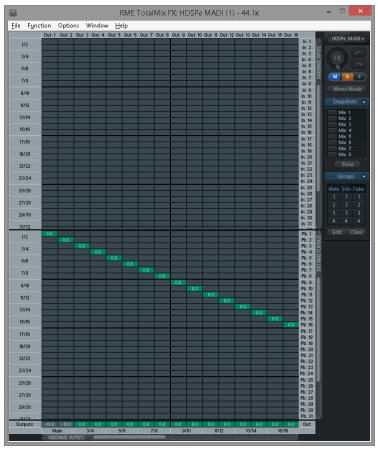




In the system tray of the taskbar you might find the hammer icon which opens the Settings dialog and the Mixer or FX icon which opens the TotalMix dialog. In case your taskbar is hidden due to the default behavior of the PB Menu [2097], open the "Configuration" menu and click "Taskbar On".

The Configuration of the 64ch MADI card is done via its own settings dialog. It can be opened by clicking on the hammer symbol in the system tray.
The first image shows the dialog "Hammerfall DSP Settings" for the 64ch MADI card which offers the following:
- Input selection
- Level of analog I/Os
- Configuration of digital I/Os
- Synchronization behavior
- State of input and output
- Current sample rate
- Latency

Regarding the Clock Source drop-down list in the Clock Mode section, the 64ch MADI can be configured to use its own clock (Internal = Master), or one of the input signals (Word, MADI, TCO, Sync In). If the selected source is not available (NoLock), the unit will change to the next available one (this behaviour is called AutoSync). If none is available then the internal clock is used. The card is equipped with SteadyClock(TM), RME's sync and clock technology. With this, the device becomes a sync reference for other devices too. SteadyClock refreshes clock signals, removes jitter, and takes permanent care of optimal conversion quality. SteadyClock allows the MADI card to control the sample rate freely on its own.
The Expansion Board's bracket has the word clock input and output. Next to the input BNC socket, a green LED displays the word clock input's LOCK state. Between the BNC sockets, 75 Ohm word clock termination can be activated and verified by a yellow LED.



Any changes made in the Settings dialog are applied immediately - confirmation (e.g. by clicking on OK or exiting the dialog) is not required. However, settings should not be changed during playback or record if it can be avoided, as this can cause unwanted noises. Also, please note that even in 'Stop' mode, several programs keep the recording and playback devices open, which means that any new settings might not be applied immediately. The status displays at the bottom of the dialog box give the user precise information about the current status of the system, and the status of all digital signals.

The digital real-time mixer, the "Hammerfall DSP Mixer" (TotalMix) can be opened by clicking on the mixer icon in the system tray.
TotalMix allows for practically unlimited mixing and routing operations, with all inputs and playback channels simultaneously, to any hardware outputs. Every single input channel, playback channel and hardware output features a Peak and RMS level meter, calculated in hardware (hardware output is Peak only).

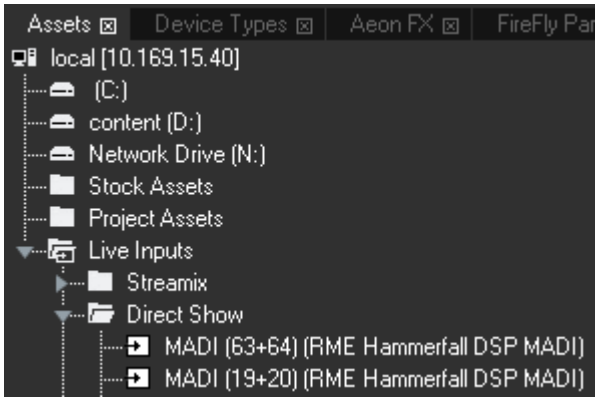
For more information about the Settings and Mixer dialog please see the audio card's manual: http://www.rme-audio.de/download/hdspemadi_e.pdf

Working with Audio In- and Outputs in Pandoras Box

To work with audio in- and outputs requires that the sound card is set up in the Configuration tab under ASIO Audio [166]. The linked chapter explains in detail how to do so and includes information regarding synchronization, timecode, sample rate setup and sample accurate audio and master volume.

To output ASIO audio from Pandoras Box, simply assign audio files from the Project tab [278] to the dedicated Audio tracks [661] in the Device Tree [173] or Sequence [292]. Audio tracks [661] allow to playback ASIO WAV files and inputs while being synchronized to a master clock. The tracks play mono and stereo PCM Wave files (16, 24 or 32bit) and they refer to the sample rate set in the Configuration tab. In order to assign a track to a specific audio channel of your ASIO sound card, please refer to the Track Inspector [217].

The Pandoras Box Software License offers unlimited Audio Tracks. Note that in a larger master-client-setup it is possible to use either a single audio card or multiple ones and it is possible to use the audio card from the Master or from the Client.



To input audio, you just need to go to the Asset tab [131] > local (or Client) > folder "Live Inputs" > folder "Direct Show" and drag the input to the Project tab. Then assign the live input to an Audio Track in the Device Tree the same way as using wave files.

It is also possible to record the incoming signal. However, the Client computer must then be started in stand-alone mode, that is with the Pandoras Box Master software. Please see the chapter " Video Recording [137]" for more information.

8.9.6 Sensor Link

Please note, that Christie Digital Systems has discontinued the Sensor Link, please refer to the NET Link [2000] with the according Encoder Input Board [2006].

The Sensor Link is a unique highspeed sensor interface that is designed to provide input signals from standard industrial encoders, distometers as well as analog potentiometers and contact closures.

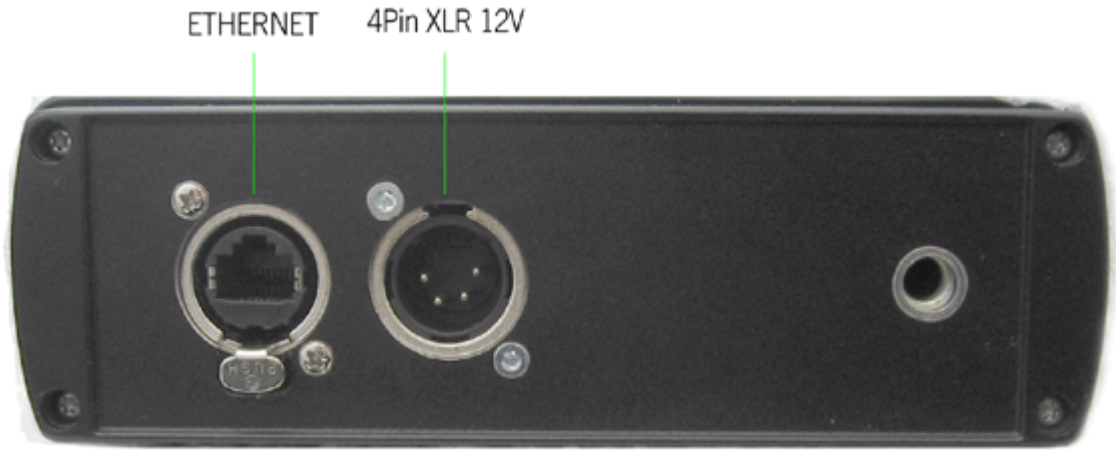
8.9.6.1 Technical Features

From one unit there can be used simultaneously:

2x 12V industrial encoder [2035] inputs with internal or external reset (GPI) with 5-pin M12-Connectors,
2x 0-10V analog inputs [2035] and
8x GPI contact closures [2035]

Since the SensorLink is based on TCP/IP communication (outputs TCP/IP at variable frequency between 50 and 5000 Hz), multiple units can be used at the same time.

The internal sample rate reads incoming data at 7 kHz to allow high resolution encoder inputs to be used for accurate motion sensing.



The front panel shows the Neutrik Ethernet input jack as well as the 4 pin XLR 12V power input.

On the rear side you will find the connection for encoders (E1 & E2, please use 5-pin M12-Connectors, Pin Connections: 1= +Volt, 2=B, 3=0Volt, 4=A, 5= N Index), analog devices (A1 & A2) and contact closures S1 – S8).

RESET
As the unit can be programmed for various encoder step count settings or remote rest indexing a reset button is provided to restore the factory settings of the unit.

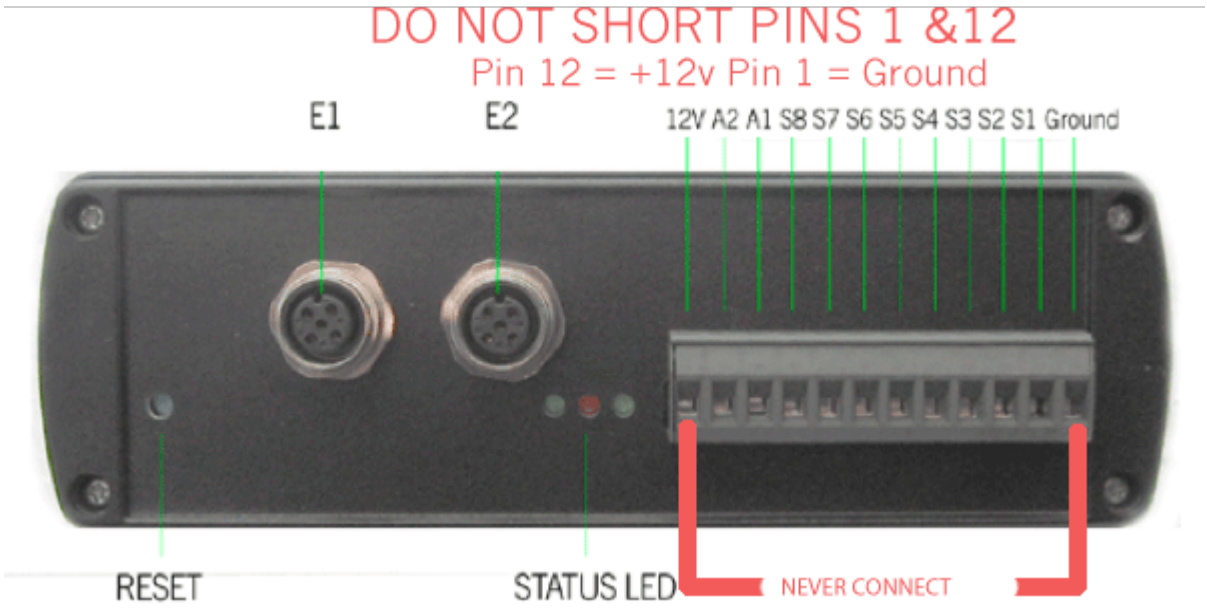
STATUS LED
Three status LEDs give feedback information of the current state of the unit.

8.9.6.2 Connecting Encoders and Sensors

You may connect the following inputs to the Sensor Link Device:

Analog inputs with their own voltage supply of 0-10 Volt [2035],
Switch Inputs with or without their own voltage supply of 12 V [2035],
Incremental Encoders, using 5-pin M12-Connectors [2035],

Please never connect Pin 12 and Pin 1 as this could cause a short circuit!



8.9.6.2.1 Analog Inputs A1 and A2

Connect – to Ground and the gate input to A1 or A2. The sensor link will read out a voltage range of 0-10 Volt and will transfer this range in 512 steps. They have 30kΩ input impedance, resulting in a minimum of 0,5A.

Please make sure that you are using the same grounding in order not to get any potential equalization!

8.9.6.2.2 Switch Inputs S1-S8

It is recommend to use potential-free relay contacts as triggers. Connect these to 12V and to one of the Switch Inputs S1-S8.
If you are using triggers with their own voltage supply (11-13 V) connect – to Ground and the gate input to one of the Switch Inputs S1-S8.
Note that the contacts should not be switched faster than twice per second to be recognized correctly. 4 times (4 Hz) could be reached under circumstances, but maximal 2Hz is recommended.

Please make sure that you are using the same grounding in order not to get any potential equalization!

8.9.6.2.3 Encoder Inputs E1 and E2

You may connect incremental Encoders to input E1 or E2, using 5-pin M12-Connectors.

Pin assignment female connector:

Pin 1 = positive power supply, +12V or +24 V
Pin 2 = Pulse signal B
Pin 3 = negative power supply, Ground
Pin 4 = Pulse signal A
Pin 5 = Pulse signal N, Reset

Guideline for choosing the right Encoder:

In order to specify the right encoder for your application, a couple of parameters need to be determined first.

- What kind of encoder should it be? Hollow Shaft Encoder or Shaft Encoder
- The diameter of the axis where the encoder should be mounted to
- Minimal and maximal speed of the axis. Measured in rotations per second

In general you might say: the more steps the encoder has, the better is the result, BUT it is important to know that the our Sensor Link is limited to a maximum of 7000 steps per second.

If the encoder turns faster and generates more steps per second, the Sensor Link will lose track of the absolute step count of the encoder. This can lead to drifting and wrong values, if the Sensor Link is being used for accurate Screen Tracking.

To avoid this, it is recommended to specify an encoder that will run at a maximum amount of 6000 steps per second.

!!! Please be aware that 6000 steps are not equal to rotation per second or Pulses per Revolution!!!

Due to the output circuit data of the encoder (which is a HTL) there are four shoulders for one circuit. That means for example: An encoder with 1500 PPR (Pulses per Revolution) generates 6000 steps per second at one full rotation per second. (4x1500 PPR = 6.000 steps/sec.)

To choose the right encoder for your application you will also need to take into consideration the following:

Is there any further gear mounted on the device which is turning?
Is the shaft, where the rotation is coming from, bigger than the axis of the encoder?

This is why you need to know the environment of where the encoder is going to be used as well as how many rotations per second the axis of the encoder will be turning.

Sensor Link has been tested with Wachendorff Encoders (www.wachendorff.com).
You may get similar encoders from other manufacturers as well, but the naming of the different signals and product specifications may vary.

Here is an example of how to determine the right product code for a Wachendorff encoder suitable for Sensor Link.

Please Note: All the encoders are custom-built. These are not stock items!

Example encoder: **WDG40S 360 ABN H24 SB5**

WDG40S	- is the type of the encoder – here it means that it's a shaft encoder.
360	- is equal to 360 PPR (Pulses per Revolution)
ABN	- is the channel type
H24	- determines an HTL Output circuit
SB5	- determines the connector type, for Sensor Link a 5-Pin, M12-sensor Connector

Although encoders can be ordered with a fixed mounted cable, this is not recommended for service flexibility.

The connection cable between encoder and Sensor Link is available in various lengths.

Below graphic shows an example from Wachendorff:

Ordering information: Please see our general technical data at: www.wachendorff-automation.com/gtd

Output circuit:

Resolution PPR	Power supply VDC	Output circuit		Light reserve warning	Order Key
up to 2500	4.75 - 5.5	⊓⊔	TTL	-	H05
			TTL, RS422 comp., inverted	-	R05
	10 - 30	⊓⊔	HTL	-	H24
			HTL inverted	-	R24

Channels: AB, ABN (up to 2048 PPR)

Pulses per revolution PPR:
4, 9, 10, 15, 20, 25, 28, 30, 36, 40, 50, 60, 90, 100, 120, 125, 128, 150, 160, 180, 200, 235, 250, 300, 314, 318, 360, 400, 500, 600, 625, 635, 720, 900, 1000, 1024, 1080, 1200, 1250, 1500, 1800, 2000, 2048, 2500.

Other PPRs on request

Electrical connections:

Order key	Outgoing	Description	ABN inv.
Cable: (Length 2 m standard)			
K1	radial	shield not connected (IP50)	•
K2	axial	shield not connected	•
L2		shield connected to encoder housing	•
K3	radial	shield not connected	•
L3		shield connected to encoder housing	•
Connector:			
SI5	axial	5-pin, connector	-
SI6	axial	6-pin, connector	-
SI8	axial	8-pin, connector	•
SI12	axial	12-pin, connector	•
S2	axial	7-pin, connector	-
SB4	axial	4-pin, M12-sensor-connector	-
SB5	axial	5-pin, M12-sensor-connector	-
SB8	axial	8-pin, M12-sensor-connector	•
SB12	axial	12-pin, M12-sensor-connector	•

Options:

Empty	=	Without option
ACA	=	Low-temperature -40 °C up to +80 °C
AAC	=	Low-friction bearings
AAO	=	Shaft sealed to IP67
In decimetres	=	Cable length

Order No.:						
Example	WDG 40S	2500	ABN	H24	K2	
Your encoder	WDG 40S	360	ABN	H24	SB5	

8.9.6.3 User Interface

The Sensor Link UI acts as the interpreter of the incoming data from a Sensor Link to a Pandoras Box Master System.

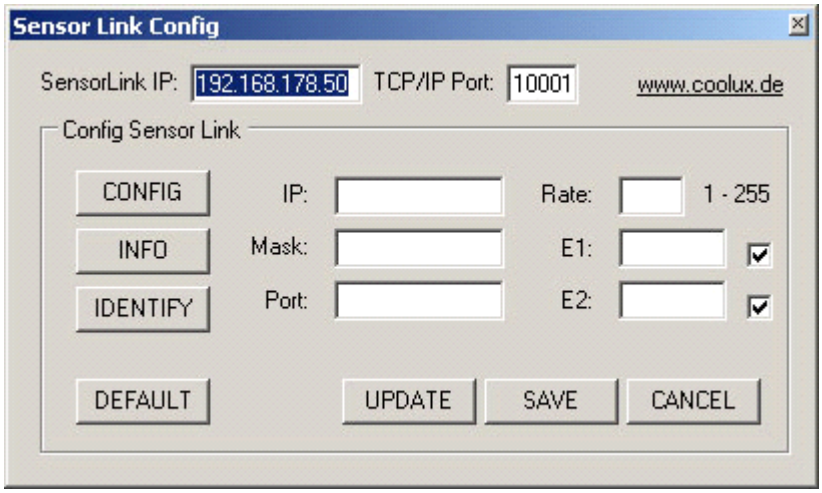
There are two different ways to use the Sensor Link to control the Pandoras Box Master software:
- Using the Sensor Links Configuration Tool [2037] and the Widget Designer PRO (recommended!).

This allows you to use the Sensor Link's whole functional range.
- Using the User Interface of the Sensor Link Software [2039] that will be delivered with the Sensor Link.

8.9.6.3.1 Configuration Tool and Widget Designer Pro

For the best flexibility and to be able using the whole function range to control Pandoras Box we recommend using the Widget Designer as processing tool for the data coming from the Sensor Link.

Please start the Sensor Link Configuration Tool from our Download-Center.



To configure the Sensor Link device, please enter the Sensor Link IP address and port and press [CONFIG]. You may now change the IP or Encoder settings if necessary:

SensorLink IP and TCP/IP Port:
Enter here the Sensor Link's IP address and TCP/IP Port.

[CONFIG]:
Press [CONFIG] to switch the unit to configuration mode.

[INFO]:
Press [INFO] to download the current settings from the unit.

[IDENTIFY]
Press [IDENTIFY] to identify the units, the right LEDs will blink for 3 seconds.

Changing the Sensor Links Settings

[DEFAULT]
Press DEFAULT to load the default settings in the settings fields.

IP:
Please enter the new IP address that you want to assign to the unit here.

MASK
Please enter the new Subnet Mask that you want to assign to the unit here.

PORT
Please enter the new TCP/IP Port number that you want to assign to the unit here.

RATE
The Rate setting changes the transmission rate, the default of 14 is recommended as this is ~70 Hz refresh cycle in order to be ahead of any displays refresh cycle. Faster rates (<14) are possible but may cause your network or processor to slow down.

E1
Please set the step count of your encoder 1 here. If the encoder provides e.g. 1440 steps, enter 1439 because the counting starts with zero.

E1 Checkbox
If you need to use an external index GPI switch to reset the encoder 1 counter to 0 you may check this box and enable switch S7 to be the index switch.

E2
Please set the step count of your encoder 2 here. If the encoder provides e.g. 1440 steps, enter 1439 because the counting starts with zero.

E2 Checkbox
If you need to use an external index GPI switch to reset the encoder 2 counter to 0 you may check this box and enable switch S8 to be the index switch.

[UPDATE]:
Press [UPDATE] to upload the new settings to the unit.

[SAVE]:
Press [SAVE] to store the new settings in the unit.

[CANCEL]:
You may press [CANCEL] at any time to leave the config mode directly.

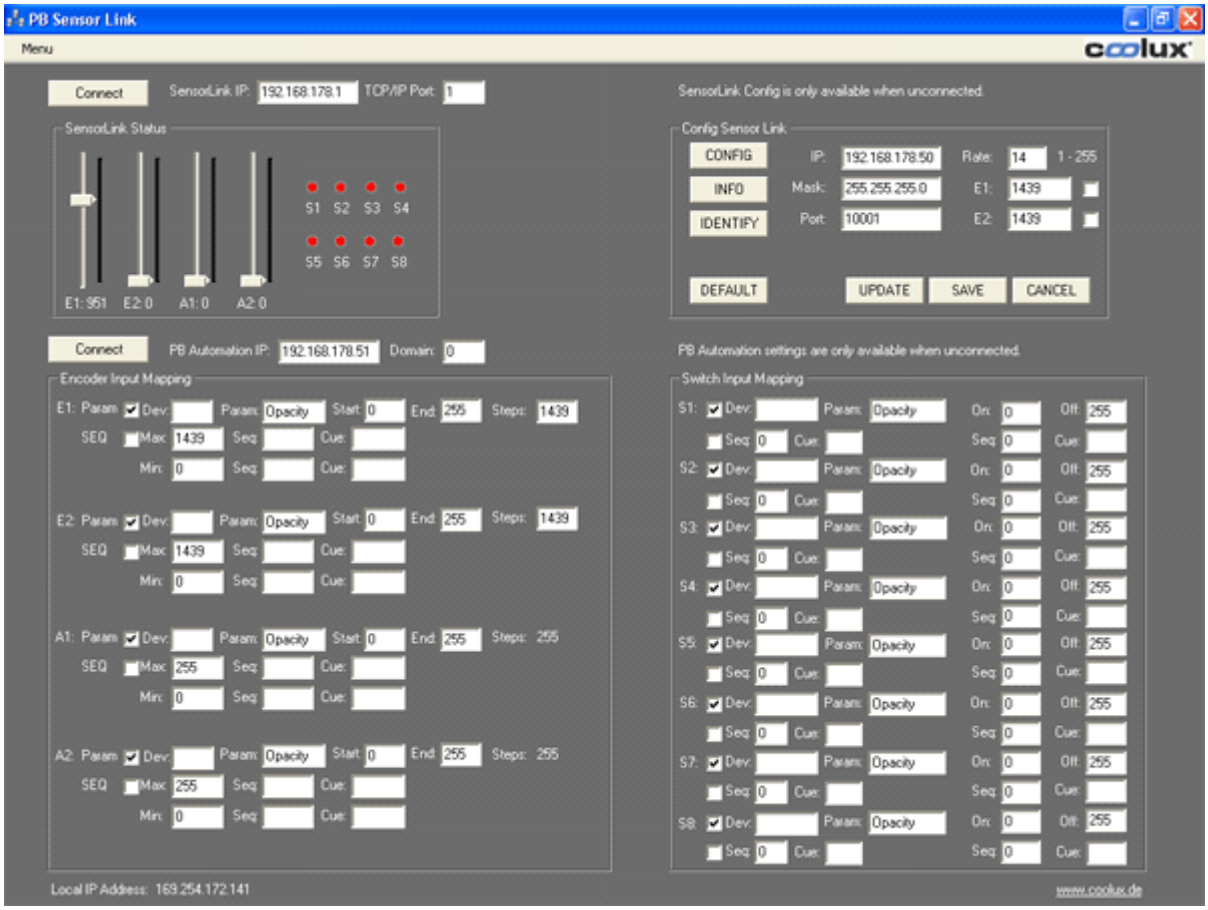
No access?
If you can not access the unit anymore, press the blue reset button for 5 seconds during power on.

The unit is set by default to
IP 192.168.178.50
Subnet 255.255.255.0
Port 10001

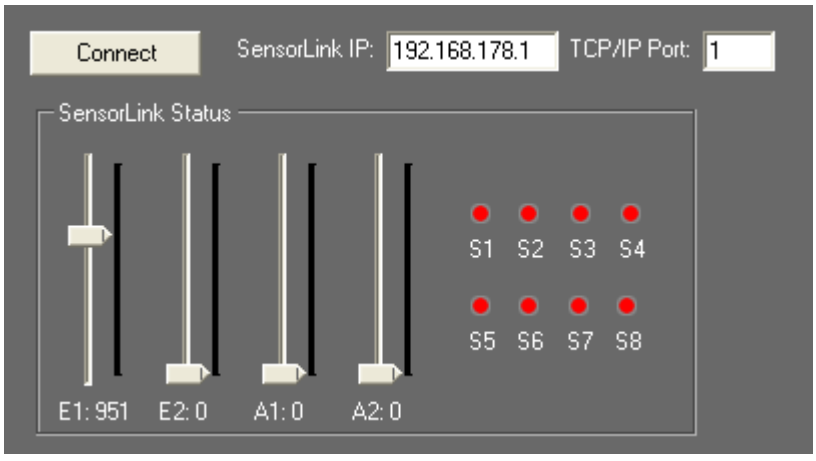
Important:
To use the Sensor Link now please powercycle the unit first!

If this setup is done you can start to read out and process the data coming from the Sensor Link Device. To do this create a Sensor Link Input Node [1026] and route the data to any filter or output nodes.

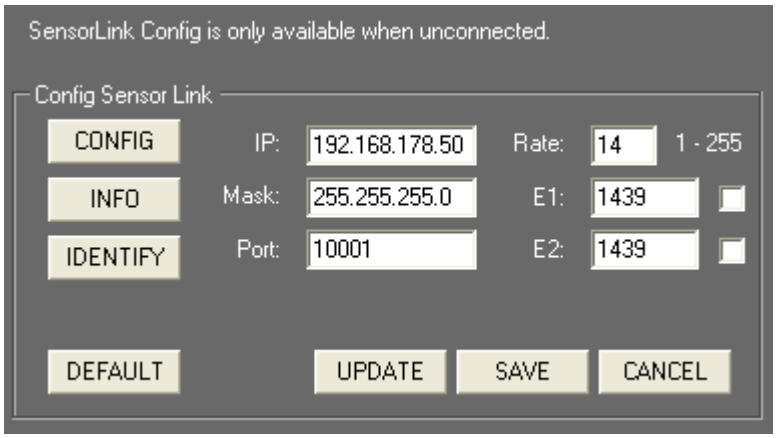
8.9.6.3.2 Stand-Alone Software



To configure the Sensor Link device, please enter the Sensor Link IP address and port in the top left fields, do not press connect at this time.



Press [CONFIG] in the ConfigSensor Link box.



[CONFIG]
When [CONFIG] is pressed the left LED will blink and the Sensor link is switched to config mode.

[INFO]
Press [INFO] to retrieve the current settings of the Sensor links into the settings fields.

[IDENTIFY]
Press [IDENTIFY] to identify the units, the right LEDs will blink for 3 seconds.
Changing the Sensor Links Settings

[DEFAULT]
Press [DEFAULT] to load the default settings in the settings fields.

IP
Please enter the new IP address that you want to assign to the unit here.

MASK
Please enter the new Subnet Mask that you want to assign to the unit here.

PORT
Please enter the new TCP/IP Port number that you want to assign to the unit here.

RATE
The Rate setting changes the transmission rate, the default of 14 is recommended as this is ~70 Hz refresh cycle in order to be ahead of any displays refresh cycle. Faster rates (<14) are possible but may cause your network or processor to slow down.

E1
Please set the step count of your encoder 1 here.

E1 Checkbox
If you need to use an external index GPI switch to reset the encoder 1 counter to 0 you may check this box and enable switch S7 to be the index switch.

E2
Please set the step count of your encoder 2 here.

E2 Checkbox
If you need to use an external index GPI switch to reset the encoder 2 counter to 0 you may check this box and enable switch S8 to be the index switch.

UPLOAD a new Configuration to Sensor Link

To upload new settings into the unit, press [UPDATE].

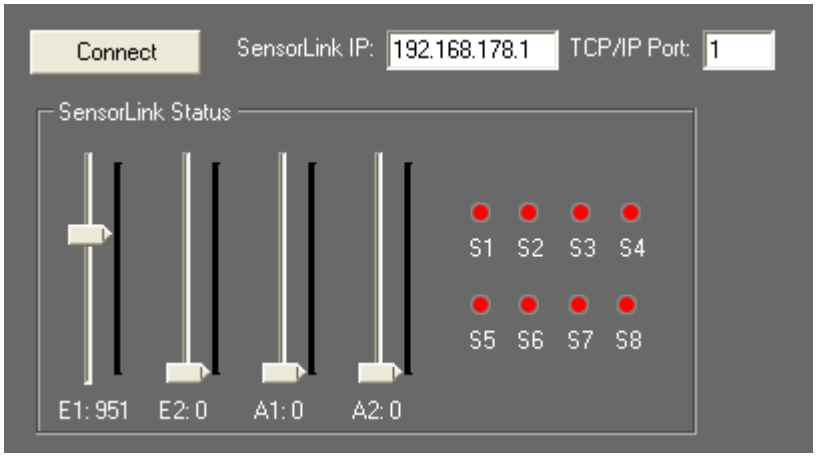
Press [SAVE] to store the new settings into the unit.
The save process takes several seconds and will reboot the unit.

Once all LEDs light up again the unit exits config mode and is ready to being connected for data transmission.

You may press [CANCEL] at any time to leave the config mode directly.

RUN Mode

Once all settings are stored into the unit, you may connect to the Sensor Link device for data transmission by entering its IP address and port and pressing the [Connect] button.



If you do not know the current IP address, you might want to use the default IP 192.168.178.50 and Port 10001.

Please note that your PC must be in the same IP address range in order to communicate with the Sensor Link Device.

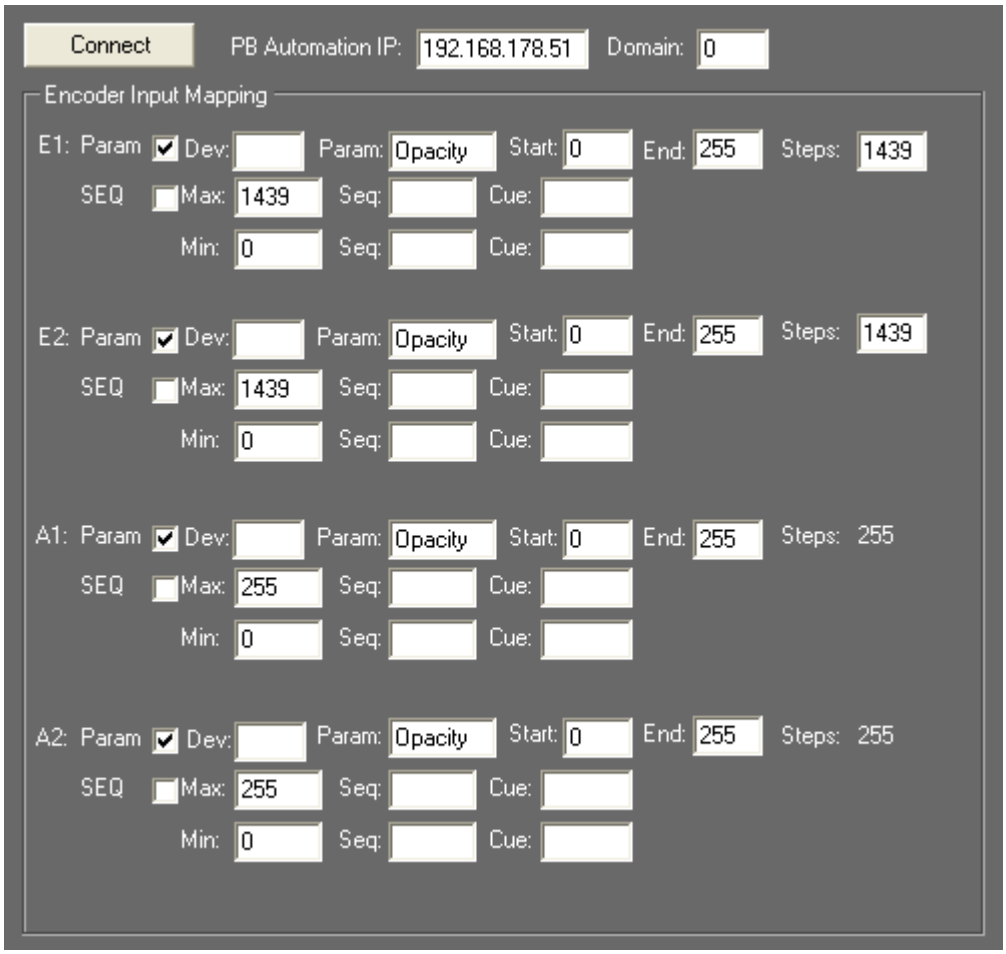
Once connected, you should be able to read the incoming values in the Sensor Link Status box. To check the switches, you may want to connect the 12V pin with any of the switch pins, the switch status S1-8 will turn green once the contact is closed.

Please do not short circuit the 12V (left most pin) with the ground pin (right most pin), this might damage the unit.

Triggering Pandoras Box from Sensor Link

In order to communicate with Pandoras Box, you need to define the function of the incoming data via Sensor Link. For each Encoder and each switch you may choose between parameter and sequence trigger mode. Use the checkboxes to set each input up for the according mode.

Encoder and Analog Inputs



Parameter Mode:
In Parameter Mode you may enter one or multiple devices to be linked to the incoming values.

Device Selection
To enter a single device like Server1 Layer 4 enter in the Dev field: "1,4".
To enter multiple devices, leave a space between each device like: "1,4 2,4 3,4" this setting will route all incoming data to layer 4 of machine one, two and three.

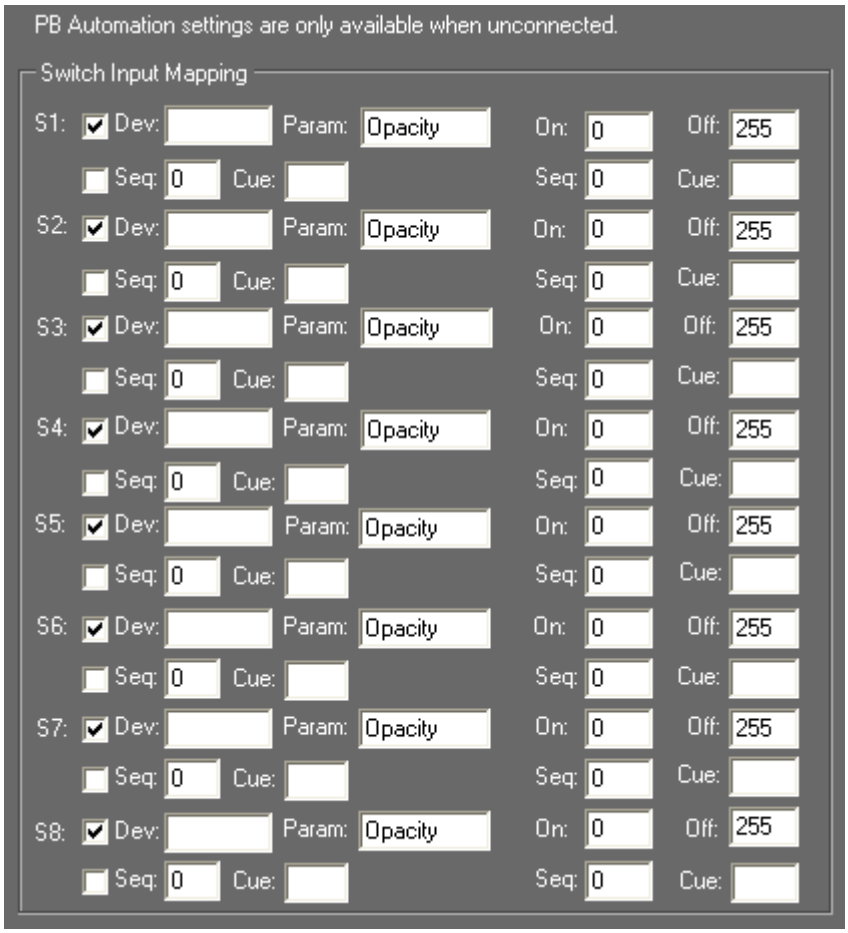
Parameter Setup
Please enter here the desired parameter name that you want to link to, like "Opacity" or "X Pos".

Start and End values
Please enter here the start and end value range that you want to map your incoming data to.

Steps
Please enter the amount of steps of your connected device (for encoders only). While the Encoder Step count is determined by the hardware connected to the Sensor Link, any analog input device can be read between 0-10V with a step count of 0-255.

Sequence Mode:
In sequence mode you may enter the desired sequence and cue to be triggered above or below the incoming min. and max. values of the connected Encoder or analog input device.

Switch Inputs



Switch inputs are handled similar to encoder inputs, besides the fact that you can setup only an on and off value in parameter mode. In sequence mode you may choose a sequence and cue separate for the on and off state of the switch.

Sensor Link Data Transmission to Pandoras Box

In order to connect to a Pandoras Box Master System your PC must be in the same IP address range and you will need to enter the Masters IP and Pandoras Box domain in the according setting fields and connect to Pandoras Box.

Please be aware that once the connection to Pandoras Box is enabled the Sensor Link user interface will pause all interface updates of faders and switch states for optimum live performance.

Resetting the Unit
To reset the Sensor Link, please press the reset button on the rear side of the unit and hold it for 5 sec while power-cycling:
Power off the Serial, press the reset button, power it on again and hold the button for 5 sec pressed.

When the unit is reset the following settings will be set:
IP: 192.168.178.50
Subnet-Mask: 255.255.255.0
Port: 10001

Troubleshooting:

Problem: The incoming data comes in slowly and does not update fast enough.
Solution: Press cancel to exit config mode and reconnect to the Sensor Link unit or power-cycle the unit to exit config mode as well.

Problem: You caused a short circuit because of connecting Pin 12 (+12V) and Pin 1 (Ground).
Solution: There is an internal fuse inside the Sensor Link that has to be replaced. Please contact Support.pandorasbox@christiedigital.com

8.9.7 Serial Link

Please note, that Christie Digital Systems has discontinued the Serial Link. Please refer to JLCooper Electronics and their product eBOX.

The Serial Link is a general purpose interface box that converts 4 serial communication ports and 24 GPI (General Purpose Interface) inputs and outputs to 100/10baseT Ethernet. The serial ports can be configured in the field to appear as EIA/TIA RS-232E and CCITT V.28 or as EIA/TIA RS-422A, RS-423 and Federal Standards 1020 & 1030 ports. Additionally, the port direction can be configured as DTE or DCE on each port independently.

The Serial Link communicates over standard TCP/IP which allows is be used with any host computer running any operating system that uses TCP/IP protocol. The Serial Link can also be connected to other Serial Links to allow longer runs than traditional serial and GPI cables. Since the Serial Link uses TCP/IP, traffic can be routed over internal LANs, wireless LANs, MANs, WANs and even over the public Internet.

Most configurations can be accomplished through a web page server built into the Serial Link. Items such as port speed, parity, IP address, remote IP address and TCP port are set using a standard web browser. Settings are stored in nonvolatile memory.

Typically, the Serial Link functions as a Server, passively waiting for Client devices to connect to it. The device can be a computer or another Serial Link configured as a Client. When the Serial Link is configured as a Client, it will actively attempt to connect to the Server Serial Link. Once this is accomplished, the Serial Link will either pass data received in the serial or GPI ports to the remote Serial Link. If there is no data received in the Serial Link, the Serial Link will not send any TCP packets.

In addition, the Serial Link can operate as a GPI to Serial Converter or GPI to Ethernet Converter. In these modes, the Serial Link will convert GPI input triggers to deck commands.

SERIAL LINK SETUP

The serial control will give you access and control to most routing switchers, projectors, shutters or other device parameters that are remote controllable via the common RS 232 and 422 serial protocols.

In order to control external serial devices over RS 232 or RS 422 you will need to add a serial link device to the device tree. Any serial ASCII or hex string can be stored and copied to any point in the timeline and be recalled at any time the time cursor hits the stored serial key. Once you set the first key you can assign the IP address and port ID in the keys properties as well as the ASCII or hex command for that key.

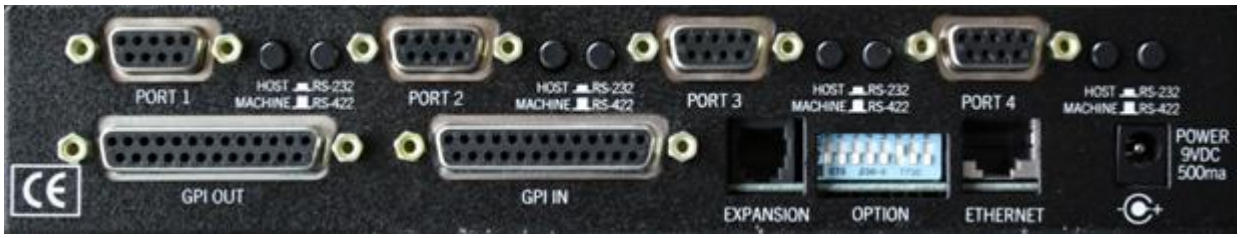
You will need a serial link device to receive and output serial commands. See further instructions in the serial link manual for port setting baud rates and parity settings.

Please find below the .clib code for the basic Serial Link device:

```
<?xml encoding="yes" ?>
<descripDevice type="fixtureSerial" artNetIndexScope="1">
<descripModule type="paramSerial" name="Com" default="0" artNetIndexOff="0" />
</descripDevice>
```

Connections

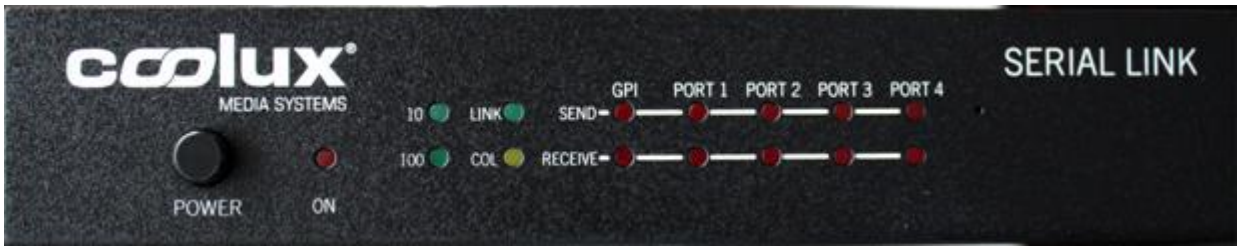
The Serial Link connections are straightforward:
1. Plug the power supply into the Serial Link.
2. Plug a network cable into the Ethernet jack.
3. Connect serial cables into ports 1-4.



The rear side

ETHERNET
This Serial Link port is just like an Ethernet port on a computer, to connect it to a hub, switch or router, use a straight through cable. To connect it to another Serial Link or computer, use a crossover cable.

The Serial Link supports IEEE 802.3u clause 28 Auto-Negotiation which automatically senses the Ethernet port speed & duplex operation and chooses the highest performance settings.



The front panel

Four LEDs on the front panel indicate various operating conditions of the Ethernet port. These LEDs are:
• Link
• 100BaseT activity
• 10BaseT activity
• Collision

CONFIGURATION

Operating Modes
The Serial Link has five distinct modes of operation that are set by the rear panel DIP switches. The DIP switches are read only at power on so the Serial Link must be power cycled for the changes to take effect.

SW8	SW7	SW5	
On	X	Off	Access web page at 192.168.254.102
Off	Off	Off	Serial Link is server at address specified on next page
Off	On	Off	Serial Link is client at address specified on next page
Off	On	On	Serial Link performs GPI to serial conversion
Off	Off	On	Serial Link performs GPI to Ethernet conversion
X = Don't care			

Configuration Mode
On the Configuration Web page, various settings such as port rate and parity, IP addresses and ports, and password can be modified.

Note: The Serial Link will not send Serial/GPI messages when the Serial Link is set to configuration mode.

To access the Configuration Web Page of the Serial Link Device, please set SW8 and SW6 to ON and all others to OFF. The PC Firewall and an Antivirus system have to be turned off!

This page is accessed by setting SW8 to the ‘On' position and typing "http://192.168.254.102" into your web browser (use Internet Explorer, Safari and Firefox do not work). During this time, the normal operation of the Serial Link is suspended.

Note: The configuration web page is always set to 192.168.254.102 regardless of the state of the DIP switches.

Note: In the configuration mode, the subnet mask is set to 255.255.255.0. This means that the Serial Link will only see traffic from computers with and IP address of 192.168.254.nnn. You will need to change your computers IP address to 192.168.254.nnn where nnn = any number except 0, 102 and 255. This will allow your computer to access the configuration page of the Serial Link.

Make any changes that are necessary for your system and click on SUBMIT. These changes are stored in non-volatile memory and are loaded at power up.
Default settings: Timeout = 100ms and max buff size = 255.

A Page with a Pink background should now be loaded into Internet Explorer to tell you to power cycle the unit, but first:
- Switch DIP8 to OFF position.
- Wait approx 10 – 15 sec !
- Now power cycle the Serial Link!

To verify the Settings:
-Power Off Serial Link
-Set DIP 8 to ON again
-Power Cycle the Serial Link
-Launch internet explorer and make sure to empty the browser cache by clicking on Safety -> Delete Browsing History… -> check all check boxes and click on the delete button
-Now type in 192.168.254.102 in the address field of Internet Explorer.

To use Serial Link with a custom IP address please make sure to set DIP 1,2,3 to ON and 8 to OFF

DIP 6 should always remain in the ON position as this is disables the internal password protection that is currently not used.

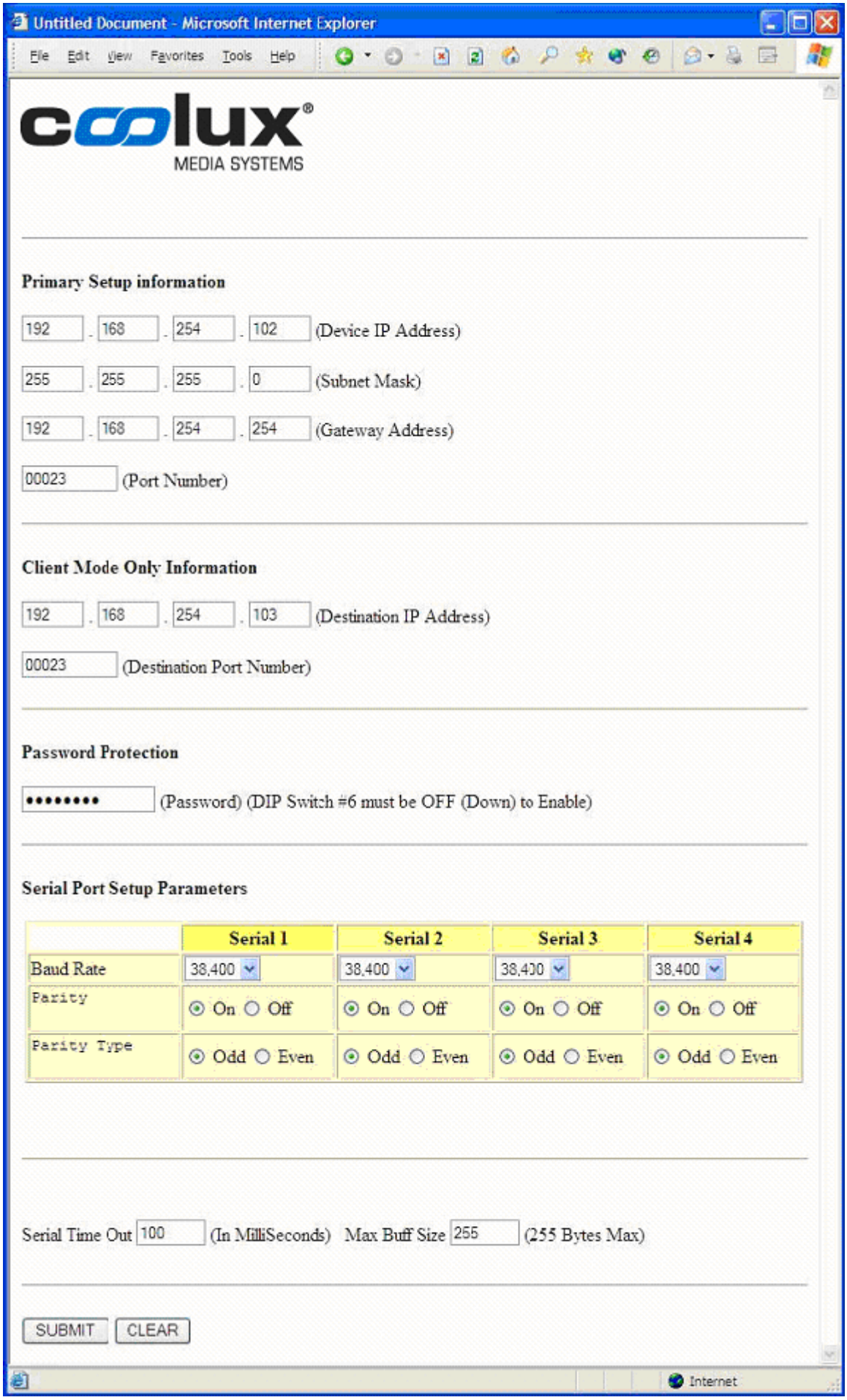

Untitled Document - Microsoft Internet Explorer
File Edit View Favorites Tools Help
coolux
MEDIA SYSTEMS
Primary Setup information
192 . 168 . 254 . 102 (Device IP Address)
255 . 255 . 255 . 0 (Subnet Mask)
192 . 168 . 254 . 254 (Gateway Address)
00023 (Port Number)
Client Mode Only Information
192 . 168 . 254 . 103 (Destination IP Address)
00023 (Destination Port Number)
Password Protection
•••••••• (Password) (DIP Switch #6 must be OFF (Down) to Enable)
Serial Port Setup Parameters
Serial 1
Serial 2
Serial 3
Serial 4
Baud Rate
38,400
Parity
On Off
Parity Type
Odd Even
Serial Time Out 100 (In MilliSeconds) Max Buff Size 255 (255 Bytes Max)
SUBMIT
CLEAR
Internet

Image: Configuration Web Page

Operational Settings

Device IP Address:
IP address of this Serial Link when SW1, SW2 and SW3=On.

Subnet Mask:
The mask is a binary pattern that is matched up with the IP address to turn part of the host ID address field into a field for subnets.

Gateway Address:
IP address of gateway router which connects to other networks.

Port Number:
TCP port of this Serial Link when SW1, SW2 and SW3=On.

Destination IP Address:
IP address of remote Serial Link when SW1, SW2 and SW3=On. This is used when Serial Link is configured as a client (SW7=On).

Destination Port Number:
TCP port of remote Serial Link when SW1, SW2 and SW3=On. This is used when Serial Link is configured as a client (SW7=On).

Password:
Eight character alphanumeric password that is embedded in the Ethernet packet that prevents unauthorized Serial Links from passing unintended packets. Both Serial Links must have the same password and have password protection turned on (SW6=Off) for this feature to work.

Baud Rate:
Sets the port speed of the individual serial ports.

Parity and Parity Type:
Enables or disables parity and sets parity type of the serial ports.

Serial Time Out:
Sets the time that the Serial Link will wait for data from the serial ports.

Max Buff Size:
Sets the maximum buffer size of the serial ports.

Serial Link IP Address:
The IP address of the Serial Link can be set by the rear panel DIP switches or by the internal web page. As above, the DIP switches are read only at power on so the Serial Link must be power cycled for the changes to take effect.

Here is a table of how IP address and the TCP port are set with the DIP switches in Serial Link Server and Serial Link Client modes.

SW3	SW2	SW1	SW7	Mode	IP Address	Port
Off	Off	Off	Off	Server	192.168.254.102	23
Off	Off	On	Off	Client	192.168.254.103	23
Off	Off	On	Off	Server	192.168.254.104	23
Off	Off	On	On	Client	192.168.254.105	23
Off	On	Off	Off	Server	192.168.254.106	23
Off	On	Off	On	Client	192.168.254.107	23
Off	On	On	Off	Server	10.0.0.128	23
Off	On	On	On	Client	10.0.0.129	23
On	Off	Off	Off	Server	10.0.0.130	23
On	Off	Off	On	Client	10.0.0.131	23
On	Off	On	Off	Server	10.0.0.132	23
On	Off	On	On	Client	10.0.0.133	23
On	On	Off	Off	Server	172.16.0.128	23
On	On	Off	On	Client	172.16.0.129	23
On	On	On	Off	Server	Set by user	
On	On	On	On	Client	Set by user	

IP Address Configuration for Serial Link Server and Serial Link Client modes

Electrical Connections

Power
The Serial Link requires a 9 volt DC supply capable of delivering at least 500 milliamps. The unit comes with a power supply appropriate for the country in which the unit was sold. If you need a power supply specific to your location, please contact us or your local distributor.

Warning: Using a power supply other than the units specified in the above table can result in damage to the Serial Link and/or other equipment which is not covered by warranty.

Christie
Pandoras Box

Part 9

Tools and Other Software

9 Tools and Other Software

This chapter describes all tools available with a Pandoras Box installation:

Art-Net Monitor 2051
Dome Master 2055
Image Converter 2069
Leica 3D Disto 2076
Matrix Patcher 2077
Splitter 2120
Quicktime Converter 108 (in Encoding chapter)
Warper 2129

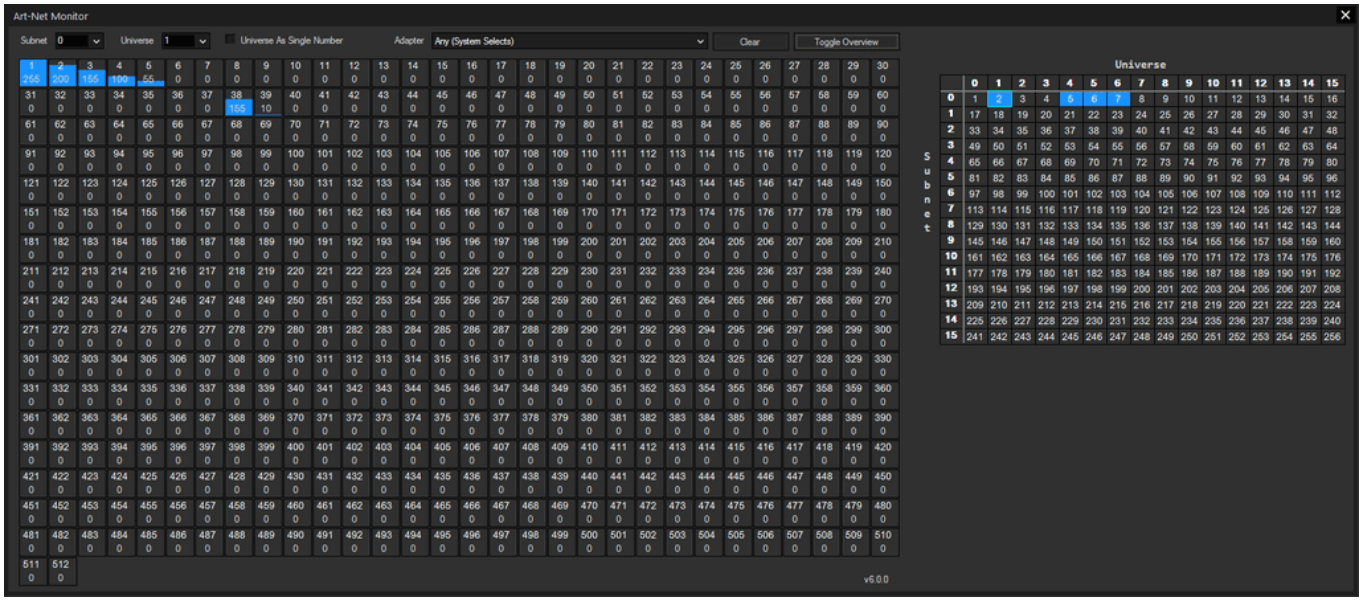
In addition, the following tools are explained that can be found in our "Download-Center" or are pre-installed on our hardware:

CITP Desktop Streamer 2052
Factory Reset Tool 2060
PB Menu 2097
Server Management Application 2109

The last chapter covers Third Party Software 2177.

9.1 Art-Net Monitor

The Art-Net Monitor is a very useful tool if you would like to monitor Art-Net data on the network or simply find out if the DMX channels you are looking for have the right values.

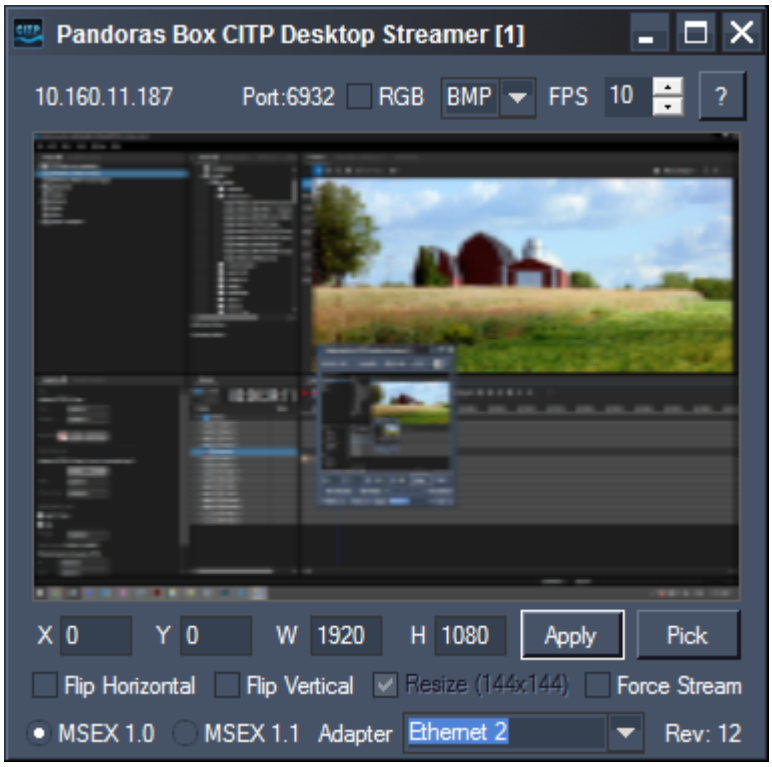


The button "Toggle Overview" shows (or hides) a table with all Universes and Subnets. The table can also be clicked to see all channels from a respective universe. Alternatively, you can choose the universe with the drop-down list(s) on the left side. You can either choose a "Universe" and "Subnet" or a single "Universe #" in case the option "Universe As Single Number" is checked.
The left table shows all 512 channels belonging to the selected universe. If no values are sent on the universe, "---" is displayed per channel. As soon as one channel receives a value, it is displayed with a blue bar graph as seen above. In the universe table on the right side, the universe is also depicted in blue. Note that all other channels for the universe are displayed with a 0 value as Art-Net always sends the entire universe, not only one channel.
If channel values cannot be received anymore (e.g. because the connection is interrupted) the displayed blue values turn gray. In other words, gray data is just an information what was sent before but is not present anymore. In that case you can erase them with the "Clear" button. Now, "---" is displayed per channel until values are received again.

The drop-down list "Adapter" lets you select the network adapter that is used for Art-Net.

If you use WD to send Art-Net and you do so by broadcasting the values, you will be able to see these values in the Art-Net Monitor too. If you are unicasting the values, please switch to another Art-Net Monitor, e.g. the one in the PB Menu [2097]. More information [2083] about broadcast and unicast...

9.2 CITP Desktop Streamer



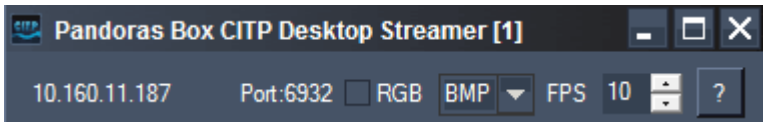
The Pandoras Box CITP based Desktop Streamer is a versatile tool to stream any Windows desktop region, for example the Preview section of Pandoras Box or any application content, into WYSIWYG or Capture Polar directly. This way you can project your favorite video content directly into WYSIWYG without using any capture cards. It is designed for pre-programming and can also be set up at different frame-rates to match the systems performance.

The CITP Desktop Streamer works with Windows version 7 and 8.1. You may use several instances at the same time.

We also offer another tool, the StreamiX Desktop [724] (since V5.1) which is a TCP based Desktop Streamer that works as a Live Input in Pandoras Box,

The user interface is divided into three sections: the Streaming settings [2052], the Preview Window [2053] and the Capture Settings [2053]. This chapter also explains the Capture Polar Settings [2053] and WYSIWYG Settings [2054].

Streaming Settings



Window Title
You may start several instances of the Desktop Streamer at the same time to pass several CITP streams into your WYSIWYG application. The number shown in the Window Title [] displays the stream number: Pandoras Box CITP Desktop Streamer [1] = Stream 1.

IP / Port
On the top left the computer's IP address and the used TCP/IP-Port are displayed. Please make sure to have a static IP address while using the streamer in combination with WYSIWYG and Capture.

RGB
The "RGB" option allows you to match to the CITP color space of your WYSIWYG version. If this option is disabled, the color space BGR is used for CITP streaming. Enable it and the color space will be changed to RGB. Note: Up to WYSIWYG Rev.25 the color space BGR (according to CITP 1.0) is used. Up from WYSIWYG Rev. 26 the color space RGB (according to CITP 1.2) is used.

BMP/JPG
The drop-down list offers two image formats to be used for streaming: Bmp and Jpeg.

FPS
The Framerate can be set to values between 1 and 30 fps, according to your system's performance.
Please note: you don't have to press [Apply], your changes are applied immediately.

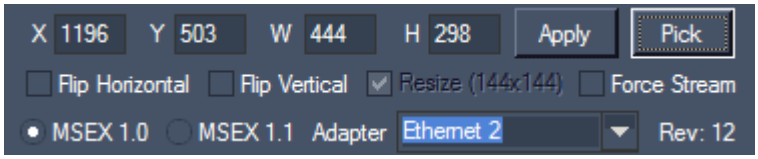
? (Help)
See here information about the settings you will need in WYSIWYG in order to use the CITP stream, see WYSIWYG settings [2054].

Preview

The Preview shows the part of your desktop that is sent via CITP. According to the CITP specifications the size of the streamed media is 144x144 px.

Please note: The Preview will not show modifications like image flipping or a change in the color space. This will be seen in WYSIWYG only.

Capture Settings



X / Y / W / H / Apply / Pick - Defining the capture region
There are two possibilities to define the region of your desktop that is going to be captured for CITP streaming:
1. enter the region's width and height (W/H) as well as its starting pixel (X/Y) into the text fields and press "Apply".
2. press "Pick" and span the desired desktop region with the mouse. However, this works only on the primary monitor. Use the first method to stream a region that is on a secondary monitor.

Flip Horizontal / Flip Vertical
Depending on the texture mapping in WYSIWYG, these two options offer you to flip the CITP stream horizontally and / or vertically.
Please note: You don't have to press "Apply", your changes are applied immediately. The Preview will not show modifications like image flipping.

Force Stream
Depending on your streaming tool, this option might be needed to start sending the stream according to frame rate set up above.

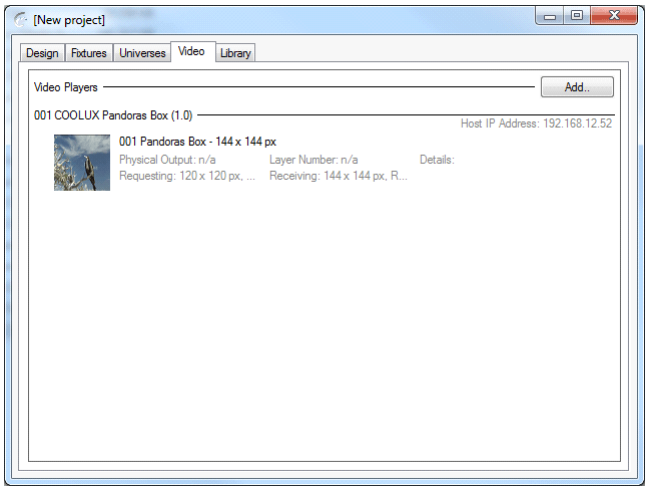
MSEX version
Choose which MSEX version should be used: 1.0 or 1.1

Adapter
This drop-down list offers support for multiple network adapters, simply choose the desired NIC that should be used for streaming but make sure to have a static IP address.

Capture Polar Settings

In order to stream your desktop or parts of it over to Capture Polar, please follow these steps:

- Start Capture Polar.
- Open a new or existing project.
- Start the Pandoras Box Desktop Streamer.
- Go to the Video Tab.
- The COOLUX Pandoras Box stream should be displayed in the list.



- Go to the Design tab.
- In Materials you add a new material".
- Select the Coolux Pandoras Box stream, in the right window next to Video.
- Add this material as a texture to the desired objects.
- The Video Stream can be stretched manually or automatic to one or more objects.

WYSIWYG Settings

In order to stream your desktop or parts of it over to WYSIWYG, please follow these steps:

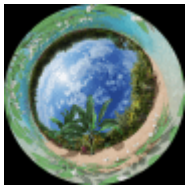
- Start WYSIWYG.
- Go to "Options - Application Options" tab "Additional Interfaces" and check "Enable CITP".
- Close and restart WYSIWYG.
- Within a project open the Video Manager.
  (this can be found in the menus Tools/Design/Live depending on your Working-Mode).
- Add a new video source.
- Check "CITP Video Stream" and click browse.
- Select the COOLUX Pandoras Box stream from the device list.
- Assign the stream as a texture to the desired objects.
- Open the "Video Tool" from "Live/Design".
- Select the CITP stream from the drop-down list and click the play-button.

9.3 Dome Master

The "Dome Master" tool converts dome master files to spherical map files.

The Dome Master is installed automatically when you install a Pandoras Box application. To open it, you can either choose it from the Tools menu [122] from your Pandoras Box master software or open it stand-alone (on the Master or Client), e.g. by choosing it from the Pandoras Box folder in the Windows Start menu. The installation path is for example: `C:\Program Files\Christie\Pandoras Box 8.0.0\data\tools\PandorasBoxDomeMaster.exe`

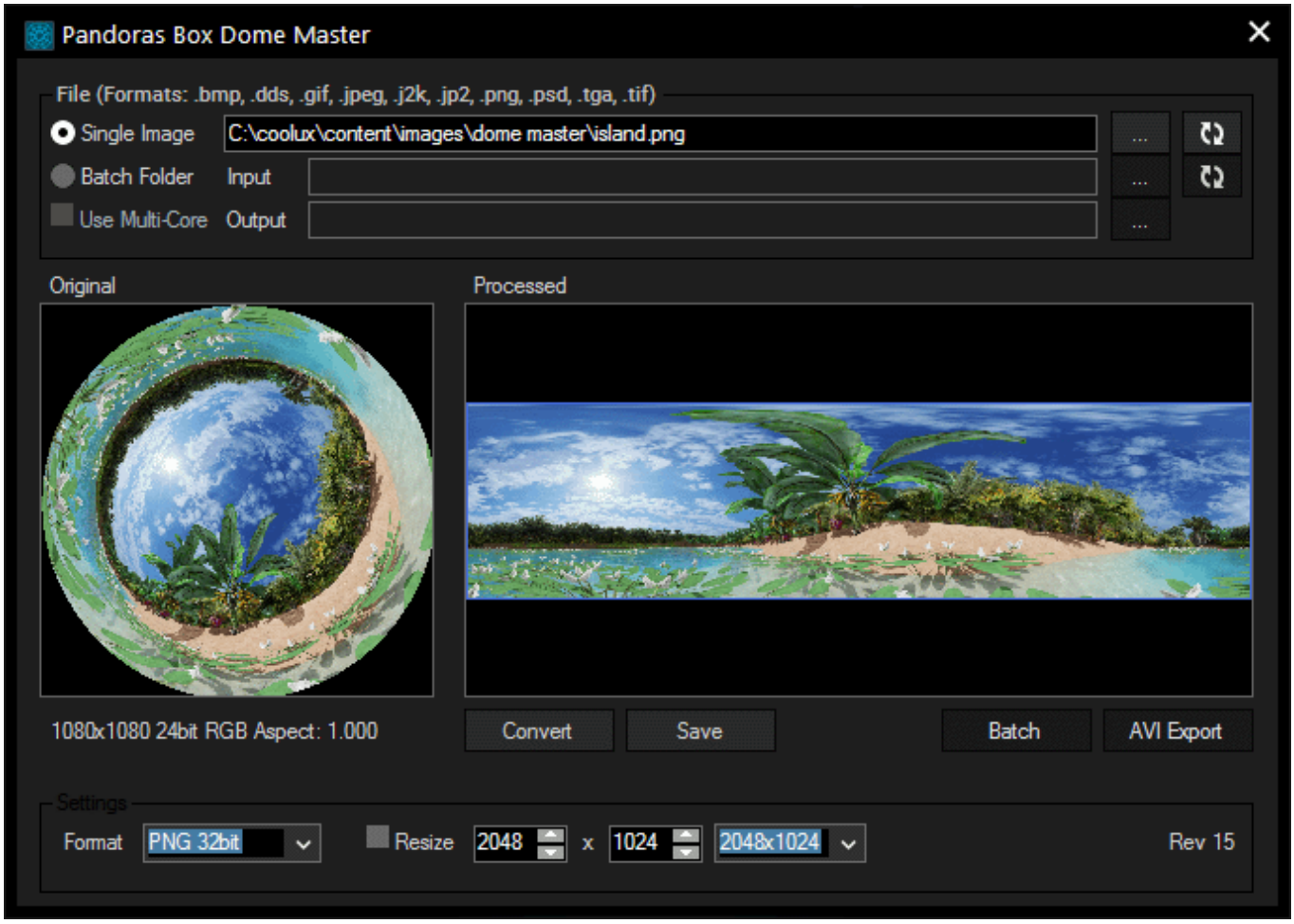
The Dome Master Converter allows a 1:1 conversion via pixel blending to achieve the best quality for the spherical map. Its sub-pixel accuracy allows optimum image sharpness for large scale projections. Due to this process it can take a longer time for the conversion.



A Dome Master file describes a sphere or a hemisphere as circular image within a square. The outer edge of the circle corresponds to the dome's equator, the circle's center point corresponds to the dome's pole. For content creators, this is a very common file type.

To map content to 3D spheres or Dome Warpings however, it is necessary to convert Dome Master files to spherical maps. These spherical maps will have the aspect ratio 2:1 (full sphere) or 4:1 (hemisphere). Especially, in case of a larger multi-projection with many Servers, the image can now be split up or sliced, e.g. using the Splitter tool [2120]. Each slice can then be spread to the dedicated playback machine and warped separately. This saves a lot playback performance or makes playback possible in first place, depending on the resolution.

The following chapters explain how to use the Dome Master for...:
- Single Image Conversion [2057]
- Batch Folder Conversion



Input Formats

You can import a single image or a folder containing an image sequence. The maximum master file size is 8192 x 8192 px if using Win8 or above. The following list shows the supported input formats.

BMP, GIF, JPG, JP2, J2K, PNG, PSD, TGA, TIF(F), DDS or SNP

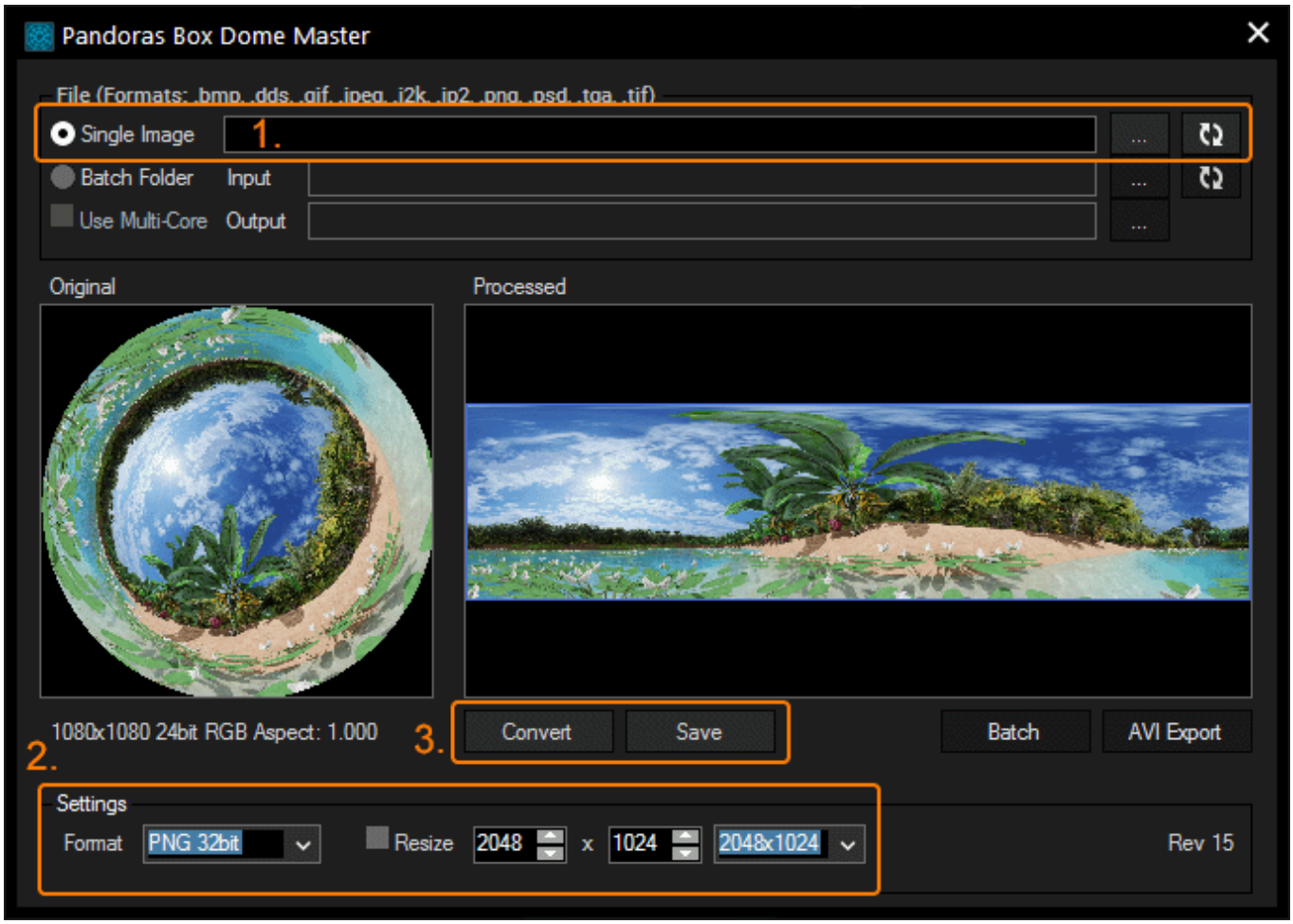
Output Formats

The Dome Master converts the master file and exports it as a single image, image sequence or video based on your settings. The following lists show the supported output formats.

AVI as video format
BMP 24/32bit, PNG 24/32bit, DDS RGB(A) (snp), JPG as image formats

9.3.1 Single Image Conversion

This chapter explains how to convert single images with the Dome Master. Please see the introductory chapter [2055] for general information about this tool and links to other topics.



1. Load File

First, choose your input file in the top section. Make sure "Single Image" is selected and click on the […] button to browse to your file's location
Alternatively, paste the path manually into the text field and press Enter or click the refresh button

The loaded file is now displayed on the left side in the Image section, together with information regarding: Size, Color Depth and Aspect Ratio.
The image size in the example above is 1080 x 1080 px, its color depth is 24 bit (containing RGB, no Alpha) and the aspect ratio is 1 which is common for dome master files.

2. Settings

Second, choose your conversion settings. The default settings mean that the image is converted to the file format "PNG 32bit" which contains an alpha channel and that the processed file will have a 4:1 aspect ratio.

Format
You can choose these formats: BMP 24/32bit (snp), PNG 24/32bit, DDS RGB(A) (snp) or JPG as image formats.

The formats labeled "24bit" or "RGB" do not support an alpha / transparency channel whilst the versions with "32bit" or "RGBA" do support it.
See the chapter "Image Formats [93]" for information about the formats (including snappy) and when to choose which format.

Resize
Without enabling "Resize", the converted image will have a 4:1 aspect ratio whereas the height is half the height of the original image. If your master file has a resolution of 4096 x 4096 px, for example, the result would have 8192 x 2048 px.
You can choose other size options after ticking "Resize".

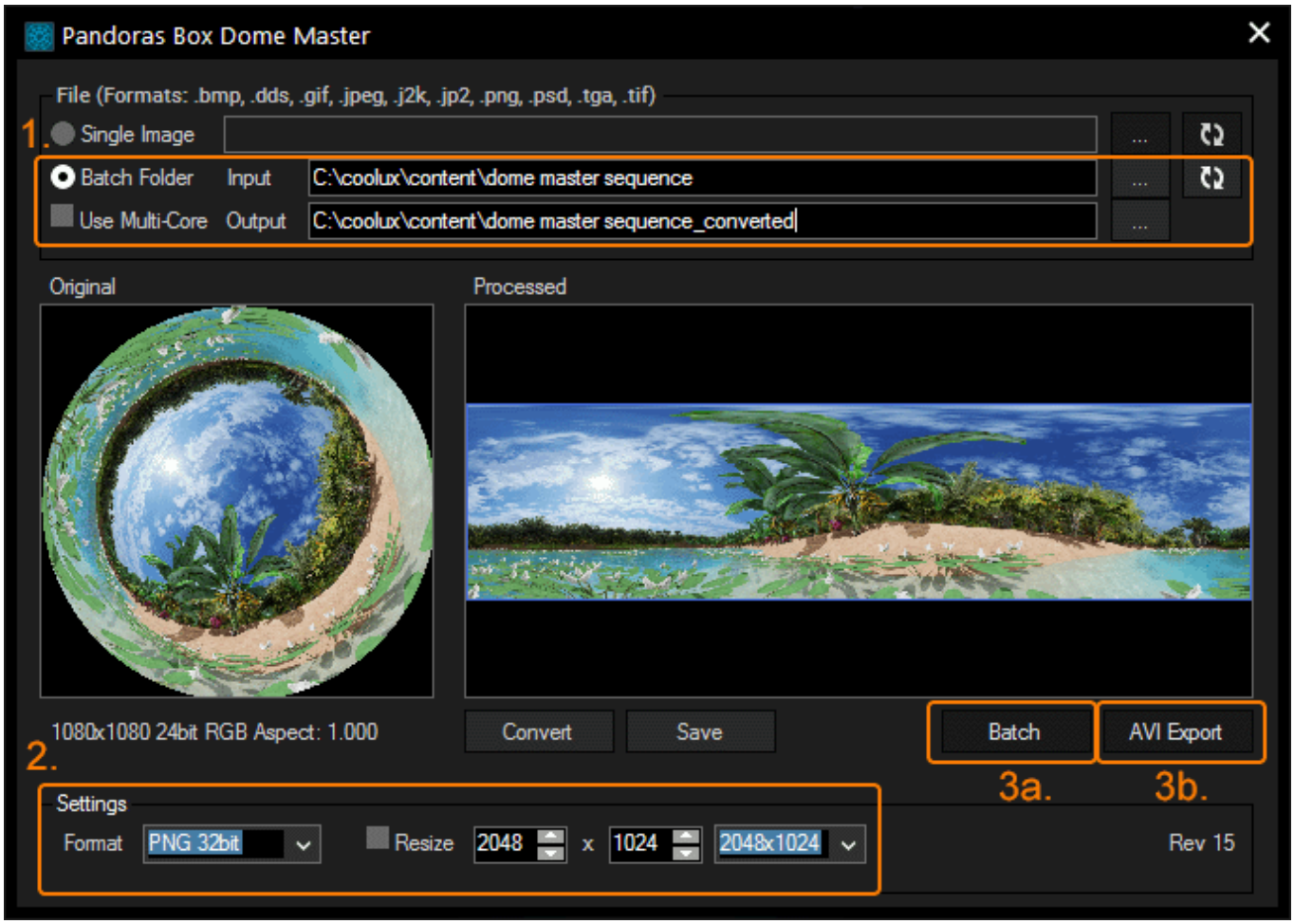
3. Convert and Save

Third and last, click the "Convert" button to see the processes image on the right side and then the "Save..." button to choose where the single image should be saved.

You can now go on and choose the next image or close the Dome Master.

9.3.2 Batch Folder Conversion

This chapter explains how to convert multiple images as a batch conversion with the Dome Master. Please see the introductory chapter [2069] for general information about this tool and links to other topics.



1. Setting up the Input and Output Folder

First, select the option "Batch Folder" and click on the […] button to choose the folder where your Dome Master image sequence is located.
Alternatively, paste the path manually into the text field and press Enter or click the refresh button
Note that the folder should be updated with the refresh button if the image sequence has been changed, e.g. an image frame was replaced, deleted or added.

Next, define an output folder, where the converted files should be stored.

Use Multi-Core
Optionally, you can enable "Use Multi-Core". This splits the conversion process evenly over all available cores of your computer which results in a faster conversion.

The first file of the first folder is now displayed on the left side in the Image section, together with information regarding: Size, Color Depth, Aspect Ratio and total file count.
The image size in the example above is 1080 x 1080 px, its color depth is 24 bit (containing RGB, no Alpha), the aspect ratio is 1 (which is common for dome master files) and the chosen folder includes 160 files.

2. Settings

Second, choose your conversion settings. The default settings mean that the image is converted to the file format "PNG 32bit" which contains an alpha channel and that the processed file will have a 4:1 aspect ratio.

Format
You can choose these formats: BMP 24/32bit (snp), PNG 24/32bit, DDS RGB(A) (snp) or JPG as image formats.

The formats labeled "24bit" or "RGB" do not support an alpha / transparency channel whilst the versions with "32bit" or "RGBA" do support it.
See the chapter "Image Formats [93]" for information about the formats (including snappy) and when to choose which format.

Resize
Without enabling "Resize", the converted image will have a 4:1 aspect ratio whereas the height is half the height of the original image. If your master file has a resolution of 4096 x 4096 px, for example, the result would have 8192 x 2048 px.
You can choose other size options after ticking "Resize".Second, choose your conversion settings. They are explained in detail in the chapter Settings [2075]. The default settings mean that the image is converted to the file format "PNG 32bit" which contains an alpha channel. If you have not chosen another size option, the converted image will have a 4:1 aspect ratio whereas the height is half the height of the original image. If your master file has a resolution of 4096 x 4096 px, for example, the result would have 8192 x 2048 px.

Third and last, choose how the converted files should be saved. Optionally, you can use the "Convert" button to convert the left image and the "Save..." button to choose whereto the (one!) converted file should be saved.

Now you have the option to convert all input images as single images to the target path using the "Batch" button, or convert them to an AVI video file.

3a. Batch

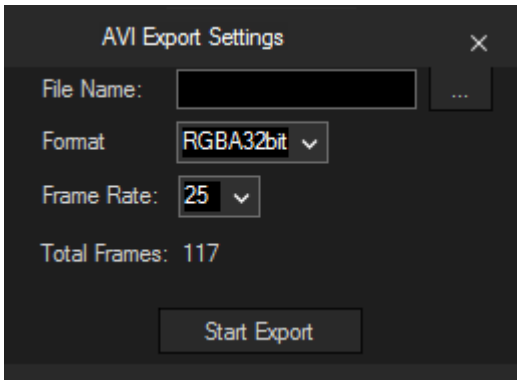
"Batch" will start the batch conversion according to the Format and Resize Settings and save the results to the target path chosen above.

While the batch conversion process is running, the message "Batching files …" is displayed in the image fields.

To skip the batch conversion process press "Stop".

3b. AVI Export

Choose "AVI Export" if you like to create a video out of all files located in the Input folder.



File Name
Choose a file name for your AVI video and use […] to browse to the location where it should be stored.

Format
Choose the format: RGBA32bit, RGB24bit, RGB555, RGB565, YUY2(4:2:2), UYVY(4:2:2) or IYUV(4:2:0).

Frame Rate
Choose the frame rate in FPS (frames per second). Possible values are 24, 25, 30, 50 and 60.

Total Frames
Displays the total frame number.

Now press the "Start Export" button and your AVI video will be created.

9.4 Factory Reset - Acronis True Image

We provide a "factory reset" disk image with the Software "Acronis" that represents the system's condition when it has been shipped out with all recommended settings and drivers!

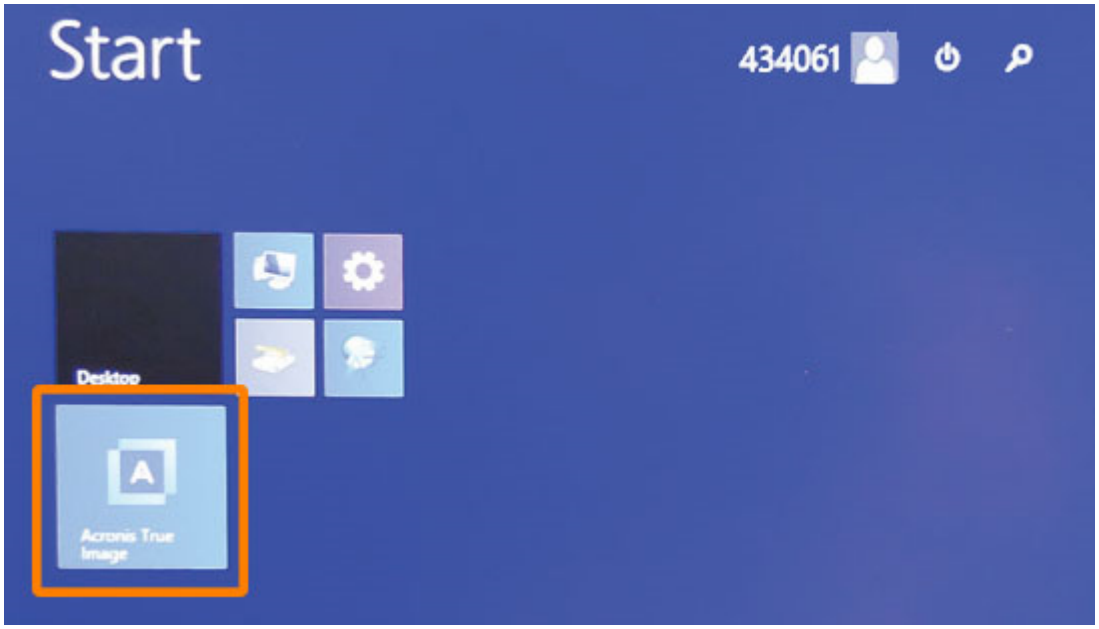
The software "Acronis True Image 2017 Home (ATH)" is installed on all Server R4 and Player R4 hardware. On the one hand it allows you to create disk images i.e. a backup of the system including all files, programs, driver versions, computer names and all other structures and settings. On the other hand it allows you to restore the system by loading created images, either ours or custom ones, which is especially interesting for rental machines. Please read the Acronis manual if you are interested in these things.

This manual focuses on the explanations:
- Creating a USB Boot Stick
- Loading a Recovery Image

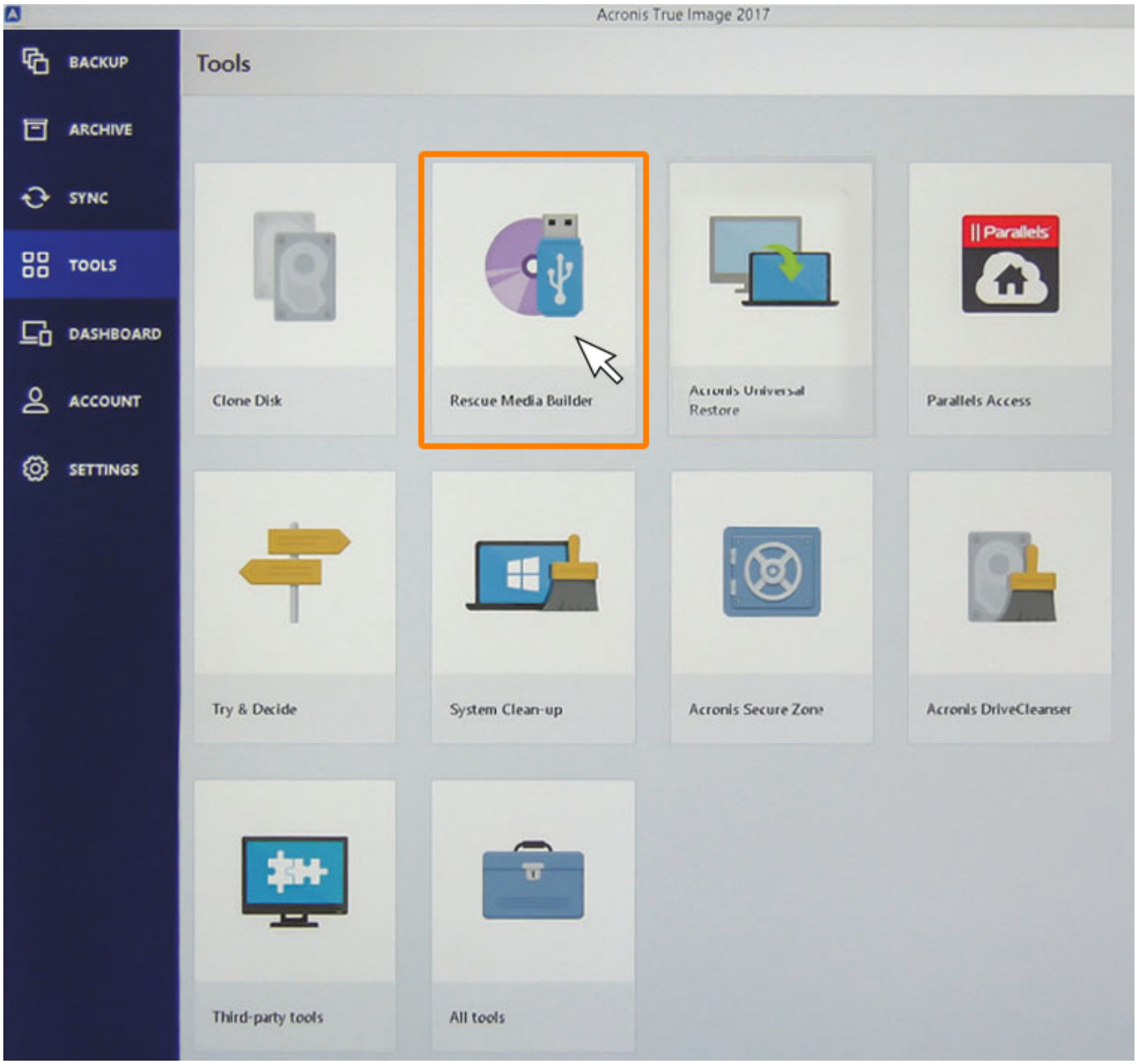
Creating a USB Boot Stick

A USB Boot Stick lets you boot your computer in case Windows is damaged so much, that it is not able to boot the computer itself. It is always a good idea to have a USB boot stick handy, in case something happens to your system. You can also create a USB stick from another computer (and the Acronis software!) and use it for the damaged one.
To create one, you need an UBS Stick (USB 1,2 or3) with at least 1GB of storage room. When you plug it in later, it will be formatted automatically.

To start "Acronis" click on the according start tile or search for "Acronis".

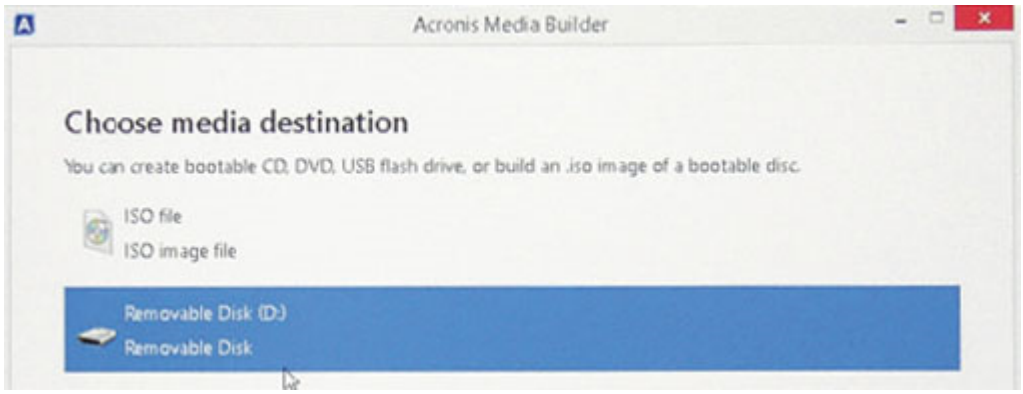


This is the Acronic surface. In the "Tools" menu, click on "Rescue Media Builder".



Before you start the process, the stick needs to be plugged into the device!
Choose "Acronis Bootable Rescue Media".

Chose "Removable Disk" as a media source.

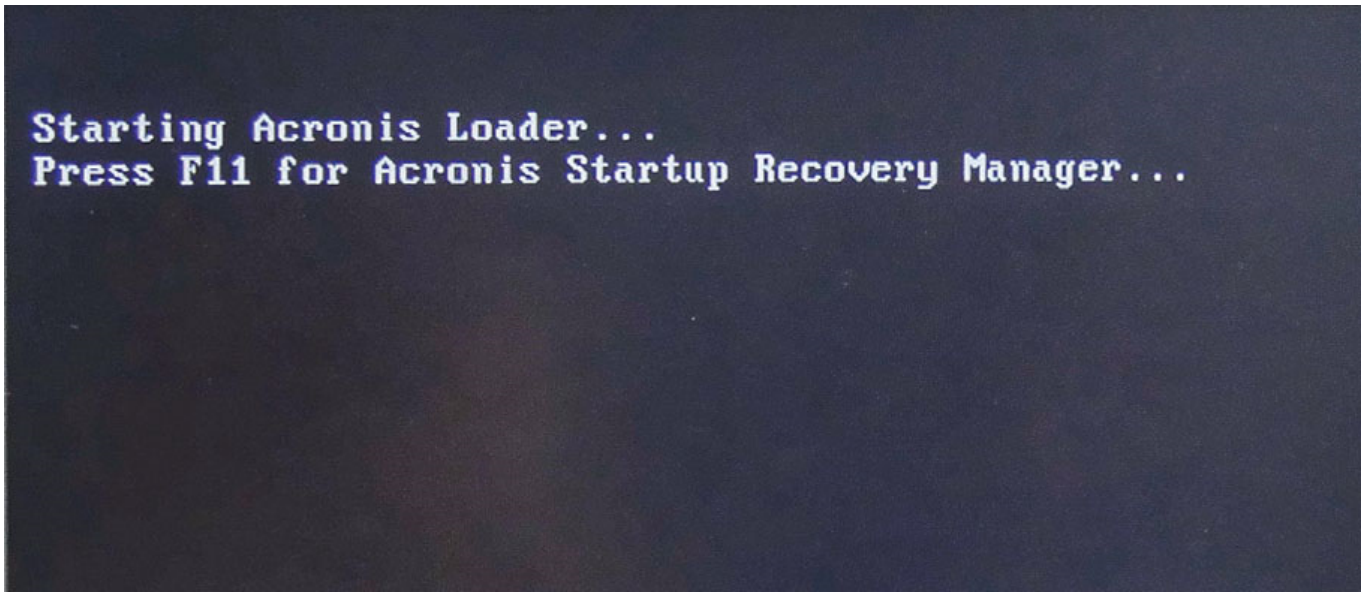


Press "Proceed" and finish the process with the "Close" button. Now you have a boot stick.

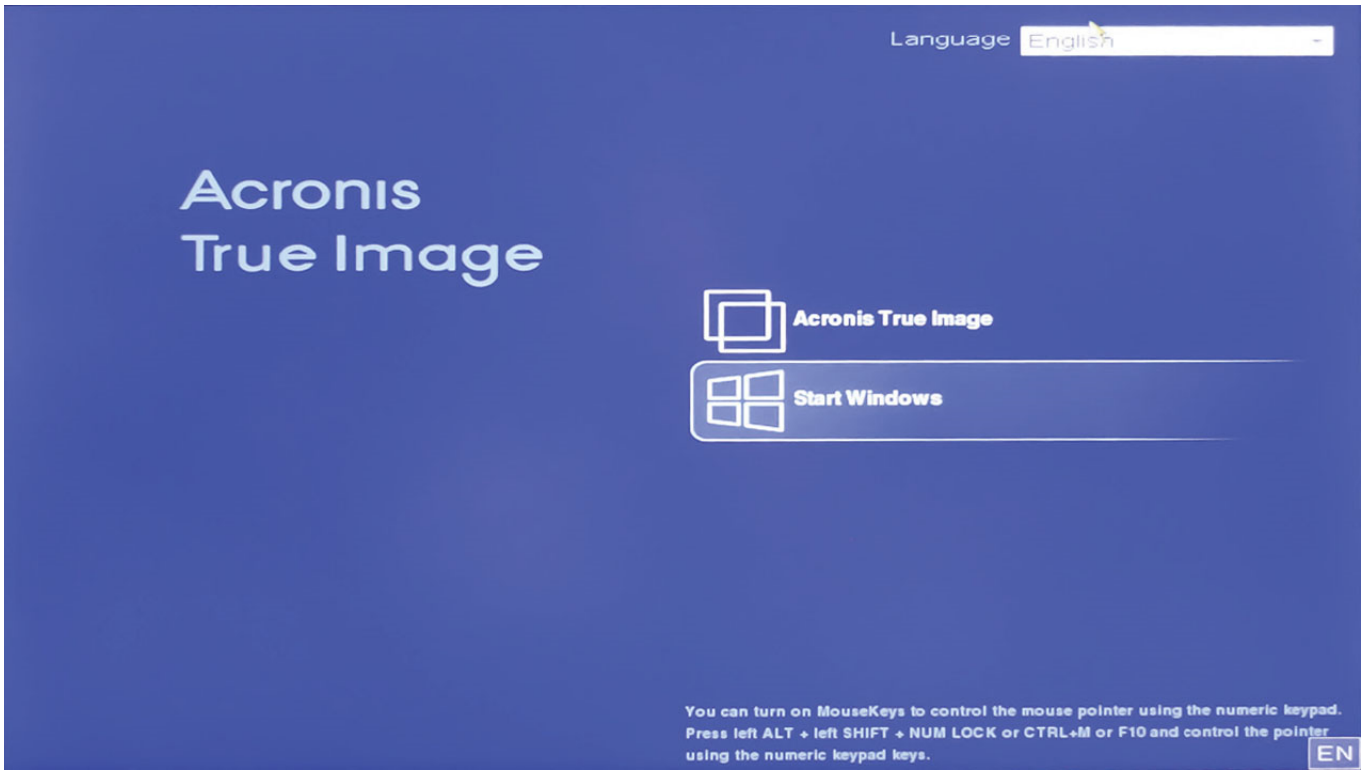
Loading a Recovery Image

You can either start the recovery from a system that still boots by itself (a) or with a USB boot stick (b).

a1) If your computer is still able to boot, start it and wait until you see the following screen. Then press [F11] immediately. If you see a window "Enter Linux kernel command line" you can simply click on "OK".

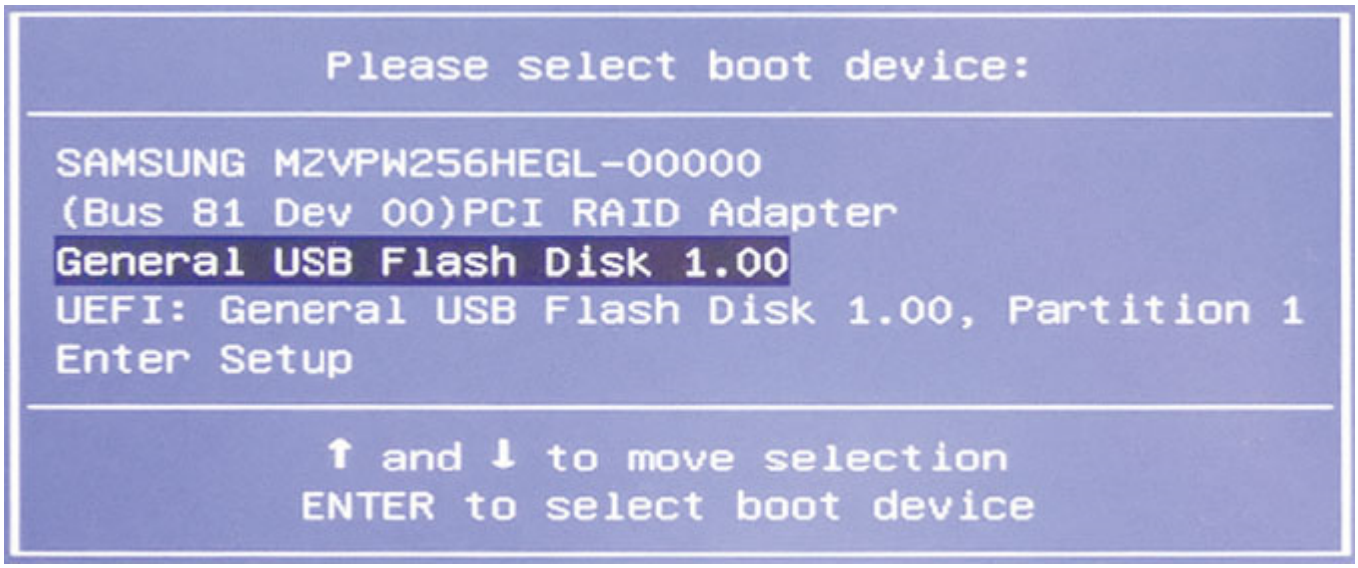


a2) Click on "Acronis True Image" and afterwards on "Recover" in the menu.

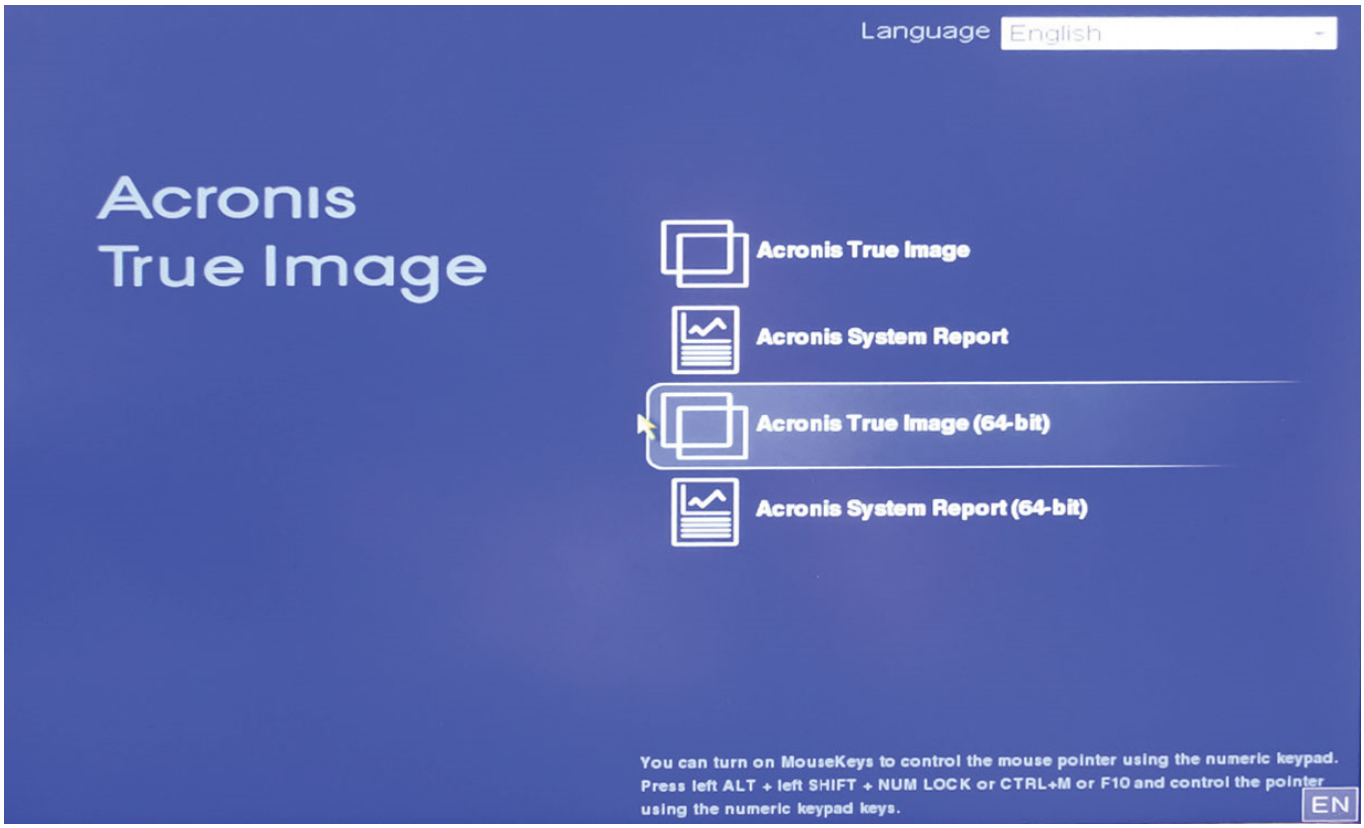


b1) If your system does not boot by itself, insert the USB boot stick and start the system. Then press [F8] instead of [F11].

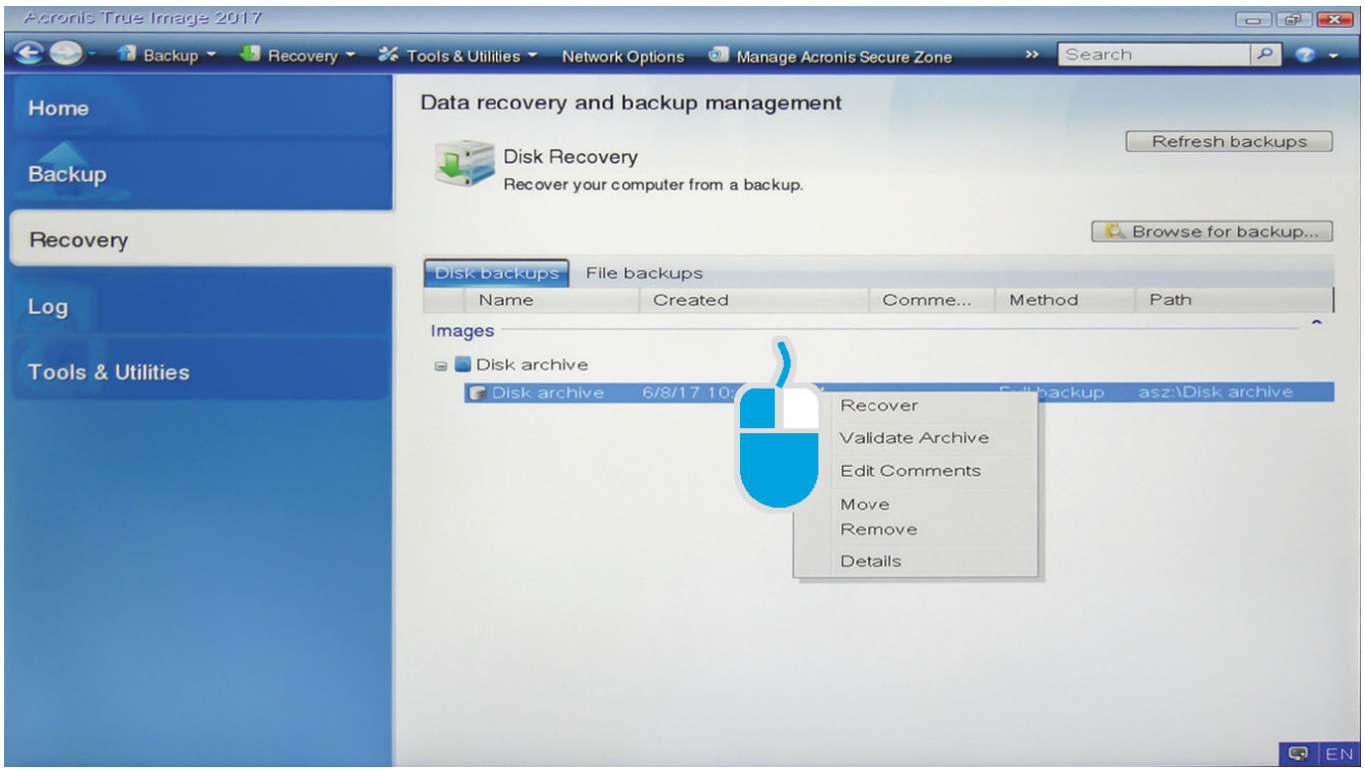
b2) In the upcoming menu, press: "General USB Flash Disk".



b3) Click on "Acronis True Image (64-bit)".

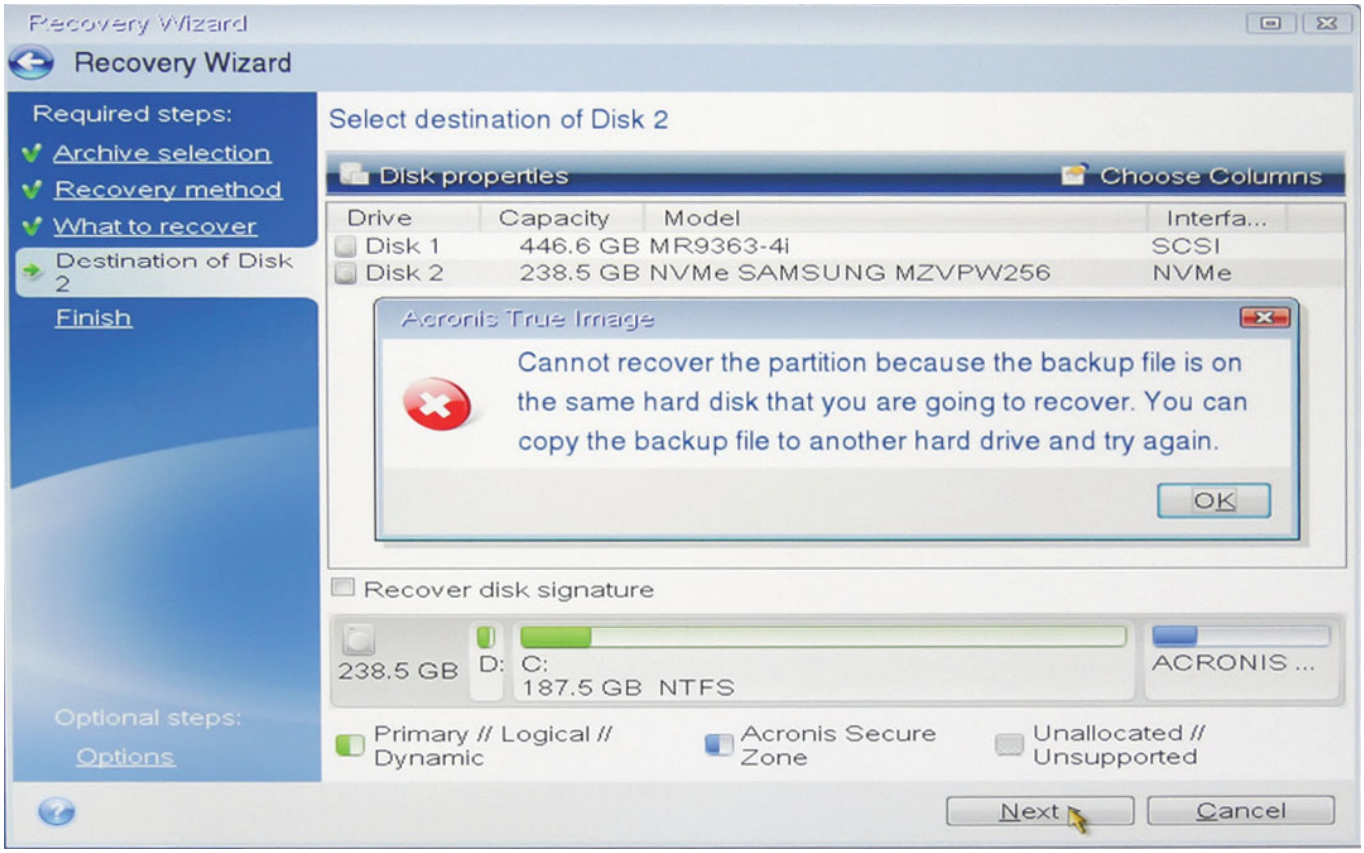


In both cases you see the following menu. Press "Recovery" and right-click (!) on "Disk archive" to get the "Recovery" menu!

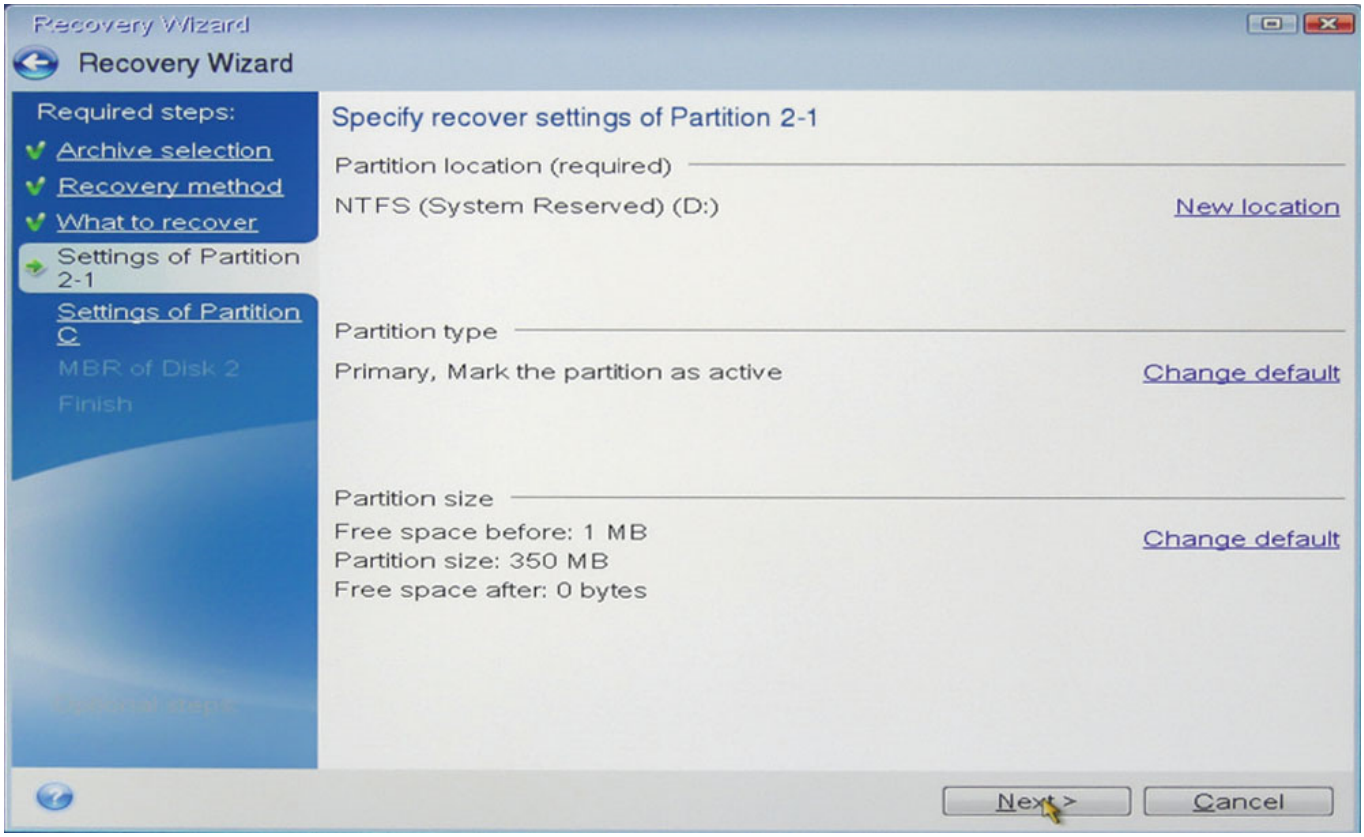


Check "Recover whole disks and partitions" and press "Next".

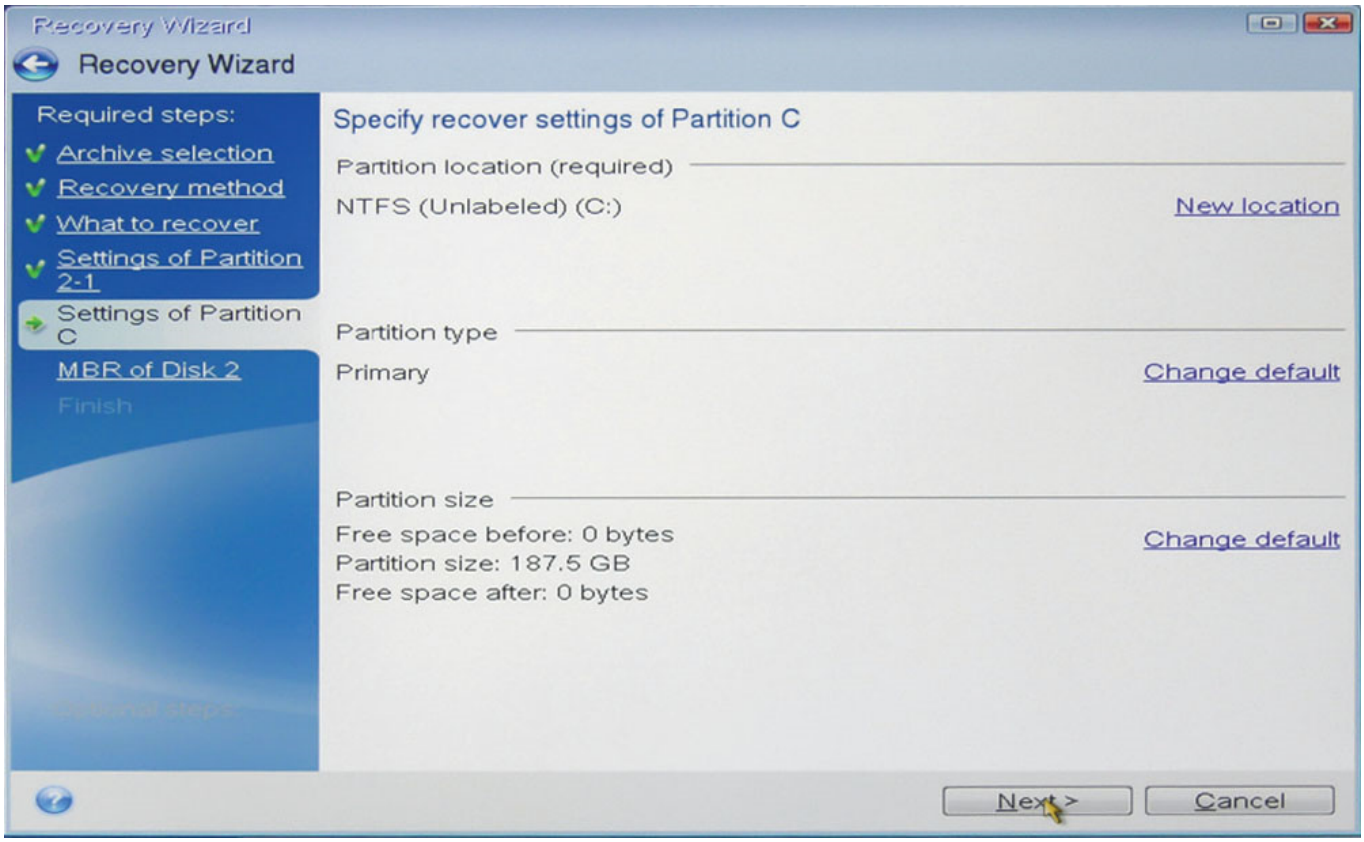
In the section "What to recover" you click on check box "Disk 2" and uncheck the check box "Acronis Secure Zone" otherwise you will get the following error:



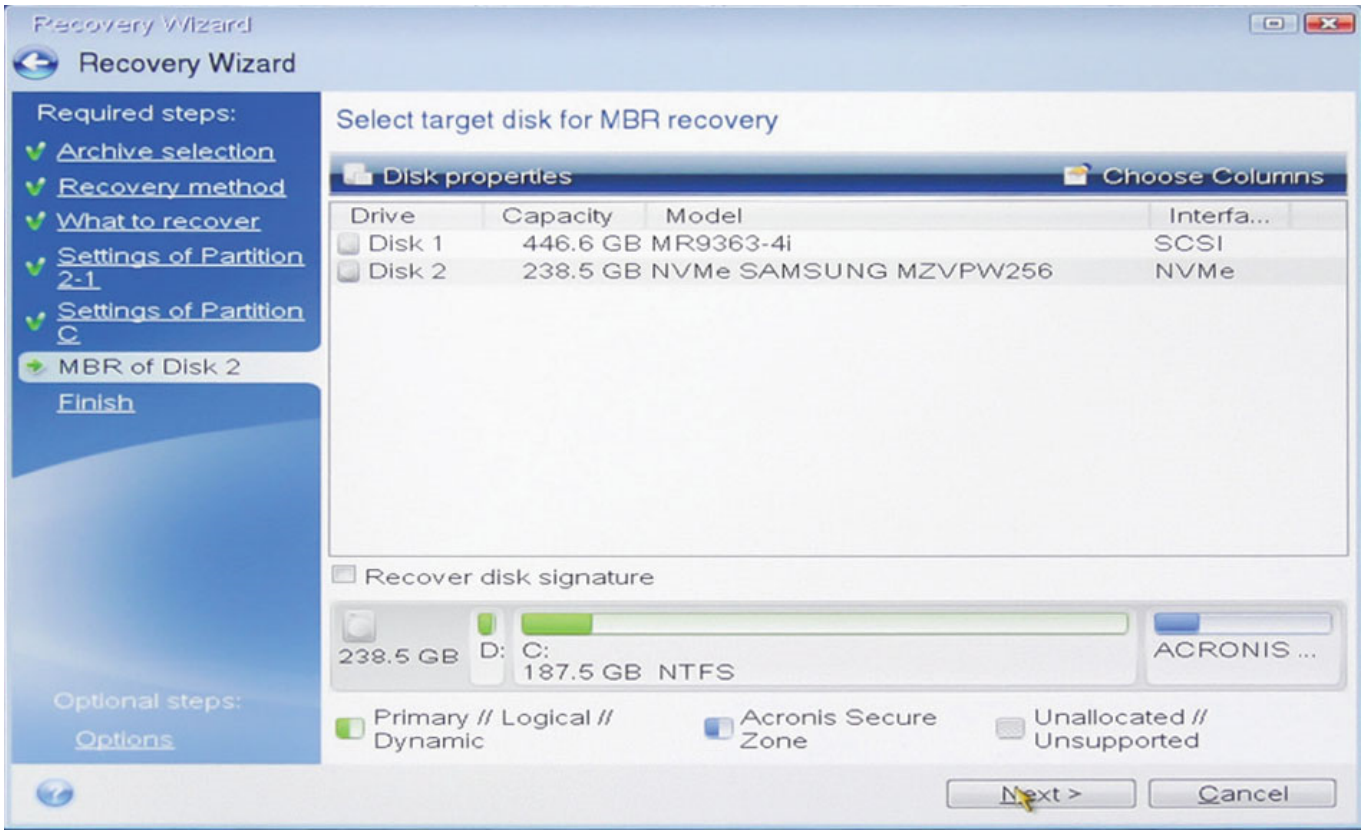
Now, you just click "Next"...



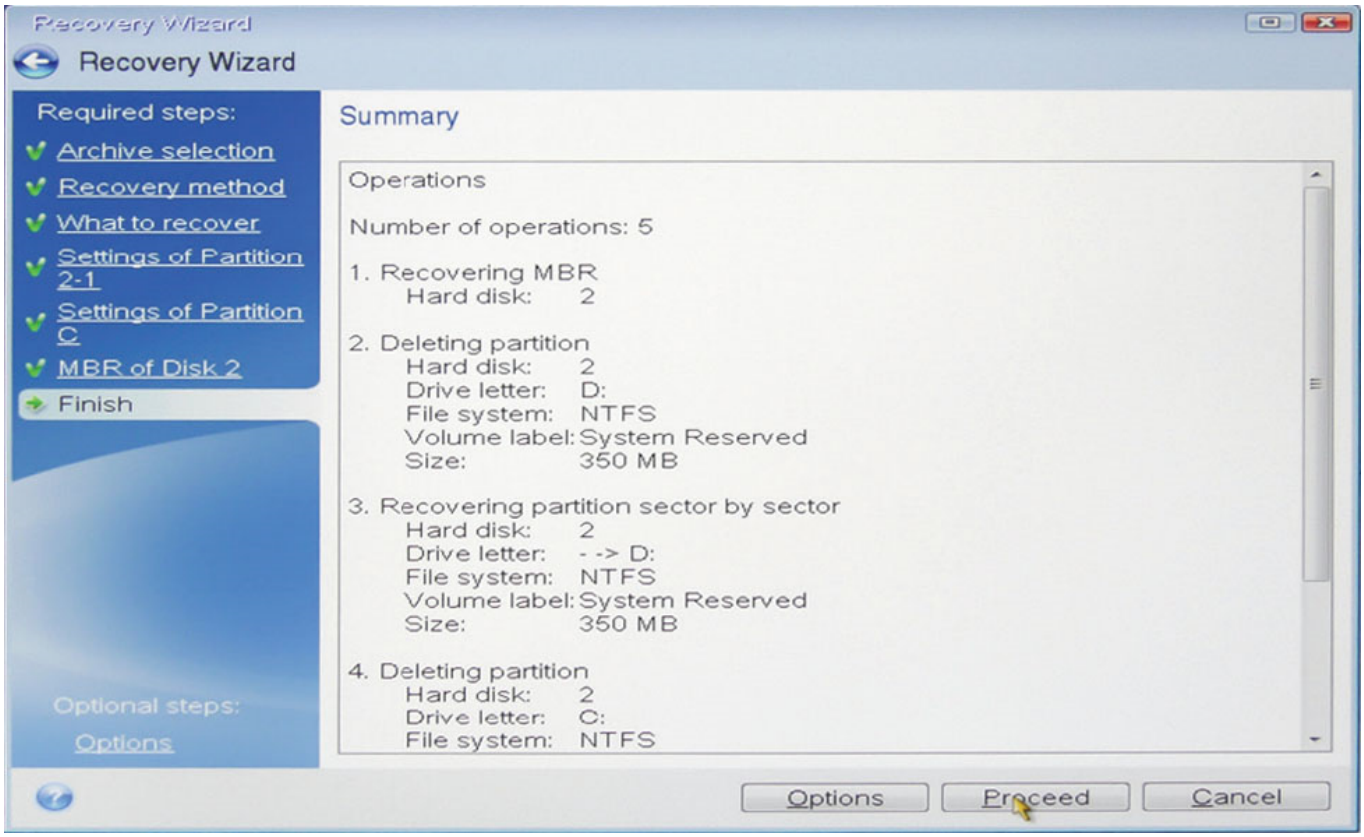
..and "Next"...



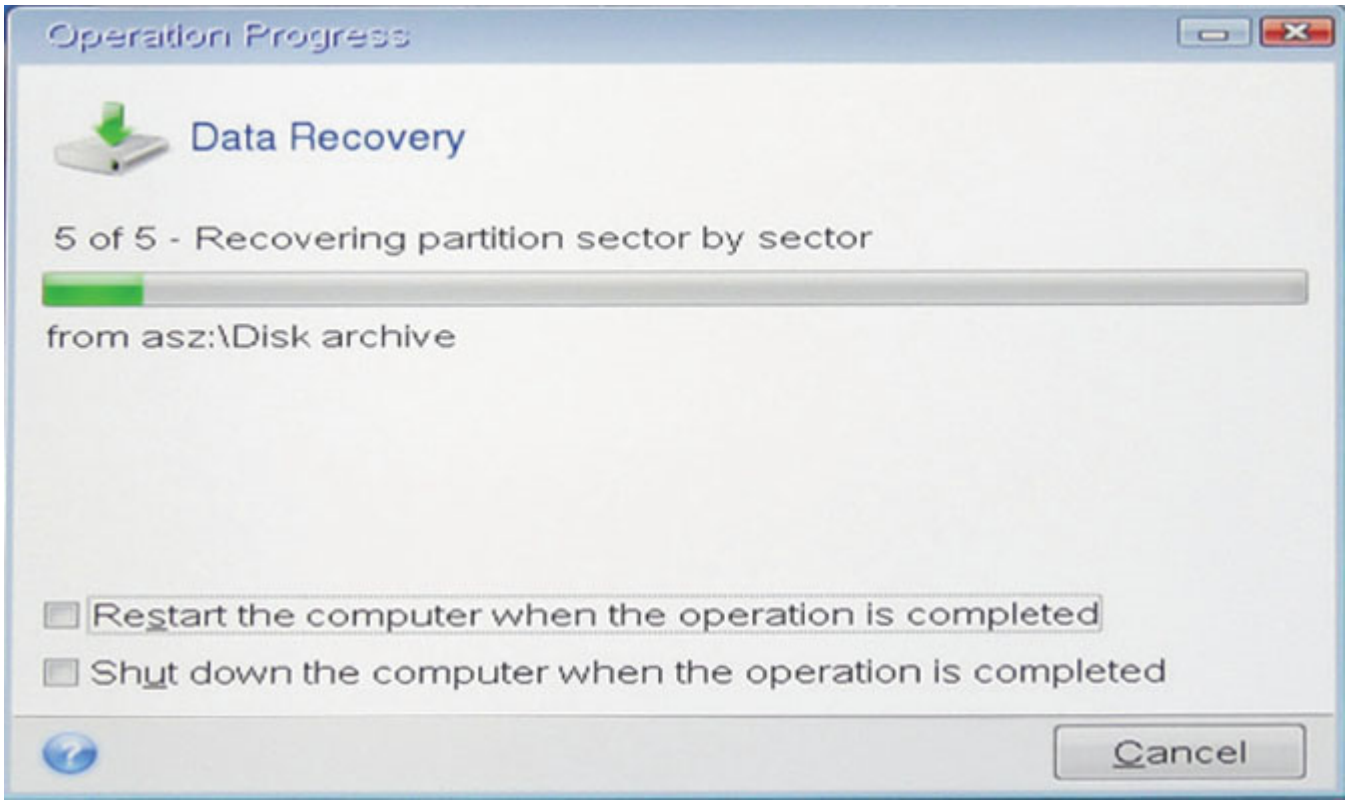
... and "Next" again.



Now click "Proceed" to finish it off.



The last action is to click on the check box "Restart the computer when the operation is complete"



9.5 Image Converter

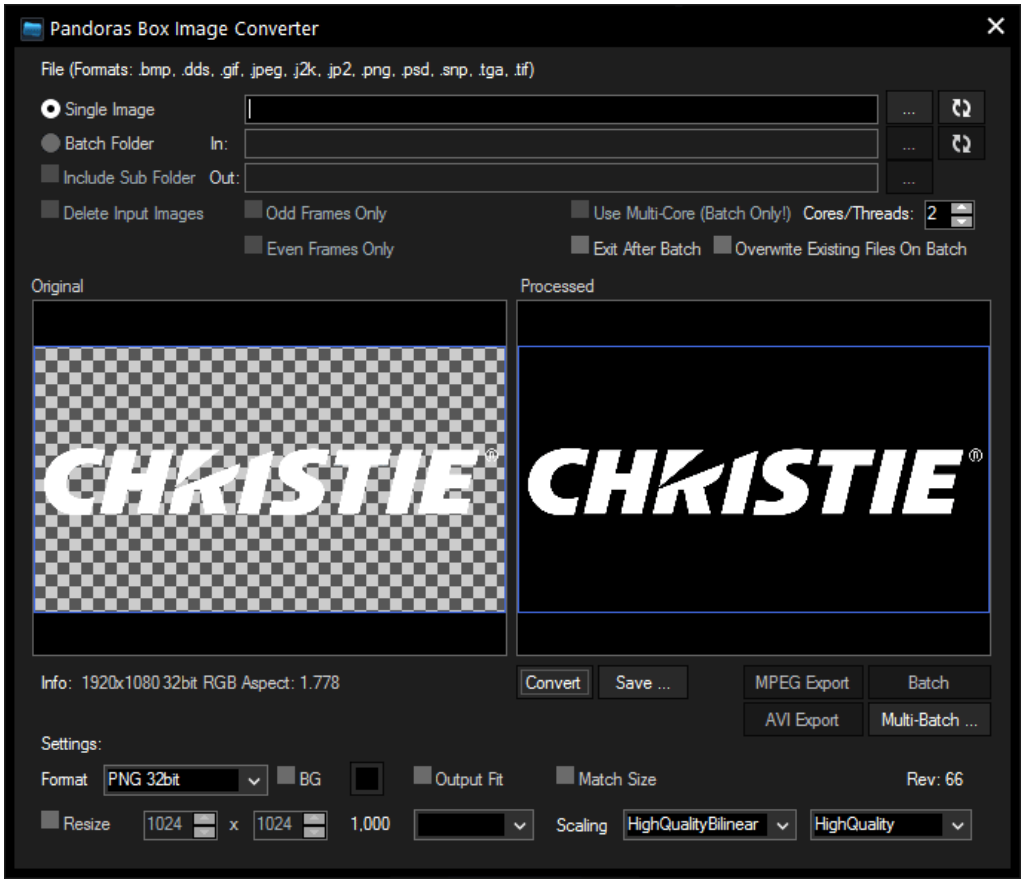
The Image Converter is a tool for content preparation and modification, and can be used to increase the performance of the PB Playback system. The tool is installed automatically when you install a Pandoras Box application. To open it, you can either choose it from the Tools menu [122] from your Pandoras Box master software or open it stand-alone (on the Master or Client), e.g. by choosing it from the Pandoras Box folder in the Windows Start menu. The installation path is for example: `C:\Program Files\Christie\Pandoras Box 8.0.0\data\tools\PandorasBoxImageConverter.exe`

The Image Converter allows to process either single images [2070] or - as it supports batch processing -entire image sequences [95]. It resizes images to smaller file sizes or it converts the format. In addition it may export an image sequence as a MPEG or an AVI video. You may choose to read the frames forwards or backwards.
Since version 5.1 it is possible to extract an additional black and white movie to be used as a mask for keying in Pandoras Box. This is the most recommended workflow when an alpha (transparent) channel is needed in videos!

The possible input formats are: BMP, DDS, DPX, GIF, JPEG, J2K, JP2, PNG, PSD, SNP, TGA, TIF(F)
All formats can be imported when they are saved with a color depth of 8bit per channel with the following exception: PNG and TIF(F) files can provide up to 16bit/channel and DPX can provide 8, 10 or 12bit.

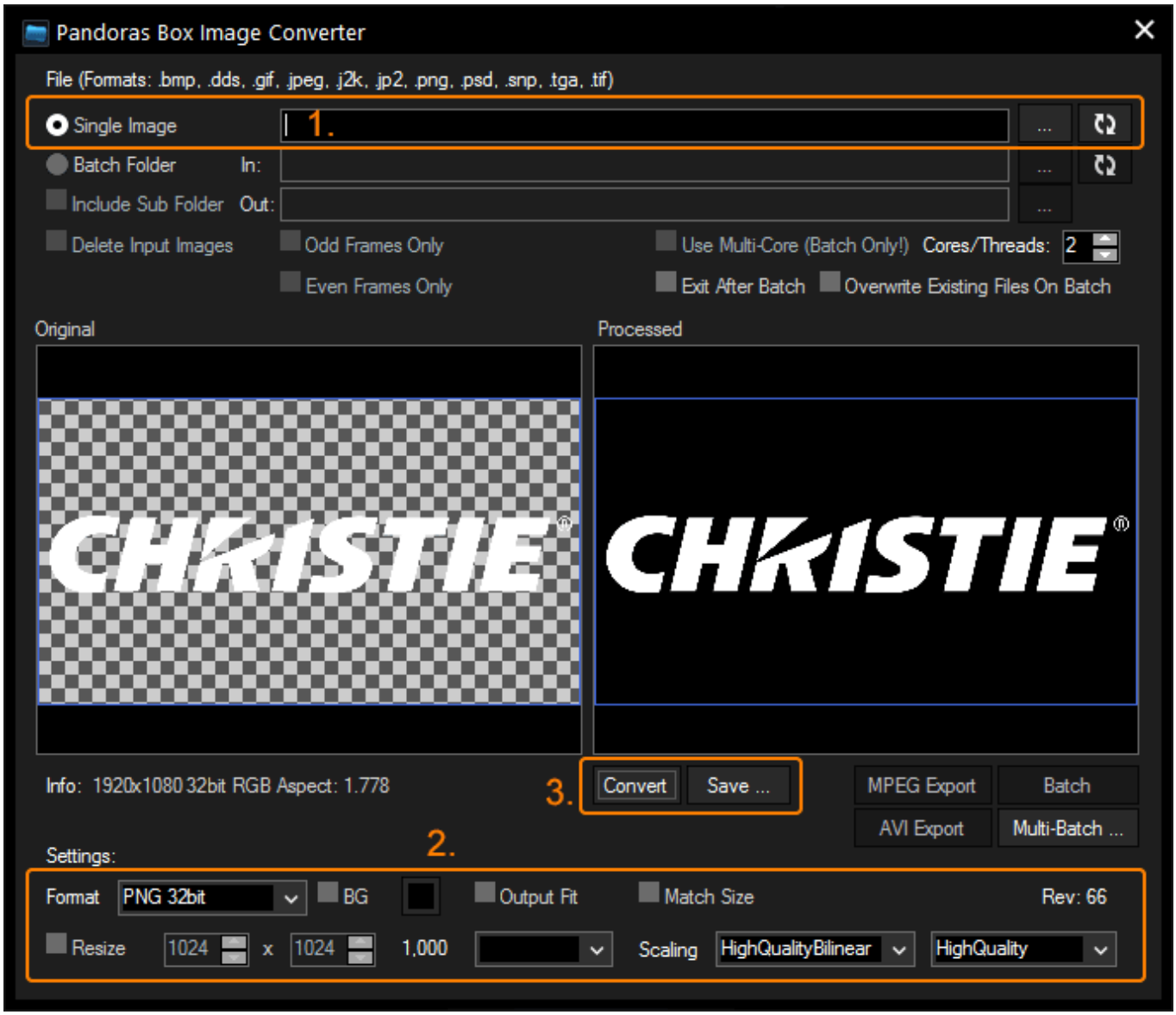
When you work with large, high resolution multi-softedged projections, you might be interested in a tool that splits images in order to have separate content per Client. For this, have a look at the Splitter [2120] tool.

The following pages explain the user interface:
- Single Image Conversion [2070]
- Batch Folder Conversion [2071]
- Image Converter Settings



9.5.1 Single Image Conversion

This chapter explains how to convert single images with the Image Converter. Please see the introductory chapter [2069] for general information about the Image Converter and links to other topics.



First, choose your input file in the top section. Make sure "Single Image" is selected and click on the […] button to browse to your file's location
Alternatively, paste the path manually into the text field and press Enter or click the refresh button

The loaded file is now displayed on the left side in the Image section, together with information regarding: Size, Color Depth and Aspect Ratio.
The image size in the example above is 1920x1080 px, its color depth is 32 bit (containing RGB + Alpha) and the aspect ratio is 1,778 (=16/9).

Second, choose your conversion settings. They are explained in detail in the chapter Settings [2075]. The default settings mean that the image is not resized but only converted to the file format "PNG 32bit" which contains an alpha channel.

Third and last, click the "Convert" button and then the "Save..." button to choose whereto the converted file should be saved.

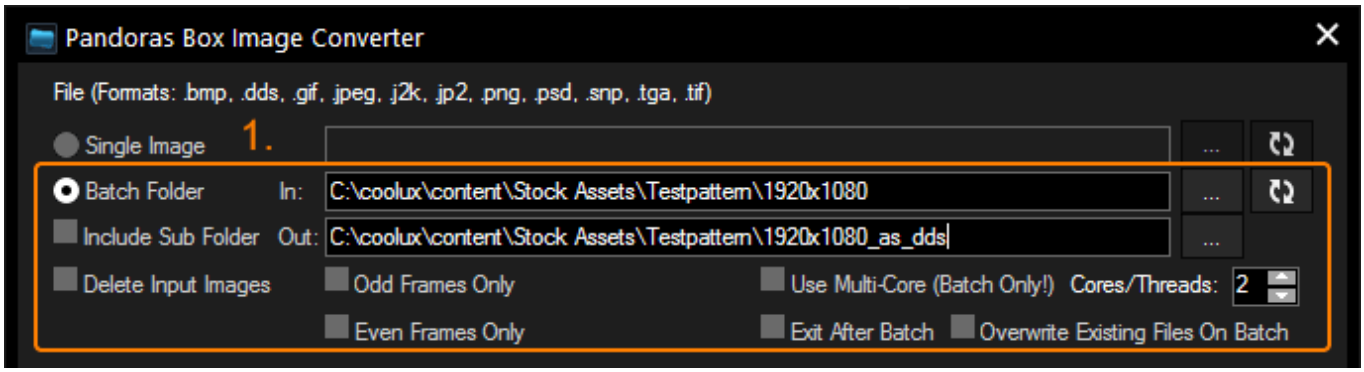
You can now go on and choose the next image or close the Image Converter.

9.5.2 Batch Folder Conversion

This chapter explains how to convert multiple images as a batch conversion with the Image Converter. Please see the introductory chapter [2069] for general information about the Image Converter and links to other topics.

Setting up the Input and Output Folder

The top section allows to choose your input and output folder and set up the batch conversion in more detail.



First, select the option "Batch Folder" and click on the […] button to browse to your folder's location. Alternatively, paste the path manually into the text field and press Enter or click the refresh button

Note that the folder should be updated with the refresh button if the image sequence has been changed, e.g. an image frame was replaced, deleted or added.

Next, define an output folder whereto the converted files should be stored and tick the following options if needed.

Include Sub Folder
This converts images in sub folders recursively. The existing folder structure on the input side is created also on the output side.

Delete Input Images
Removes the source images after conversion.

Odd / Even Frames Only
Tells the Image Converter to convert only odd or even images. This is useful when getting an image sequence that is produced for 3D-content where the first, third, fifth... frame is for the right eye and the second, forth, sixth... frame contains the information for the left eye. A typical frame rate would be for example 120Hz. With this feature you may separate the right and left content and render two videos to be played back with 60Hz.

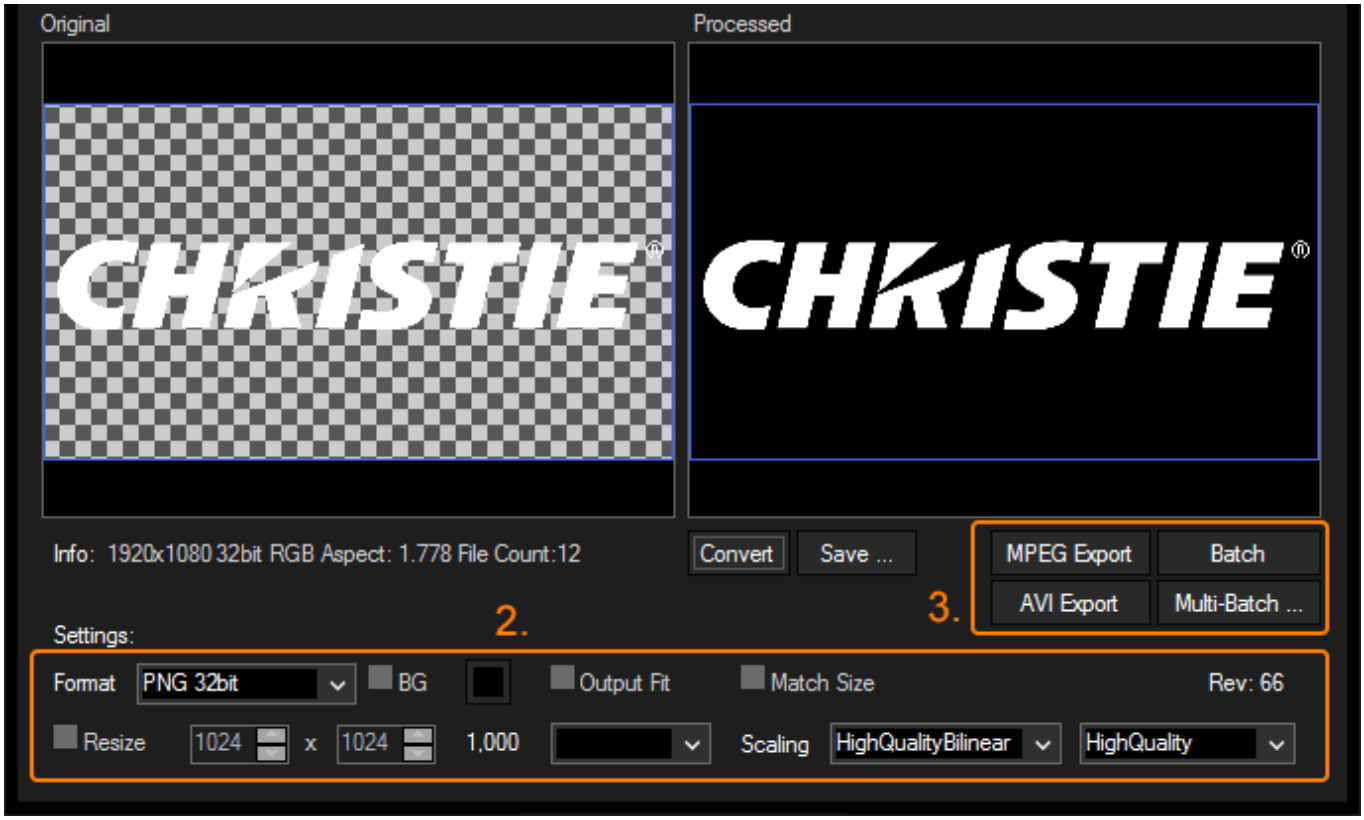
Use Multi-Core
Enables the Image Converter to use multiple processor cores. This uses more system performance but is accordingly faster. With the number field "Cores / Threads" you can set up how many cores should be used.

Exit After Batch
The Image Converter window will be closed as soon as the batch conversion is completed.

Overwrite Existing Files on Batch
Overwrites the existing files in the current output folder during the conversion.

Saving Converted Images

The first file of the first folder is now displayed on the left side in the Image section, together with information regarding: Size, Color Depth, Aspect Ratio and total file count.
The image size in the example above is 1920x1080 px, its color depth is 32 bit (containing RGB + Alpha), the aspect ratio is 1,778 (=16/9), and the chosen folder includes 12 files.



Second, choose your conversion settings. They are explained in detail in the chapter Settings [2075]. The default settings mean that the image is not resized but only converted to the file format "PNG 32bit" which contains an alpha channel.

Third and last, choose how the converted files should be saved. Optionally, you can use the "Convert" button to convert the left image and the "Save..." button to choose whereto the (one!) converted file should be saved.

Now you have the option to convert all input images as single images to the target path using the "Batch" or "Multi-Batch" button, or convert them to an AVI or MPEG video file.

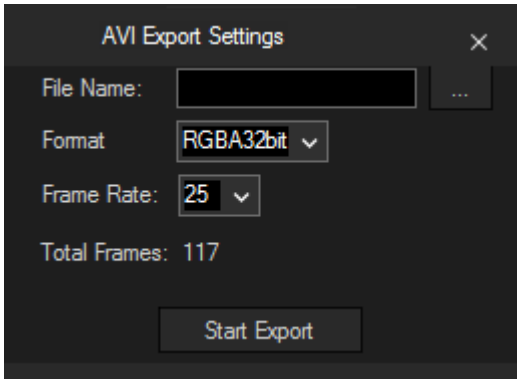
Batch and Multi-Batch

Both buttons will start the batch conversion according to the below Settings [2075] and save the results to the target path chosen above.
The difference is that "Multi-Batch" shows a dialog first where you can select the number of instances which will be used for the conversion. If you select "8", you will see that indeed eight Image Converter windows will take care of the batch conversion. This option is recommended if you are converting images with a high resolution (4K and above) because this quickly reached the RAM limit of one instance under Windows.

While the batch conversion process is running, the message "Batching files …" is displayed in the image fields.

To skip the batch conversion process press "Stop". To pause the batch conversion process press "Pause" and press "Play" to continue the process.

AVI Export



File Name
Choose a File Name for your AVI video and use […] to browse to the location where it should be stored.

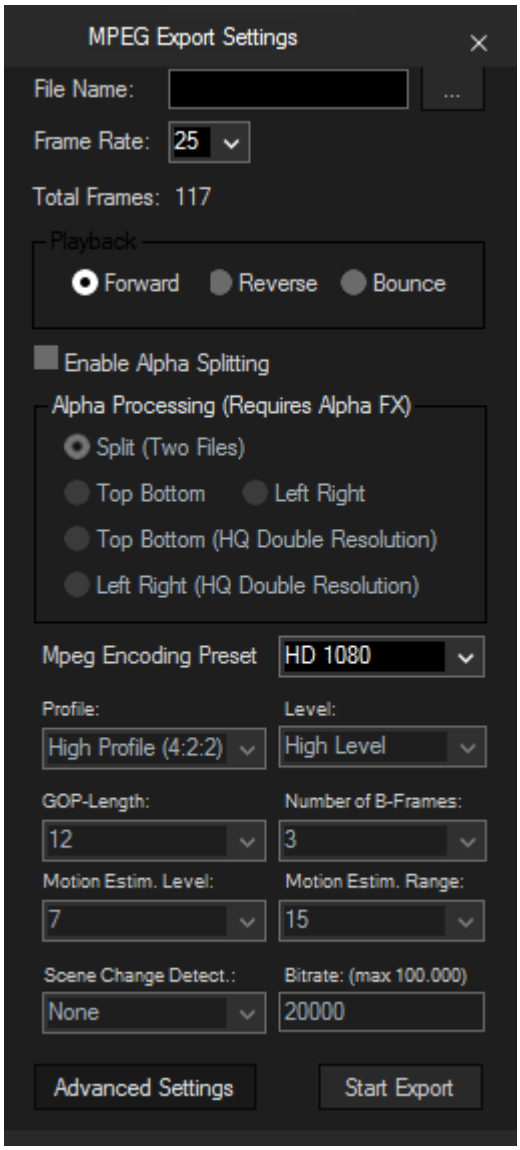
Format
Choose the format: RGBA32bit, RGB24bit, RGB555, RGB565, YUY(4:2:2), UYVY(4:2:2) or IYUV(4:2:0).

Frame Rate
Choose the framerate in FPS (frames per second). Possible values are 23.976, 24, 25, 29, 29.97, 30, 50, 50.94, 60, 100 and 120.

Total Frames
Displays the total frame number.

Now press the "Start Export" button and your AVI video will be created.

MPEG Export



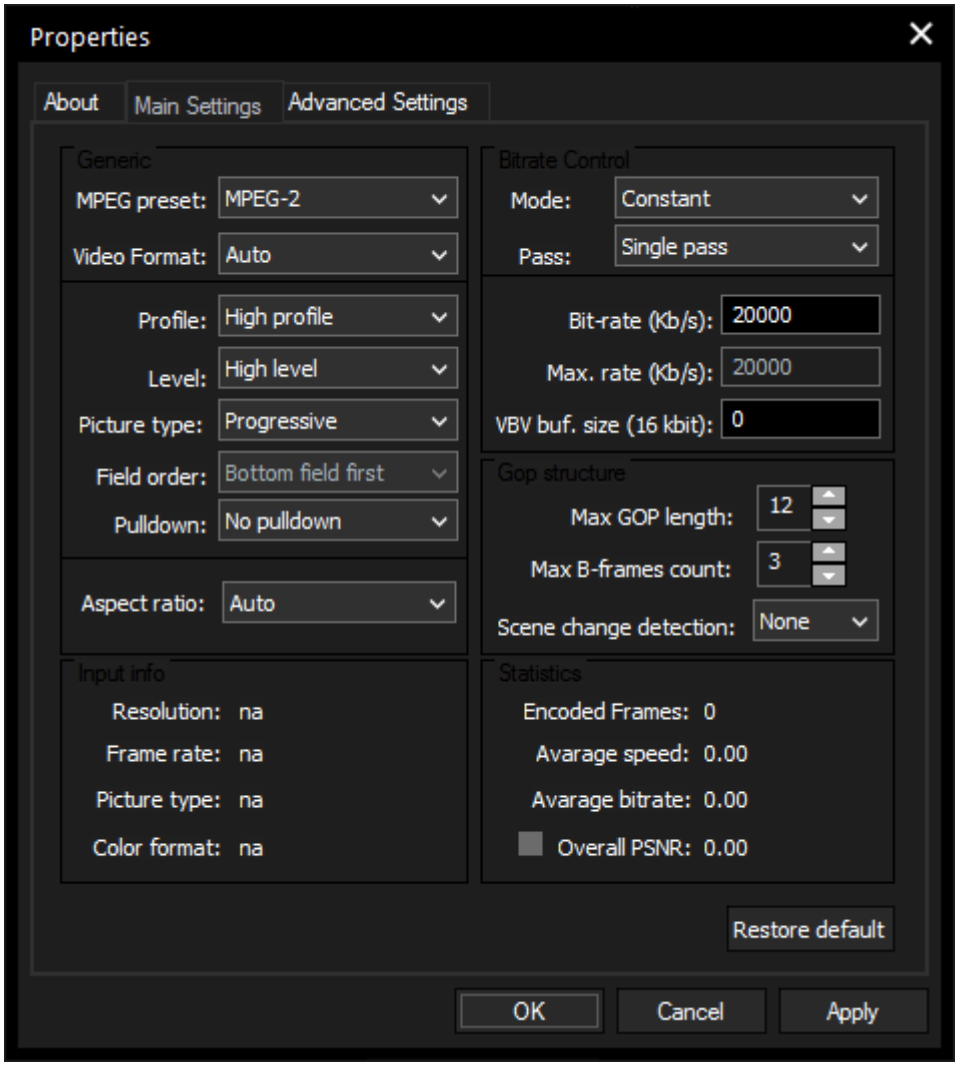
The MPEG export feature allows you to render a sequence of images to a MPEG file.
Enter a file name and click the [...] button for determining the entire file path.
Choose the appropriate frame rate and the playback direction. "Bounce" will play all images forward and then once again in reversed order.

The alpha splitting section gives you the possibility to maintain the alpha channel when converting to the MPEG format. The MPEG format was originally specified as a codec not supporting the alpha channel. Therefore the Image Converter enables the following workflow: First the main mpg video that naturally supports RGB only is generated, then an additional MPEG is saved which holds the original transparency information in terms of a black and white mask. In Pandoras Box dedicated effects merge the main video and the mask video resulting in one video with transparency.

Split	Splits color and alpha information into two separate tracks. In Pandoras Box, use the effect Alpha Media as Mask [546]
Top Bottom / Left Right	The color information is rendered to the top/left half of the video. The other half contains the alpha information. Please note that this option lowers the image quality but saves playback performance. In Pandoras Box, use the effect Alpha Top Bottom [546], respectively Alpha Left Right
Top Bottom / Left Right HQ Double Resolution	The video height/width is doubled. The alpha information is stored as a black&white video attached at the bottom/right of the video. Other than the above option, the quality is preserved. On the other hand the playback takes more processing power!

Choose a "Mpeg Encoding Preset" or define a custom setting. The encoder is based on the Pandoras Box Encoder Extension, please find more detailed descriptions for the encoder settings [107] here, including recommendations regarding the bitrate setting.

Click "Start Export" to confirm all settings. When the process is finished a dialog will pop up.



The advanced settings make in depth configuration of the MPEG encoder possible.

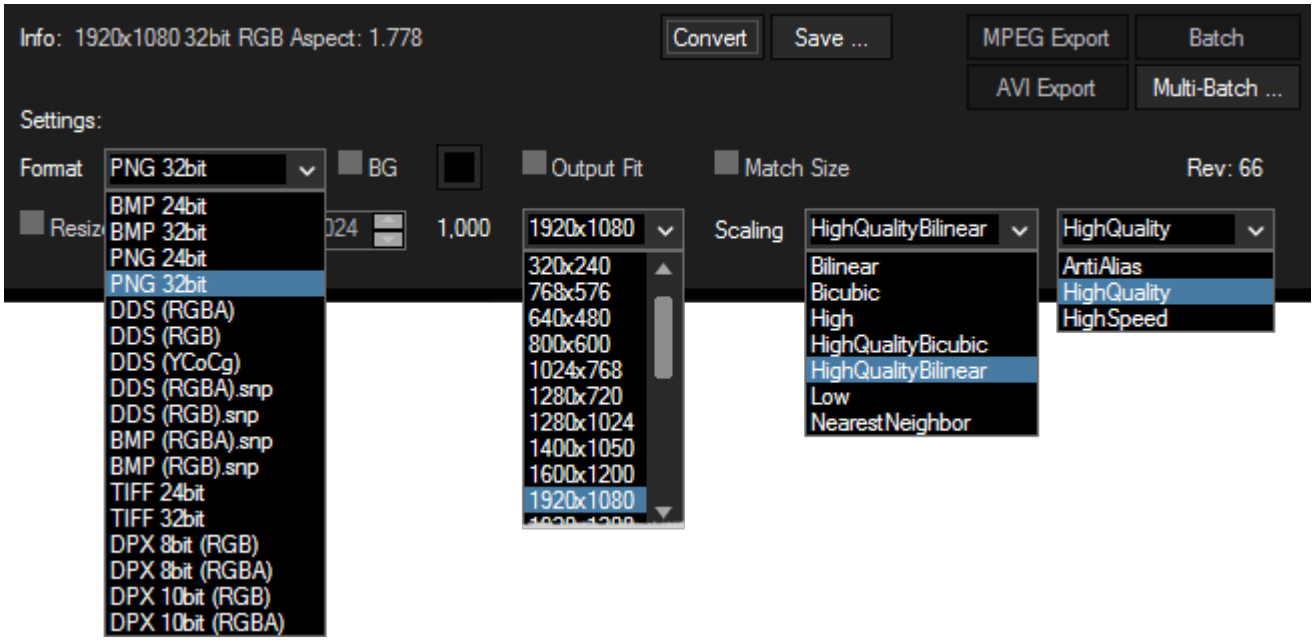
The maximum resolution for an MPG2 video is 4094x2800 px, though it is recommend to set 4080x2800 px as this conforms with the MPEG standard. The encoding process depends on available graphics card RAM.

For further information regarding the encoder settings please see the Encoder Extension[100].

9.5.3 Image Converter Settings

This chapter explains which settings you can choose when converting images with the Image Converter. Please see the introductory chapter [2069] for general information about the Image Converter and links to other topics that explained how to load and save images.

Remember to click the "Convert" button if you change a setting and like to see the result in the image on the right side.



Format

Choose an output image format from the drop-down list : BMP, PNG, DDS, TIFF or DPX (8bit, 10bit). The formats labeled "24bit" or "RGB" do not support an alpha / transparency channel whilst the versions with "32bit" or "RGBA" do support it.
See the chapter "Image Formats [93]" for information about the formats (including snappy) and when to choose which format.

Resize and Background Options

By default this option is disabled which means that converted images keep their original size. If you like to resize images, enable "**Resize**" and either enter new values for the width and height manually in the text fields or choose a setting from the drop-down list. You can also change the scaling mode which is set to "**HighQualityBilinear**" and "**HighQuality**" by default. Note that the aspect ratio from the image(s) is displayed next to the "**Width**" and "**Height**" fields. So if you enter a size of 2000x1000px, the new aspect mode will be 2,0.

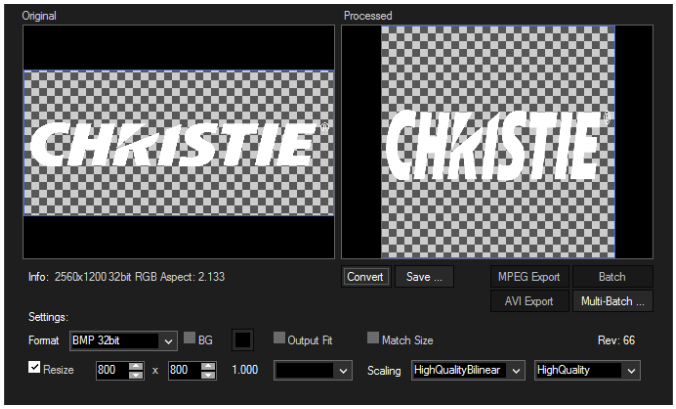
If you like to keep the native aspect without calculating both sizes, enable the check box "**Match Size**". Now enter a "Width" or "Height" and click the "**Convert**" button. A new text appears to the right side and shows which size was automatically calculated, based on the smaller number that was entered in the Width/Height text fields. For example: If the original aspect is 2,0 and the Width/Height fields were set to 1000x200, the calculated size will be 400x200. If the Width/Height fields were set to 200x1000, the calculated size will be 200x100.

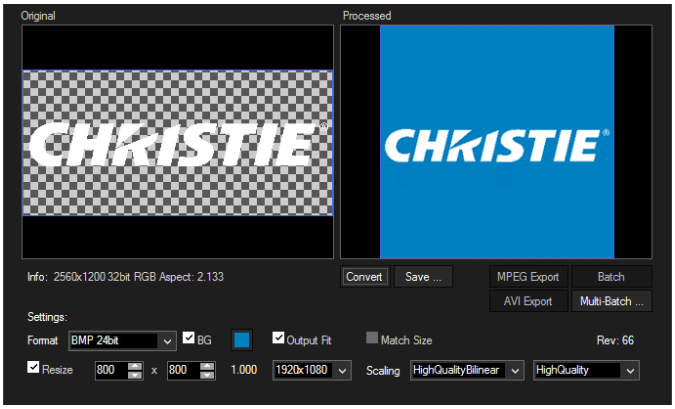
Alternatively you can set the option "Output Fit" which allows to resize an image to a size with a different aspect ratio without stretching the content. The "old" image is placed within the canvas with the "new" size in a way that it fits either the height or width. The rest of the canvas is either transparent or filled with a **background color** if "**BG**" is checked. This option allows to replace transparent image parts with a background color. Enable the option "**BG**" and click on the small black box to choose another background color. The background color is also applied if you are converting an image with transparency to a format that does not support it, see above.

Examples

The first example on the left side shows that the input image (size 2560x1200 px) is resized to 800x800 px. The original aspect ratio is not preserved and the image is stretched. The output format is set to "BMP 32bit", which is an uncompressed format supporting transparency.

For the second example on the right side, the same size is applied. But this time, the option "Output Fit" is checked so that the image is resized but not stretched. The background option "BG" is checked and a light blue color is chosen. The output format is set to "BMP 24bit", which is an uncompressed format not supporting transparency.





9.6 Leica 3D Disto

The Leica 3D Disto tool comes automatically with the installation of Pandoras Box. It is accessible in the "Tools" menu. The tool requires that the Leica hardware, specifically the 3D Disto, is connected and the drivers are installed. Please note, that the distometer cannot be purchased through your local distributor but we are happy to forward contact details.

In short, the motorized distometer measures points with great accuracy and in return can laser to any point, measured before. It draw our attention as it can be used in complex 3D scenarios to measure projector positions and object position. It reduces setup times significantly.

9.7 Matrix Patcher

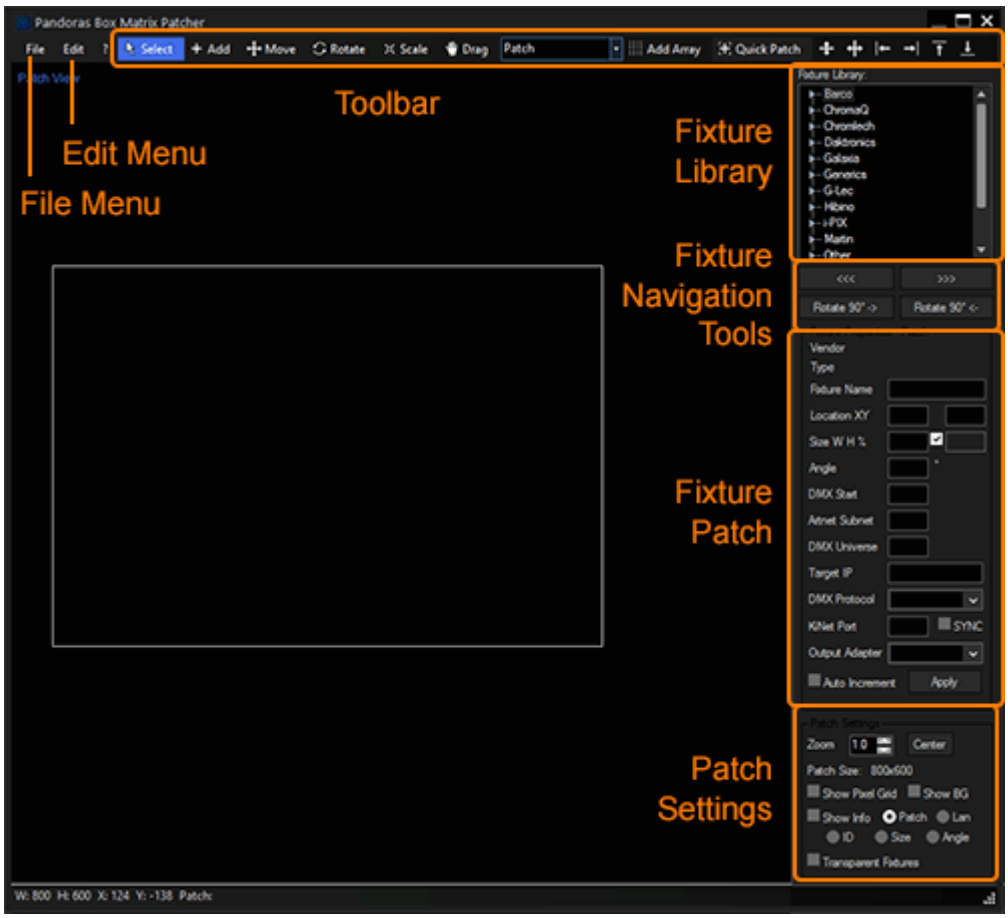
For special applications where LED or lighting fixtures are required to be patched to individual pixels, you will need to use the Matrix Patcher to set up your patch. With the saved patch file in your Pandoras Box project, the output will be additionally rendered to multiple Art-Net universes.

The Matrix Patcher plays also an important role if working with LED video tiles that are connected via DVI and a LED video processor. Most LED video processors are capable of creating only horizontal and vertical layouts for the video tile alignment. As soon as any of the tiles are rotated (in any angle), content mapping can be very difficult and complex to setup.
For those creative applications Matrix Patcher and a new Pixel Re-Mapping FX were designed.

Please see the topics below for working with the Pandoras Box Matrix Patcher:
Matrix Patcher User Interface [2077]
Fixture Editor [2085]
Patching Guide [2089]
ReMapping Guide [2093]

9.7.1 Matrix Patcher User Interface

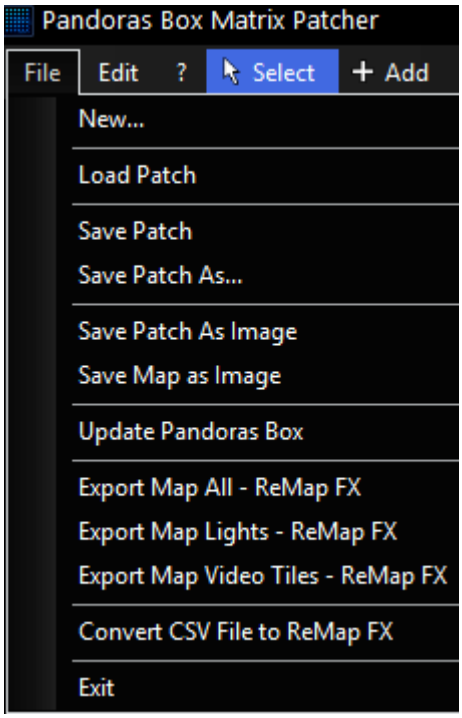
The Matrix Patcher User Interface is described in the following sections:



- File Menu [2078]
- Edit Menu [2079]
- Toolbar [2081]
- Fixture Library [2082]
- Fixture Navigation Tools [2083]
- Fixture Patch [2083]
- Patch Settings [2085]

9.7.1.1 File Menu

This chapter explains the File Menu of the Matrix Patcher [2077]. For other sections of the user interface see the introductory chapter [2077].



New…

Select this option to create a new Patch. In the pop-up dialog select the size when creating a new patch. Please make sure to match the display resolution of the Pandoras Box Client!
You may change the size anytime later as well via "Resize Patch" in the Edit menu.
Please keep in mind, that Pandoras Box outputs Art-Net only if a display is connected, or at least an EDID manager that leads the graphics card to believe there is one.

Load Patch
Loads a saved Matrix Patcher project file as a PBX file.

Save Patch
Saves the current Matrix Patcher project file as a PBX file and adds the patch to an open Pandoras Box project (and spreads it to all connected PB Clients). If the patch has been already saved, this command updates the Matrix Patcher project status and the file in the Pandoras Box project in real-time.

Save Patch As...
Saves a new copy of the current Matrix Patcher project file as a PBX file and adds the patch to an open Pandoras Box project.

Save Patch As Image
Saves the current Matrix Patcher Patch file as PNG image with the previous entered patch file resolution.

Save Map as Image
Saves the current Matrix Patcher Map file as PNG image with the previous entered patch file resolution.

Update Pandoras Box
Saves the current Matrix Patcher project file and adds the patch to an open Pandoras Box project (and spreads it to all connected PB Clients). If the patch has been already saved, this command updates the Matrix Patcher project status and the file in the Pandoras Box project in real-time.

Please note: Since revision 49 the CSV format is an obsolete format. From now on, Pandoras Box version 5.3 can read the Matrix Patcher file directly. However, if you still have copies of those CSV files and want to work with them, simply drag them into the project manually. From here, Pandoras Box treats a CSV like a PBX file.

Export Map All - ReMap FX
Exports a PNG file which can be used in Pandoras Box as a media for the ReMap FX [580].

Export Map Lights – ReMap FX
Exports a PNG file with all LED fixtures included which can be used in Pandoras Box as a media for the ReMap FX [580].

Export Map Video Tiles – ReMap FX
Exports a PNG file with all video tiles included which can be used in Pandoras Box as a media for the ReMap FX [580].

Convert CSV File to ReMap FX
Imports a CSV file with the output [x,y] and the source [x,y] positions in pixels and exports a png file which can be used in Pandoras Box as a media for the ReMap FX [580].

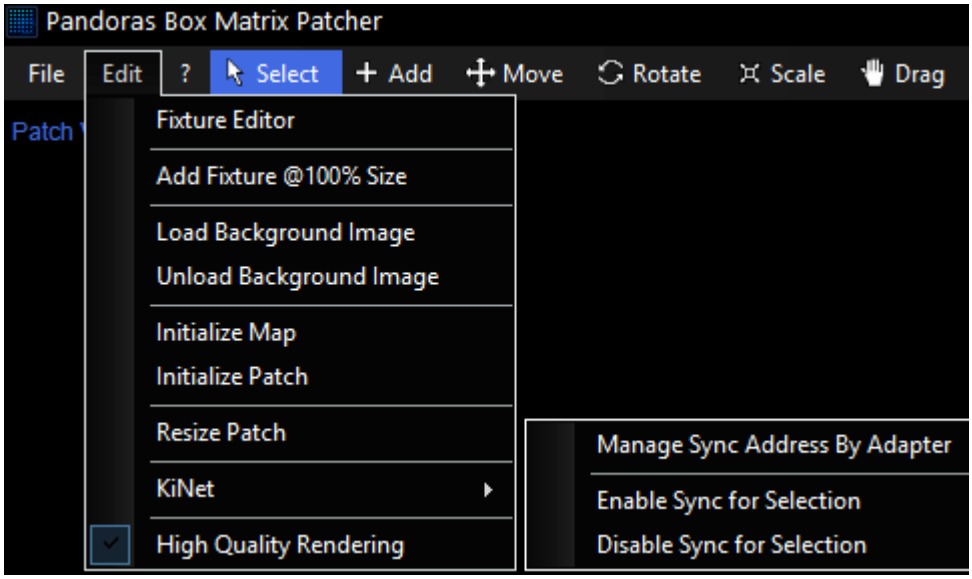
The CSV file format should be as in the following example:

Size;1024,768;
Output X; Output Y; Source X; Source Y;
0;0;315;73
0;1;312,72
0;2;311;71
0;3;309;69
0;4; […]

Exit
Quits Pandoras Box Matrix Patcher.

9.7.1.2 Edit Menu

This chapter explains the Edit Menu of the Matrix Patcher [2077]. For other sections of the user interface see the introductory chapter [2077].



Fixture Editor
From within the Edit Menu you have access to the Fixture Editor [2085] to create or modify your fixtures.

Add Fixture @ 100% size
Adds new tiles with a scaling factor of 100% - per default it is set to 500%. This command is especially useful when working with Video Tiles.

Load Background Image
This option opens a Windows explorer dialog where you may select an image to be loaded as background image in order to simplify creating the patch. To display the image the option [Show BG] in the Patch Settings [2085] has to be enabled!

Unload Background Image
Unloads the background image.

Initialize Map
Redraws the Map view[2081]. It discards all changes made there and applies all position / rotation / scale information from the Patch view[2081]. For more information please have a look at the ReMapping Guide[2093].

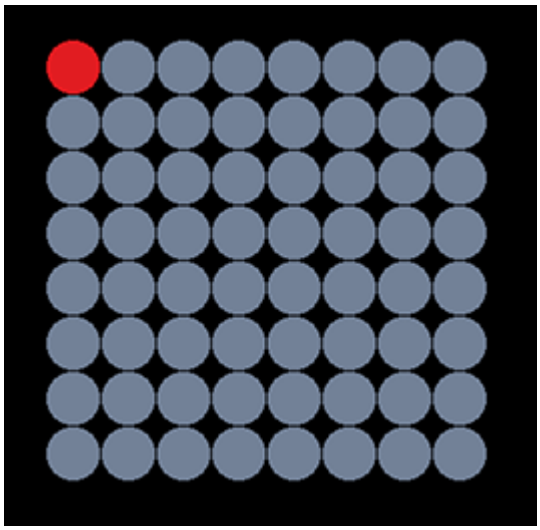
Initialize Patch
Resets the Patch view. It discards all changes made there and resets all position / rotation / scale information.

Resize Patch
Choose the new patch size (Width and Height, in px) in the pop-up dialog. Please make sure to match the display resolution of the Pandoras Box Client exactly!

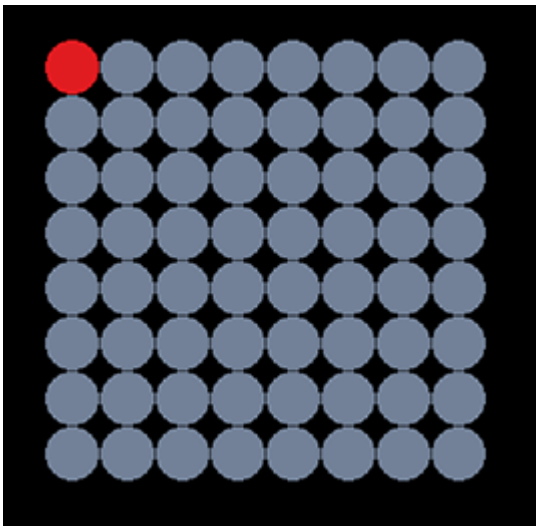
KiNet > Enable Sync for Selection
Ticks the checkbox Sync for KiNet Port in the Fixture Properties – Patch.

Disable Sync for Selection
Unticks the checkbox Sync for KiNet Port in the Fixture Properties – Patch.

High Quality Rendering
This option displays the fixtures either with high or low rendering quality, as seen in this example with the fixture 'ColorWeb 125':



"High Quality Rendering" enabled



"High Quality Rendering" disabled

9.7.1.3 Toolbar

This chapter explains the toolbar of the Matrix Patcher [2077]. For other sections of the user interface see the introductory chapter [2077].

File
Please see File Menu [2078].

Edit
Please see Edit Menu [2079].

Help
This shows the current Matrix Patcher revision number.

Select
In the "Select mode" you can click on a fixture to view or change the current values in the Fixture Properties - Patch [2083]. If you like to delete a fixture, select it and press the delete key on the keyboard.

Add
In the "Add mode" you can add fixtures to the patch. Choose one of the various LED fixtures in the Fixture Library [2082] first. Every click into the patch will create one fixture of the selected fixture type.

Move / Rotate / Scale
The Move, Rotate and Scale Modes allow to move, rotate or scale the fixtures on the patch. Just click on one fixture and move your mouse. You can also multi-select several ones first, either by holding the Ctrl Key and clicking on them or by drawing a selection frame.

To rotate all selected fixtures in 90° steps, you may as well use the Buttons [Rotate 90° ->] and [Rotate 90° <-] from the Fixture Navigation Tools [2083] (underneath the Fixture Library).

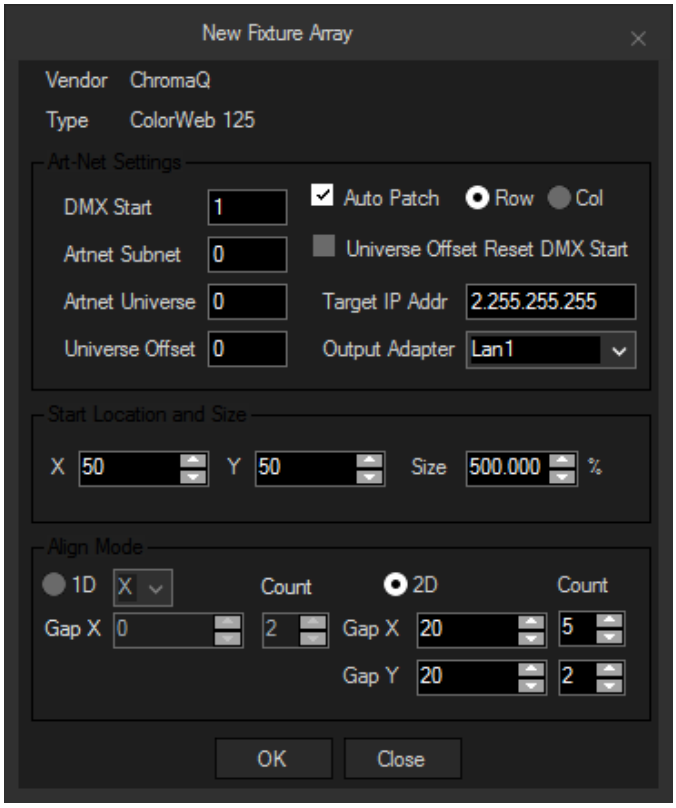
For accurate positioning, rotating or scaling of one fixture you can also enter values in the Fixture Properties – Patch [2083].

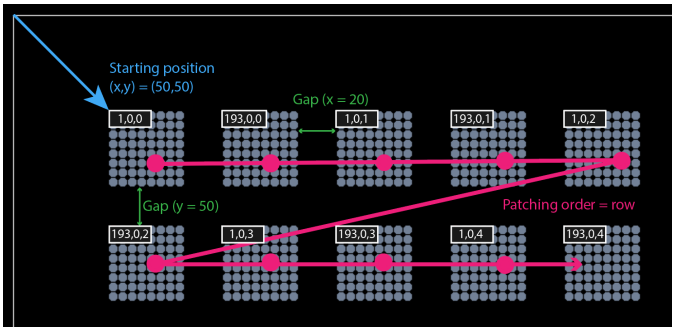
Keep in mind, that you can also Align [2082] fixtures, or create them using "Add Array" [2081].

Drag
In Drag Mode, the whole Patch can be moved to reach every part of the patch. The section Patch Settings [2085] (right bottom corner) offers a button called "Center" which moves the center of the patch to the center of the user interface. There, you can also set another Zoom factor alternative to using the mouse wheel.

Patch / Map Mode
This mode is of interest when working with rotated LED video tile connected via DVI, HDMI or DP and a LED video processor.
For more information please have a look at the ReMapping Guide [2093].

Add Array
After selecting a fixture from the Fixture Library this command will help you to add not only ONE copy to your current patch file (as the "Add" command does), but several ones. A dialog opens where you can set up how many copies you would like to add and the way how the array is inserted.
The Art-Net settings are of course only available if a DMX fixture was chosen from the library. The options are descriped here [2083].



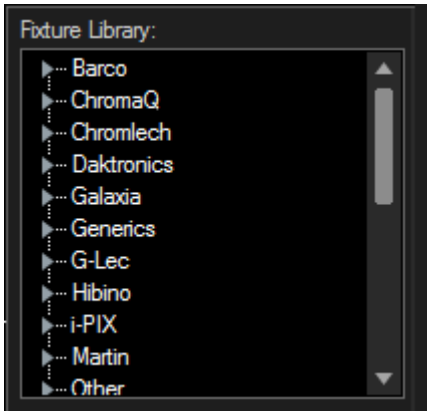


Quick Patch
The "Quick Patch" mode allows to patch fixtures quickly that were added already (in the Add Mode [2081]) to the patch. A dialog opens where you can enter your patch settings and a start address. Then, simply click on fixtures you want to patch. All fixtures will be patched one after another in an ascending order automatically.

Align
The Align mode allows to align the fixtures in vertical or horizontal orientation. To multi-select several fixtures, either hold the Ctrl Key and click on them or draw a selection frame.

9.7.1.4 Fixture Library

The Fixture Library of the Matrix Patcher [2077] contains a stock of common LED-Fixtures, sorted by vendor name. For other sections of the user interface see the introductory chapter [2077].



The Fixture Library also shows those fixture that were created by you with the Fixture Editor [2085].

To add one of the fixtures to the patch, select the fixture and choose [Add] from the toolbar. Then click inside the patch field.
If you need to add more copies, the "Add Array" [2081] function is a great time saver.

9.7.1.5 Fixture Navigation Tools

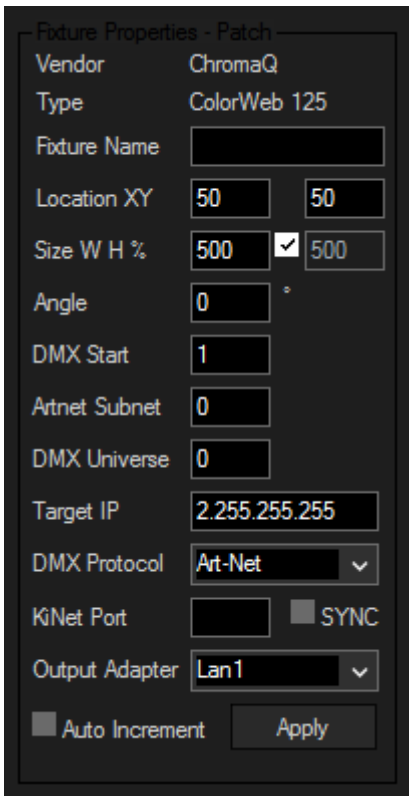
This chapter explains the Fixture Navigation Tools of the Matrix Patcher [2077]. For other sections of the user interface see the introductory chapter [2077].

These buttons select the next or previous fixture that was already added in the patch. The selected fixture turns blue and is loaded into the Fixture Patch [2083] section.

These buttons rotate the selected fixture(s) in 90° steps.

9.7.1.6 Fixture Patch

The Fixture Properties Patch of the Matrix Patcher [2077] displays information about the selected fixture. For other sections of the user interface see the introductory chapter [2077].



Vendor and Type
See here the Vendor and Type of the currently selected fixture.

Fixture Name
Here you may add a unique name to the currently selected fixture.

Press [Apply] to assign the modification to the fixture.

Location X /Y
This is the X and Y pixel location of the selected fixture. To change the position, enter new values and hit Enter or press the Apply button. Alternatively, you can use the Move Mode from the Toolbar [2081].

Size
This is width and height of the selected fixture (in %). Any new fixture added on the patch will be scaled to 500% by default, unless you activated the option "Add Fixture @ 100% size" in the Edit menu [2079].
To change the size, enter new values and hit Enter or press the Apply button. Alternatively, you can use the Rotation Mode from the Toolbar [2081].

Angle
This is rotation angle of the selected fixture (in °). To change the size, enter a new value and hit Enter or press the Apply button. Alternatively, you can use the Scaling Mode from the Toolbar [2081] or the 90°-Rotation Buttons (see Fixture Navigation Tools) [2083].

DMX Addressing
Using the DMX Addressing (including DMX Start address, Art-Net Subnet and Art-Net Universe) you may set up the fixtures in two different ways:

1. on the one hand it allows to modify the addressing of the selected fixture (press [Apply] to assign the modification to the selected fixture).

2. On the other hand you may enter the start address for the fixtures that are going to be added to the patch. Don't press [Apply] in order to use this function! Otherwise the modified DMX addressing is going to be assigned to the currently selected fixture.

++ Background information regarding DMX Addressing and network performance ++

One Art-Net-Universe contains 512 DMX channels (from 1-512). One Art-Net-Subnet contains 16 Art-Net-Universes (from 0-15). There can be 16 Art-Net Subnets (from 0-15). So in theory you may be able to address 512x16x16 = 131.072 DMX channels within one patch. As one pixel is represented by its color information, e.g. RGB, it takes up three channels, resulting in 43.690 pixels.

But when putting this into practice, we have to take the transfer rates of the transmitting network components and receiving devices into account too. This is the true limitation regarding the question how many pixel data in

form of Art-Net universes can be transferred. As a rule of thumb, we do not recommend to output more than 60 universes per network.

Let´s have a closer look: One Art-Net universe takes up 572bytes. As soon as you send only one channel, the entire universe needs to be transmitted. The default transfer frequency is 44Hz (40Hz, 33Hz and 25Hz are common too, especially for older devices). Thus each components needs to transfer 0,1-0,2 Mbit/s. This applies to switches as well as to the receiving devices. Most of the time these ones have lower specifications, e.g 10Mbit/s, which limitates us to approx. 60 universes. With 60 universes you can transfer approx. 100*100px.

Per default, Art-Net is set up as a broadcast protocol which means that all data is send to each Client within the network. Considering the above mentioned, this means that 100*100px is the limitation for the entire network, not only per device.

Since revision 49 the Matrix Patcher gives you the possibility to apply multi- or even unicast methods. The advantage is, that you may decide where the information is actually send to. Thus you may use your network resources more efficiently.

To do so, set up the "**Target IP Address**" per fixture.
2.255.255.255 - example for fixture 1 - the default address is a broadcast address.
2. 0 .255.255 - example for fixture 2 - the multicast method saves network resources.
2. 0 . 0 . 255 - example for fixture 3 - the multicast method saves network resources.
2. 0 . 0 . 5 - example for fixture 4 - the unicast method uses the network resources to its full capacity.

If we had the following devices connected via a switch to the Pandoras Box Client:
2. 0 . 0 . 5 - Device A receives the information from fixtures 1, 2, 3, 4
2. 0 . 0 . 22 - Device B receives the information from fixtures 1, 2, 3
2. 0 . 11. 5 - Device C receives the information from fixtures 1, 2
2. 3 . 11. 5 - Device D receives the information from fixture 1

As you can see, the broadcast information is sent to all devices. Even though it transfers information only for ONE panel, the other panels have to read and discard this information. This lowers the remaining amount of data that can be sent to this panel per second. If programming the patch to sent the information only via unicast, the receiving devices in the network do not need to discard unnecessary information and can be used more efficiently.

A downside of this method is, that you loose flexibility. In case the entire network routing has to be changed or only a single device needs to be set to a different IP address, keep in mind that the software patch needs to be updated as well. In conclusion, the multicast method combines efficiency and flexibility.

DMX Protocol
Choose the DMX protocol: Art-Net, KiNet V1 or KiNet V2.

With the protocol KiNet V2 the option SYNC will be activated and is selectable.

KiNet Port
Enter the KiNet Port. If the destination device has no specific port then enter -1, otherwise enter the current port number.

Network adapter
Choose the name of the network adapter that should output the Art-Net information. If an explicitly named network adapter is not found, the "any" adapter will be used. This is to be decided by the operating system and depends on your network adapter settings.
In case of processing a lot of Art-Net data it is very much recommended to use one network adapter for the communication with the Pandoras Box Master software and one separate network adapter for the Art-Net communication. All Pandoras Box Servers are shipped with a dual network card providing this working method.

Auto Increment
Use this function to automatically increment the DMX addressing when adding several fixtures to the patch. If checked, the application will calculate the next free DMX and Art-Net Address and patch added fixtures to it.

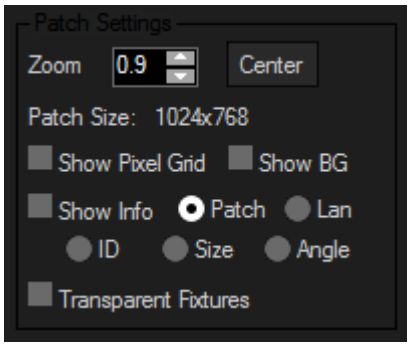
Example:
You want to add fixtures with a total channel count of 120 channels. You set the addressing to (1,0,0): DMX Start address = 1, Art-Net Subnet = 0 and Art-Net Universe = 0. Auto Increment is checked.
- The 1st fixture gets the Start address (1 , 0 , 0),

- the 2nd fixture gets the Start address (121, 0 , 0),
- the 3rd fixture gets the Start address (241, 0 , 0),
- the 4th fixture gets the Start address (361, 0 , 0)
- the 5th fixture gets the Start address (1 , 0 , 1).

The 5th fixture with its 120 channels won't completely fit any more into the Universe 0, so it gets assigned to the next higher Universe.

9.7.1.7 Patch Settings

The section "Patch settings" of the Matrix Patcher [2077] allows to change the Patch size and how it is displayed. For other sections of the user interface see the introductory chapter [2077].



Zoom
By default the zoom factor is set to 1. Increase the value to zoom into the patch, decrease it to zoom out.

Center
Press [Center] to move the patch's center point into the center of the user interface. This can be helpful after dragging [2081] the patch.

Patch Size
This simply displays the current pixel size of the patch. To edit it, open the Edit menu [2079] and choose "Resize Patch."

Show Pixel Grid
Check this option if you like to see a pixel grid in the patch. Note that the Zoom factor should be above 1.

Show BG
A background image (for example a floor plan) may be displayed in patch. To load a background image, open the Edit Menu [2079] and choose "Load BG image".

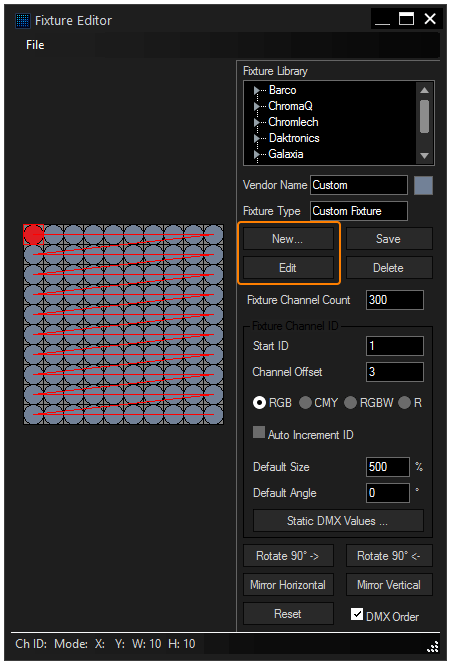
Show Info
When enabling this option, one of the following information will be shown on the patch for each fixture.
- Patch, i.e. DMX Addressing
- Lan IP-address
- ID
- Size [%] or
- Rotation Angle [°]

Transparent Fixture
Enable this option to display all fixtures half transparent.

9.7.2 Fixture Editor

The Fixture Editor from the Matrix Patcher [2077] allows to create and modify fixtures that can be used in the patch. Open it via the Edit Menu [2079] from the Main Matrix Patcher User Interface.

The File Menu [2087] and Fixture Library and Channel ID [2087] settings are explained in full detail on the following pages. Below you can find a guide of how to create and edit fixtures.



To begin with, press the "**New** 2087" button underneath the Fixture Library and create a "DMX Fixture" with e.g.: Vendor = _Custom, Type = Test1, Width=10, Height=10 and Offset=3.

Alternatively, you can also edit an existing fixture. Simply choose it from the Fixture Library and press "**Edit**". Now you can modify it as if you created a new one.

As seen left, for **DMX fixtures** (controlled via DMX or Art-Net), a default channel order is already applied. The fixture's top left pixel is highlighted in red because it is the first channel. From there the red line shows the DMX addressing order.

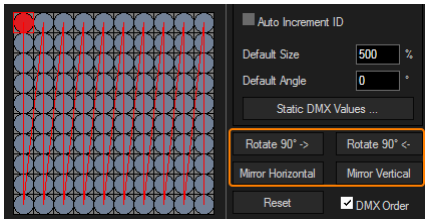
If you hover your mouse over the fixture, the **bottom bar** displays according pixel and fixture information including the pixel's channel ID.

If you are working with **Video Tiles** (controlled via DVI, HDMI or DP) the pixels are displayed as squares. Of course no DMX order needs to be displayed and no further adjustments have to be made. The ReMapping Guide 2093 might be of interest.

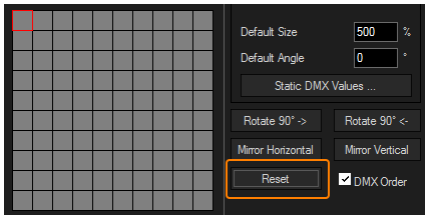
If the order of the channels and the other settings 2087 are alright, press the "**Save**" button. In case you edit an existing fixture, change the "Vendor" or "Type" name before saving if you do not like to overwrite the existing fixture.

After closing the Fixture Editor, you can find your fixture in the Fixture Library to add 2081 it to your patch.

Changing the Channel Order of a Fixture

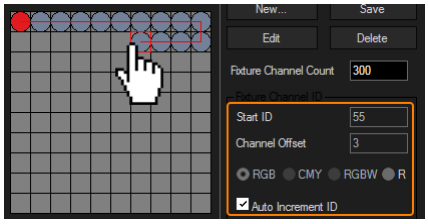


If the default channel pattern is alright but has another orientation in your LED fixture, the "**Rotate**" and "**Mirror**" buttons could be helpful. Remember to save changes to the fixture.



If you need to set up a completely different channel pattern or addressing, first press the "**Reset**" button.

Now, there are no addressed channels in the fixture and you can apply a custom DMX channel order.



Set the "**Start ID**" to 1 and check whether the "**Channel Offset**" and **Color Mode** are alright, e.g. 3 and "RGB".

Then check the option "**Auto Increment ID**" and draw the new order in the fixture - simply click into the fixture and drag the line.

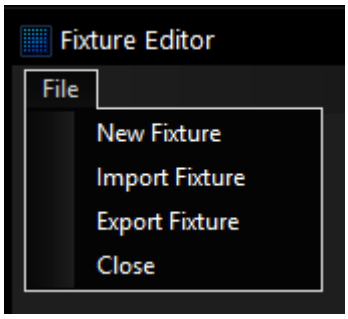
If you made an error, you can stop drawing at any time. Click on a patched pixel to unpatch it again. If you start drawing again, the line red continues automatically. However, keep in mind, that the ID increments automatically. If you like to "go back" some channels, uncheck the increment option, set the correct "Start ID" and check the option again.

Alternatively, you can work without the "Auto Increment" option. In that case, set the "Start ID" any time before clicking into the fixture. The red line connects the pixels automatically in the order of ascending IDs.

Remember to save changes to the fixture before closing the Fixture Editor.

9.7.2.1 File Menu

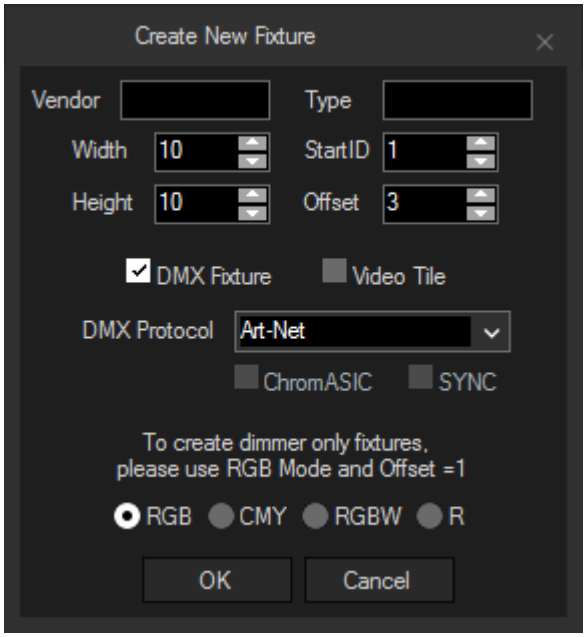
This chapter explains the File Menu of the Fixture Editor inside the Matrix Patcher [2077]. For other sections of the Fixture Editor [2085], see here.



New Fixture
Creates a new fixture, see dialog below. Please note that any prior work won't be saved.

Import Fixture
Imports existing fixtures (in MFX format) into the library.

Export Fixture
Exports fixtures from the library to be saved for later use (as MFX files).



In the pop-up dialog you may enter a **Vendor** Name, fixture **type**, its **width** and **height** (in px), the **Start ID** and an **Offset**. An offset of 3 (by default) will create 3 DMX channels (RGB or CMY) per pixel.

DMX Fixture / Video Tile
If you wish to create a fixture that should be controlled via DMX or Art-Net, tick the check box "DMX Fixture".
If you wish to create a fixture that should be controlled via DVI, HDMI, DP, etc., tick the check box "Video Tile".

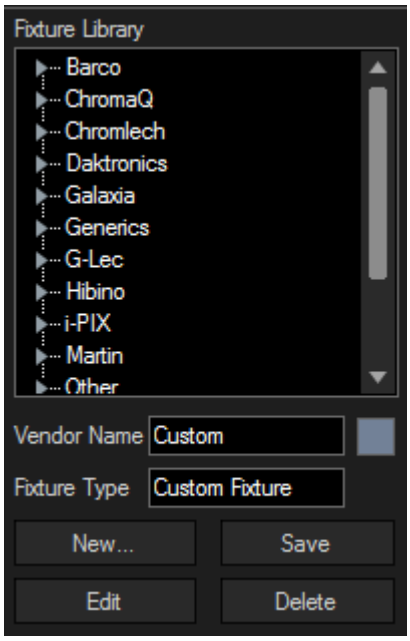
DMX Protocol
Choose the DMX protocol: Art-Net, KiNet V1 or KiNet V2.

With the protocol KiNet V2 the options ChromASIC and SYNC will be activated and are selectable.

Color Mode
With the radio buttons RGB, CMY, RGBW and R you can setup the color mode according to your fixture. Note that the offset should be set accordingly, except for dimmer-only fixtures. In that case please use RGB Mode and Offset=1.

9.7.2.2 Fixture Settings

This chapter explains the Fixture Library and Channel ID settings of the Fixture Editor inside the Matrix Patcher [2077]. For other sections of the Fixture Editor [2085], see here.



The Fixture Library contains a stock of common LED-Fixtures, sorted by vendor name. Every Fixture that was created with the Fixture Editor will appear here as well.
The section below allows to create a new custom fixture and edit or delete any existing fixture.

Vendor Name / Fixture Type
As soon as a fixture is created or set to be edited, the Fixture's Vendor Name and Type are displayed in the text fields.

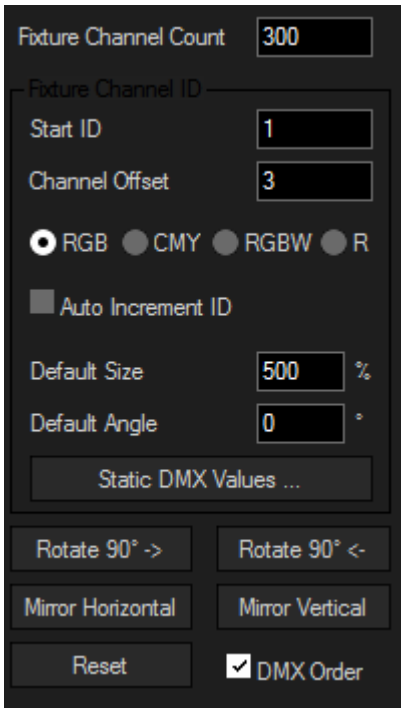
Fixture Color
The small colored box next to the Vendor Name allows to choose the fixture's background color. It will open a colour-picker once you click on it

New...
Allows to create a new fixture, see File Menu [2087]. Please note that any prior work won't be saved automatically. The Fixture will be loaded into the Workspace now for further editing.

Edit
To start editing an existing fixture, please select the fixture in the Fixture Library and press [Edit]. The Fixture will be loaded into the Workspace now for further editing.

Save
Saves the changes in the fixture under the Vendor Name and Fixture Type currently displayed.

Delete
Deletes the fixture currently selected in the fixture library.



Within the Fixture Channel ID section the DMX Addressing for the selected fixture will be set up.

Fixture Channel Count
This is the amount of channels the fixture contains. It is calculated by the Fixtures Width and Height multiplied with the Channel Offset. For example, the fixture is 4 px wide and 4 px high and has an offset of 3 channels (for RGB), then the Fixture Channel Count is 4x4x3 = 48 channels. This Channel Count is used to calculate the start address for the next fixture when setting up the patch.

In order to create fixtures with unpatched pixels you need to manually modify that number, so that the start address for the next added fixture is correct when using auto increase while patching the matrix.

Start ID
This is the DMX Start Address for the next pixel.

Channel Offset
Amount of DMX channels per pixel.

RGB / CMY / RGBW / R
Defines if the color values of the pixels will be interpreted as RGB, CMY, RGBW or R.

Auto Increment ID
When activated the starting channel of the pixels will automatically be incremented according to the chosen offset.

Default [Size
This is the fixture's size when added on the patch. By default it is set to 500%, you may change this size when editing a fixture.

Default Angle
The default angle refers to the fixture's angle when it will be added on the patch.

Rotate 90°
To rotate the selected fixture in 90° steps, you may simply use the Buttons [Rotate 90° ->] and [Rotate 90° <-].

Mirror
This allows to mirror the selected fixtures horizontally or vertically.

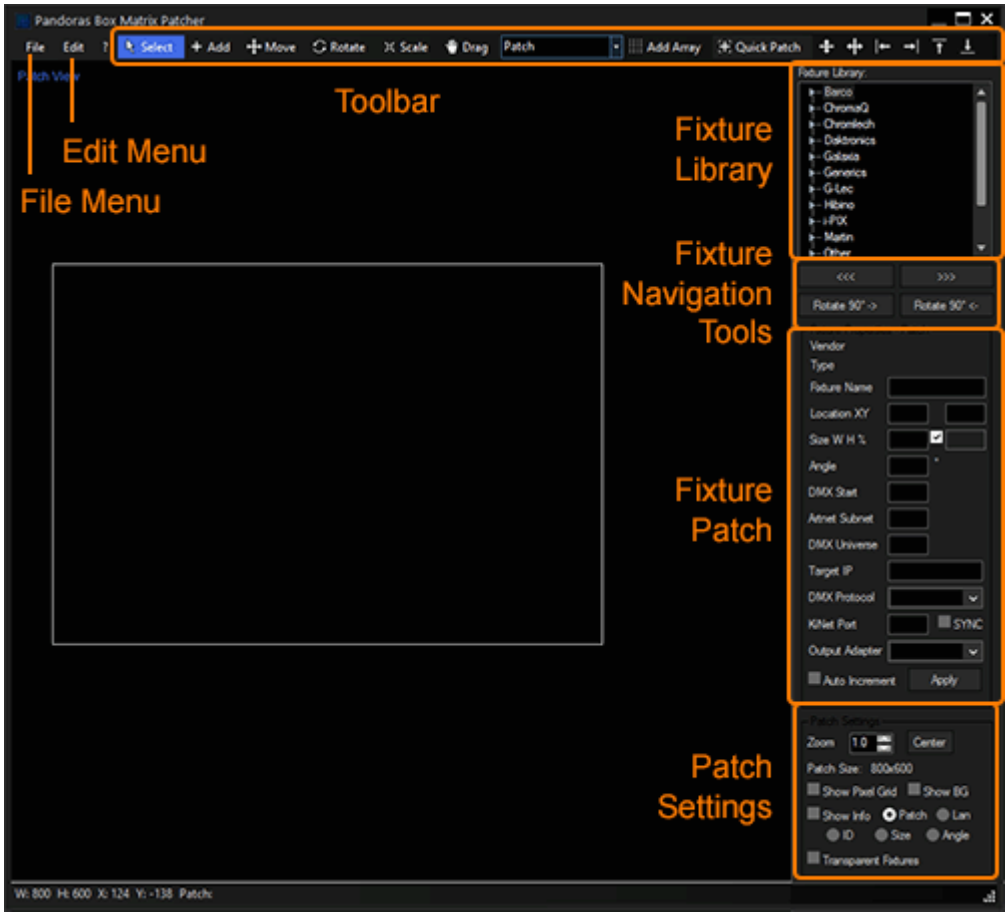
Reset
Clears the DMX addressing of the currently selected fixture.

DMX Order
When this option is checked, it shows the pixel addressing order in the fixture with a red line. The starting pixel is marked with a red dot.

9.7.3 Patching Guide

This chapter will be updated soon.

The Pandoras Box Matrix Patcher starts with the main patching window:



The Fixture Library on the top right contains a stock of fixtures by different vendors. If the fixtures you need are not included or if the fixtures need to be modified, you have to create/edit them first using the Fixture Editor [2085].

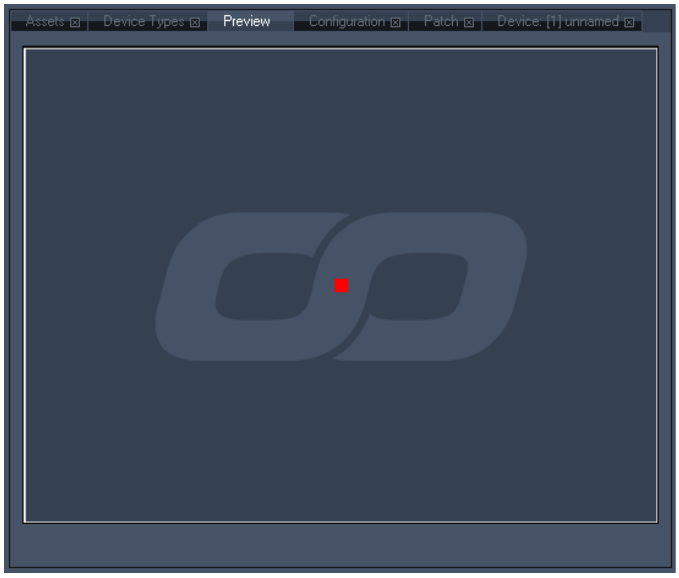
As soon as all fixtures needed are available in the Fixture Library you may start setting up your patch.

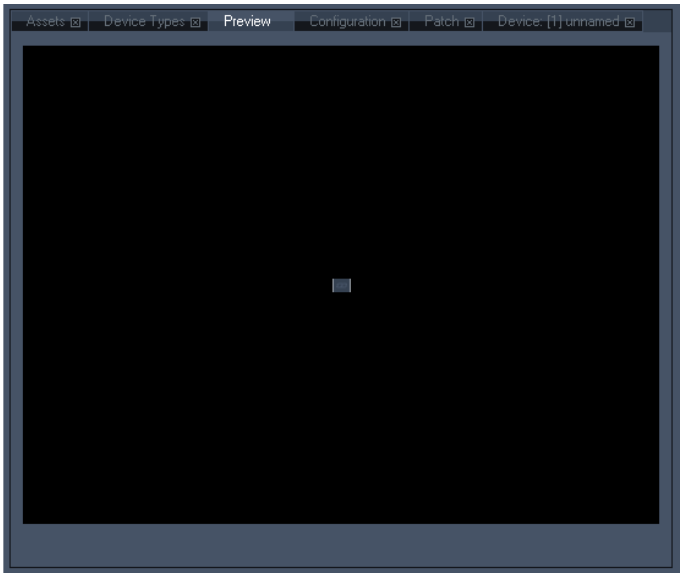
Important information about setting up the patch size

The Matrix Patcher up from Revision 12 should be used with big patch sizes, although you maybe only need a patch that is 10x10 pixels big! The reason for this is that the patch size always refers to the size of the Pandoras Box output that is used for the Matrix. Please keep in mind, that Art-Net output is only "rendered" if a display is connected, or at least an EDID manager that leads the graphics card to believe there is one.

Example 1:
The Pandoras Box output 1 of a PB Server should be used to control your LED Panels. The display attached to this output is set up to the resolution 1024x768 px. The LED Panels where the video content should be displayed on are 20 px wide and 20 px high.

a) If the patch is only 20x20 px large (according to the amount of pixels of the LED Panels), only a fraction of the whole PB layer size will be used for the Matrix, illustrated by the red square in the picture below. This way the content (in this example the infinity logo) has to be scaled very small to be completely displayed on your devices. And this is quite unhandy.





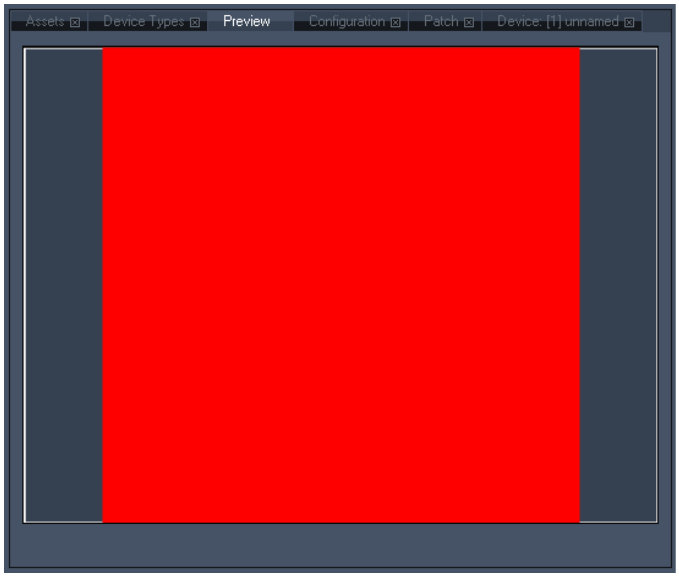
Left side:

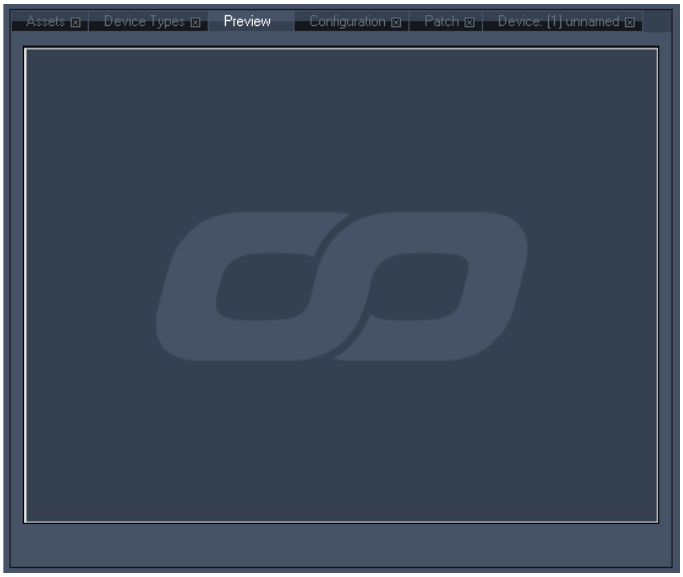
By default the video content (the infinity logo) covers the whole output (output size: 1024x768 px). A patch with the size 20x20 px would only give out DMX data for the area that is covered by the red square.

Right side:

To ensure that the logo will be shown on your LED Panels with its whole height, it has to be scaled down a lot.

b) A better and more handy workflow inside Pandoras Box is given when the patch gets close to the size of the Pandoras Box Matrix output (PB output= 1024x768 px). You may create a patch now that is 1024x768 px. The fixtures on the patch will be scaled up and arranged so that they form a square (because of the pixel aspect ratio 20:20) covering the whole patch's height. The colour values for one pixel of the patch will now be calculated as average value from a bigger area, not only from one pixel as shown in the example 1a. See picture below.





Left side:

By default the video content (the infinity logo) covers the whole output (output size: 1024x768 px). The fixtures on the patch with the size 1024x768 px cover nearly the whole output, shown by the red square.

Right side:

To ensure that the logo will be shown on your LED Panels with its whole height, it has not to be scaled down any more!

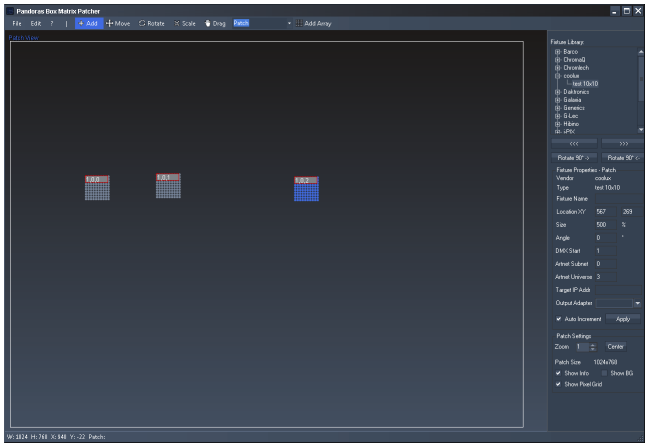
Patch Size

In the Patch Settings [2085] the current patch size is displayed. By default the size is set to 1024x768 px. Adjust this patch size to fit your PB Matrix output resolution.

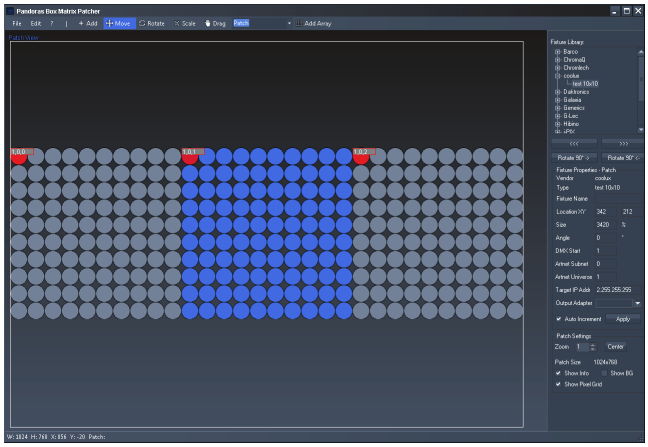
Adding Fixtures

Now that the patch is set up, you need to add one or several fixtures to your patch. Let's assume that you want to place 3 square fixtures (with the size 10x10 px) side by side in a line, covering the whole width of the patch.

Select the fixture in the Fixture Library. If it is not available set it up in the Fixture Editor [2085] first. But before adding the fixtures on the patch, think about how they should be addressed if working with DMX fixtures. Turn over to the Fixture Patch [2083] and enter the DMX starting address and the Art-Net Subnet and Universe for the first fixture. Don't press [Apply] as this should be used only when modifying DMX addresses after the fixture is created.
If the DMX addressing should be automatically incremented for all following fixtures, enable [Auto Increment]. If you want to address the following fixtures manually, you may enter the new DMX addressing in the Fixture Patch before adding the fixture to the patch.

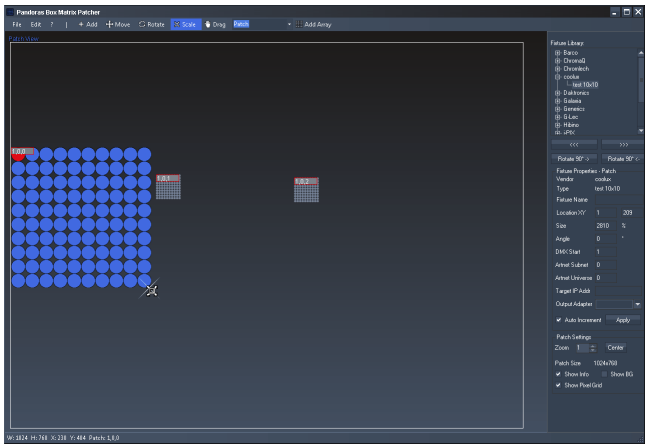


Now click on [Add] in the Toolbar [2081] or choose the add-function via the right-click menu and proceed a left mouse click in the patch area. Your first fixture will appear. Do this two more times to get all 3 fixtures on the patch.

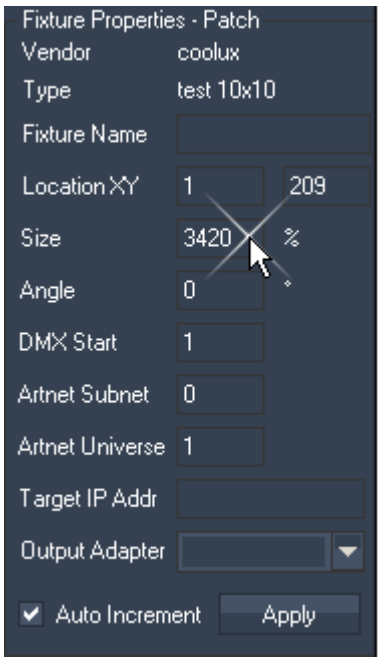


Adjusting the fixtures on the patch
As the fixtures should cover the whole width of the patch, you have to position and scale them to the correct size. So that the result looks like depicted.

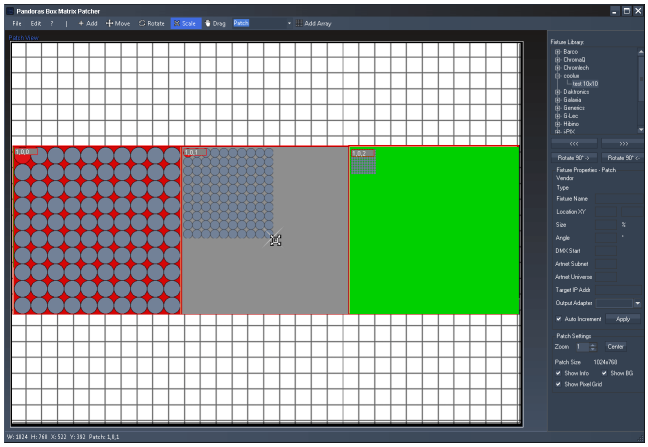
The scaling can be done in several ways: by eye, by calculating the correct zoom factor or by using a prepared background image.



1) By eye:
Turn to scale-mode using the right-click menu or pressing the scale button in the toolbar [2081]. Click on the fixture and move the mouse cursor to the right while holding down the left mouse button. Do this for every fixture. Then adjust the positions by turning into the move-mode.



2) By calculating the correct size (in %):
The patch is 1024 px wide, one fixture is 10 px wide (zoom factor = 100%). A fixture should cover 1/3 of the patches width. So 1/3 of the patch width are 1024/3 = 341,3 px. To get the new size, divide the size 100% through the fixture's width (10 px) and multiply this with the width the fixture should cover (341,3 px):
100% / 10px x 341,3px = 3413 %.
Select the first fixture when being in move-mode. Now enter the new size into the according text field in the Fixture Patch [2083] and press [Apply]. Do this for all fixtures and position them on the patch.
Please note: the size can only be applied in 10%-steps, so that you have to round the value up or down. In this case you could use a size of 3410% or 3420%.
A faster workflow using this method is: when starting from the beginning, choose the fixture from the library and press the button "Add array", set up 3420% and choose to insert a "1D" array.



3) By using a prepared background image:
You may prepare an image, that shows the correct adjustment of three items over the width of the image. This can be loaded as background image. To do so, go to Edit> Load Background Image and browse to the image path.
To show / hide this image, toggle the option [Show BG] in the Patch Settings [2085] down right.
This background image helps you now to scale and arrange the items on the patch.

Useful Hints:
You may check and modify the settings in the Fixture Patch for each fixture if necessary (e. g. the DMX addressing or the positioning).
To remove a fixture from the patch, highlight the fixture by clicking it and press [Del] on your keyboard.

Save Patch & Export Matrix:
When patching the matrix is completed, it is recommended to save this patch file in order to still be able to modify it later on. Do this by File > Save Patch. In addition this command adds an according *.pbx file as a media file to the project if a Pandoras Box Master software is running at the same time plus it spreads the file to all connected Clients. This is very important, especially when working with DMX Fixtures as the Art-Net data is generated by the Client and output through its network adapter.
Please note: In older versions (before Rev 49 and Pandoras Box 5.3) it was necessary to export a patch separately. Automatically a "pixelpatch.csv" was saved in the directory where the Matrix Patcher.exe file is located (e.g. C:\coolux\program files\Pandoras Box SERVER STD Rev xxx). This is now not necessary anymore. However, if you still have copies of those CSV files and want to work with them, simply drag them into the project manually. From here, Pandoras Box treats a CSV like a PBX file.

The next step is, to use the generated matrix file in Pandoras Box. As soon as the file is included in the project you may start programming it onto the timeline. To do so, choose which Pandoras Box Client should calculate the Art-Net data (in addition to the normal video output via DVI) and drag the matrix file on its output [682]. Please note, that effects dragged onto the output layer cannot be considered by the patch file, thus you will not see output effects on your DMX panels.
Of course each Client in your PB network may be programed with different matrix files. In addition it can be saved on the timeline and may change throughout the show.
Use the right-click menu in the Preview [245] to visualize the matrix files.

If working with Video Tiles please proceed to the ReMapping Guide [2093].

9.7.4 Remapping Guide

This chapter will be updated soon.

This chapters explains in more detail how to use the remapping feature if working with Video Tiles that are connected via DVI and a LED video processor. If you are working with DMX Fixtures connected via Art-Net feel free to skip this chapter.

BACKGROUND INFORMATION
Most LED video processors are capable of creating only horizontal and vertical layouts for the Video Tile alignment. As soon as any of the tiles are rotated (in any angle), content mapping can be very difficult and complex to setup.
For those creative applications Matrix Patcher and a new Pixel Re-Mapping FX [580] were designed.

DIFFERENCE BETWEEN PATCH AND MAP VIEW
The patch view represents the output pixels, that is how the tiles are programmed and located in the LED processor.

The map view is for relocating the individual tiles in any orientation and size according to the real setup. In other words the map view represents the source pixels of a Pandoras Box Layer Composition Space. In Pandoras Box you set up the content (and layers) according to the stage in the real world. There is no need to create special content in a complex way anymore.

The combination of both views results in the exported map png. The map stores for each source pixel the position of the output pixel. Afterwards the ReMap FX in Pandoras Box reads the map and routes automatically any source pixel to the stored output location. Thus it routes the pixels according to the patch / map setup that you may influence at any time of programming.

When creating Video Tiles you would first create a Video Tile layout as in the video processor view and then switch to the Map View and adjust the tiles as seen in the real world.

STEP BY STEP: How to use the ReMapping feature

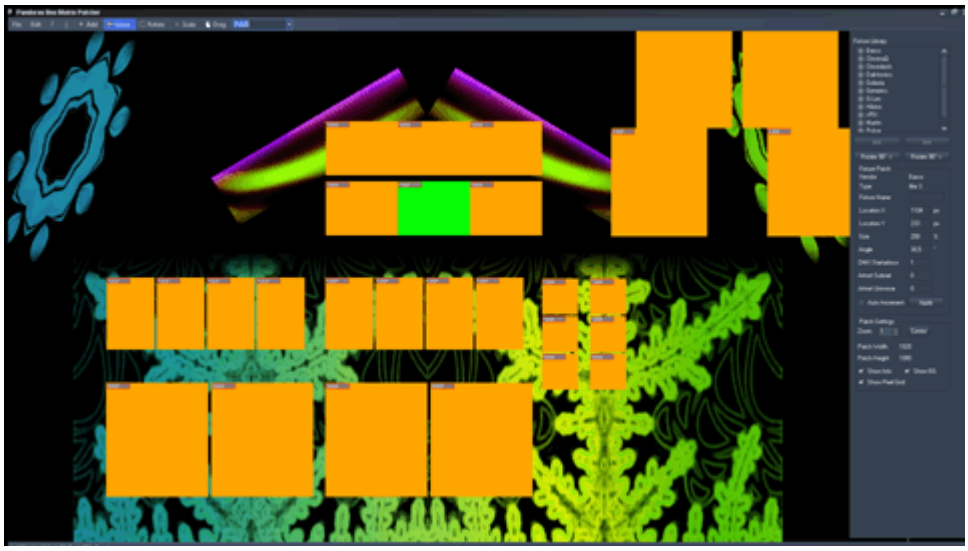
- Create a new Patch with the same (!) resolution Pandoras Box Output is set to and how the LED video processor is controlled

- If the fixtures needed are nor included in the Fixture Library, create new Video Tiles in the Fixture Editor 2085

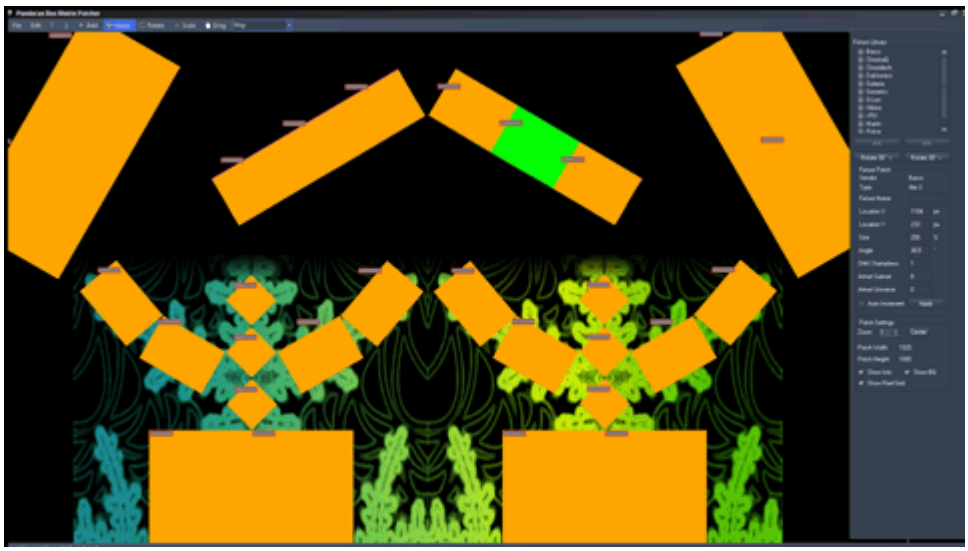
- Add these Video Tiles in the Patch View as they would be laid out in the LED processor

- Then choose "Initialize Map" from the Menu and switch to the Map View that now looks like the Patch View to start with

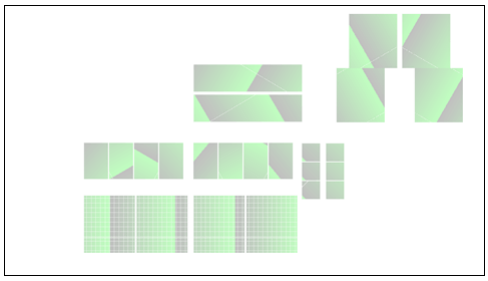
In the image below a background image was chosen. It shows a scene from the real stage setup. This is helpful for the next step.



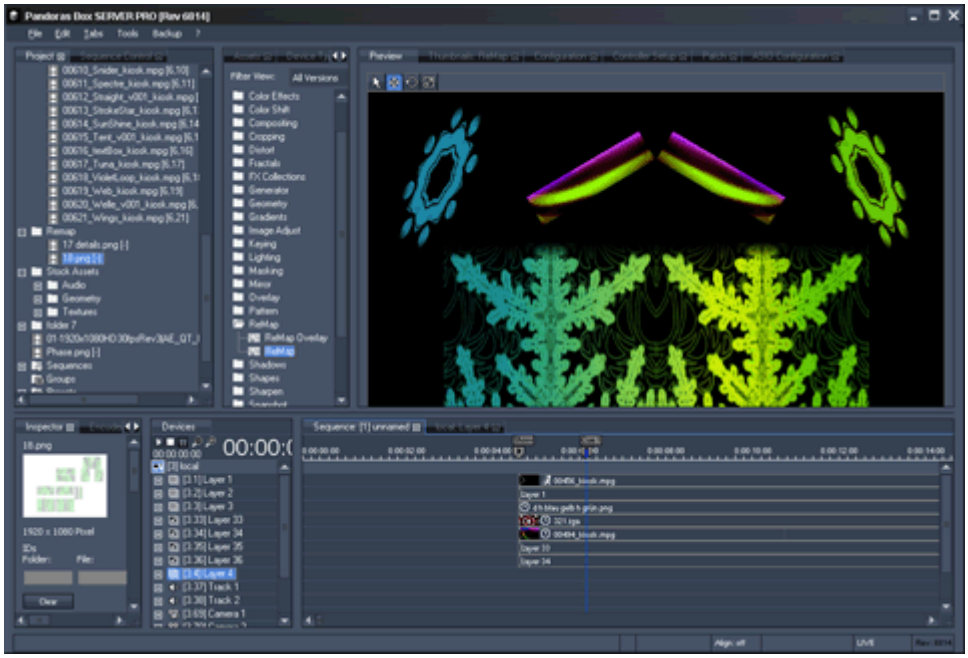
- Now re-arrange and rotate the Video Tiles in the Map View as they are in the real world. You can do this by eye or by using a prepared background image.



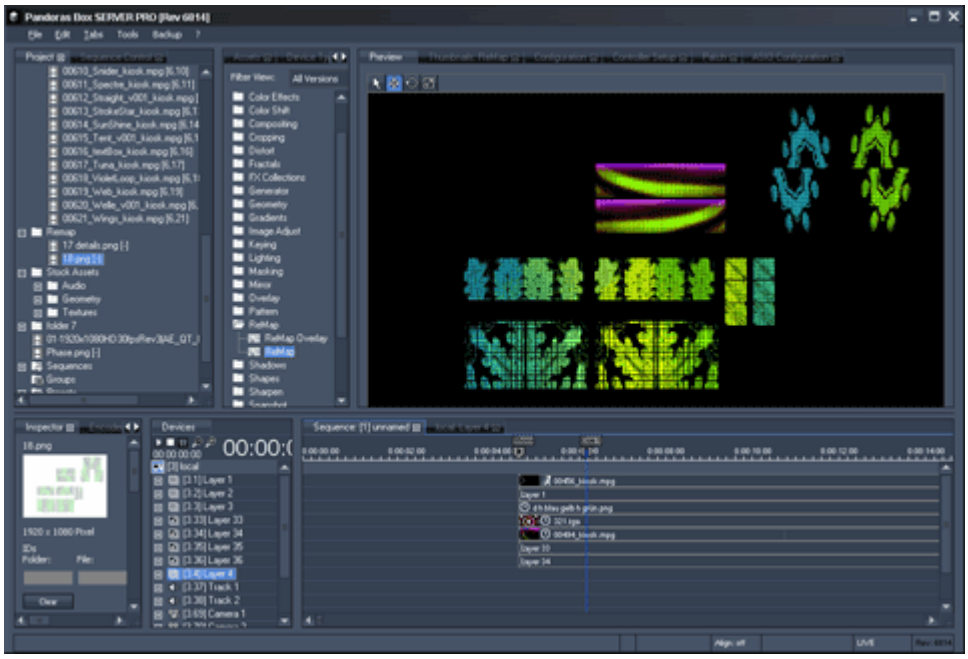
- Once you are confident with the setup, use the "Export Map" option from the File Menu in order to store the map as a PNG file for the ReMap FX in Pandoras Box.



- Load the map.png into a Pandoras Box project and make sure that anisotropic filtering is turned off. To do so, click the file in the Project Tab and have a look in the File Inspector. This step if very important for the entire process !
The image below shows a setup up without the ReMap effect turned on.



- Add the ReMap FX to the desired Output and load the Map.png into the Media Input of the ReMap FX [580] and set the Mix to 255. Now, the pixels are routed differently as seen in the Preview depicted in the image below. Note that the effect has to be assigned to an output layer and is only visible in the Preview if it is set to an output view, not the global camera.



If you need to change something regarding the fixture setup in the Matrix Patcher please keep in mind that for any exported PNG file the anisotropic filtering must be turned off in Pandoras Box.
If you have to add one more tile to the scene or if you have to replace some of them, it is the easiest to position the few tiles in the Patch View and WITHOUT using "Initialize Map" you position them in the Map View. Keep in mind that the "Initialize Map" command resets all changes made in the Map View so far.

If you need to enlarge the entire Patch, please use the "Resize Patch" command. Do not scale the exported Map. It is very critical to keep every single pixels' information.

9.8 PB Menu

This chapter describes the launch application "PB Menu" that is pre-installed on all Pandoras Box hardware like Compact Players [1944] and Servers [1939]. It starts automatically when booting. It covers the Windows desktop and offers quick access to the most relevant actions, e.g. starting the PB Master or Client software or opening the Windows Explorer (File Browser). Further, the PB Menu displays crucial information regarding the network, disk space and output state.

The PB Menu includes another software called the Server Management Application [2109]. Its main purpose is to manage multiple hardware systems and to establish a VNC connection to another computer to remote control it. The installation includes both the VNC Server and Viewer (Client). The VNC Server service can be found in the SysTray from Windows.

You can download the PB Menu from the Download-Center on our homepage. Even though the PB Menu is primarily designed for our hardware you can also install it on custom Windows hardware running Windows 8.1 or Windows 10. Since version 6 a 64-bit environment is necessary.

It might be of interest that Widget Designer [786] offers two input nodes displaying software and hardware information of remote computers. The "PB Menu [1062]" Input Node receives information from the PB Menu and "Engine Health [1057]" from Pandoras Box. Please use the PB Menu version 7 with WD version 6.5.

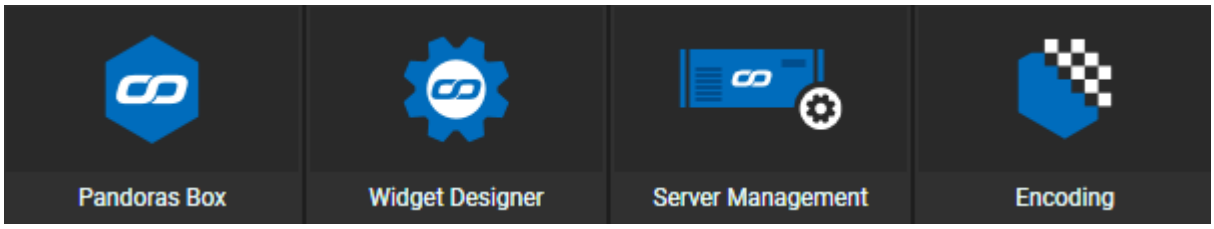
Installation

The PB Menu installer can be found in the Download-Center on our website. Before installing, please deinstall first any PB Menu version if it is older than version 7. This is not necessary if you are updating within version 7. Note that in this case, your "Settings [2101]" are kept.

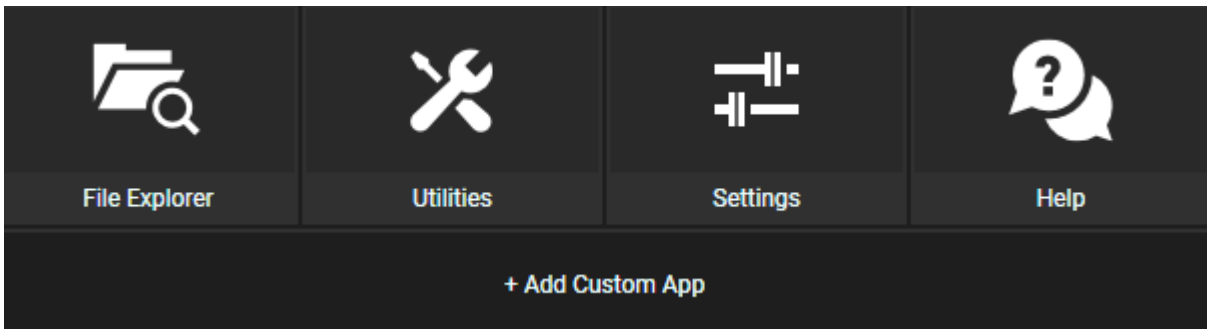
Since PB Menu version 6 you can also run the installer whilst being connected through a VNC connection from the Server Management Application [2109]. The installer restarts the PB Menu after installation so that you can re-connect via VNC.

The fixed installation path is `C:\Program Files\Christie\PB_Menu` whilst Test Pattern are now saved under `C:\Christie\content\Stock Assets\Testpattern` (or `C:\coolux...`).

When the installation is finished you are asked to accept the End user License Agreements from Windows and Quicktime. From now on, when you boot the system, PB Menu will start automatically to cover the Windows desktop and task bar and show a menu instead.
Please note that automatic updates from Windows 10 are deactivated with the installation of the PB Menu, but you can activate them again under Settings [2105].



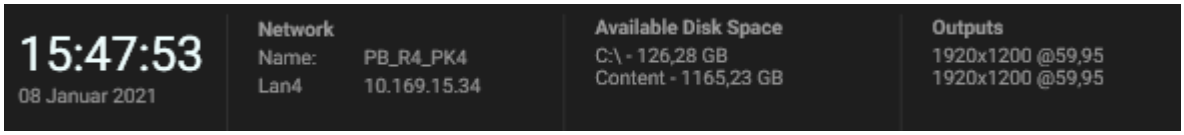
Pandoras Box	This starts the configured Pandoras Box version. To choose one, go to Settings > Pandoras Box [2101]. If PB is already running,
Widget Designer	This starts the configured Widget Designer version. To choose it, go to Settings > Widget Designer [2102]. There, you can also activate, whether you like to start multiple WD instances when pressing this button.
Server Management	This starts the Server Management Application [2109].
Encoding	For the time being, this is not implemented yet. This will start the Encoding GUI.



File Explorer	This opens the Windows Explorer (File Browser).
Utilities	This opens the Utilities menu [2099].
Settings	This opens the Settings menu [2101].
Help	This opens the helpfile from the installation path chosen in "Settings > Pandoras Box / Widget Designer".
Add Custom App	This feature allows to add a quick access button for another executable file, including EXE or BAT files. Simply choose your file from the dialog and click "Open". To setup another label or icon or if you like to launch the file automatically each time the system is started, go to Settings > Custom Apps [2103].

Taskbar	When switched on: - the Windows Taskbar and desktop are visible - the PB Menu can be repositioned by dragging its title menu When switched off: - the Windows Taskbar and desktop are hidden underneath a full screen test pattern - the PB menu is placed at position 0/0 You can save the initial behavior, i.e. whether the taskbar is automatically hidden, under Settings > Taskbar [2104].

Shutdown — This opens a pop-up menu where you can choose:
- Shutdown: shuts the system down
- Restart: restarts the system
- Reset Menu: restarts the PB Menu, the included VNC Server and the LCD Display (for PB Server hardware) - note that all Settings [2101] and the customizable buttons are not reset



Time, Date — This shows the current system time and date.

Network — This shows the Computer name as well as the names and IP addresses from three available network adapters.
You can change IP addresses etc. under Settings > Network [2107] or of course within the Windows network setup.

Disk space — This shows the available storage space for four disks as chosen under Settings > Info Section [2104].

. If there is a folder named "`C:\Christie\content`" or "`C:\coolux\content`" it is shown instead of the C:-drive.

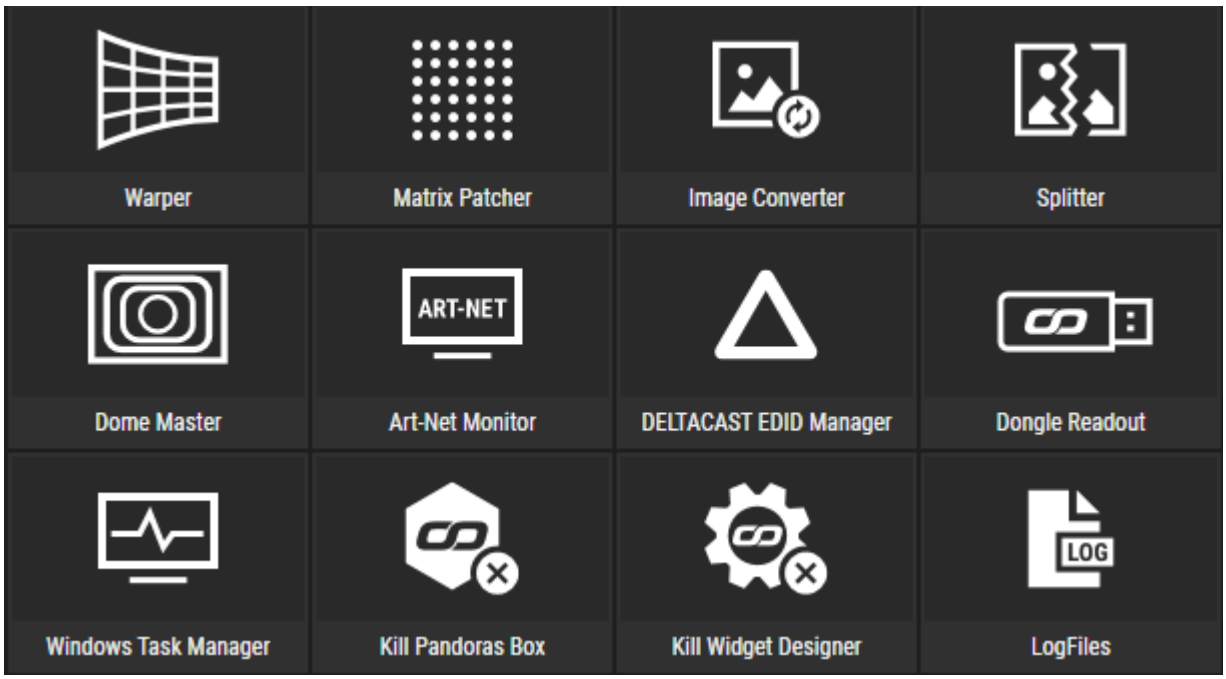
Outputs — This shows the number of available outputs with their respective output resolution and frame rate for supported NVIDIA graphics cards (i.e. Quadro M4000, P1000, P4000, RTX 4000, RTX 6000).
If the mosaic mode is activated in a NVIDIA card, a single resolution will be shown including the number of used outputs.

You can change output settings under Settings > Display Setup [2106] or of course within the NVIDIA control panel.

The next chapters describe the Utilities [2099] and Settings Menu [2101].

9.8.1 Utilities Menu

This chapter described the Utilities Menu from the PB Menu [2097]. It opens when the Utilities button is clicked.

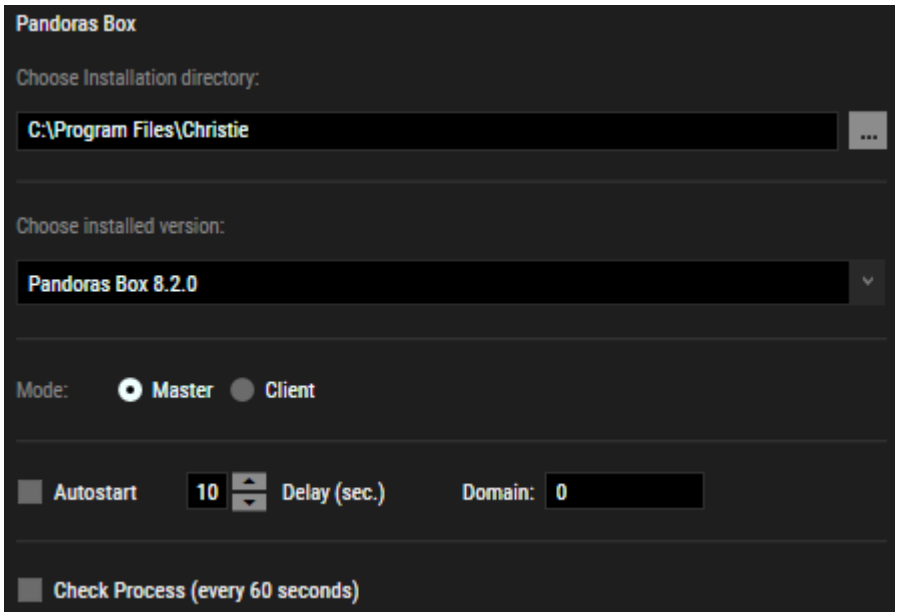


Warper	This starts the PB tool "Warper 2129". As its version this depends on the installed Pandoras Box version, choose one via "Settings > Pandoras Box 2101".
Matrix Patcher	This starts the PB tool "Matrix Patcher 2077". As its version this depends on the installed Pandoras Box version, choose one via "Settings > Pandoras Box 2101".
Image Converter	This starts the PB tool " Image Converter 2069". As its version this depends on the installed Pandoras Box version, choose one via "Settings > Pandoras Box 2101".
Splitter	This starts the PB tool "Splitter 2120". As its version this depends on the installed Pandoras Box version, choose one via "Settings > Pandoras Box 2101".
Dome Master	This starts the PB tool "Dome Master 2055". As its version this depends on the installed Pandoras Box version, choose one via "Settings > Pandoras Box 2101".
Art-Net Monitor	This starts the PB tool "Art-Net Monitor 2051". As its version this depends on the installed Pandoras Box version, choose one via "Settings > Pandoras Box 2101".
Deltacast EDID Manager	This starts the Deltacast EDID Manager GUI.
Dongle Readout	This starts the Dongle readout tool which allows you to read information from your attached USB dongle or to write new information on it which is of interest when you updated to a new version for example or purchased another license.
Windows Task Manager	This opens the Windows Task Manager. It provides detailed information about computer performance and running applications, processes and CPU usage, commit charge and memory information, network activity and statistics, logged-in users and system services.
Kill Pandoras Box	This ends the process of Pandoras Box without asking to save a projects.
Kill Widget Designer	This ends the process of Widget Designer without asking to save a projects.
LogFiles	This opens two Windows Explorer windows to show the log files from Pandoras Box and Widget Designer. Select the PB and WD version first using the "Settings 2101" menu.

9.8.2 Settings Menu

This chapter describes the Settings Menu from the PB Menu [2097]. It opens when the "Settings" button is clicked and is divided into the following tabs.

Pandoras Box



In the "Pandoras Box" tab you can set up how Pandoras Box starts when the "Pandoras Box" button is pressed in the main PB Menu screen or when the Autostart function is switched on.

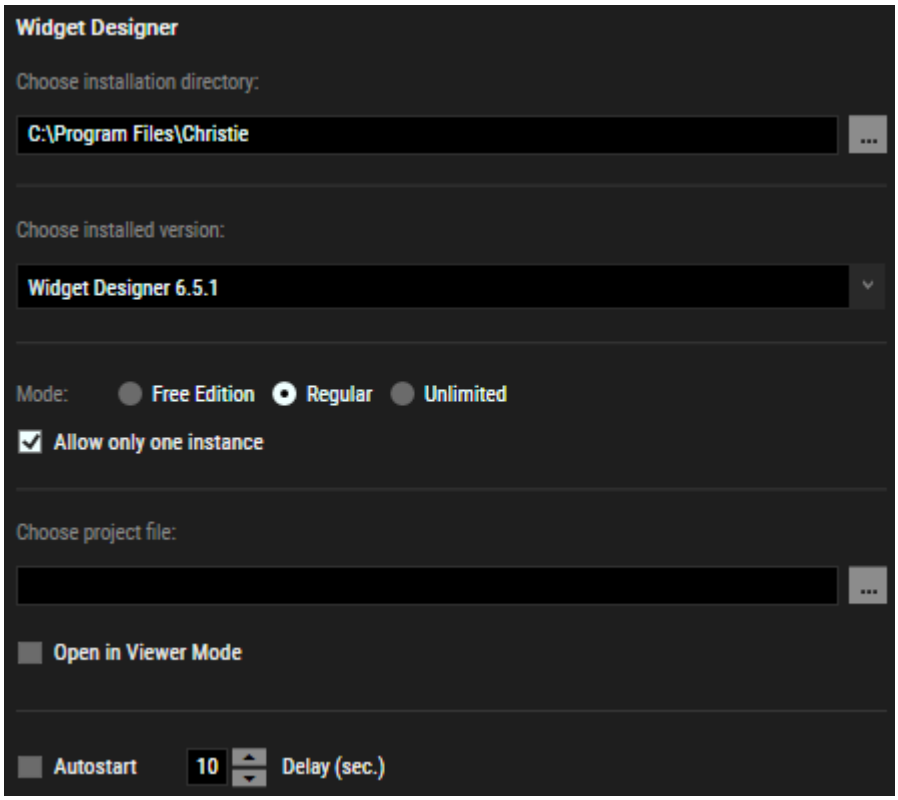
In case you have changed the default installation path C:\Program Files\Christie when installing Pandoras Box, you can change the "Installation Directory" with the [...] button to the new location.
The drop-down list below, offers all Pandoras Box versions installed there. Choose the version and decided whether it should start in Master or Client mode.
In other words: If you downloaded and installed a new version, simply choose this version in the drop-down list. In return, you can also choose another older version if you like to start it, e.g. if you have a show file saved in this version and you want to load it with this version too.

Per default, the Domain is set to be 0 and should only be changed if necessary. For more information, please see the chapter Configuration > Network [147] where you can change the Domain in a PB Master when already started. A PB Client [316] also offers an according "Domain" text field.

If you like to start Pandoras Box automatically each time the system is started or rebooted, check the Autostart "Enabled" check-box. Per default, there is a delay of 10 seconds (after starting the PB Menu) to ensure that the computer finished the boot process and all USB devices (especially the dongle) are ready.

The "Check Process" option checks whether the Pandoras Box process replies and restarts the application if there is no response for 60 seconds. If you like to use this feature, activate the option and start Pandoras Box via the Menu, which means via the Autostart feature or the Menu button "Pandoras Box".

Widget Designer



In the "Widget Designer" tab you can set up how Widget Designer starts when the "Widget Designer" button is pressed in the main PB Menu screen or when the Autostart function is switched on.

In case you have changed the default installation path C:\Program Files\Christie when installing Widget Designer, you can change the "Installation Directory" with the [...] button to the new location.
The drop-down list below, offers all Widget Designer versions installed there. Choose the version and decide which edition should start: Free, Regular or Unlimited.
In other words: If you downloaded and installed a new version, simply choose this version in the drop-down list. In return, you can also choose another older version if you like to start it, e.g. if you have a show file saved in this version and you want to load it with this version too.

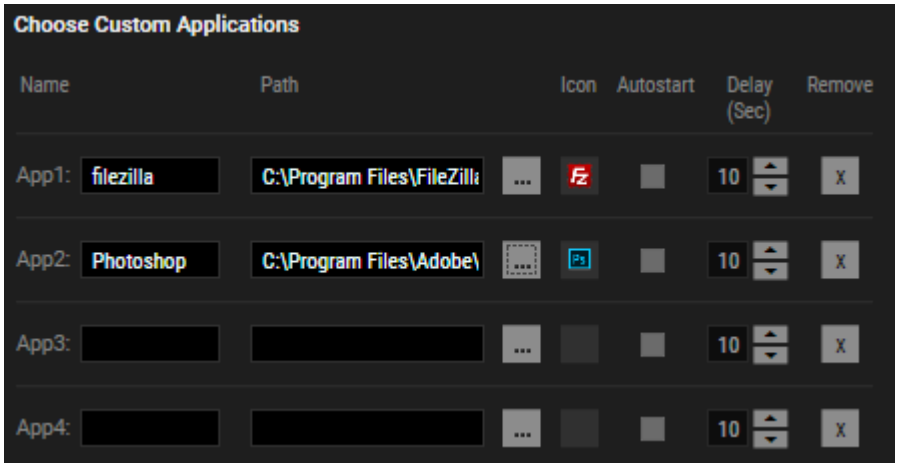
The option "Allow only one instance" is checked per default which means that Widget Designer cannot be started multiple times by pressing the "Widget Designer" button on the main page.

In addition, you can choose a project file that should be loaded. Choose a new project path by clicking on the [...] button to add it to the drop-down list.

The check box "Open in Viewer Mode" starts the chosen edition in the Viewer mode [787], which restricts the user from changing the project.

If you like to start Widget Designer automatically each time the system is started or rebooted, check the "Autostart" check-box. Per default, there is a delay of 10 seconds (after starting the PB Menu) to ensure that the computer finished the boot process and all USB devices (especially the dongle) are ready.

Custom Apps



In the "Custom Apps" tab you can choose four additional quick access buttons that should be displayed in the main PB Menu screen. This can be also done by clicking the "Add Custom App [2097]" button.

Click the [...] button to choose your file, e.g. an executable file like a EXE or BAT file and click "Open".

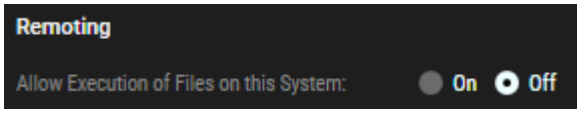
Icons are automatically retrieved from the app. Click the "Icon" button to choose a custom icon. Regular images as well as *.ico are allowed.

If you like to launch the action automatically each time the system is started or rebooted, check the "Autostart" check-box. Per default, there is a delay of 10 seconds (after starting the PB Menu) to ensure that the computer finished the boot process and all USB devices (especially the dongle) are ready.

Change the "Name" if you like to change the text that is displayed in the customized button.

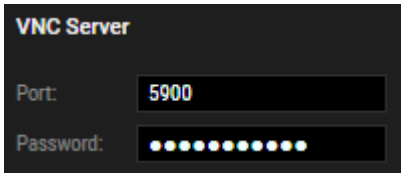
App buttons are filled from left to right. If a path is left blank, the entry is simply ignored.

Remoting



In the "Remoting" tab you have one option "Allow Execution of Files on this System". Per default, it is set to "Off", which means that the PB Menu will not trigger any actions send via the command "PBApplicationStart [1566]" from Widget Designer.

VNC Server



The "VNC Server" tab allows to setup the VNC Server settings.

The default port for the VNC Server is 5900, change it only if necessary.
The password is "PandorasBox" per default. It can be changed, or left empty.

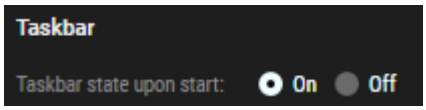
In the Server Management Application [2109], you can use the syntax "IP:port", e.g. "2.0.0.1:5905" to specify another port than 5900.

Info Section

The bottom part of the main screen shows network and disk information. Per default, PB simply chooses three active network adapters (in the order as they are delivered from the system) and either the C: drive or preferably the folder "`C:\Christie\content`" or "`C:\coolux\content`".The "Info Section" tab allows to change the shown **Network Adapters** and **Disk Space Locations.**

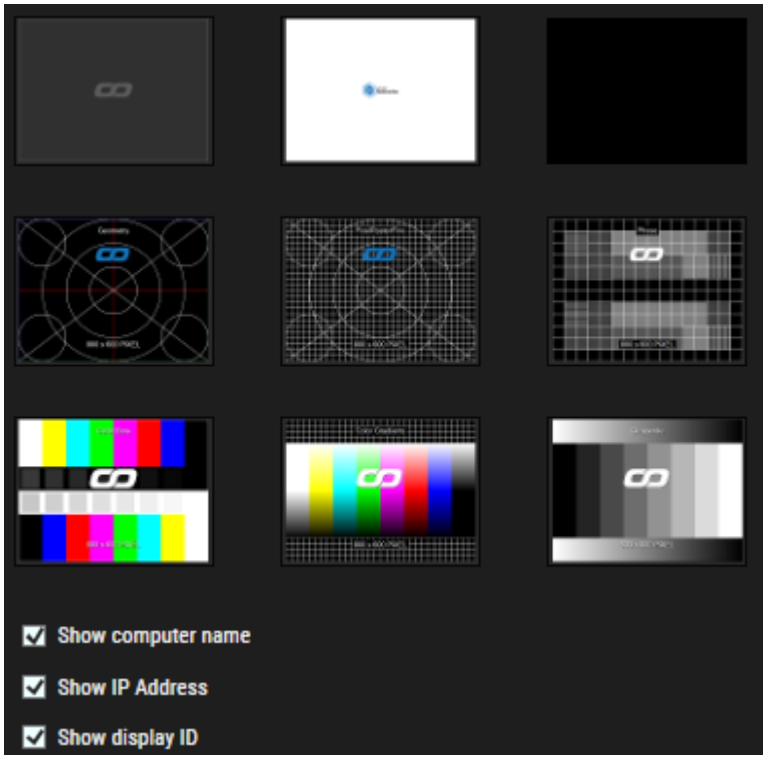
To change network addresses etc., open the section "Network Setup [2107]".

Taskbar



The "Taskbar" tab shows one option "Taskbar state upon start". Per default, it is set to "Off", which means that the Windows taskbar and desktop are hidden because the PB Menu overlays it.

TestPattern

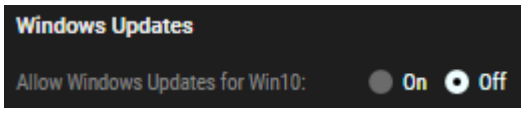


The "TestPattern" tab allows to choose a testpattern that is displayed as a fullscreen background on all connected (single) displays. This is especially interesting when setting up displays or projectors and a testpattern is needed but neither the network nor the Pandoras Box master are setup yet. You can also choose a testpattern using the LCD Menu on PB Server hardware.

Test-patterns are automatically created on-the-fly when starting the system, changing resolutions or opening the Testpattern dialog. Hence, they always match current and possible arbitrary resolutions. The images are saved under `C:\Christie\content\Stock Assets\Testpattern` (or `C:\coolux`...). Note that this will not overwrite existing testpattern which need to be deleted manually to have new ones generated.

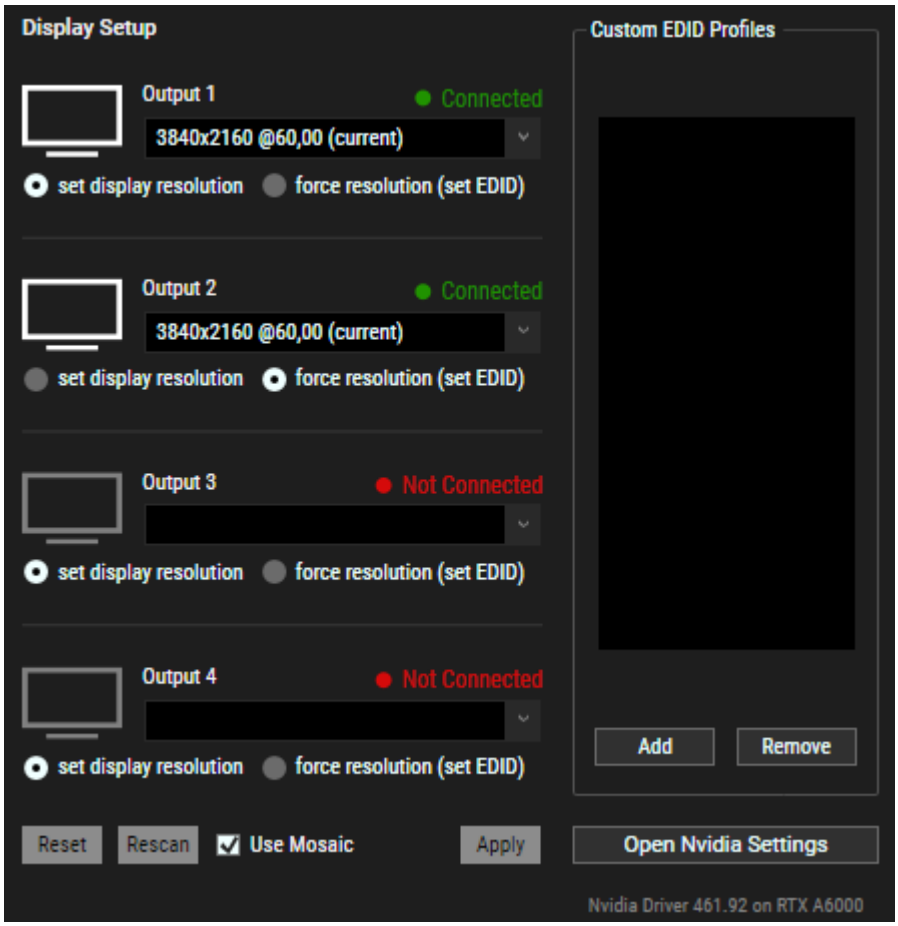
The three check boxes below allow to hide / show further information regarding the Output ID, IP address and Display ID which is shown with some testpattern.

Windows Updates



In the "Windows Updates" tab you have one option "Allow Windows Updates for Win10". Per default, it is set to "Off" which means that automatic updates from Windows 10 are deactivated with the installation of the PB Menu. This is due to the high number of supported issues with hardware with automatic updates. We recommend to work with our hardware as it was delivered and to use Media Servers in their own network without permanent internet connection. Hence, deactivating those updates is not a threat. If you like to activate them again, simply choose "On"- The change takes effect when restarting the computer. You might need to restart it twice to actually start the update process.

Display Setup



The "Display Setup" tab allows to setup display output settings.

You can use the PB Menu directly, if you have a Quadro M4000, P1000, P4000, RTX 4000, RTX 6000 graphics cards from NVIDIA. Otherwise click the "Open Graphics Card Settings" button or use the "Screen Resolution" dialog from Windows.

Note that testpattern are automatically created when changing the output resolution. They are available with the Testpattern tab under Settings > TestPattern [2105].

The "Display Setup" tab shows the current display configuration (resolution and frame rate) of all connected displays. The numbering corresponds to the output port numbering of NVIDIA. This means that "Output 1" refers to the lowest port when looking at the graphics card and "Output 4" refers the topmost one.

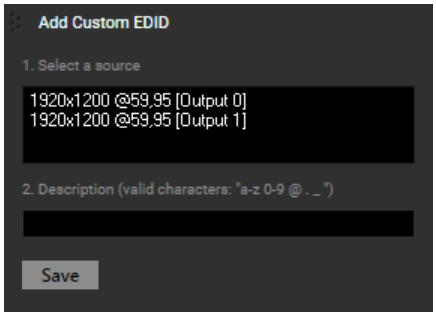
Choose how you like to set a resolution. With the option "**Set Resolution**" the drop-down list shows all timings that the display communicates to the graphics card including its preferred timing. Keep in mind that it is possible that devices switch to another timing when they are dis- and reconnected or when the signal chain is interrupted.

With the other option "**Force Resolution**" the drop-down list shows standard EDID timings and custom EDIDs (described further down). When applying a timing, the EDID information sent from the monitor has no effect anymore. The used EDID is now based on the information written in a special edid file. Please make sure, that you only pick a timing that is supported by the display device.
Managing the EDID in this way has several advantages. Sometimes the display EDID cannot be received due to switchers and / or cable extenders. Also, it is safer during hot plug events especially when they would reconfigure your display. Lastly, with the EDID being read from file the operating system will always think the display is attached and you can even fake a display being attached for testing.
By the way, the NVIDIA settings will show what option you have chosen here. In the "System topology" it will say "EDID (Monitor)" for the first possibility and "EDID (File)" for the second one.

In summary it can be said that using an edid file is a safer way to setup up timings as they are more reliable and can not change that easily. An even better way is to use external EDID managers which generate and hold an EDID constantly. Therefore the graphics card really thinks that there is a monitor attached that sends the (emulated) EDID. In addition they cannot be reset as easily by mistake or by the need to reinstall the graphics driver.

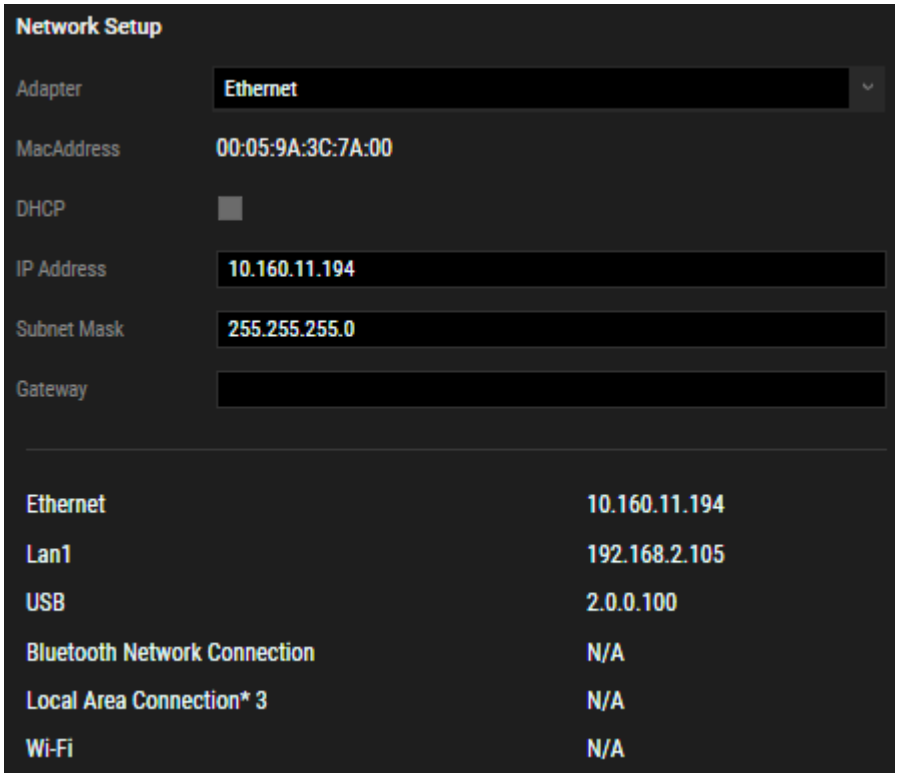
Confirm your settings with the "Apply" button. This will apply the timings to the display devices and possibly rearrange the desktop layout so that the output with the lowest number is on the left side of the desktop and the monitor with the highest number is on the right side. Tick the check box "Use mosaic" if you like to set the NVIDIA mosaic mode for all displays. For more information, please see the description in the Graphics Card Settings [1948] chapter under Multiple Displays [1952].

The other buttons at the bottom are pretty self-explanatory: "**Open Nvidia Settings**" opens the according dialog in case you like to check there something or use it instead of our menu. "**Reset**" discards all forced EDIDs and sets all displays to the "Set Resolution" mode again. The "**Rescan**" button updates the displayed settings to the current ones reported from the NVIDIA graphics card.



Lastly, the right side from the dialog allows to save and delete custom EDIDs. Click "**Add**" and the depicted "**Add Custom EDID**" dialog opens. Here, the top shows the current timings reported from the connected outputs. Choose one, enter a name (using valid characters: a-z 0-9 @._) and save the EDID. The custom EDID will be displayed on the right side (just for your overview). When you now choose the "force resolution" mode for any output, your custom EDID will be listed in the drop-down menu.

Network Setup



The "Network Setup" tab allows to set individual IP addresses for Ethernet adapters.

First, choose the network adapter's name in the "Adapter" drop-down menu. With the check-box below you can setup whether DHCP is enabled for the selected adapter. If there is no DHCP server in the network, disable the

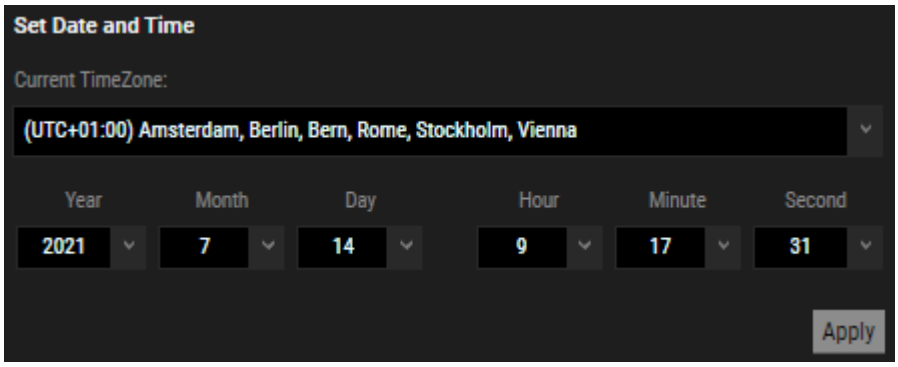
option and enter a static "IP Address", e.g. 2.0.0.1, a "Subnet Mask", e.g. 255.0.0.0 and a "Gateway" if required, e.g. 2.0.0.100.

At the bottom of the network dialog, you see available Lan ports with their according IP address. The PB Menu itself displays three ports on the main page which can be chosen in the tab "Info Section [2104]".

Raid Controller

This works only on Pandoras Box hardware and opens the installed RAID tool or its web interface via the Windows Internet Explorer. The tool allows collecting information about the computer's raid system and gives the possibility to manage it. Please note: Do not touch the raid system settings unless you are well skilled in doing this. If there are hard drive / raid problems, please contact Support.pandorasbox@christiedigital.com or your local distributor for support!

Date/Time



The "Date/ Time" tab allows to set up the timezone, date and time for the local system. It might be of interest that the Server Management Application [2109] offers date settings influencing all systems in the network.

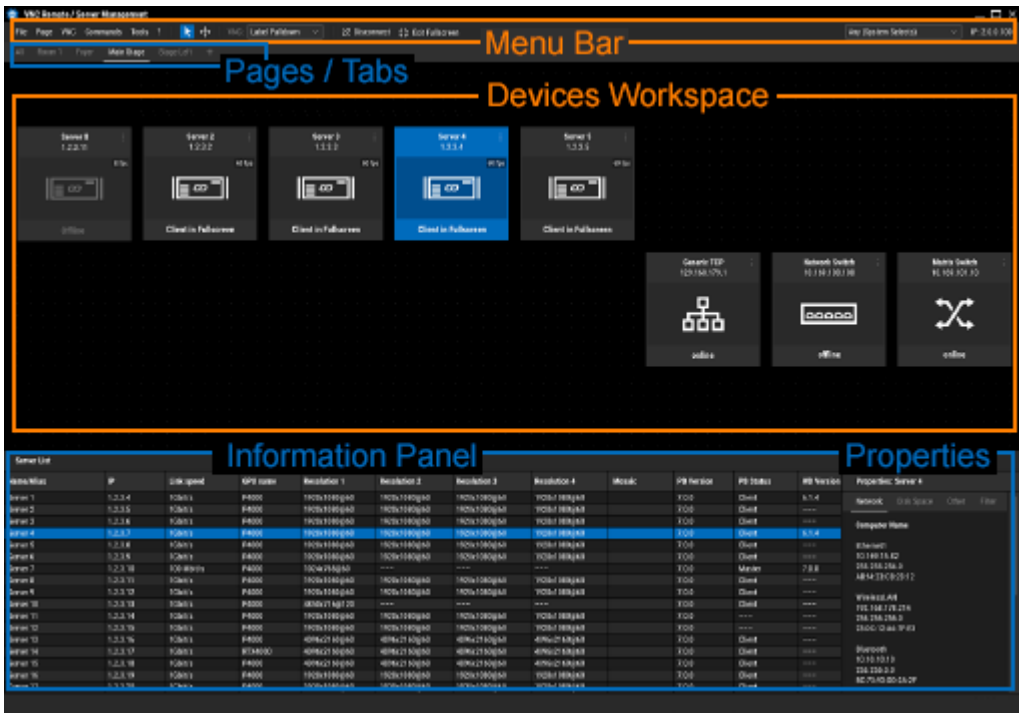
9.9 Server Management Application

The Server Management Application allows you to manage multiple hardware systems and to keep track of their status. As seen in the image below, you can depict computers and other devices as icons on a page showing necessary information and status updates. This provides a good overview of important devices in the network.

Furthermore you can access your entire system from a single point for updating Pandoras Box [67] or Widget Designer [786] along with gathering log files of all connected machines instead of manually remoting into each system, one after another. Read more about this in the Tools Menu... [2112]

As the former "VNC Remote" the new Server Management Application offers a dedicated VNC connection that allows you to remote control another computer. You can connect to any computer in the network that launched the corresponding VNC Server, which automatically runs in the background when the PB Menu is started. You simply enter the according IP address and see the desktop of that computer in a window. Now you can use your local mouse and keyboard to interact with the remote desktop.

The Server Management Application is part of the PB Menu [2097] but is also available as a stand-alone version in the Download-Center on our homepage. Please make sure that the PB Menu is installed on all computers that should be manged in order to transmit information properly. The PB Menu receives request from the Server Management Application and sends information back.



Menu Bar
The Menu Bar shows various menus and VNC controls that allow to start or stop a VNC connection to a certain IP address. On the right side, a drop-down list lets you to choose the network for the Server Management Application and the according IP address is displayed. More info... [2110]

Pages
The first page "Active" shows all systems with a running PB Menu that are online. The Pages Bar allows to add more pages to the project if you like to organize your systems more. Each page switches to a new Devices Workspace with its according icons. More info... [2116]

Devices Workspace
The Workspace can be filled with devices or hardware systems which are depicted as rectangular icons showing a specific image surrounded with information. More info... [2116]

You can choose various icons of Pandoras Box hardware, but also generic or custom icons.

The icon shows the following information if applicable:
- Alias name
- IP address
- online state = white icon ; offline state gray icon
- Render engine frequency and state (e.g. Client in Fullscreen) in case Pandoras Box runs on this system

All icons can be placed into "slots" or aligned to a grid.

Information Panel
The lowest section of the user interface is the Information Panel or "Server List". It shows system information in a spreadsheet style for all devices belonging to the displayed page. To the right, a "Properties" tab shows even more information of a selected device.

The panel can be enlarged to cover the entire screen, minimized to a single line or placed at any height on the screen. More info... [2118]

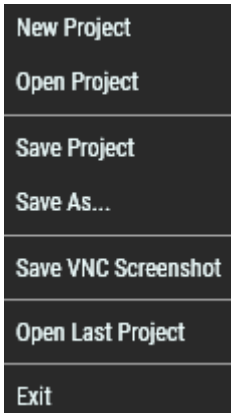
9.9.1 Menu Bar

This chapter describes the Menu Bar of the Server Management Application [2109].

The Menu Bar shows:

- various menus on the left side: File [2110], Page [2111], VNC [2111], Commands [2111], Tools [2112]
- VNC controls [2115] that allow to start or stop a VNC connection to a certain IP address
- a drop-down list that lets you to choose the network [2116] for the Server Management Application and the according IP address.

File Menu



New Project
This opens a new empty project.

Open Project
This opens a file dialog where you can pick an already saved project with the file extension .json.

Save Project
This saves the project. When saved for the first time, a file dialog opens to choose a name and pick a location to save the current project.

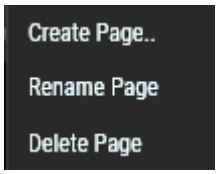
Save As...
This opens a file dialog to choose a (new) name and pick a (new) location to save the current project.

Save VNC Screenshot
This command is only available if you established a VNC connection to another computer. It opens a file dialog to choose a name and pick a location to save a screenshot from the remote desktop.

Open Last Project
Check this option if the Server Management should open the last opened project (which was also saved) whenever it starts again.

Exit
This closes the current project and application. Remember to save the project first if necessary.

Page Menu

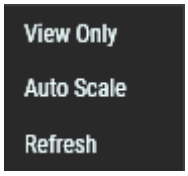


Create Page
This opens a dialog to specify a name and adds a new page tab afterwards. Alternatively, you can click the +-icon in the Pages Bar [2116].

Rename Page
This opens a dialog to re-specify the page name.

Delete Page
This deletes the current page.

VNC Menu



View Only
This influences the VNC connection. Per default, you see the remote desktop in a window and send your local mouse and keyboard data to interact with the remote desktop. The "View Only" option stops the mouse and keyboard data. To send them again, untick the option once more.

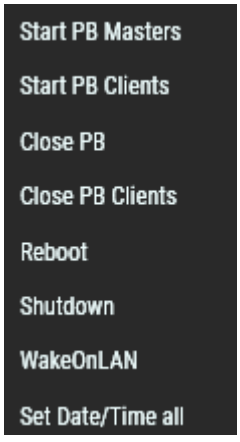
Auto Scale
Per default, the remote desktop is transmitted with the native resolution; it is not scaled to fit into the local VNC window. Check the option and the remote desktop is scaled to fit horizontally or vertically by keeping its aspect ratio.

Refresh
This refreshes the current VNC Connection.

Commands Menu

The Commands Menu is separated into three sections:

Commands from the "Systemwide" menu are sent to all systems in the network running the PB Menu.
Commands from the "Current Page" menu are sent to all systems from the current page running the PB Menu. This is only activated when no VNC connection is established.
Commands from the "Connected Device" menu are sent only to the single remote device to which a VNC connection is currently established. Hence the options are only available when a VNC connection is established.



Commands Menu > Systemwide > ... or Commands Menu > Current Page > ...

Start PB Masters
Launches PB in Master mode on applicable systems where Master is selected in the PB Menu settings [2101].

Start PB Clients
Launches PB in Client mode on applicable systems where Client is selected in the PB Menu settings. [2101]

Close PB
Stops both, the PB Master and Client process on applicable machines (without saving the project!)

Close PB Clients
Stops only PB Client processes on applicable machines.

Reboot
Reboots all machines where PB Menu is installed.
If the command was chosen from the "Systemwide" menu, it includes the PC running the Server Management Application!

Shutdown
Shuts down all machines where PB Menu is installed.
If the command was chosen from the "Systemwide" menu, it includes the PC running the Server Management Application!

WakeOnLAN
This wakes all machines via WOL if their MAC Addresses were stored in the system information and WOL is supported.

Set Date/Time all
This is only available from the "Systemwide" menu and opens the "Date/Time Settings" dialog to adjust the timezone, date and time for all systems.

Commands Menu > Connected Device > ...

Start PB
Starts PB in the mode which is selected in the PB Menu [2101].

Close PB
Stops the current PB Master or Client process.

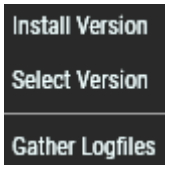
Start WD
Starts WD in selected version as specified in PB Menu. [2102]

Close WD
Stops the current WD process.

Reboot
Reboots the system.

Shutdown
Shuts down the system.

Tools Menu

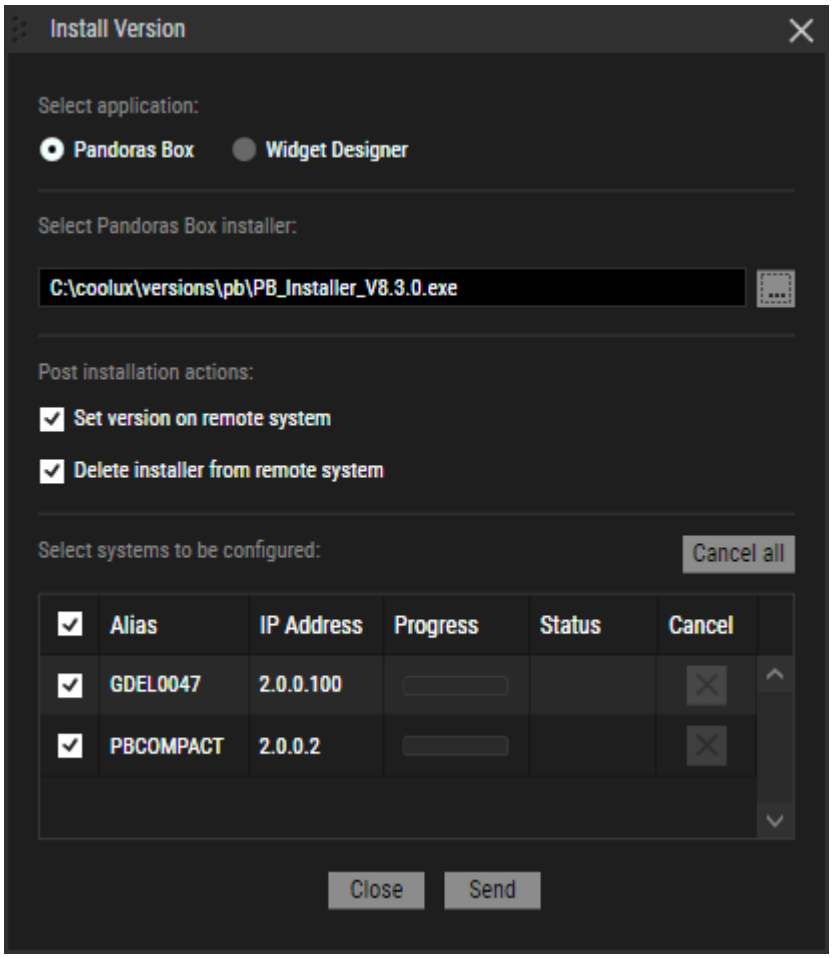


Install Version
This opens a new dialog where you can choose to install Pandoras Box or Widget Designer on multiple devices. See below [2112].

Select Version
This opens a new dialog where you can select - for multiple devices - which version (or other starting options) of Pandoras Box or Widget Designer should be chosen. See below [2113].

Gather Logfiles
This opens a new dialog where you can select from which systems you like to gather log files. See below [2114].

Install Version
This opens a new dialog where you can choose to install Pandoras Box or Widget Designer on multiple devices.



First, you choose the installer with the [...] button. You can select Pandoras Box 8.3 or above or Widget Designer 6.5.2 or above.

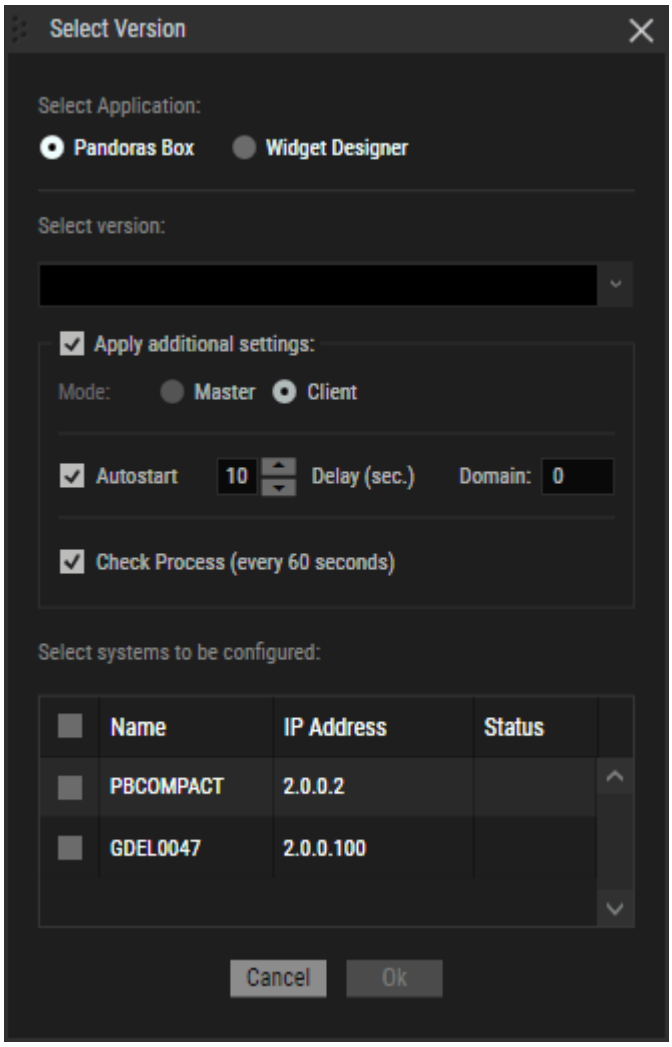
The option "Set version on remote system" selects the newly installed PB or WD version in the "Settings [2101]" menu from the remote PB Menu. Whenever you start PB or WD then, it will be the new version.
The option "Delete installer from remote system" removes the installer when the update process is finished.

Below, you mark the systems to be updated. By the way, if you select devices in the Workspace [2116] and open the dialog from the right-click menu, only these devices will be marked.

When the "Execute" button is clicked, the "Progress" and "State" column show information for each system:
Request Sent - Checks or creates target path (either `C:\Christie\content\versions` or respective "coolux" folder, according to logic of PB's file spreading [193])
Sending installer - Copies installer via network
Installing - Executes installer remotely
Done - Update process is finished including above described post-installation actions
Failed - Update process not finished. Possible errors could be: not enough disk space for the installer or application, network connectivity, firewall settings etc.
Canceled - User clicked "Cancel" button

In case you like to remote install the Offline License, e.g. for a training center, simply add "_Offline" to the name so that it says for example: PB_Installer_V8.2.0_Offline.exe

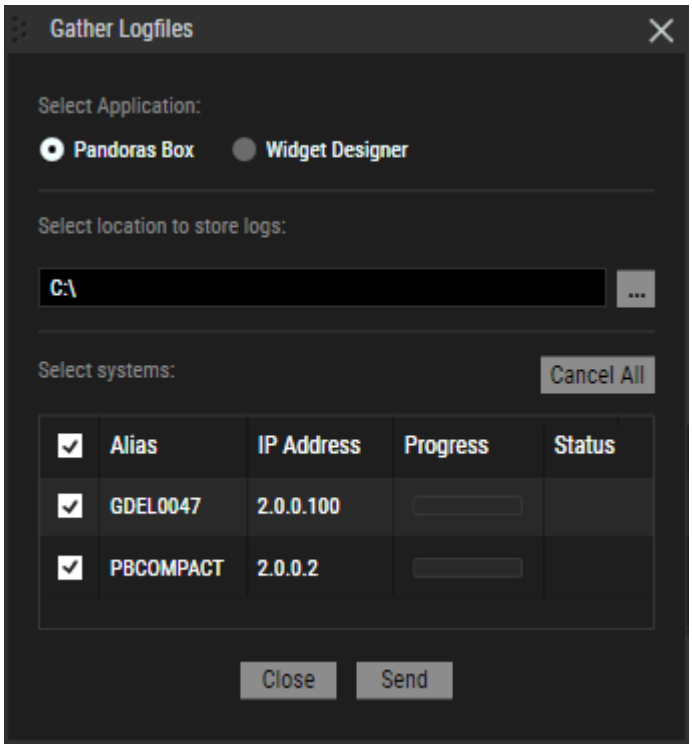
Select Version
This opens a new dialog where you can select - for multiple devices - which version (or other starting options) of Pandoras Box or Widget Designer should be chosen.



Essentially, this dialog offers all settings which you also find in the PB Menu under Settings > Pandoras Box [2101] or Settings > Widget Designer [2102]. Hence, all options are explained in the linked chapters. The huge advantage is, that you set those starting options from one single dialog. First, select the version for PB or WD. Then, activate the "additional settings" in case you like to overwrite the settings from the remote PB Menu settings. Last, mark your devices in the list and hit the "Execute" button. Note that this does not start PB or WD on the selected systems, it just sets *how* they start the next time when the applications are called.

By the way, if you select devices in the Workspace [2116] and open the dialog from the right-click menu, only these devices are marked.

Gather Logfiles
This opens a new dialog where you can select from which systems you like to gather log files.

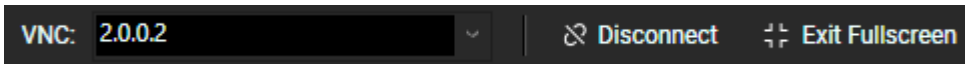


First select whether you like to gather the log files of Pandoras Box or Widget Designer. Then select a location and name for the final zip file and mark from which devices the log files should be requested. When pressing "Execute" the "Status" column displays current information for each device. When all files are stored in sub folders named with the IP address, the file browser opens and displays the final zip file.

By the way, if you select devices in the Workspace [2116] and open the dialog from the right-click menu, only these devices are marked.

Info: As log files from Pandoras Box and Widget Designer are saved in folders depending on the running version, please make sure, that this version in question is selected in the Settings Menu [2101] from PB Menu. For this you can also use the above described "Select Version" dialog. If no version is selected, no log files can be gathered which is displayed in the "Status" column.

VNC Controls



To connect to a remote computer, you do not have to add it to the workspace first. Simply enter or choose an IP address for the "VNC" IP entry field and press [Enter]. You might need to enter a password to establish the VNC connection. The default password is "PandorasBox". You can change it in the PB Menu under Settings > VNC Server [2103]. If you leave the password blank, you can establish a VNC connection without the need to enter a password.

Click the "Disconnect" button to cancel the current connection.

Click the "Exit Fullscreen" button to send a "leave fullscreen" command to the remote computer. This makes sense if the remote machine runs Pandoras Box and renders in full screen mode. The fullscreen mode cannot be captured via VNC, hence you see the desktop area underneath the Pandoras Box application but you cannot select there anything.

Network Connection

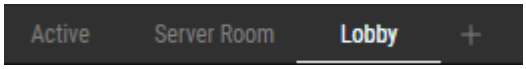
If necessary, select a specific NIC from the drop-down list at the right side of the Menu Bar. This network adapter is then used to send commands and establish connections. The according IP address is shown next to it.

The next chapter describes how to work with "Pages and Workspace" [2116].

9.9.2 Pages and Workspace

This chapter describes the the Pages Bar and Workspace of the Server Management Application [2109].

Pages



The Pages Bar shows all available pages as tabs for quick access and overview.
Click on the +-icon to create a new page with an empty workspace. A dialog opens to specify a name.
Alternatively, you can create a new page via Pages Menu [2111] > Create Page. The new tab will be added to the far right after existing tabs. You can drag and drop tabs to resort them. Please use the Pages Menu [2111] if you like to rename or delete pages.

Per default, the first tab, called "Active", shows all systems that are currently available in the network and that are running the PB Menu [2097]. The systems are listed in the workspace in ascending order of their IP addresses and hence cannot be resorted. The icons show the device name and not their alias name. If a device goes offline, it will be removed from this screen after 30 seconds. New devices will sort in the list according to their IP address. The Active tab cannot be deleted or renamed.

Workspace

The Workspace can be filled with devices or hardware systems which are depicted as rectangular icon showing a specific image surrounded with information.

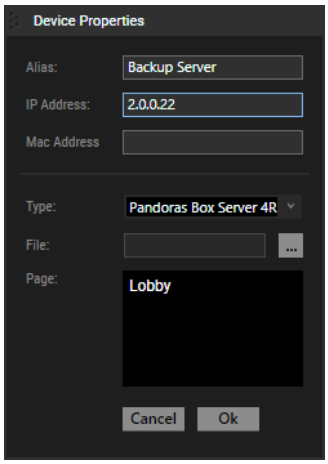
To add a new device, simply right-click into the empty Workspace background to open the context menu and choose a device. After specifying details (as described below [2117]), it will be placed in the slot where the context menu was triggered.

Icons are either white which indicates an online state or gray which indicates an offline state. The status is determined based on communication with the PB Menu in case of PB Devices or a successful ping-reply in case of TCP devices which is inquired every 10 seconds.
While TCP Devices can only be online/offline, PB devices can have the following state: Master Windowed, Master in Fullscreen, Client windowed, Client in Fullscreen

You can also copy a device to another page, e.g. from the "Active" tab. Simply right-click an icon and choose "Add to Page" from the context menu, or "Move to Page" respectively.

You can select an icon with a left-click or make a multi-selection via [Ctrl] or [Shift] + left-lick . Alternatively, select devices in the below Information Panel [2118]. Selected devices are highlighted in blue in the Workspace and Information Panel.

On custom pages, a selected device can be moved to another empty slot within the workspace.



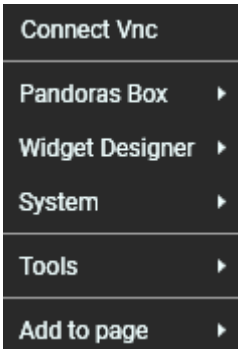
Alias
Enter a name that should be displayed underneath the icon.

IP Address
Enter the IP address from the computer you like to add to the page.

Mac Address
Enter the Mac Address if you like to use "WakeOnLan" commands later.

Type & File
The type is simply the icon that either depicts a tower PC, or a PB Server etc. You can also choose "Custom" and a custom image with the [...] button next to "File". Note that an icon is scaled up or down to fit the icon size of 150x130px. In addition, it makes sense to choose an image with a transparent background so that the offline/online status is better visible and bear in mind that the information text added to the top and bottom of the image has a white font.

Page
This field shows all pages where the device is present.



This context opens, when a PB device is right-clicked.

Connect Vnc
This establishes a VNC connection which can also be triggered with a double-click as described above.

Pandoras Box > ...
Start PB
Starts PB in the mode which is selected in the PB Menu.

Close PB
Stops the current PB Master or Client process.

Enter Fullscreen
This sends a "enter fullscreen" command to the remote PB device.

Leave Fullscreen
This sends a "leave fullscreen" command to the remote PB device.

Widget Designer > ...
Start WD
Starts WD in selected version as specified in PB Menu.

Close WD
Stops the current WD process.

System > ...
Reboot
Reboots the system.

Shutdown
Shuts down the system.

WakeOnLAN
This wakes the device via WOL if the MAC Address was stored in system information and the device supports WOL.

Reset Menu
This closes and restarts the PB Menu.

Reset VNC
This closes the according "VNC Server" process and restarts it again.

Tools > ...
Install Version
This opens a dialog where you can choose to install Pandoras Box or Widget Designer on multiple devices. See previous page 2112.

Select Version
This opens a dialog where you can select - for multiple devices - which version (or other starting options) of Pandoras Box or Widget Designer should be chosen. See previous page 2113.

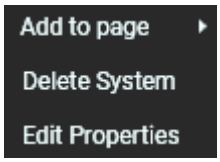
Gather Logfiles
This opens a dialog where you can select from which systems you like to gather log files. See previous page 2114.

Add to page
This adds the device to the first available slot on the selected page.

Move to page
This moves the device to the first available slot on the selected page.

Delete System
This deletes the system from this page. Other pages remain untouched.

Edit Properties
This recalls the "Edit Properties" dialog of the device.



This context opens, when a TCP device is right-clicked.

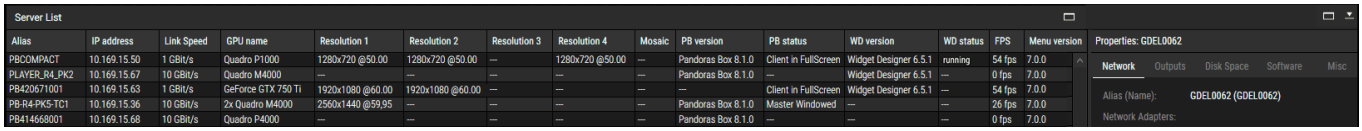
Add to page
This adds the device to the first available slot on the selected page.

Delete System
This deletes the system from this page. Other pages remain untouched.

Edit Properties
This recalls the "Edit Properties" dialog of the device.

9.9.3 Information Panel

This chapter describes the the Information Panel of the Server Management Application 2109.



The Information Panel shows additional system information in a spreadsheet style for all devices belonging to the displayed page. To the right, the "Properties" tab displays verbose information if one device is selected. It lists miscellaneous information about all network adapters, details about the graphics card settings including the used driver and shows remaining as well as the total space of RAM and hard drives.

The panel can be enlarged to cover the entire screen, minimized to a single line or placed at any height on the screen. Buttons are available for these actions.

A click onto a line in the table highlights this entry in blue and also selects the according icon in the Workspace and vice-versa. You can also make a multi-selection via [Ctrl] or [Shift] + left-lick.

The following information is displayed:

Alias
Shows the alias when on user pages; shows computer name when on "active" page.

IP Address
Shows the IP address.

Link Speed
Shows the speed of the network connection.

GPU name
Shows the GPU name.

Resolution 1
Shows resolution of Output 1, even when in mosaic. Only for supported NVIDIA graphics cards (i.e. Quadro M4000, P1000, P4000, RTX 4000, RTX 6000).

Resolution 2
Shows the resolution of Output 2, even when in mosaic mode.

Resolution 3
Shows the resolution of Output 3, even when in mosaic mode.

Resolution 4
Shows the resolution of Output 4, even when in mosaic mode.

Mosaic
Shows "x" when the mosaic mode is set; nothing when not set.

PB Version
Shows the PB version that is selected in the remote PB Menu; "---" when none is selected.

PB Status
Shows the same status as on device workspace (Master windowed, Master in Fullscreen, Client windowed, Client in Fullscreen, "---").

WD Version
Shows the WD version that is selected in the remote PB Menu; "---" when none is selected.

WD Status
Shows the status of WD ("---" or "running").

FPS
Shows the frame rate reported from Pandoras Box.

PB Menu
Shows the Menu version; "---" for TCP Devices or when not available.

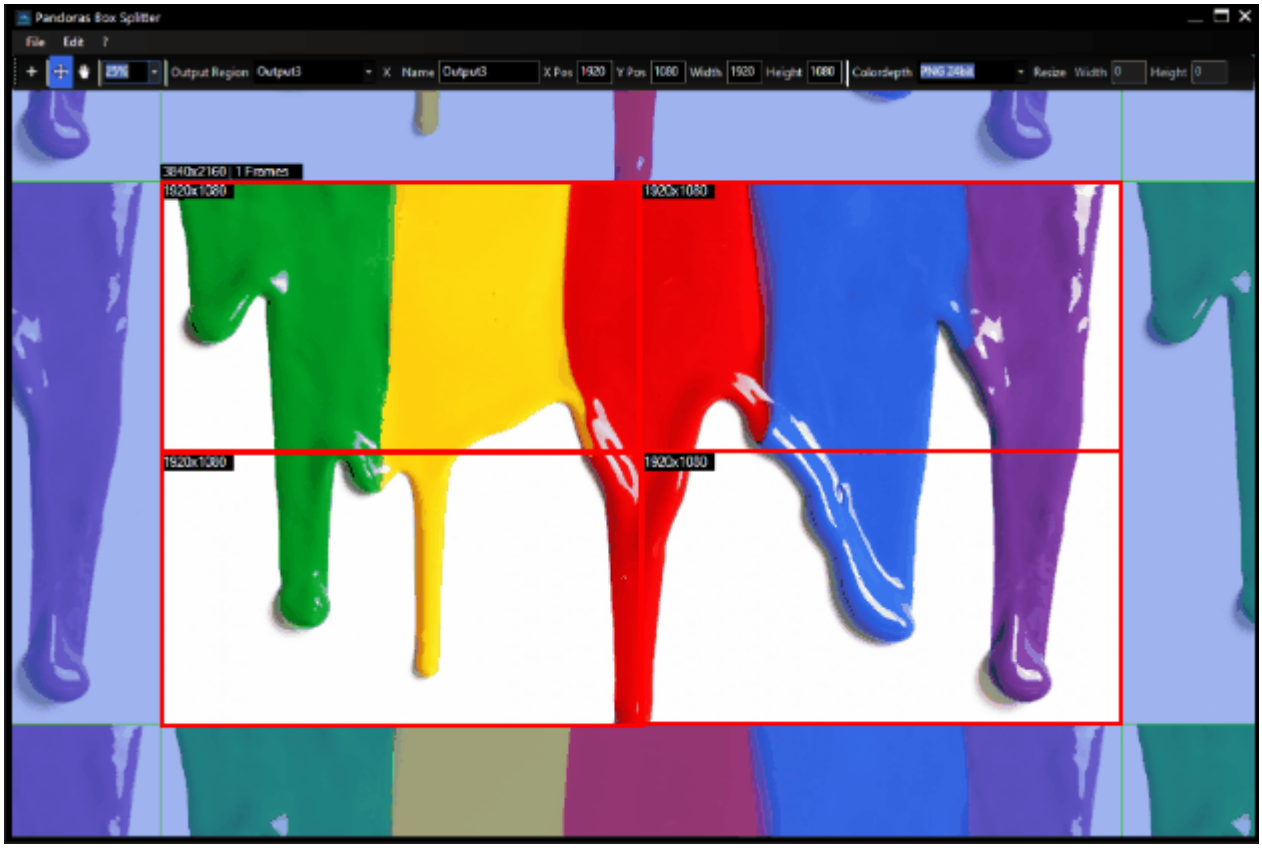
9.10 Splitter

The Splitter is a tool for splitting one big master-file into several smaller split-files for Multi-Display or Multi-Projection purposes. This allows a good performance and capacity on the playback systems, with native playback of Video Content of nearly any size.

The tool is installed automatically when you install a Pandoras Box application. To open it, you can either choose it from the Tools menu [122] from your Pandoras Box master software or open it stand-alone (on the Master or Client), e.g. by choosing it from the Pandoras Box folder in the Windows Start menu. The installation path is for example:
`C:\Program Files\Christie\Pandoras Box 8.0.0\data\tools\PandorasBoxSplitter.exe`

Using the Splitter primarily makes sense when the content has to be created before the technical setup is defined, e.g. without knowing how many projectors will be used or how big the overlap between the projectors will be. This way the content producers may work independently from the technical setup. Only the total image resolution has to be defined.

The User Interface is divided into the File Menu [2121], the Edit Menu [2123], the Toolbar and main Window [2125]. The Splitter Workflow [2127] is summarized on the last page of this chapter.



Input Formats

You can import a single image or a folder containing an image sequence. The maximum master file size is 65535 x 2900 px if using Win8 or above. The following list shows the supported input formats.

BMP, GIF, JPG, JP2, J2K, PNG, PSD, TGA, TIF(F), DDS or SNP

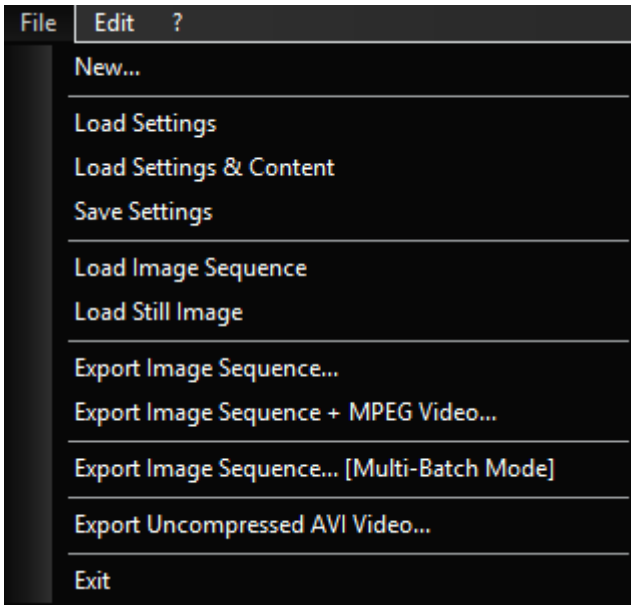
Output Formats

After setting up the output regions, the Splitter tool exports them as separate image sequences and/or videos. The following lists show the supported output formats.

MPEG, AVI as video formats
BMP 24/32bit (snp), PNG 24/32bit, DDS RGB(A) (snp), JPG as image formats

9.10.1 File Menu

This chapter explains the File Menu of the Splitter tool. Please see the introductory chapter for an overview about the Splitter tool [2120].



New…
This starts a new project. Please save your project first if you like to keep it. All Output Regions set up before will be removed in the new project.

Load Settings
This loads the settings from a previously saved Splitter project (.vsp) but does not load the original content (as the next option does).

Load Settings & Content
This loads the settings and the content from a previously saved Splitter project (.vsp). Please make sure that the file name has not changed and the content is still located under the same path which was used before.

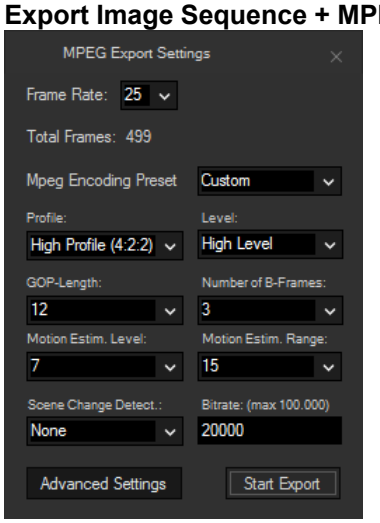
Save Settings
This saves your current Splitter project as .vsp file so that you can load the settings (and the content) later.

Load Image Sequence
Choose this option to load an image sequence as your master file in the current project. Next, setup the Output Regions and choose an "Export" option. If you created already some Output Regions, they are applied to the new content and not discarded.
The maximum master file size is 65k x 65k px if using Win8 or above. The following list shows the supported output formats: BMP, GIF, JPG, JP2, J2K, PNG, PSD, TGA, TIF(F), DDS or SNP

Load Still Image
Choose this option to load a still image as your master file in the current project. This is helpful, if you have a drawing, e.g. a testpattern showing all planned outputs. If you created already some Output Regions, they are applied to the new content and not discarded.
The maximum master file size is 65k x 65k px if using Win8 or above. The following list shows the supported output formats: BMP, GIF, JPG, JP2, J2K, PNG, PSD, TGA, TIF(F), DDS or SNP

Export Image Sequence…
Choose this option if you like to export an image sequence for each Output Region you created before. The image format should be set up beforehand in the Toolbar [2125]. A dialog opens where you can browse to an export folder. During the export, a sub folder is created automatically for each Output Region with the same name as in the Splitter, e.g. "Output0". A pop-up window informs you when the export is finished.

Export Image Sequence + MPEG Video…

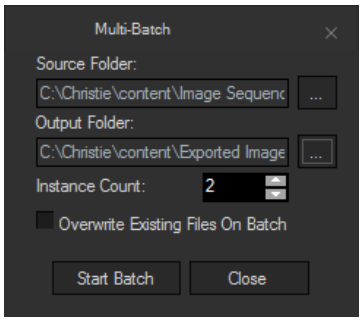


Choose this option if you like to export an image sequence and an additional MPEG video file for each Output Region you created before. The image format should be set up beforehand in the Toolbar [2125].

First, a dialog opens which allows to define the MPEG settings. The MPEG's source is the according image sequence, which therefore also defines the MPEG's resolution. Per default, MPEGs are limited to an output size of 1920 x 1080 px. However, our proprietary MPEG format "MXL" allows a size of 4094 x 2800 px, though it is recommend to set 4080 x 2800 px as this conforms with the MPEG standard. Make sure, that the image settings in the Toolbar [2125] are not larger than this. According to the MPEG specifications, the image width (in pixels) should also be divisable by 16 and the height by 8, see Edit Menu > Override Mpeg Size Limits [2123].

Choose a "Mpeg Encoding Preset", define a custom setting, or choose "Advanced" to unlock the button "Advanced Settings" at the bottom. The encoder is based on the Pandoras Box Encoder Extension. The chapter "Encoder Extension" [103] provides a detailed description for the encoder settings, including recommendations regarding the bitrate.

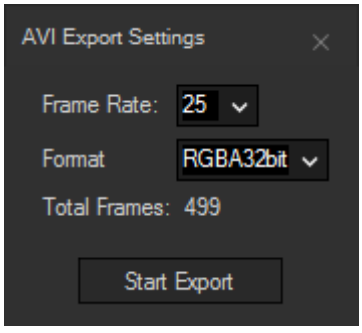
Click "Start Export" and a second dialog opens where you can browse to an export folder. During the export, a sub folder is created automatically for each Output Region with the same name as in the Splitter, e.g. "Output0", wherein the according images are saved. The MPEG videos are saved directly in the export folder, again with the same region name, e.g. "Output0.mpg". A pop-up window informs you when the export is finished.



Export Image Sequence…[Multi-Batch Mode]
Choose this option if you like to export an image sequence for each Output Region you created before. This option is recommended if you are splitting images with a high resolution (4K and above), because this quickly reaches the RAM limit of one instance under Windows.

A dialog opens where you can select the source and output folder and the number of Windows instances which will be used for the splitting process. Tick the check box "Overwrite Existing Files on Batch" when the existing files in the current output folder should be overwritten.

During the export, a sub folder is created automatically for each Output Region with the same name as in the Splitter, e.g. "Output0". A pop-up window informs you when the batch export is finished.



Export Uncompressed AVI Video…
Choose this option if you like to export an uncompressed AVI video file for each Output Region you created before. The video size depends on the image format settings in the Toolbar [2125].

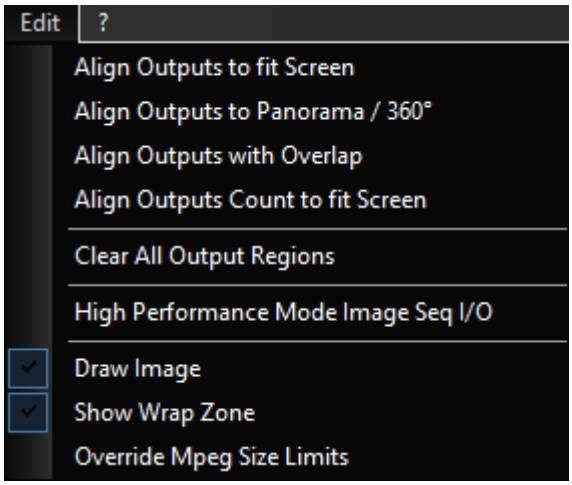
A dialog opens where you can browse to an export folder. Later, the AVI videos are saved directly in this folder with the same name as in the Splitter, e.g. "Output0.avi". A pop-up window will inform you when the export is finished.

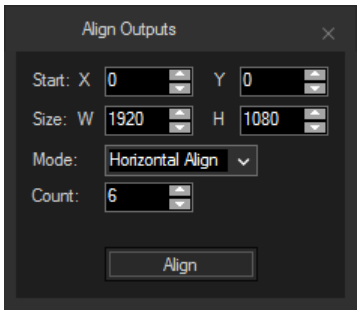
The upcoming "AVI Export Settings" dialog allows you to set the frame rate (24, 25, 30, 50 or 60 fps) and the format (RGBA32bit, RGB24bit, RGB555,RGB565,YUY2,UYVY or IYUV). It also displays the total amount of input frames.
Press [Start Export] to render the video files.

Exit
This closes the Splitter tool. Save your project first if necessary.

9.10.2 Edit Menu

This chapter explains the Edit Menu of the Splitter tool. Please see the introductory chapter for an overview about the Splitter tool [2120].



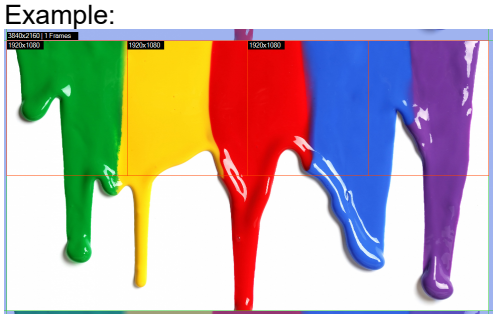


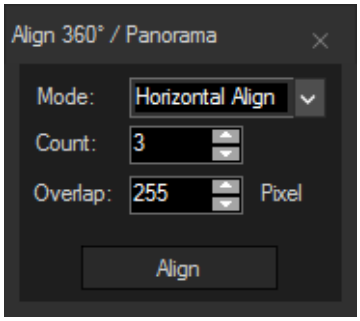
Align Outputs to fit Screen
This option allows to specify:
- a fixed number of output regions (= **Count**)
- with a fixed resolution (= **Size W and H**)
- to align either horizontally or vertically (= **Mode**)
- starting at a certain position (= **Start X and Y**), whereby the position (0,0) is the upper left corner.

The Splitter tool then aligns the output regions accordingly within the boundaries of the master file.

Check whether the resulting overlap area is sufficient, e.g. if you have a softedge projection. Otherwise, choose the option "Align Outputs with Overlap" where you can specify an overlap area too.

Example:



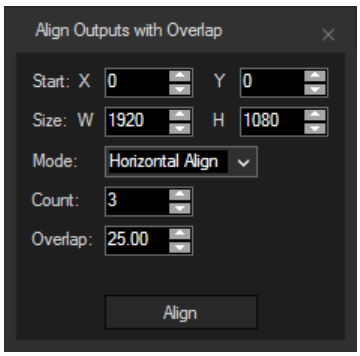


Align Outputs to Panorama / 360°
This option is useful for Panorama/ 360° setups and allows to specify:
- a fixed number of output regions (= **Count**)
- to align either horizontally or vertically (= **Mode**)
- with a certain overlap in pixels (= **Overlap**)

The Splitter tool then aligns the output regions accordingly starting at the upper left corner of the master file. The last output region will overshoot the master file's boundaries by the specified overlap so that it can blend into the first region.

Check whether the resulting resolution for each output region is sufficient, especially when exporting DDS image sequences [2126] or MPEG videos [2122], as there are restrictions regarding the size. See also the option "Override Mpeg Size Limits [2125]" below.

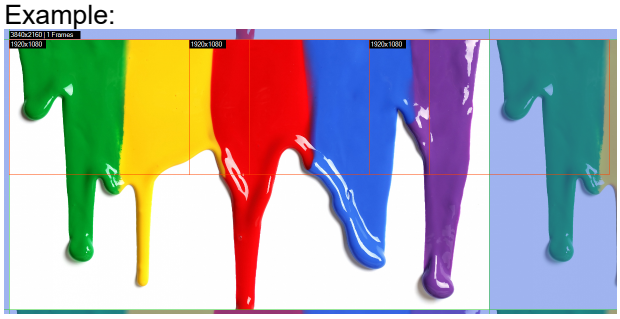
Example:

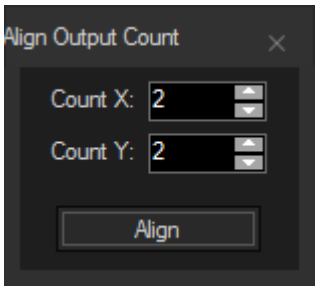


Align Outputs with Overlap
This option allows to specify:
- a fixed number of output regions (= **Count**)
- with a fixed resolution (= **Size W and H**)
- to align either horizontally or vertically (= **Mode**)
- with a certain overlap in pixels (= **Overlap**)
- starting at a certain position (= **Start X and Y**), whereby the position (0,0) is the upper left corner.

The Splitter tool then aligns the output regions accordingly. The last output region could overshoot the master file's boundaries. If the output regions should fit inside the master file instead, choose the option "Align Outputs to fit Screen" which does not ask for an overlap area.

Example:



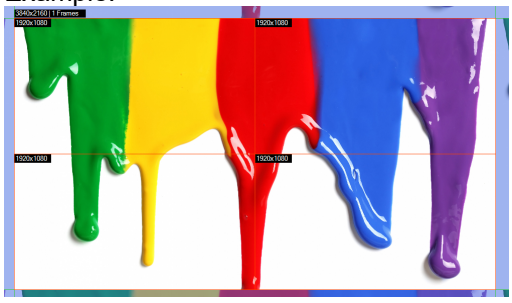


Align Outputs Count to fit Screen
This option is useful when no overlap is needed and allows to specify:
- a fixed number of output regions horizontally (= **Count X**) and vertically (= **Count Y**)

The Splitter tool then aligns the output regions accordingly starting at the upper left corner of the master file. The output regions will not overshoot the boundaries of the master file and no overlap is applied.

Check whether the resulting resolution for each output region is sufficient, especially when exporting DDS image sequences [2126] or MPEG videos [2122], as there are restrictions regarding the size. See also the option "Override Mpeg Size Limits [2125]" below.

Example:



Clear All Output Regions
This option deletes all output regions. The next region you create, will become "Output0" again.

High Performance Mode Image Seq I/O
This option enables / disables the High Performance Mode for exporting Image Sequences.

Draw Image
This option displays/ hides the master image file in the main window.

Show Wrap Zone
This option displays/ hides the wrapping zone around the master image file in the main window.

Override Mpeg Size Limits
This option is of interest when working with still images, image sequences or AVI. With the activated option "Override Mpeg Size Limits" you are free to set up any output region size.

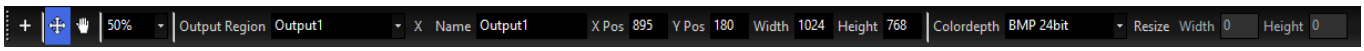
According to the Mpeg specifications, the image width (in pixels) must be dividable by 16 and the height by 8. If you enter a region that does not conform to this specification, the Splitter corrects it automatically to the nearest smaller number allowed. For instance, a 33 pixel wide and 15 high region results in 32 x 8. If overriding the size limits, the output region can be 33 x 15. When using the Option in the File Menu to "Export Image Sequence + MPEG Video..." now, the exported MPEG file is being encoded without the specified settings and might cause unexpected issues when being played back.

9.10.3 Toolbar and Window

This chapter explains the Toolbar and main Window of the Splitter tool. Please see the introductory chapter for an overview about the Splitter tool [2120].

Toolbar

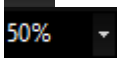
The toolbar allows you to create and manage the Output Regions.



Create Region
Click the left mouse button and span the desired region. This command is also accessible via the right-click menu [2127].

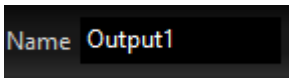
Move Region
Select the desired region and move it to it's correct place. This command is also accessible via the right-click menu [2127].

Drag View
Use this mode to drag the view around. This command is also accessible via the right-click menu [2127].



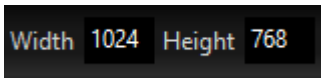
Zoom
Choose one of the Zoom factors to adjust the Preview window. You may enter an individual zoom factor as well. You may use the keys + / - on your keyboard as well.

You may choose an Output Region from the list in order to delete it, to adjust its name, position or size.

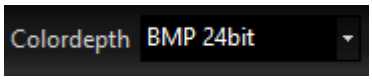


Name
Select an output region with this list. To change the name simply enter the new name and press [Enter]. To delete it, click the X-button.

X / Y Pos
The Pixel values inside these two text fields display the position of the region's top left corner. To change its position to an exact place please enter new values for X and Y here.
To move a region with the mouse please use the option "Move Region", as explained above.



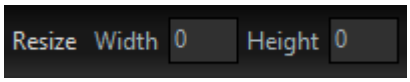
Width / Height
These two values display the size of the selected Output Region. To change its size please enter the new Width and Height into the text fields.



Colordepth
Choose here the color depth for the Image Sequence. You may choose between: BMP 24/32bit (snp), PNG 24/32bit, DDS RGB(A) (snp) or JPG

The formats labeled "24bit" or "RGB" do not support an alpha / transparency channel whilst the versions with "32bit" or "RGBA" do support it.
Please note that the height and width resolution of DDS images needs to be divisible by 4. The BMP and DDS format can also be saved as compressed SNP files. See the chapter "Image Formats [93]" for information about the formats (including snappy) and when to choose which format.

This setting does not affect an exported AVI or MPEG file, it will always be exported as Uncompressed AVI or MPEG-2 (24bit RGB).

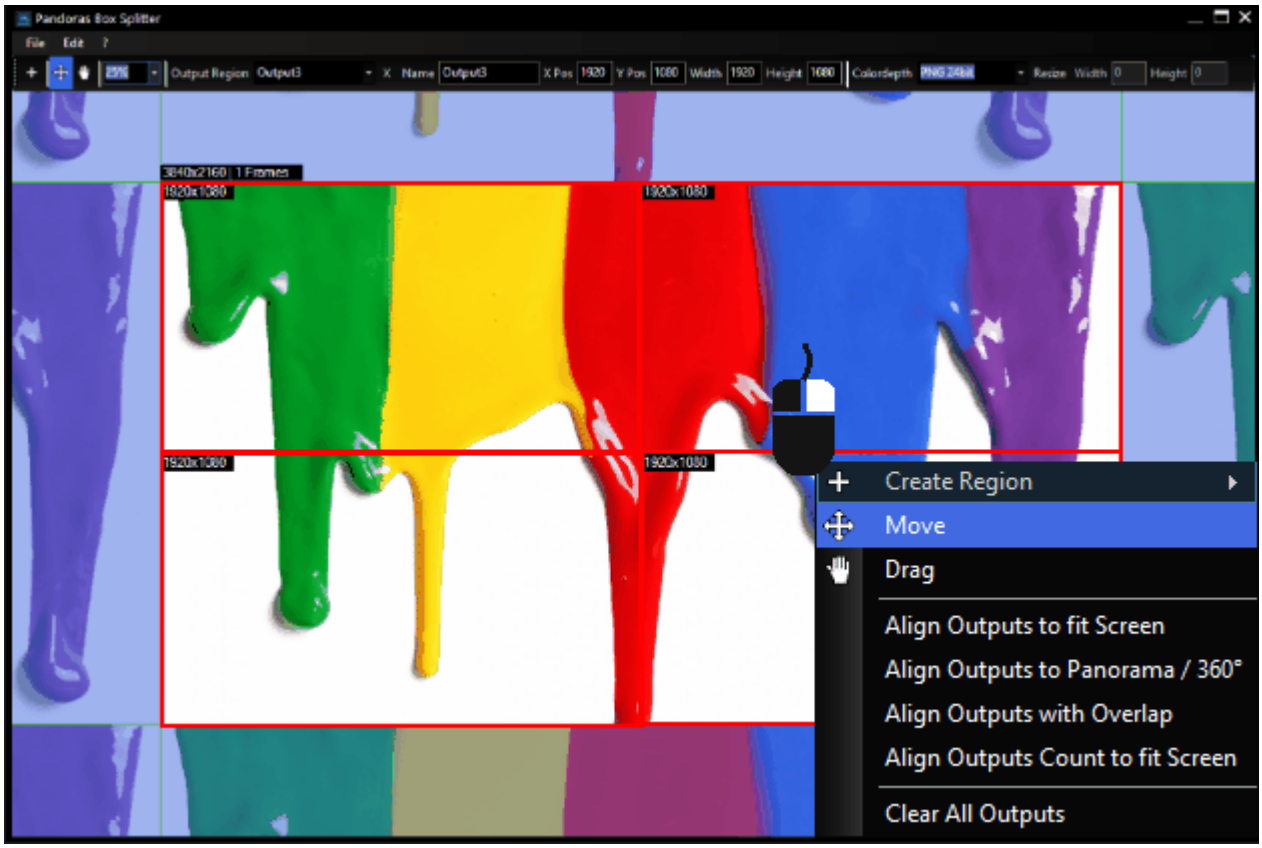


Resize
By default the Resize option is disabled, click the "Resize" button down to enable it. Enter the width and height to resize the selected output region for the exported image sequence / video. Keep in mind that DDS image sequences [2126] and MPEG videos [2122] have restrictions regarding their size.

Main Window

In the middle of the Main Window the loaded master file is shown. It is repeated at the left, top, right and bottom side to allow wrapping Output Regions around which is important for 360° Panorama / Sphere content overlap setups. You can disable this feature in the Edit menu > Show wrap zone.

You may set up any amount of Output Regions using the right-click menu, the "Create Region" button from the Toolbar or the automatic "Align" commands from the Edit menu. In the example, four Output Regions were created for a Multi-Display setup, whereby each region has a resolution of 1920x1080 px. When exporting the master file (via the File menu), it will be split into those four regions and four separate image sequences and/or videos will be created.



Context Menu

From the right-click menu you can choose one of the following options:

- Create an Output Region: Choosing Custom Size will enable you to span a rectangle in the Main Window. When choosing one of the pre-defined sizes, you just have to define the position of the Output Regions top left corner by clicking once into the Main Window.

- Move existing Output Regions to its correct place

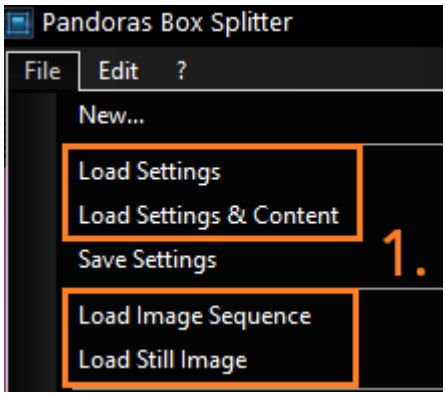
- Drag the whole view around (incl. Masterfile and Output Regions).

- Align and clear the Outputs, see Edit Menu [2123].

9.10.4 Splitter Workflow

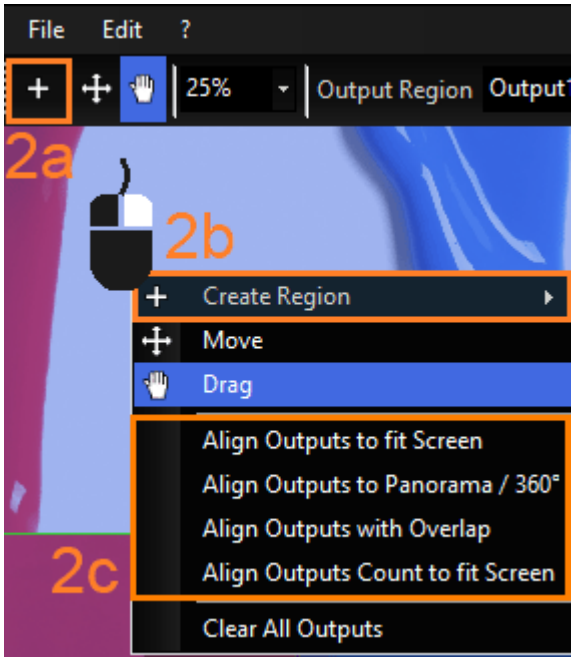
This chapter describes a possible workflow when using the Splitter. Please see the introductory chapter for an overview about the Splitter tool [2120].

1. Loading the Master File



Load an Image/ Image Sequence or an existing Splitter Project (.vsp) using the File Menu [2121].

2. Creating Output Regions

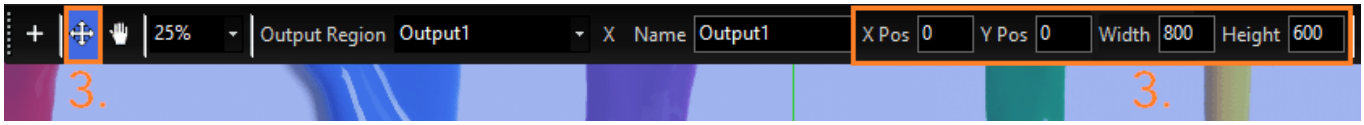


Create as many Output Regions as you need:
a) using the "Create Region" function in the Toolbar [2125]...

b) using the "Create Region" function from the right-click menu [2127] inside the Main Window

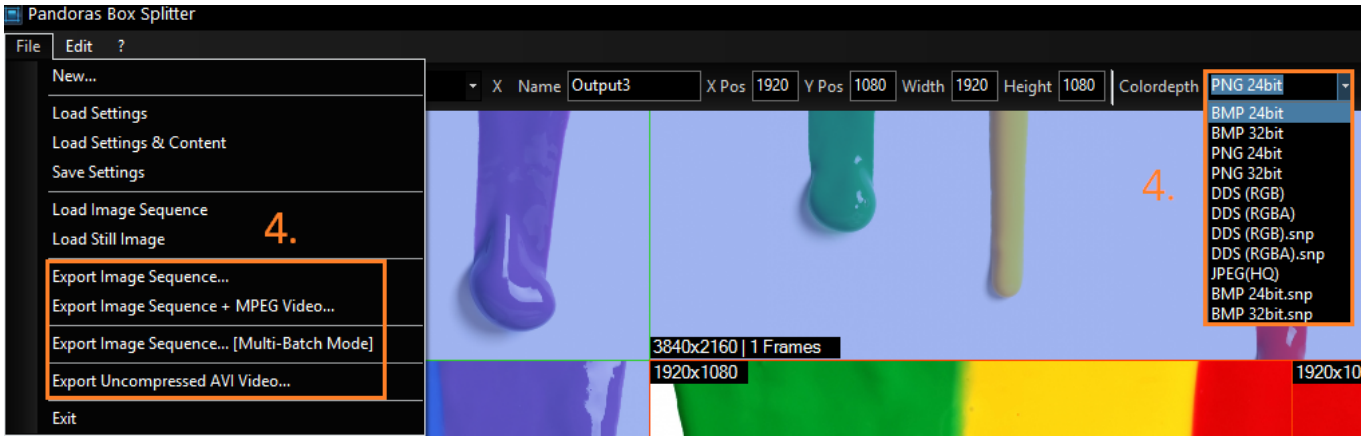
b) using the "Align" functions from the right-click menu inside the Main Window which can also be found in the Edit menu [2123]

3. Adjusting Output Regions



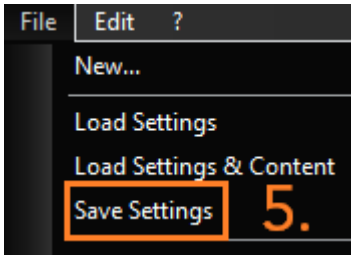
If any Output Regions need to be adjusted in their position and/or size, you have two choices. Moving the Output Regions with the mouse (using the "Move"-option from the toolbar) is the fast but unprecise option. To do an exact modification, select the Output Region (by name in the Toolbar or by clicking on it when being in Move-Mode) and enter the Pixel values of the top left corner in the "X Pos" and "Y Pos" fields and/or the "Width" and "Height" fields.

4. Exporting the split files



When the setup for the Output Regions is done you may export the separate split files now as image sequences or as video files using the File Menu [2121]. When exporting an Image Sequence choose the Image Sequence format in the Toolbar first (more info about formats [2126]).

5. Saving the current project



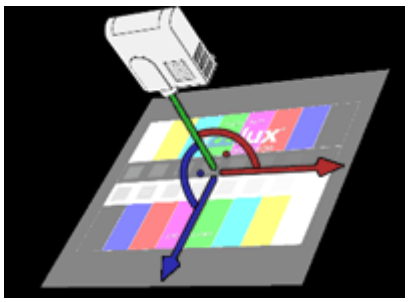
If you like to modify the Output Regions sometime later, it a good idea to save the project via the File Menu > Save Settings.

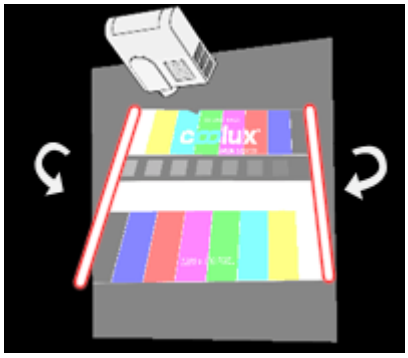
9.11 Warper

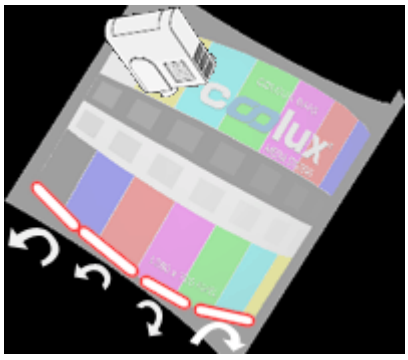
The Warper enables you to project on any screen shape. The Warper is a specialized, basic 3D modeling software using custom shapes with scalable free-form-deformers (FFD [2131]).
The application is included in the Pandoras Box installation and requires an attached dongle when started. Since Pandoras Box version 8 the full 3D edition is available for everybody. Before, the PB Player came with the 2D edition that had restricted features, e.g. no Z axis, object import, live warping, markers, camera settings etc.

More advanced objects can be imported from many 3D modeling programs as described in the topic " Third Party Software [2177] ".

Solutions for projecting on bended surfaces and complex objects



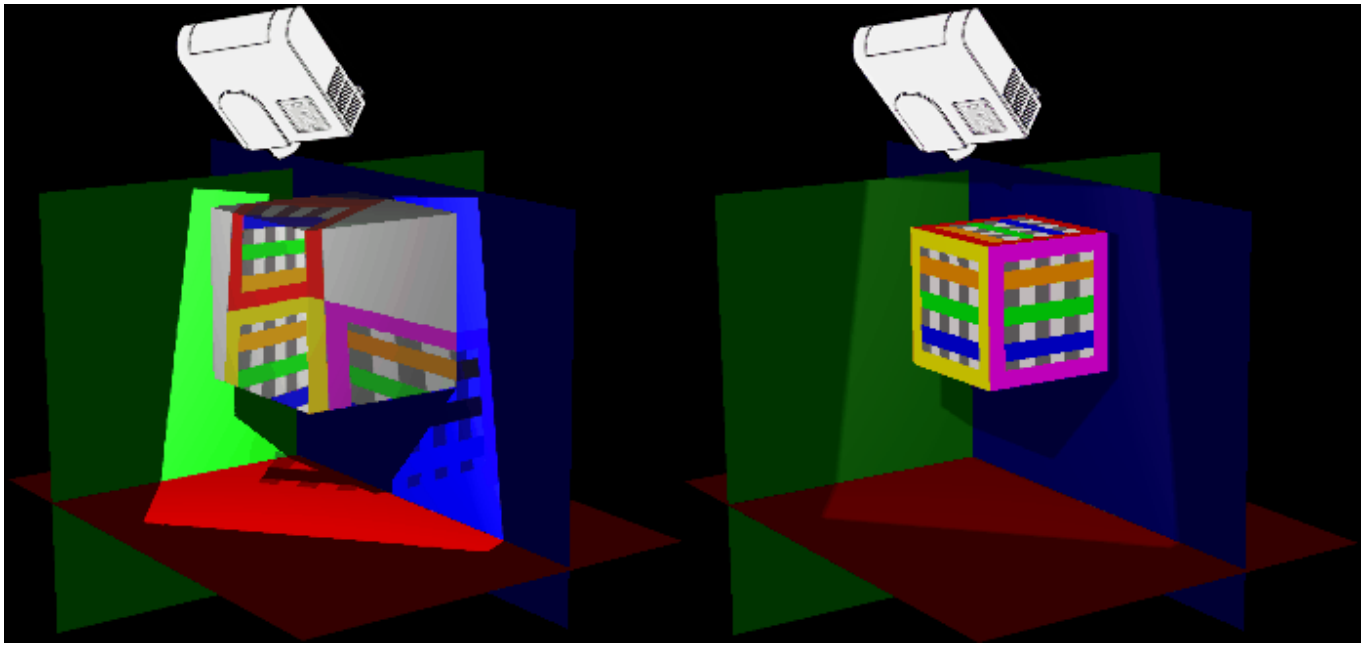




The main problem with projection surfaces that are not positioned perpendicularly to the projection axis is that the light source from the projector does not have the same distance to all projection points any more. A rectangular image is not rectangular any more and furthermore, the distances between particular points in the original image may be very different in the projected image. In short the projected image is distorted. The distortion relates to the shape of the screen (or object) and to the way the projector "looks" at it which includes the orientation and distance as well as all lens settings.

The challenge is to output an image that will look correct, considering the viewpoint of the audience. On simple planar screens, the keystone parameters for the top, bottom, left and right side of the image can already achieve a satisfactory results (see left, middle image).

As soon as the projection surface is not a single plane anymore, e.g. it is a bended screen or a more complex object like a cube the keystoning reaches its limit.
Those setups can be mastered in two ways. Which solution you choose depends on your setup, time as well as experience and knowledge of the team.



projection before ... .

... and after warping

2D Warping

The first approach is to create a 2D plane offering for example a grid with 100 intersection points. The entire plane and each individual point can be moved, scaled and rotated until the original image is distorted in a way that the projected image looks correct. The plane is then exported as an X file and applied to the PB output layer [682] (or another layer) and acts like a map for each pixel of the original content.
This workflow can not be prepared beforehand. It is indispensable that you work in the Warper with the real projector connected and distort the 2D image whilst looking at the real screen. When something is changed in the setup the chances are high that ALL points must be adjusted.

The advantage of the workflow is that you do not need to have any knowledge about 3D objects. You simply need to move the plane's intersection points until all lines are straight and all distances between certain points are equal. Measuring tools, tape to mark distances and (rotary) lasers can help you in that process and simplify and speed up the workflow.

3D Warping

The second approach is to represent the real 3D setup as a virtual 3D setup. It includes to import a 3D object into the Warper; the 3D model must match the real screen or object as accurately as possible. Then, the virtual camera in the Warper is set up in the same way as the projector looks at the real screen. For this, you need to adjust the camera's and the object's position, orientation and lens settings. Two ways to do this are manual measurement, described in the Camera/View tab [2159] and the automatic camera calibration based on markers, described in the Marker tab [2161]. This feature is available since Pandoras Box version 5.5.

In addition, fine-tuning adjustments with the object's vertices can be done too. As soon as the virtual model and the virtual camera accord with the real screen and real projector the projected image looks correct.
As the final steps, the Warper's camera values are transferred to the PB Camera Layer [675] and the object is exported as an X file and applied to a individual Video Layer [647].

This workflow can be prepared beforehand. You may for example create the screen model using a 3D modeling program like 3ds Max [2197] and Maya from Autodesk, blender [2198] from Blender Foundation or CINEMA 4D from MAXON Computer GmbH. It is possible to setup the virtual settings for XYZ position of the object and the camera in accordance to available plans.
The advantage of the workflow is that you can nevertheless adopt changes conveniently. If, for example, the real setup is changed or not as accurate as scheduled you can adopt these changes easily for the virtual setup in the Warper.

For more information about how to use the Warper please refer to the following pages:
General 3D Modeling Terms [2131]
User Interface, [2136]
Warping Guide, [2166]
Keyboard Shortcuts [2176]

9.11.1 General 3D Modeling Terms

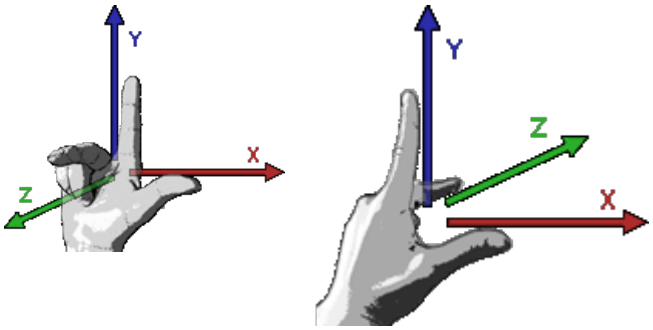
This chapter covers the common 3D modeling term definitions:
- coordinate system
- generic units
- Vertices, Pivot Point, Edges and Faces
- UV texture mapping
- mesh versus FFD

Please make yourself familiar with these terms. They will be used in the following chapters when the Warper interface and workflow is being explained. As well, the explanations might help you when communicating with 3D artists who prepare 3D models for you or when getting in touch with one of the third party applications [2177] yourself.

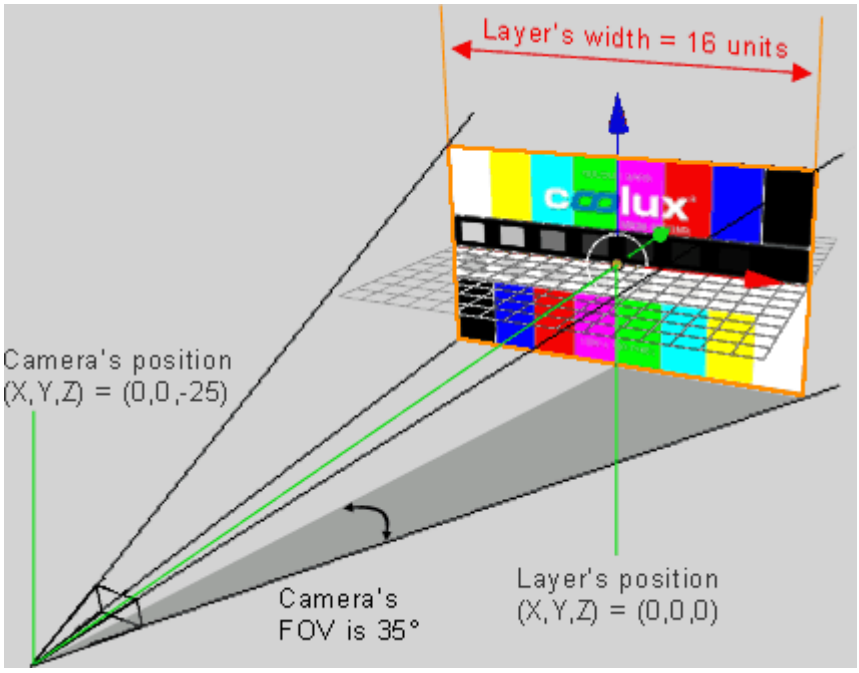
Coordinate System

Each 2D and 3D world needs a coordinate system which defines the position and orientation of each object. There are two possible systems: the right-handed and left-handed system. In both systems the positive X and Y direction are the same and show to the right and upper side (as seen in the picture below). The Z axis makes the difference, it shows either to the back or to the front.

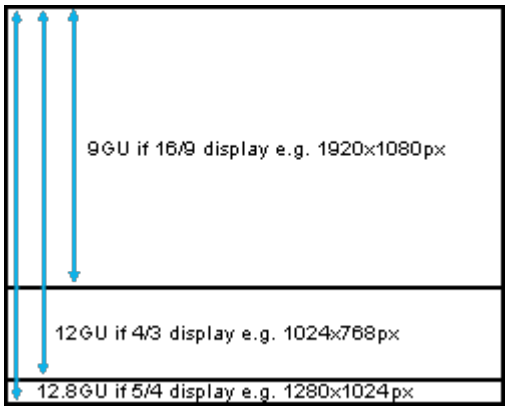
Pandoras Box and the Warper are based on the left-handed system. If you import an object from an application that is based on another orientation make sure to adapt it.



The origin of the 3D space (X,Y,Z) = (0,0,0) is situated in the exact middle of the screen when starting Pandoras Box and the Warper with default settings. The origin is the reference point for every object. The camera has a position of (0,0,-25) and a FOV (field of view = opening angle) of 35,489 degree (or 56,251 mm).



Generic Units

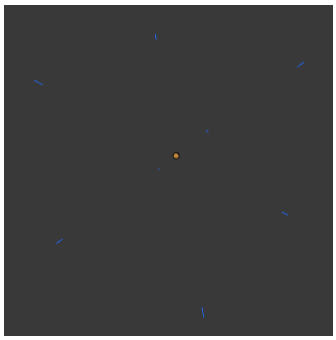


Next to the coordinate system's orientation, units are of great importance. An object has a defined position and size which is defined and saved in units.

Many 3D modeling programs can save an object with so called generic units (GU), e.g. a cube may have a height/width/depth of 8 GUs, rather than being measured in metric or pixel units. The advantage is that GUs adapt to the current screen, they "generate" a relative size and must not be scaled when the resolution changes. Per default a screen width is exactly 16 GUs, the height is calculated by the aspect ratio. Hence, a 4:3 display is 16 GUs wide and 12 GUs high, whereas a 16:9 display has the same width but a height of 9 GUs.
For example: A 3D modeler works an a 1024x768 pixel display and creates a cube that covers half of his display, i.e. it is 8GUs wide. If you import the cube into the Warper it will have the same relative size (half of *your* display) no matter which aspect ratio or resolution is chosen at your computer.
The height will always be the same size as the width, it will never be "squeezed", however it might cover less or more space than on the modelers screen.

The Warper and Pandoras Box are based on generic units since version 5. The coordinates (X,Y) = (0,0) are situated in the middle of the display. If a layer or object moves 8 units to the right its center will be exactly on the right edge of the screen.

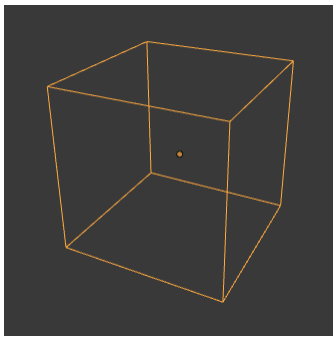
Vertices, Pivot Point, Edges and Faces



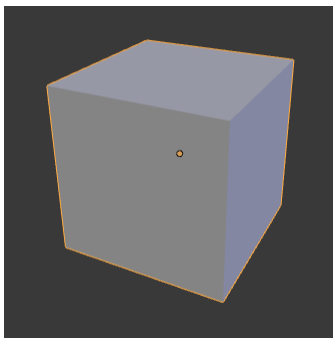
Every 3D model and geometric shape consists of corners and intersections. These special points are saved as so called "vertices" (singular: "vertex"). In a 3D space a vertex must consist of three coordinates, X,Y and Z to definitely mark the position of a corner. A cube, for example must be defined by a minimum of 8 vertices. In the example, these are the small blue points.

The default mesh in the Warper consists of vertices as well, the more mesh segments you set up, the more vertices are generated by the Warper.

In most 3D modeling programs there is a so called pivot point. In the left image it is depicted as a round orange point. The pivot starting point is the center of a transformation like rotation and scaling. If the cube is scaled, all vertices still would have the same distance to the pivot point. For more examples, please see the topics Scale Pivot [655] and Rotation Pivot in Pandoras Box [654].
In the 3D modeling program itself you may position the pivot point where you like but please keep in mind that file formats like X, FBX and 3DS do not support pivot points. These formats always save the coordinates of the vertices as absolute coordinates. That means that the position of the object and all its vertices will be kept. But when importing the object in the Warper, the center of rotation and scaling is the center of the object itself. This cannot be influenced. Pandoras Box, on the other hand, supports pivot points itself. Thus, when importing the object, the pivot point is set to the origin of the coordinate system (0,0,0) and can be moved to the position of your choice with the parameters of the layer.



The connections between two vertices are called "edges". Here, they are depicted as orange lines. A cube consists of 12 edges.

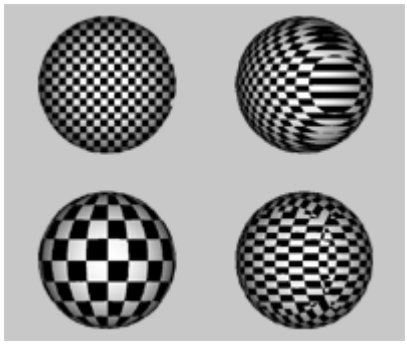


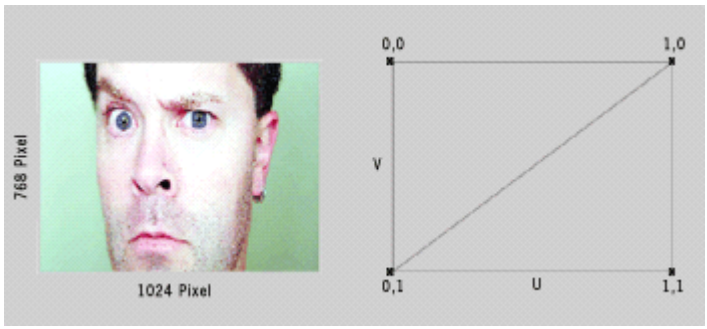
Some 3D modeling programs define that exactly three vertices make one face. Thus faces are always triangular.

The next hierarchy is then called "polygon" whereas one or more face(s) form one polygon. The outlines from a polygon are the edges.
In the example image they are 6 gray squares forming a cube; 6 polygons. If faces exist, there would be at least two faces per polygon.

No matter how your 3D modeling program works, either faces or polygons can be applied with a texture.

UV Texture Mapping

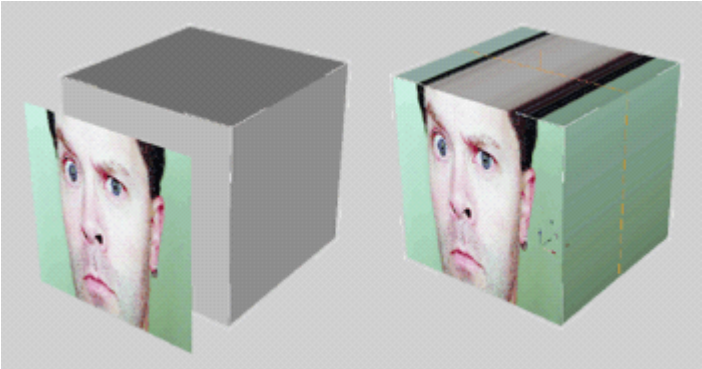




UV mapping is a process of defining how to represent a 2D image on a 3D model.
The UV mapping transforms a 2D source image (in our case an image or video) into an image buffer called a texture.
In contrast to "X", "Y" and "Z", which are the coordinates for the rendered 3D object, "U" and "V" are the coordinates of the texture. The UV map stores for each X,Y,Z- coordinate a defined U,V-coordinate.

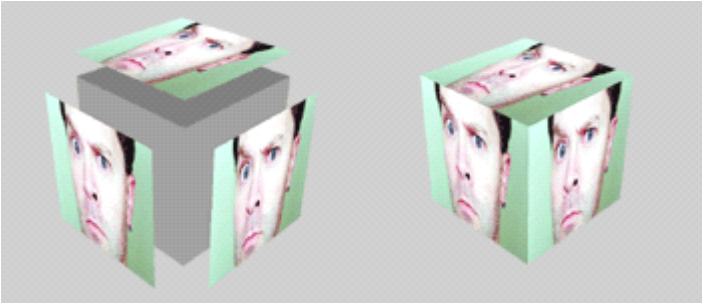
This creates the effect of painting the image onto the surface of the 3D object, or in other words, how to wrap or stretch the image around the object. As explained below there are different ways how to do that, hence the chosen UV map is stored as a property of the 3D model.

If a 3D model has no UV map, neither the Warper nor Pandoras Box can paint a texture on it which leads to the fact that the object is invisible and cannot be displayed. The Warper has tools that can influence the UV Map, e.g. scale or move it [2153]. When not importing 3D models from third party programs but working with meshes in the Warper you can define how to apply a texture on it. A texture can cover more than one mesh as well. This is described in the chapter "Edit Menu [2142] ".
On the Pandoras Box side, there are effects influencing the UV map [631] as well.

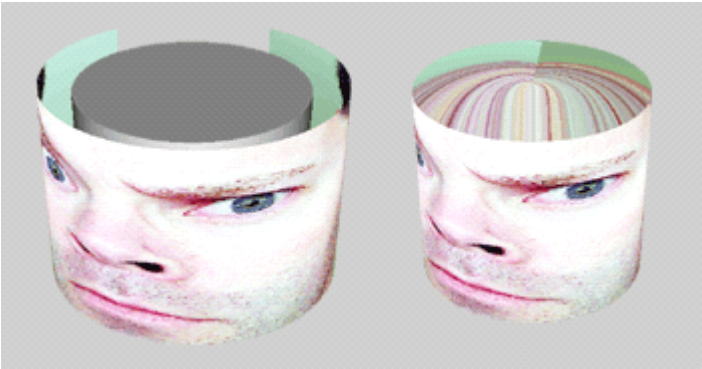


There are several standard mapping techniques available to map a texture onto an object:
- planar
- cubic or box
- cylindrical and
- spherical mapping.

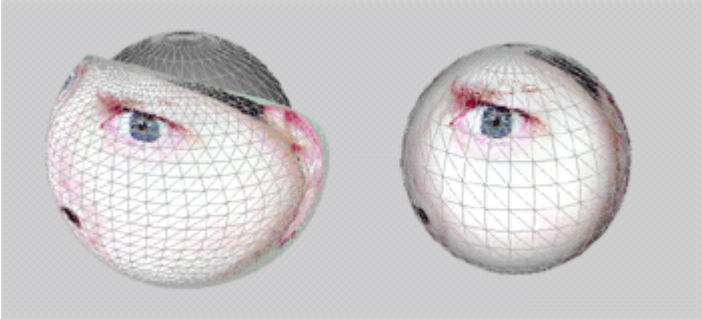
Planar mapping can be referred to an image projection from one side onto an object.



A cubic mapping for example maps the texture to all sides of a object like a box.



The cylindrical mapping wraps an image around an object like a cylinder, the left and right edge will join each other.



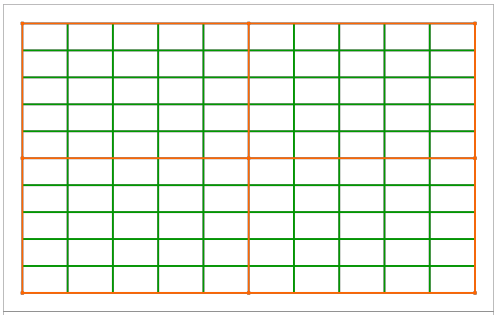
Spherical mapping wraps the image all around an object as a sphere. Please be aware that the top edge of the texture shrinks down into the top north pole and the bottom edge in the south pole.

For proper spherical mapping textures with an aspect ratio of 2:1 apply best to a spherical object.

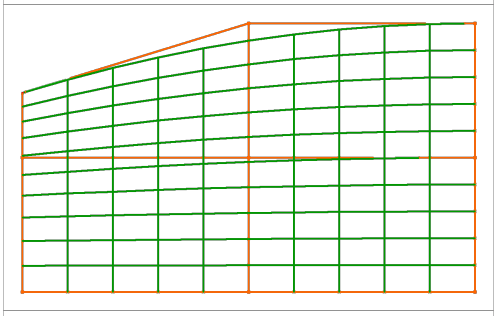
Mesh versus FFD

This paragraph describes the difference between a mesh point and an FFD point, thus it is covering a fundamental function of the Warper.
The pictures below show a 2D plane with a green-colored 10x10 mesh and orange-colored 3x3 FFD.

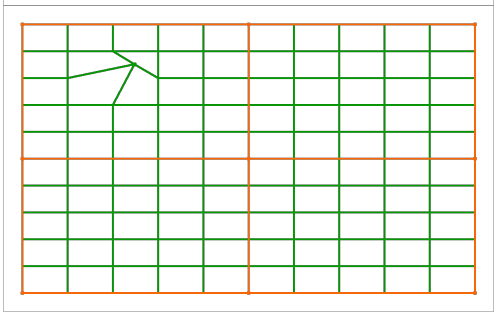
See here the differences between moving a FFD control point and moving a mesh point. Please note that the FFD is only a helping tool to set up the mesh, you won´t see the FFD in the exported object.



a. The grid without any editing. The green lines represent the mesh, the orange lines the FFD.



b. The top left FFD control point is moved further down. The whole mesh is affected by this change: the horizontal lines are bend together on the top left side, the meshes outline gets curved.
This effect can be of advantage or disadvantage. In the beginning of the warping process it can simplify and accelerate the workflow as it is not necessary to move each individual mesh point. The further the warping process develops the more it is necessary to apply changes to particular pixels only. At this point the FFD is not sufficient any more as it affects large areas of the grid.

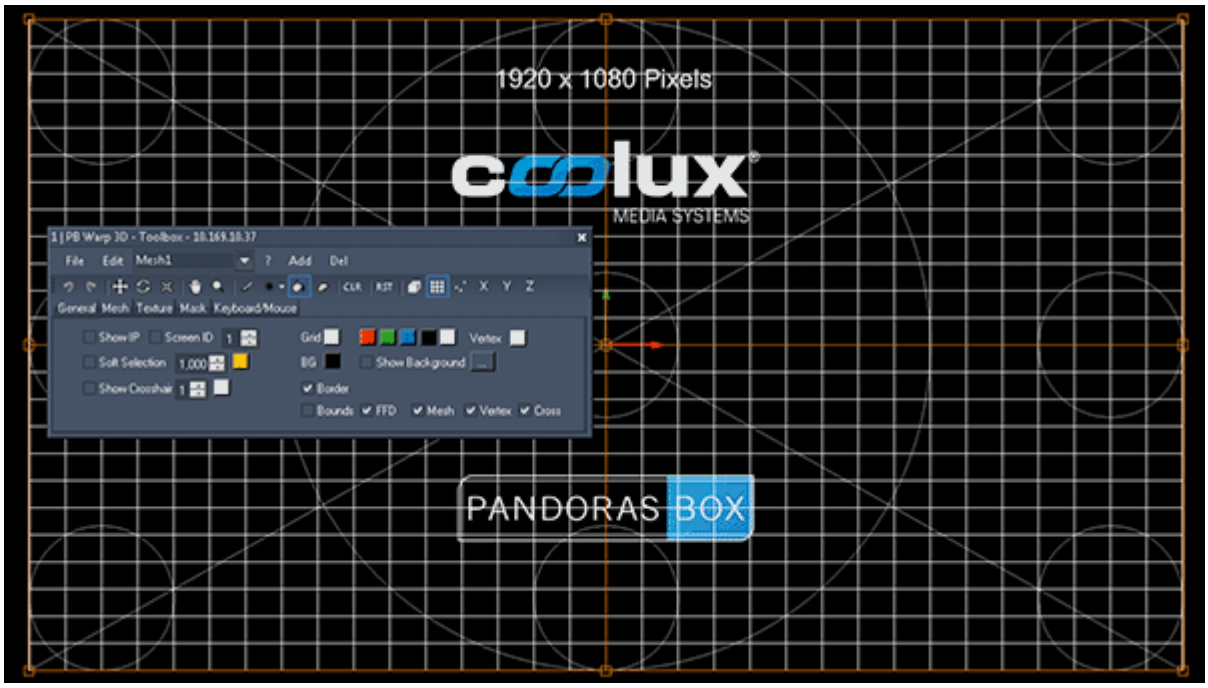


c. One mesh point is moved. Only the segment lines between the moved point and the four neighbor mesh points are affected by this. The more you are experienced with warping the better you will be able to answer the question how many mesh points a grid should have. If too little points are chosen it won't be possible to apply the detail changes that are necessary. This is especially crucial when setting up meshes for a softedge projection as the pixels must overlap each other exactly in the overlapping area.
If too many points are chosen, the warping process is lengthened unnecessarily as all points must be adjusted.

For more information about how to use the Warper please refer to the following pages:
User Interface, 2136
Warping Guide, 2166
Keyboard Shortcuts 2176

If you are interested in other 3D modeling programs, please refer to the topic covering third party applications 2177.

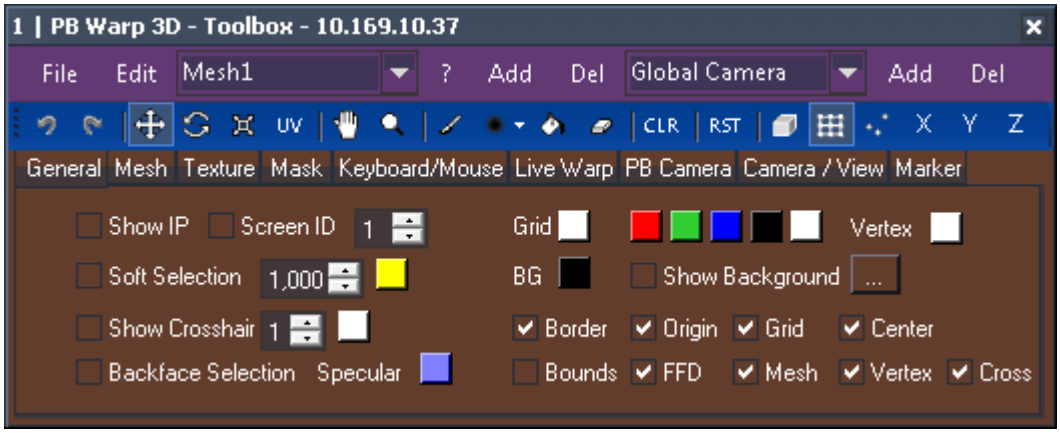
9.11.2 Warper User Interface



When opening the Pandoras Box Warper application in the background you will see a default 2D plane with orange lines (FFD - Free Form Deformers [2131]) and white lines (mesh [2131]), the grid displays a testpattern that is generated automatically according to the output resolution. How to work with the grid is explained in the chapter "Warping Guide [2166]".

In the foreground you have a gray Toolbox, wherein you set up the grid properties and other features.
Edit > Advanced Mode activates more advanced options and settings. Please note that there are differences between the 2D and 3D edition as explained in the previous introductory chapter [2129].

The Toolbox' title bar displays the ID of the application (e.g. "1 | PB Warp 3D") and the IP address, this is quite useful when working on several Warper windows at the same time, for example when setting up two meshes for a softedge projection. Note that the x-button closes the toolbox only (T brings it back), File > Exit closes the entire program.



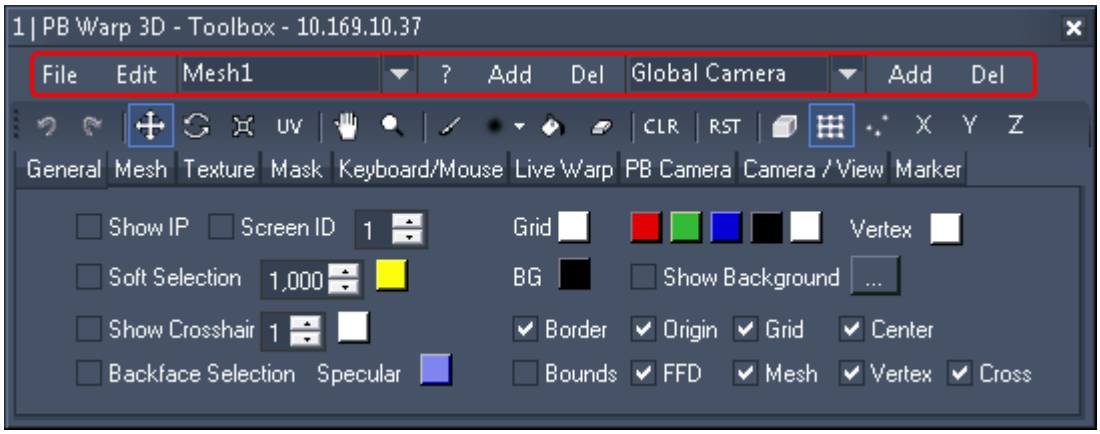
Title bar

Menu bar [2137]

Tools bar [2149]

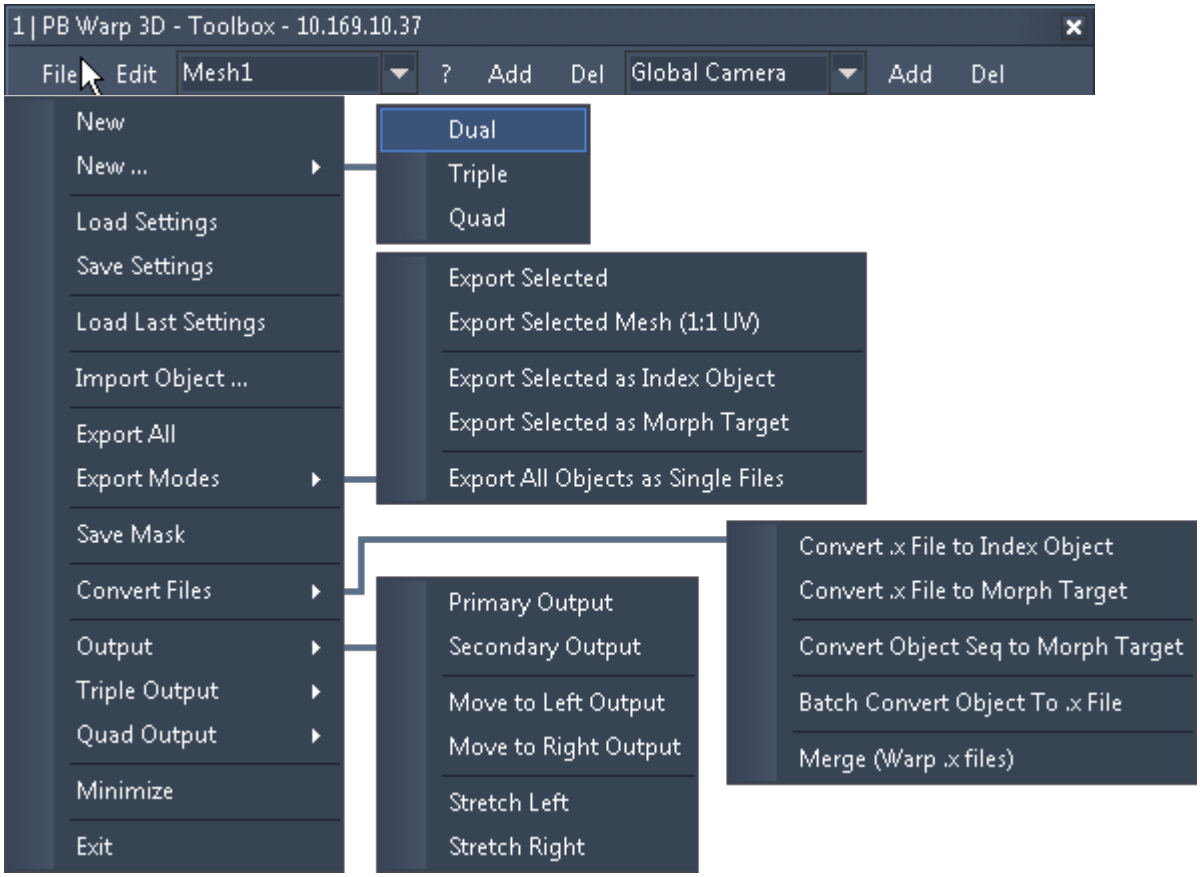
Tabs and their content [2151]

9.11.2.1 Menu Bar



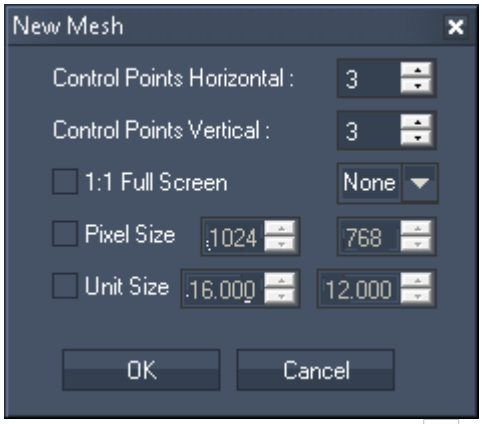
The Menu Bar is divided into the File Menu [2137], the Edit Menu [2142], the Mesh Selection [2147] and [?] - Button [2148] and the Camera Selection [2148].

9.11.2.1.1 File Menu



The depicted menu is the advanced one. Per default, the basic menu is shown, to get access to all options, go to Edit > Advanced mode

New

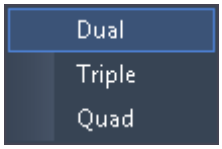


NEW
When creating a new file, this dialog box opens up. Please note that this will close your current project. (In case you simply like to add another mesh, click the Add Button [2147] next to the mesh selection drop-down list.)

Decide how many horizontal and vertical FFD control points should be applied. How many to choose depends on the shape of the screen you need to warp. For more details, there is a tutorial [2166] at the end of the Warper chapter.

Tick the check box [1:1 Full Screen Start Size] if the new mesh should cover the complete output instead of keeping a small border uncovered. You may as well decide what aspect ratio the new fullscreen grid will have. The width of 16 generic units [2132] will always cover the current output screen's width.

If you decide for a pixel size please note that it will generate values relative to your local output resolution and aspect ratio and will apply them to the object. For example: if you are working on a 1920x1080px display and choose a 1024x768px mesh, the new plane will have a width of 8.533 generic units (16GU * 1024/1920) and a height of 6.4 GU (9GU * 768/1080).
Defining a unit size for the width and height will apply those generic units directly.



NEW...
Again, this option will close your current project.

Choosing "Dual / Triple / Quad" will generate two / three / four new meshes (on the screen that is set to be the primary monitor). There is a UV Map covering all meshes.
This option is very useful when working with a device that splits *one* output into several parts. If you use this device in a softedge projection, it is recommended to create special testpatterns with a grid that is colored differently per region. That allows you to see if the warped meshes overlay each other pixel per pixel. Alternatively you may open the Warper several times and use the Triple / Quad option [2141] as described below.

The settings offered in the dialog box are the same as explained in the previous paragraph.

Load and Save Settings

LOAD SETTINGS
If you want to load an existing warping file (WRP), choose the directory and file name in the opening dialog .

SAVE SETTINGS
Choose the directory and file name in the saving dialog to save your warping file (WRP). In contrast to an exported X file object, the warping file includes all internal settings like FFD points, the Mask and all other Warper settings. It is recommended to always save in this format as well.

LOAD LAST SETTINGS
Check this option if you like to load the last project as soon as the Warper is started.

Import Object

This command allows to import an object from another 3D modeling program [2177]. A pop-up asks whether you like to merge the imported object, i.e. include it into your current setup. If "No" is chosen your current project file closes and a new project is opened containing nothing but the imported 3D object.

Choose the directory and file name in the opening dialog. The following formats are supported:
- 3DS
- FBX
- OBJ
- X

The next dialog asks you whether you would like to load the 3D scene of the chosen file in separate parts or as one united object. In case your scene has more than 32 individual objects, it is not recommended to load them as separate elements as this consumes much memory.

If your object is larger than 16 generic units [2132] a pop-up lets you rescale the object. Though, due to possible rounding errors, it is recommended to scale the object in the software it was originally created with.
Please note that X files can be written by many exporters and sometimes the same data is written differently. The build-in importer is optimized to work with the exporters provided by us [2177]. Importing files from other exporters could result in changed orientations, normals,... .

Export Modes

EXPORT ALL
Use this export option to export all meshes as one X file.
For example, if you have created two meshes and use the "Export All" option the resulting X file will behave as follows. Both meshes are covered with the same texture. If applied to a layer in Pandoras Box its media will be displayed twice, according to the scaling and the position of the two meshes. If applied to an output layer, everything that is seen by the camera layer will again, be displayed twice.

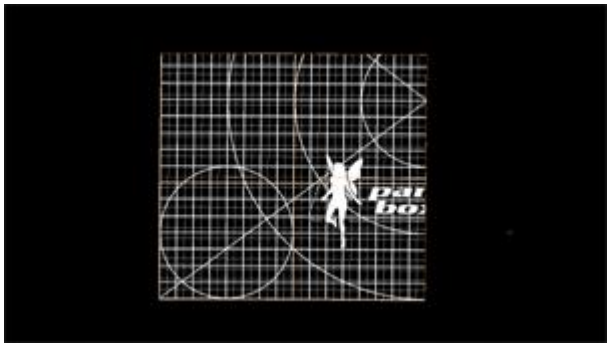
Choose the directory and file name in the saving dialog to save all meshes as one scene.

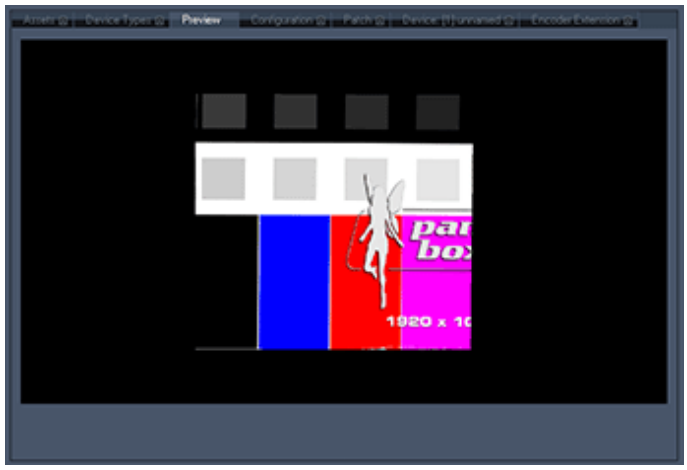
EXPORT SELECTED
Use this export option to export *only the selected* mesh as an X file.
In contrast to the above option you may now export each mesh individually and apply it to different Video Layers or different Output Layers.

Choose the directory and individual file name in the saving dialog to save only the selected mesh.

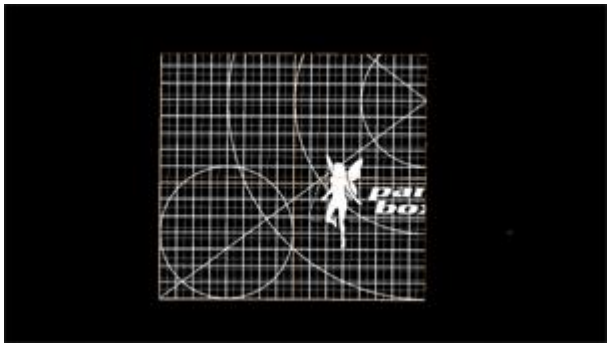
Please note: Changes done to the UV texture mapping (by editing the texture's Zoom or Offset settings in the Mesh tab [2153]) will be included in the exported file. This will influence the size or position of a texture used with this object in Pandoras Box as well! If you want to have the whole texture mapped on the object, please use the option "Export Selected (1:1 UV)" instead.

Example: The left image below shows a mesh in the Warper. The texture mapping has been modified to see only a part of the testpattern. After having exported this file via the option "Export Selected", in Pandoras Box (depicted in the right image) the object's texture will look corresponding to texture in the Warper.





EXPORT SELECTED MESH (1:1 UV)
Use this export option to export the selected mesh as an X file without the changes done in the Mesh tab [2153].
In contrast to the above option any texture's Zoom or Offset settings will not be included in the exported file - the UV map is applied with the original 1:1 setting. When the mesh is applied to a Video or Output Layer (depicted right) the object's texture will look different to the texture in the Warper (shown left).



EXPORT SELECTED AS INDEX OBJECT
Use this export option to export the selected mesh as an index object (X file) to be used for morphing or live warping within Pandoras Box.
Choose the directory and file name in the saving dialog to save the index object. In Pandoras Box, the index object can be used as any other object on a Video or Output Layer. Please refer to this chapter to learn more about morphing and live warping [2174].

Please note that a new index file must be exported each time when the FFD and mesh count is changed.
If you like convert an external X file to an index object please use the option "Convert .x File to Index Object".

EXPORT SELECTED AS MORPH TARGET
Use this export option to export the selected mesh or 3D object and its current look as a morph target (X file) to be used for morphing within Pandoras Box.
Choose the directory and file name in the saving dialog to save the morph state. Then you may alter the mesh / object deformation and save this as a new morph target. In Pandoras Box, the morph objects can be used as effect media files for various morph effects. Please refer to this chapter to learn more about morphing and live warping [2174].

Please note that a mesh / object used for morphing should not be placed outside the area that reaches from -32 to +32 generic units.

If you like convert an external X file to a morph target please use the option "Convert .x File to Morph Target".

EXPORT ALL OBJECTS AS SINGLE FILES
Use this export option to save each 3D objects from your scene as individual X files. Choose the directory and folder name in the saving dialog. Automatically, the files are named with a consecutive name like "3ds Obj1.x" and "3ds Obj2.x".

Save Mask

Use this option to export the screen mask you created in the ScreenMask tab.
Choose the directory and file name in the saving dialog to save the mask as a PNG file, which can be inserted in your Pandoras Box project and used as a mask object on a layer. Read more... [2156]

Convert Files

CONVERT .X FILE TO INDEX OBJECT
CONVERT .X FILE TO MORPH TARGET
The above described options "Export Selected as Index Object" and "... as Morph Target" refer to an object made within the Warper. If you have exported your object from another third party program as an X file and you like to use it for morphing and live warping within Pandoras Box please use the "Convert..." option.

First, choose the directory and file name from your external X file in the opening dialog. Then, choose the directory and file name for the converted index object (or converted morph target) in the saving dialog. Please refer to this chapter to learn more about morphing and live warping [2174].

CONVERT OBJECT SEQ TO MORPH TARGET
Some programs can export a movement from objects as an object sequence - a folder consisting of several 3DS,

OBJ or X files. To convert all of them in one step to be used as a morph target, choose this option.
An opening dialog opens where you can navigate to the folder and multi-select all object files that should be converted. If needed, set a resizing factor. A progress bar is shown in the right upper corner of the menu. Each object is converted to a PNG file, the new name consists of a consecutive number followed by the original name. The image files are automatically saved in the same folder.

As a next step you might copy those images into a new folder that can be used as an image sequence in Pandoras Box, for example on the Aeon Effect "Warp Target [644]" in the folder "Warp" or "Morph A-B-C [448]" in the folder "Geometry". In order to save performance, you may convert the PNG files as well into an AVI video file, e.g. using the Image Converter [2069] tool or to a lossless video format from the coolux codec [102].

BATCH CONVERT OBJECT TO .X FILE
Some programs can export a movement from objects as an object sequence - a folder consisting of several 3DS, OBJ or X files. This command allows to convert all of them to X files in one step.
An opening dialog opens where you can navigate to the folder and multi-select all object files that should be converted. If needed, set a resizing factor. A progress bar is shown in the right upper corner of the menu. Each object is converted to an X file, the original name is not changed. The files are automatically saved in the same folder and can now be imported in Pandoras Box.

MERGE (WARP .X FILES)
This option enables you to merge several warping files (X files) into one file.

In the dialog, browse to the folder and select all files you want to merge. A new dialog opens asking you under what name and whereto the merged file should be saved. The merged object may now be imported in Pandoras Box or the Warper itself.
This option might be useful when you already exported separate meshes instead of using the "Export all" option.

Output

PRIMARY / SECONDARY OUTPUT
By default the warping grid opens on your primary output.

When you have two outputs and the graphics card is in "Extended Mode" mode / "Dual View", please use this option to position the Warper's mesh on the left or right output. This is contrary to the below described graphics card setting "Horizontal Stretch".

You may open the Warper application twice whilst the first is positioned on the primary output and the second on the secondary output. This has the advantage of working on both outputs simultaneously.

MOVE TO LEFT / RIGHT OUTPUT
Choose these options if you are working with a stretched desktop consisting of more than one "display area" in order to move the warping area one step further.

STRETCH LEFT / RIGHT
When you have two outputs and the graphics card is in "Horizontal Stretch" mode (recommended display mode for soft edge projection), please use this option to position the Warper's mesh on the left or right half. This is contrary to the above described graphics card setting "Extended Mode" mode / "Dual View".

Triple Output and Quad Output

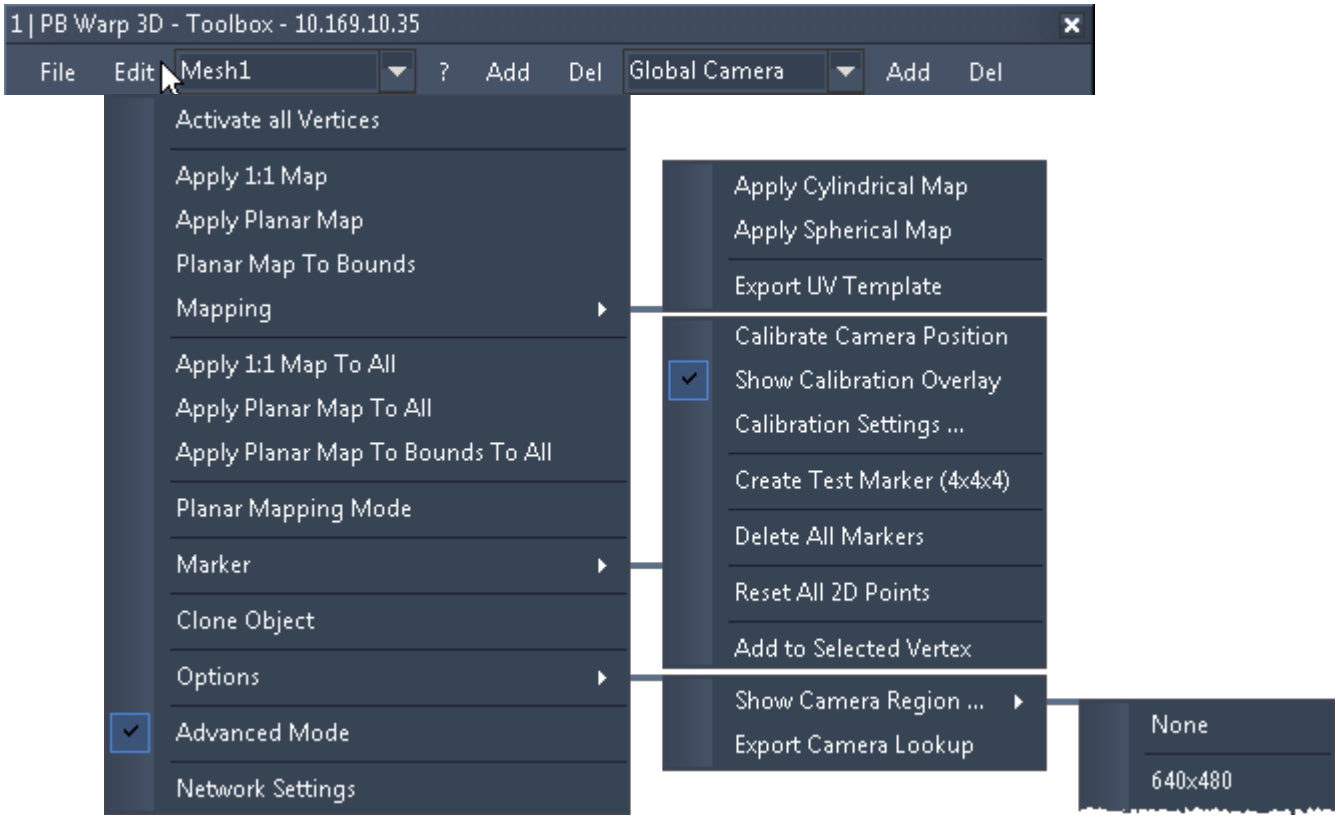
This command is useful when having three or four outputs. Open the Warper application and send it to Quad Output 1, then open the Warper 3 more times and send each to another quarter of your screen. Now you may work on all meshes simultaneously.

Alternatively you may open the Warper only once and work on several meshes using the New... command [2138] as described above.

Minimize
Minimizes the Warper window. As the Toolbox always stays in the foreground you have to close it as well or just press [H] to hide it.

Exit
Quits the application. A dialog box will appear and ask you if you want to save the project before closing. Choose [Yes] to save it, [No] for exiting without saving or [Cancel] to get back to the application.

9.11.2.1.2 Edit Menu

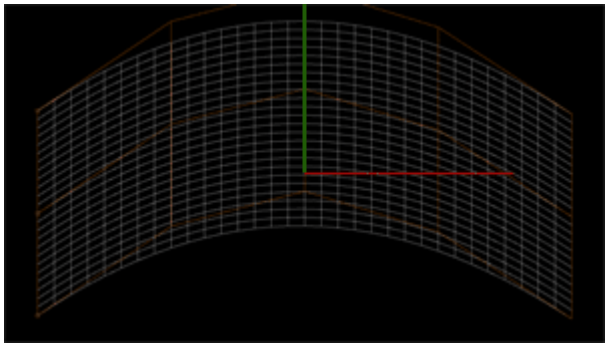


Activate all Vertices

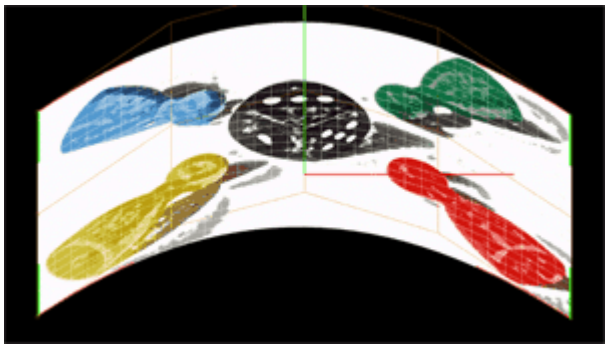
If you toggled some vertices/faces of the mesh with the shortcut [D] to be invisible, you can activate all hidden vertices again using this menu entry.

Apply Map and Mapping

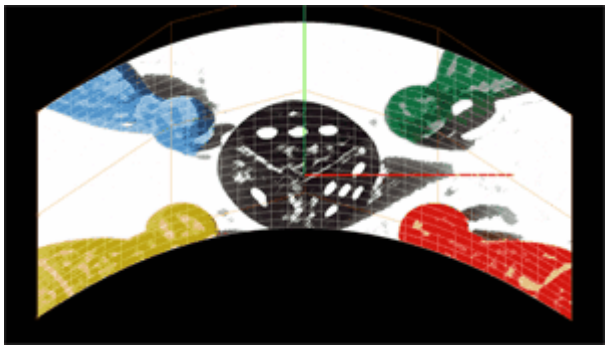
The following commands change the UV texture map for generated planes and objects. If you are not familiar with UV mapping, the chapter "General 3D modeling terms 2133" includes a short section covering it and can get you started with that important 3D modelling technique. The Mesh tab 2153 allows to offset and zoom a previously applied UV map.



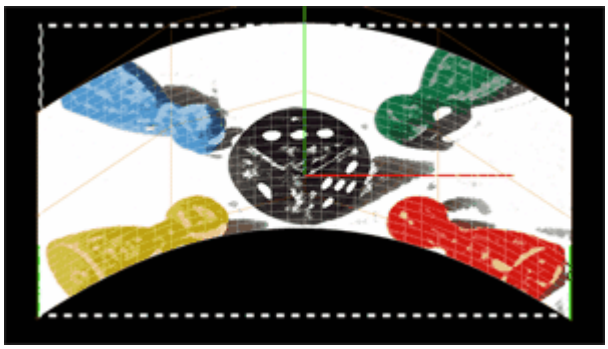
The depicted example shows one single mesh that was bended by using the FFD.



APPLY 1:1 MAP
In the Texture tab you may find the button "Load Texture...".It applies a texture 1:1 like, that is, the texture is scaled and deformed to fit exactly into the corners of the mesh. The command "Apply 1:1 Map" always brings you back to that state. It applies a texture to the currently selected mesh or object.

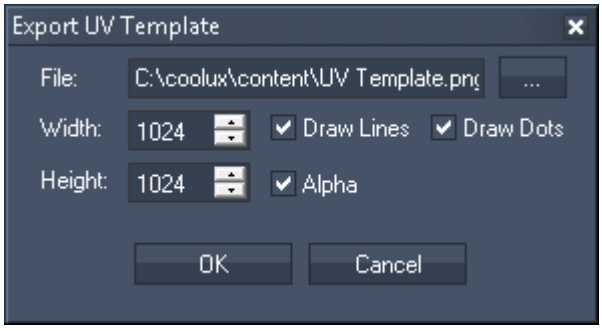


APPLY PLANAR MAP
This command applies the texture to the selected mesh in a way that it is first scaled to the current fullscreen resolution and then applied to the mesh without being influenced by its scaling, position or deformation. It stays planar.
One could imagine that the fullscreen texture is projected and burned into the mesh. If you now deform the mesh further, the deformation applies to the texture as well. If this is not required you may apply a planar map again.



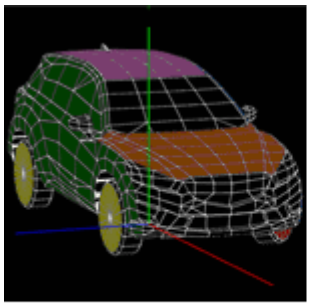
PLANAR MAP TO BOUNDS
This command applies the texture to the selected mesh in a way that it is first positioned and scaled to the bounds of the mesh. The size of the mesh is indicated in the left image by the white dashed line. Then the texture is applied to the mesh without being influenced by its deformation. It stays planar.
One could imagine that the resized texture is projected and burned into the mesh. If you now deform the mesh further, the deformation applies to the texture as well. If this is not required you may apply a planar map again.

MAPPING



If you are working with cylinder and sphere objects, select the object and apply the according special mapping techniques. The available ones include a cylindrical and spherical mapping. For more information please read the chapter "General 3D modeling terms [2133]". Please note that the mapping technique expects the object to be positioned at (X,Y,Z) = (0,0,0).

The command "Export UV Template" generates a template that can be used for creating content that should be mapped on the object. The feature unfolds an object, the result is a 2D image that can be colored in and later used as a texture on the object. For modifying the UV template you may use any graphics editing software of your choice as the UV template is saved as a PNG file.



an imported object with a texture that has been partly colored

an UV template can be used in any graphics editing software

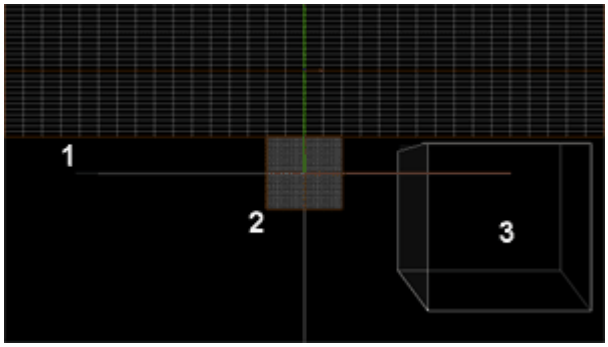
In the dialog click the [...] button to choose the directory where the PNG image should be saved.
The ideal width and height of the UV template depend on the object. The higher the resolution the more details can be shown but the more performance is drawn.
"Draw Lines" will show the edges of the faces. In the Warper you can not see faces, but the larger polygons when activating the Wireframe mode in the Texture tab [2155].
"Draw Dots" referes to the vertices themself which are shown as red crosses when activating them in the General tab [2151].

If ticking the "Alpha" check box there will a transparent background instead of a black background. Vertices and edges are always white.

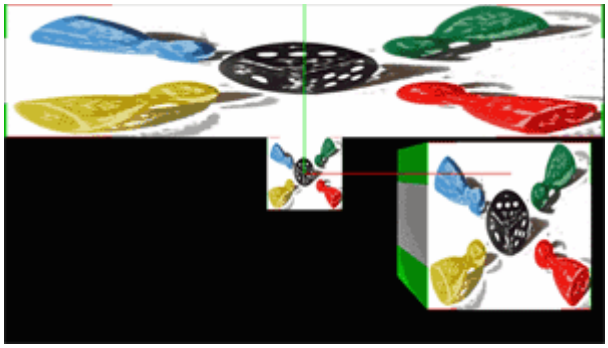
Apply Map to All

A UV map can cover several meshes and objects at the same time, that is, it applies to one mesh only partly. This is for example of interest when...
1) you like to show one layer on two separated screens
2) you have a complex screen that could be divided into several regions in order to make the warping process easier and faster. Imagine, that you want to project on a facade. Everything should be covered by the same picture. The facade probably consists of areas that are differently complex. Maybe there are quite plane areas and others, that need some more warping, e.g round pillars or stucco decoration. On these areas you can place meshes with a finer mesh grid and more FFD points. On the easier parts you can place more rough meshes. The alternative would be, that everything is covered by a mesh that has everywhere the same fine grid resolution. This would mean that you spend a lot of time on the easier parts too, as you have to work with unnecessary mesh points.

For both applications the following mapping commands can be helpful. For understanding the meaning and potential of planar mapping it is recommended to build some examples and try it out practically.

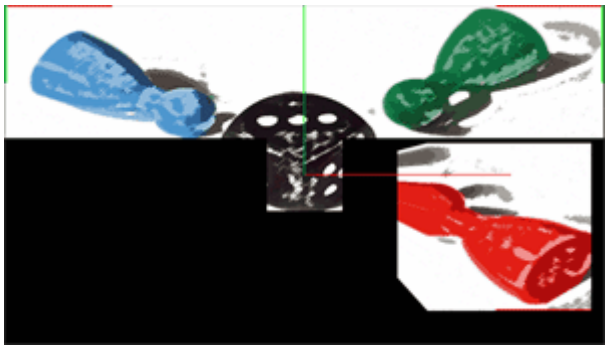


As an example, let's take this setup. At the top, there is one large mesh, a 2D plane, covering the entire fullscreen width. In the middle, there is another mesh, a square like 2D plane. As a third object, a cube has been imported as an X file and is positioned in the bottom right corner. The camera stays at the default position at (X,Y,Z) = (0,0,-25).



APPLY 1:1 MAP TO ALL
In the Texture tab you may find the button "Load Texture...".It applies a texture 1:1 like, that is, the texture is scaled and deformed to fit exactly onto each mesh and object that exists in your project. You do not need to select a mesh or object first.
The command "Apply 1:1 Map To All" always brings you back to that state. As seen in the cube object, the texture is applied in a planar way, the front face shows the image, the side face stretches the left pixel line. The same is happening on the right,top and bottom side.

APPLY PLANAR MAP TO ALL
This command applies the texture to all meshes and objects in a way that it is first scaled to the current fullscreen resolution and then applied to them without being influenced by their scaling, position or deformation. It stays planar.
One could imagine that the fullscreen texture is projected and burned into the meshes. If you now deform the meshes further, the deformation applies to the part of the texture as well. If this is not required you may apply a planar map again.



APPLY PLANAR MAP TO BOUNDS TO ALL
This command applies the texture to all meshes and objects in a way that it is first positioned and scaled to the bounds of the meshes. This can be best seen in the above described section "Planar Map To Bounds" where the size is indicated by a white dashed line.

Then the texture is applied to the meshes without being influenced by their deformation. It stays planar.
One could imagine that the resized texture is projected and burned into the meshes. If you now deform the meshes further, the deformation applies to the part of the texture as well. If this is not required you may apply a planar map again.

Planar Mapping Mode

An activated Planar Mapping Mode applies the chosen texture constantly onto all meshes and objects in the project. The texture is not influenced by their scaling, position or deformation, it stays planar until the mode is deactivated again.
This is useful for positioning and scaling meshes *before* the Warping process as described above [2144].

Marker

Please find a general description of markers in the topic describing the Marker tab [2161].

SHOW CALIBRATION OVERLAY
An activated Calibration Overlay is the default state. If you deactivate this option, the object will always be rendered in the Wireframe mode (i.e. the texture will be hidden) as soon as you enter the Calibration mode when working with Markers. This is useful when you want to position a marker without being distracted by the texture.

CALIBRATION SETTINGS
A new dialog opens where you can influence the calculation that is called to estimate the camera position after setting up markers [2161]. The calculation is based on an algorithm that runs a defined number of times until it finds a position for which the two instances of all markers match each other, allowing a certain accuracy. If the result does not meet your expectations you can set up, that the algorithm should run more often, e.g. to reach the expected accuracy. Whether a more accurate (less than 0.00010) calculation or a more tolerant one gives better results, depends on your setup.

CREATE TEST MARKER
This command creates four markers that can be used to estimate a camera position automatically as described in the Marker tab [2161].

DELETE ALL MARKERS
Choose this command if you like to delete all markers [2161].

RESET ALL 2D POINTS
After adding a marker to the project, a 3D point and a 2D point are generated. You may relocate the second instance, the 2D point, onto the real object (in the projected image). Choose "Reset All 2D Points" to discard this reposition and bring back the second instance. According to the current view, the 3D point and the 2D point share the same position again.

ADD TO SELECTED VERTEX
First, select a vertex from your 3D object, then choose this command in order to attach a marker to it. Instead of this workflow you may as well add a marker using the Marker tab [2161].

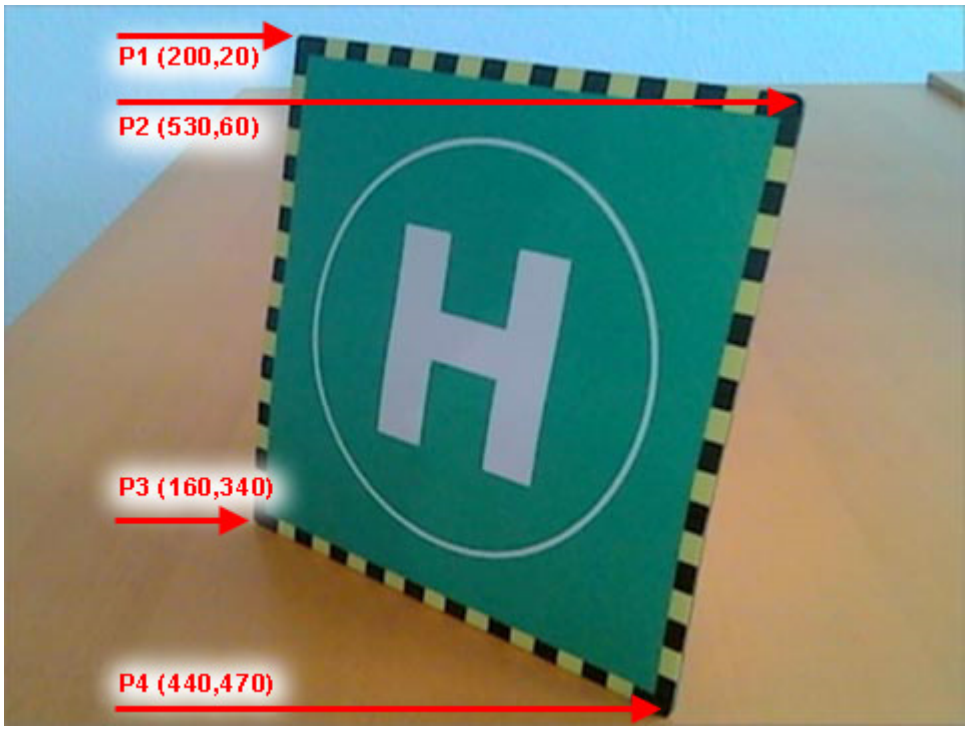
Clone Object

The command "Clone Object" allows to make a copy from the currently selected 3D object or 2D plane. If you are in the FFD mode, please make sure that all FFDs are selected. If you are in the Object mode, just select the object or plane that should be copied. Now, go into the Move mode and drag the object whilst holding down the right mouse key.

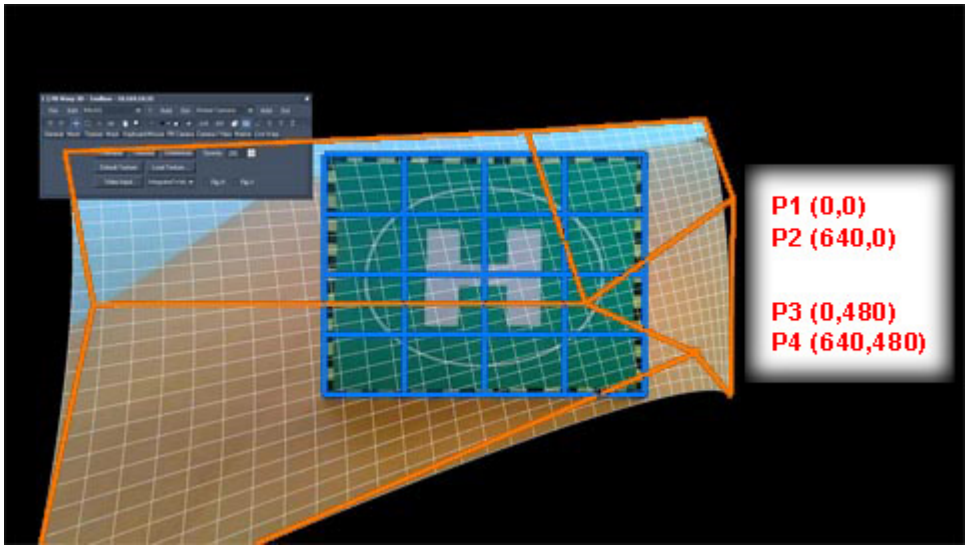
Options

In the Options menu you may find settings of interest when working with a camera tracking. A problem that you might encounter is, that the image seen by the camera is distorted due to a non-orthogonal camera position or due to lens deformation, the so called cushion effect. Same as with projectors, the (input) image needs to be keystoned or even warped. This is especially needed if the projector(s) is (are) not positioned on the same axis and output the image in a different angle.
The goal is, that the input coordinate of a tracked point can be translated into an output coordinate.

First, open the Warper on the computer where your Camera is attached to. On the default mesh, apply the video input image as a texture. You may do this is in the Texture tab [2155]. Now, open the Options menu and choose in the command "Show Camera Region" your camera resolution. Relative to your local screen resolution, a blue rectangle marks the camera size. Depending on the camera position and how it records the scene you need to deform the mesh in order to have a straightened tracked region as if the camera would look straight onto it without lens deformation, see the example depicted left. As a last step in the Warper, choose "Export Camera Lookup" and the directory whereto the *.dat file should be saved.



The camera records 640x480 px but the position is not orthogonal to the region that should be tracked. The corner points have coordinates that depend on the camera position. If a projector is not positioned where the camera is, it is very complicated to project a layer at the exact position from a tracked point. The input image needs to be keystoned and transformed.



The screen resolution in this example is 1920x1080 px. The camera region (the blue rectangle) is set up to match the input resolution of 640x480 px. The mesh's texture shows the Live Input. Then the FFD is used to deform it in a way that the tracked region fits the camera region. This is the same result as if the camera would have been installed at an ideal position.
With the lookup table, the tracked input coordinates can be translated into valuable output coordinates (for a layer).

Now, the lookup table can be used in the Widget Designer [786] for example. In the Camera Point Tracker tool [1291] you may find the check box "Use Lookup". Click the [...] button to choose the directory from the saved *.dat file. Press the "Apply" button.
If a person now walks along the top edge for example, the tracked camera point will report a movement in one axis only, e.g. from (X,Y) = (0,0) to (640,0). Without the lookup table the same movement would have resulted in an X position change and Y position change, e.g. (200,20) to (530,60). It would have been impossible to prepare a Widget Designer project without knowing the exact camera position and how the input image looks like. With the lookup table you may straighten the image according to the camera position and use prepared Region nodes [1092] or Custom Script Buttons [822] that are positioned in certain areas on your page. For example you can take a draft image from the scene and use it as a background image [805] for your Widget Designer page. With the lookup table you can as well use the Camera Input node [1067] and route the X,Y Pos directly into a Range node [1147] and PB Device Output node [1212] without further calculation.

Advanced Mode

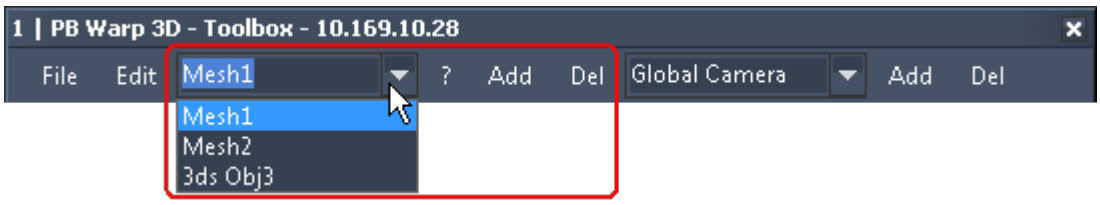
The Warper application starts per default in a basic mode, that is, certain menus, commands and tabs are hidden. Activate the advanced mode if you like to have access to all functions of the Warper.

Network Settings

If you're running the Warper on the same computer as your PB Master, the exported mesh or mask can directly be added into your current PB Project. To use this function, please make sure that the IP address and Domain of your local computer match the PB Masters settings.

The next time you export a mesh or mask, it will be added to your Pandoras Box project automatically.

9.11.2.1.3 Mesh Selection

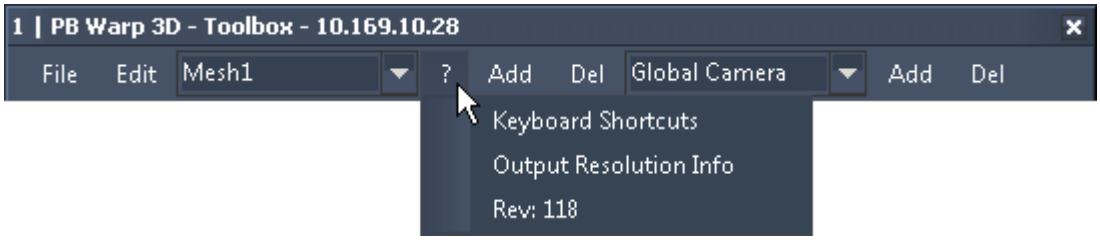


The Warper allows to deal with several mesh objects or 2D / 3D objects within one warping project. Choosing one of the existing objects from the drop-down list will select the object in the main window. Alternatively you may go into the "Object mode" [2149] (e.g. via the shortcut "O") and click on the object to select it.
The settings in the tab "Mesh" are stored per each individual mesh and apply as soon as another mesh is selected.

[Add]
If you want to add one more mesh object, press [Add]. A pop-up dialog allows to define the amount of FFD control points and the size for the new mesh. For more information, go to the chapter "File Menu [2138]".
The new mesh will be added to the main window.

[Del]
In order to delete one of the existing objects, please select it first and then press [Del]. A pop-up dialog needs a further confirmation for this action, as it can not be undone.

9.11.2.1.4 [?]-Button



Here you find a listing of all keyboard shortcuts available in the application. Alternatively, you may open this window using [F1].

The "Output Resolution Info" informs you about the current resolution, reported from the graphics card. Before you start warping please make sure that the resolution is the correct one as all meshes and settings relate to the absolute resolution!

Lastly, the current revision is depicted.

9.11.2.1.5 Camera Selection

The Warper allows to deal with multiple cameras within one warping project. This is needed when using the Warper for a 3D application, meaning to import a 3D object from the screen you are projecting on and to transfer the position from the real projector(s) to the virtual camera(s) rather than doing a 2D warping [2129].

Choosing one of the existing cameras from the drop-down list will activate it and change the current view according to the stored settings from the chosen camera.
The settings in the tabs "PB Camera [2159]" and "Camera / View [2159]" are stored individually per camera.

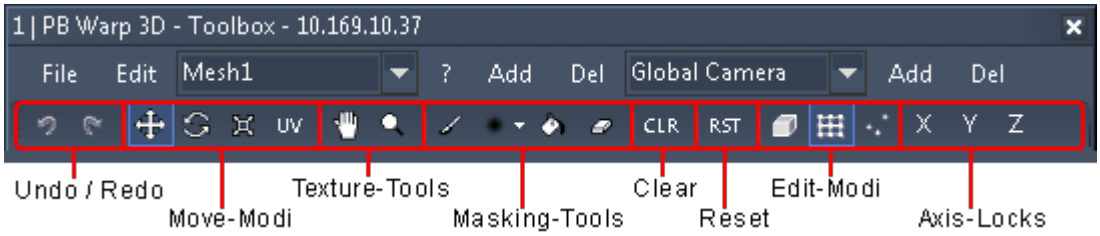
[Add]
Per default a camera with the name "Global Camera" exists. If you want to add one more camera, press [Add]. Any changes done in the tabs "PB Camera [2159]" and "Camera / View [2159]" are automatically assigned and stored to the selected camera.

As you can create as many cameras as you like, you can create the ones that you need for a 3D set up in Pandoras Box and in addition some cameras that can be used to get an overview from your 3D scene. For example, next to the "Global Cam" you may create a "Front View", a "Side View" and a "Top View". The front view is applied with a camera position (X,Y,Z) = (0,0,-25), the side view could be (25,0,0) and the top view (0,25,0).

[Del]
In order to delete one of the existing cameras, please select it first and then press [Del]. A pop-up dialog needs a

further confirmation for this action, as it can not be undone.
Please note that the "Global Camera" can not be deleted.

9.11.2.2 Tools Bar



Undo / Redo

Use this button to undo the last change done in the mesh, FFD or 3D object.
Please note: Undo only refers to moving tools. Changes done in the tabs themselves can't be undone. To enable the Undo/Redo function for the ScreenMask, please go to the tab and tick the check box

Use this button to redo the last undone change.

Move-Modi

There are 3 possibilities to transform the selected mesh / FFD point(s) or entire object: you can either...
move
rotate or
scale the selection.

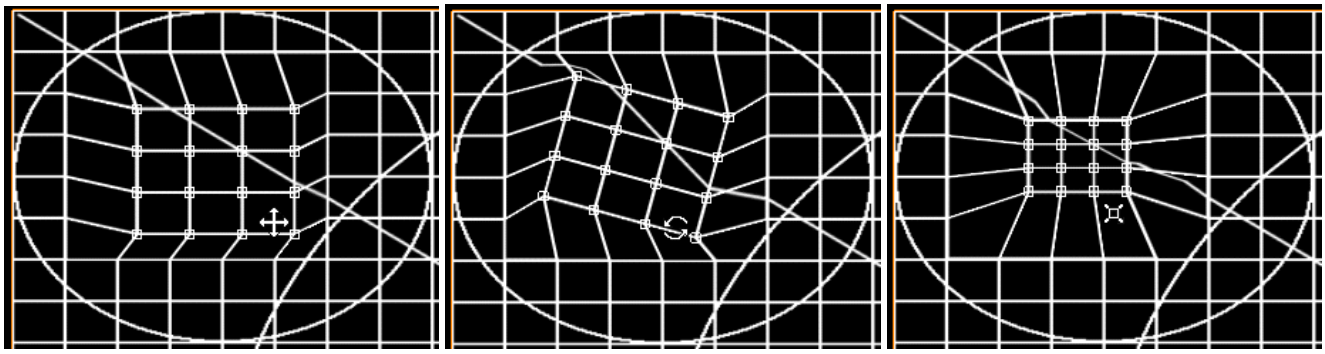
The fourth tool works for mesh points only! If you like to drag the entire texture, please use the tool "Drag Texture" as described below or the settings in the Mesh tab [2153]. The UV icon can only be seen in the advanced mode. To activate it, open the "Edit" menu and choose "Advanced Mode".
UV drags the UV map [2133] within selected mesh points.

Before moving points, decide whether you like to work with mesh points, FFD points or an object by switching to the according Edit mode [2150]. Then select some points:
- mouse: click on a single point to select it, hold down the [Ctrl] key additionally to multi-select points or drag a selection box
- keyboard: click the Up/Down/Left/Right key to select a single point and hold down the [Ctrl] key additionally to multi-select points

Now, there are 3 different ways of applying a transformation change:
- keyboard: you can move the selection with the shortcut [Shift + Up/Down/Left/Right arrow key]

- dialog: make a right-click on the move/rotate/scale icon to open the "Relative ... Dialog". There you can enter a specific numeric value and apply the change. Please note, that this works in a relative way, not absolutely! You can either undo the changes done in the dialog by hitting the Undo icon or by applying the inverse value, e.g after scaling by 50%, you may scale by 200%.

- mouse: you can move, rotate or scale the selection by holding the right mouse button pressed and moving the mouse up, down, left and right. You can switch from one transformation mode to another by clicking the respective button or by using the keyboard shortcuts. Press the following numeric characters on your main keyboard: [1] for moving, [2] for rotating and [3] for scaling. (When the Toolbox is active/selected, these shortcuts will not work. In that case, click into the Warper's main interface to deselect the Toolbox). If you like to transform along a certain axis, you may lock it with the according buttons X, Y, Z.



See here examples for a) Move Mode, b) Rotate Mode and c) Scale Mode.

Texture-Tools

Drags the UV map [2133] from the entire object. Please activate the Textured mode [2155] and then use this function to drag the visible texture on the mesh. In the Mesh tab [2153] you may adjust the UV map more precisely.

Zooms the UV map [2133] from the entire object. Use this function to zoom the visible texture on the mesh. Moving mouse up zooms in, moving mouse down zooms out. Again, the Mesh tab [2153] provides a more precise adjustment.

Masking-Tools

Please change to the Mask tab [2156] before using the following tools. This will show you a white texture instead of the texture chosen before.
The chapter covering the ScreenMask [2156] explains as well how to work with the mask in more detail.

Draws on the mask on your warping object which later will hide parts of the real texture.

Unfold the drop-down list to see all available brush types for the painting tool.

Fills the whole texture black, which means that the mask covers everything from the texture and nothing is visible. Alternatively you may use the button [Black Mask] in the ScreenMask tab [2156].

Erases the black parts and thus makes the texture visible again.

Clear

CLR After you have selected mesh or FFD control points, you can press this button to unselect the points again, that is to clear the selection.

Reset

RST Resets the position of selected mesh points. They will loose their offset (set up in the mesh mode) and go back to their origin position related to the FFD.
Please note: Changes applied to the FFD control points and the screen mask will not be reset!

Edit-Modi

Use these buttons (or the according shortcuts) to change between editing the object, the FFD or the Mesh. (When the Toolbox is active/selected, these shortcuts will not work. In that case, click into the Warper's main interface to unselect the Toolbox)

Objects can be selected and edited. (Keyboard Shortcut [O])

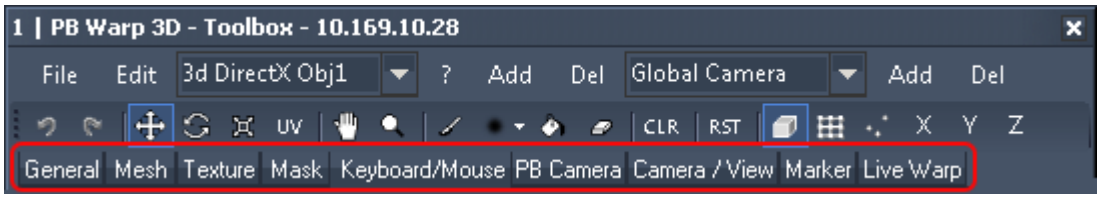
FFD control points can be selected and edited (Keyboard Shortcut [F])

Mesh control points can be selected and edited (Keyboard Shortcut [E])

Axis-Locks

Use these three buttons to lock or unlock the according axis. With a locked axis (highlighted button) any transformation like movement, scaling or rotation can solely be performed on this axis. The drawing tools are effected as well.
As an example, let's say you are in the standard camera view, have selected some FFD control points and move them with mouse. Moving the mouse horizontally makes the FFD points move along the X-axis whilst moving the mouse vertically makes them move along the Y-axis. With a locked X axis, the points can not move along the Y-axis howsoever the mouse is moved.

9.11.2.3 Tabs



The Menu Bar is divided into the following tabs: General [2151], Mesh [2153], Texture [2155], Mask [2156], Keyboard/Mouse [2158], PB Camera [2159], Camera/View [2159], Marker [2161], Live Warp [2164]

9.11.2.3.1 General

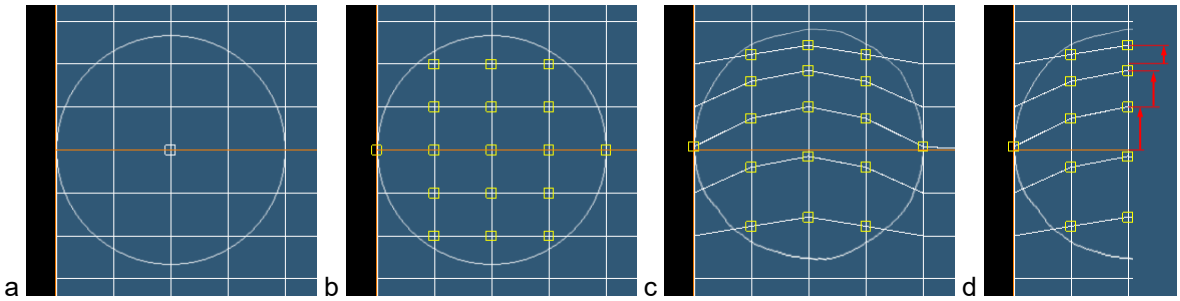
The "General" tab includes options that influence the appearance of the Warper interface. You may activate, deactivate or color certain visual objects. Some options are hidden in the basic mode and are only shown in the advanced mode. To activate it, open the "Edit" menu and choose "Advanced Mode".

Show IP and Screen ID

The "Show IP" check box displays the computer's IP address at the bottom left corner of the screen. This is helpful when working with several instances of the Warper.
The "Screen ID" check box displays the Screen ID in the center of the screen. Choose an ID number from the list on the right.

Soft Selection

Selects also neighboring mesh points within a particular radius. In the example depicted below, only one mesh point is originally selected (a), the surrounding once are soft selected automatically (b).
When the selection is moved (i.e. dragged using the mouse), the middle point is affected with the biggest possible offset. The further a point is away from the center, the smaller its offset gets (c and d). This is useful when you like to correct areas within your mesh without the result of having clear edges because the mesh points were not moved equably.
Set the radius with the numeric box. The default value for the radius is 1 generic unit [2132]. As a soft selection is affected differently than a normal multi-selection (where all points would move with the same offset), a soft selection is represented with another color. Per default it is yellow. If you like to change it, click in the color field and choose your color in the newly opened color dialog.



Show Crosshair

If this option is active, the position of the mouse cursor is clarified by showing a screen spanned horizontal and vertical line. You may set up how many pixels thick these two lines are and what color they have.

Backface Selection

When working with a three-dimensional object there might be vertices [2132] that can not be seen from the current view point as another part of the object lies in front of them. The default setting - an inactive backface selection - lets you only select those vertices that are visible in the current view. An activated backface selection will select all vertices within your selection rectangle (drawn with the mouse). This option might need more time depending on the number of vertices and the size of the rectangle.

Specular, Grid, Vertex and Background color

Click into the specular color field to assign another color for the global light source. The specular light reflection enhances the three-dimensional look when the textured mode [2155] is active.

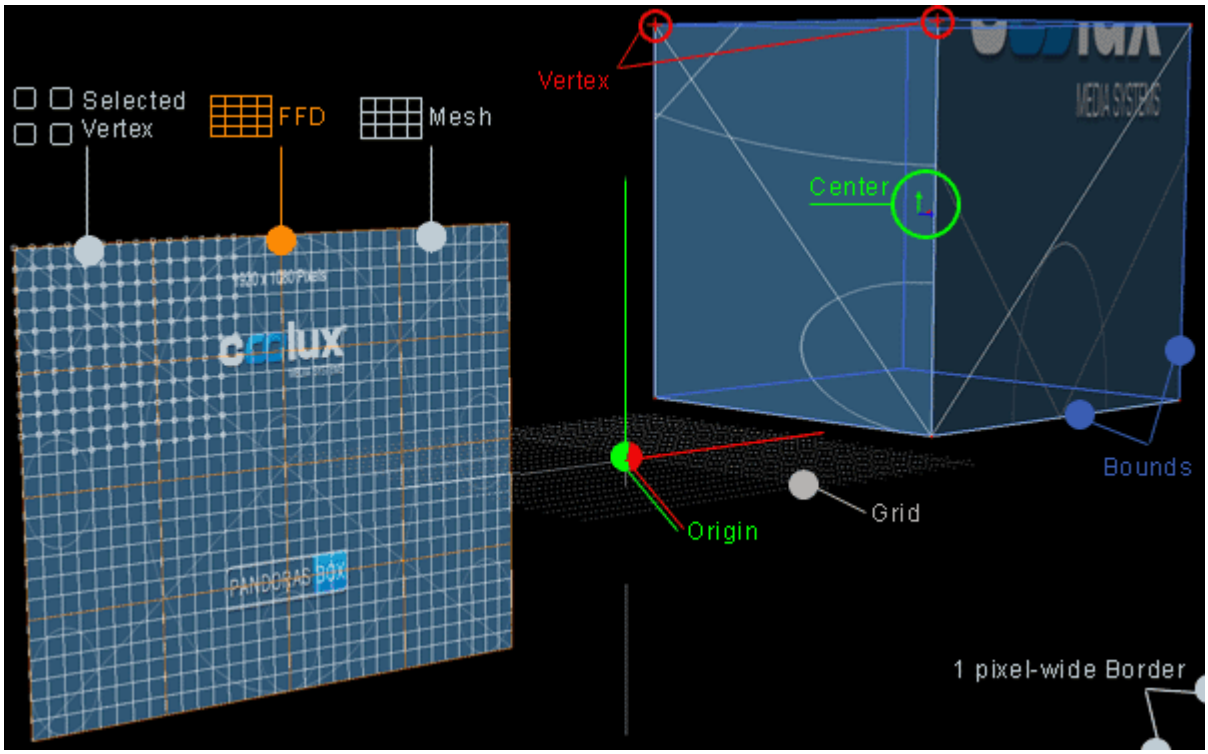
With these color fields you may change the color of the mesh lines. It is useful to work with different grid colors if you have an overlap between two warping grids. Once the lines match exactly, the color of the doubled grid lines change. This lets you work more precisely. Click into the leftmost color field to open a dialog that lets you pick a color of your choice. The five color field to the right apply the depicted color directly.

Click into the vertex color field to choose a different color for vertices / mesh points that are selected. Selected mesh points are indicated with a small rectangle.

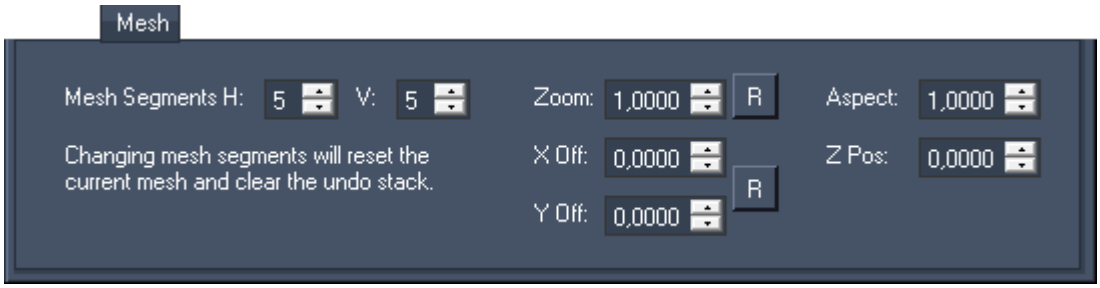
Click into the BG color field to choose a different color for the background of the Warper interface. Activate the option "Show Background" and click the [...] button to display an image instead of a solid color.

Check boxes

Check / uncheck the boxes to show / hide different visual auxiliaries.



9.11.2.3.2 Mesh



The "Mesh" tab allows to setup more mesh segments and to influence the UV texture map applied automatically to it. Please note that changes can only be applied to original meshes, not to imported objects.

Mesh Segments

Enter the amount of horizontal and vertical segments you want to use in your mesh. For more details, there is a tutorial [2166] at the end of the Warper chapter.

Important:
Adjust the amount of mesh segments BEFORE you modify the mesh, because this will reset any changes you did to the mesh before! There is no [Ctrl + Z] function. Nevertheless you may modify the mesh count as long as you are working with the FFD only.

UV Texture Mapping: Zoom, Offset and Aspect ratio

These settings influence the UV texture map [2133] and can also be adjusted using the Texture tools [2150] from the tools bar. Please activate the Textured Mode [2155] first to see the changes done here.

ZOOM and ASPECT
Use the zoom option (or the Texture Tool from the Tools Bar [2149]) to zoom into the selected texture. The zoom values range from 1 to 99. The texture is enlarged while retaining the aspect ratio given by the mesh size.

A change in the aspect ratio, influences the texture's height only. The aspect values range from 0 to 99999999.
Press R to reset the value for the zoom and the aspect ratio to the default value 1.

X and Y OFFSET

The X- and Y-Offset (or the Texture Tool from the Tools Bar [2149]) enable you to move the displayed texture in both X and Y position. The possible values range from -1 to 1, whereas 0.5 would mean for example that the texture is moved half the way to the right and upwards.

Press R to reset the X- and Y-Offset to the default value 0. Please note that the Z-Position will be set to default as well!

Z Position

The Z position does not influence the UV map but the position from the mesh / object. According to the coordinate system [2131], positive Z values move the object backwards whilst negative values move it forwards. To change the Z position is especially important when objects overlap each other.

As seen in the example below, overlapping meshes with same Z-Order values will result in Z position fighting when the objects (exported as one object) are used in Pandoras Box. As this issue also depends on the camera position, it can happen that you see a Z fighting problem only under certain circumstances and maybe not in the Warper itself but with in the exported file in another program.
To export positions that can be clearly assigned, you have to define an unambiguous Z-Order (for the overlapping meshes).

A new warping project with two meshes is created. The meshes overlap each other partially. The Textured Mode [2155] is active and Mesh 1 is selected.

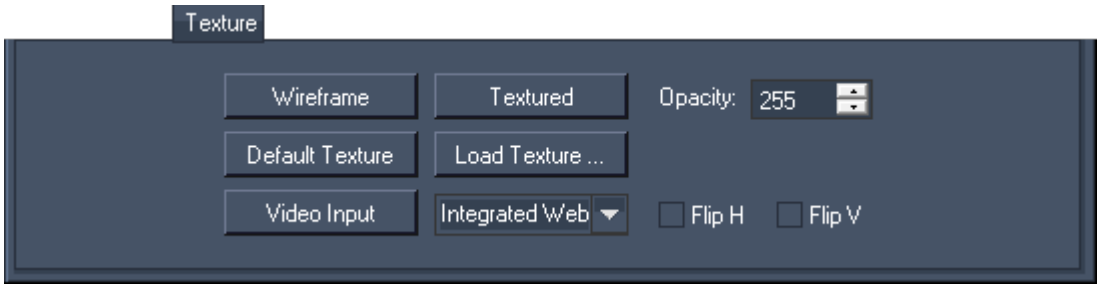
Now Mesh 2 is selected, and therefore overlaps Mesh 1.
As long as the Z-Order for none of the meshes is modified, the selected mesh will always be displayed in front of the not-selected one

The two meshes are exported using the command "Export All" and assigned to a layer in Pandoras Box. Both meshes share the same Z position, withing the overlapping area Pandoras Box can not clearly decide which polygons from which mesh are in front of other ones. The result is that both meshes intersect.

To create a clear assignment, Mesh 2 is brought to the front of Mesh 1 by changing its Z position to a lower value,

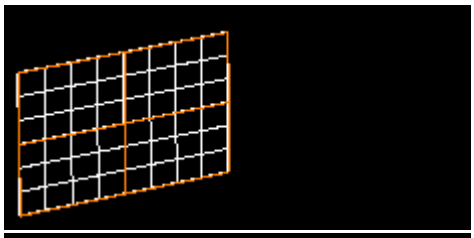
Now, it is irrelevant which mesh is selected in the Warper. Always, Mesh 2 covers Mesh 1.Also, the export in Pandoras Box shows no Z fighting any more.

9.11.2.3.3 Texture

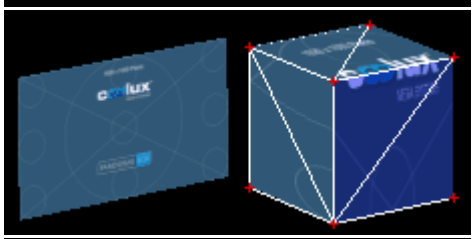


The "Texture" tab includes options that influence how meshes and objects appear in the Warper in terms of whether they are rendered with a texture or just in a wireframe mode. Some options are hidden in the basic mode and are only shown in the advanced mode. To activate it, open the "Edit" menu and choose "Advanced Mode".

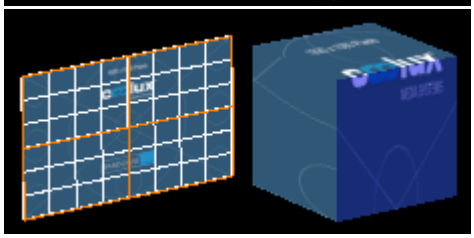
Wireframe versus Textured Mode



Click the "Wireframe" button to render meshes and 3D objects in a wireframe / grid mode only. Only the object that is currently selected is visible. Make sure that the check boxes "Mesh" and "FFD" are activated in the General tab [2151].



Click the "Texture" button to change to the additional Textured Mode. All objects are then rendered with a texture applied to them. The object that is currently selected is rendered with the wireframe on top.



In the Textured Mode you will see the same texture on all objects. You may choose between the default texture, a Pandoras Box testpattern, a picture of your choice or a live input:
- Click the "Default Texture" button to return to the default view
- Click the "Load Texture" button to browse to a picture of your choice
- Click the "Video Input" button and choose an available input from the drop-down list. If needed you may flip the input horizontally or vertically

All textures can be rendered with a full opacity (value 255) or a lower one.

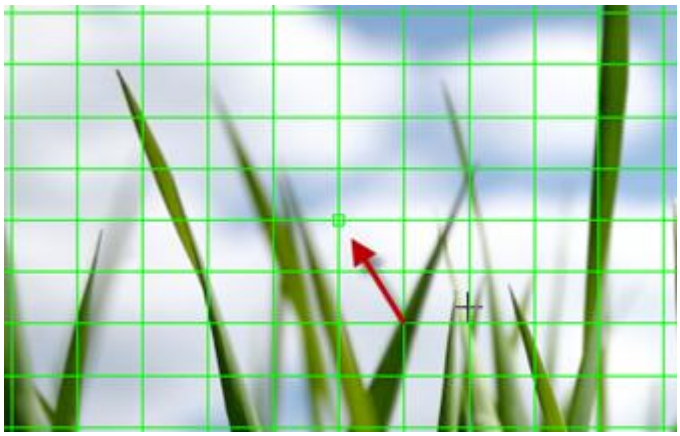
The Shortcut [G] toggles between the Wireframe/Grid and the Textured View.

Toggle Face Visibility

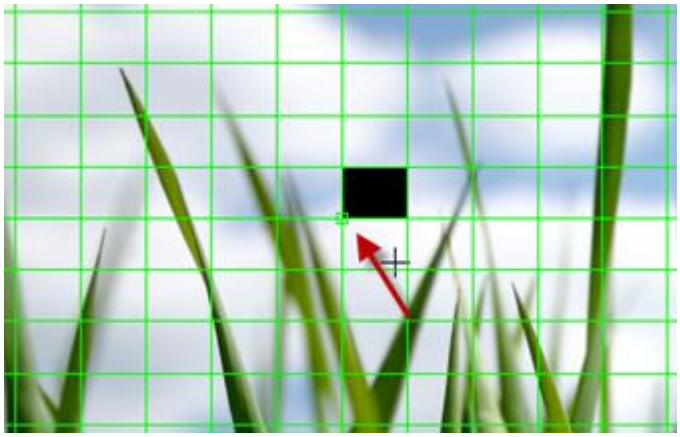
This is another function that should be mentioned in this Texture Section.
With the keyboard shortcut [D] you are able to toggle the visibility of selected faces. This could be handy if there are parts in the projection where you do not want to have anything textured, e.g. when there is the need to cut out doors on a stage design.

As you just toggle hard edged faces, this is more a rough masking function. To create a finer mask for your screen, please refer to Mask tab [2156].

Please have a texture loaded to your mesh. To turn a face (the area surrounded by four mesh lines) to be invisible, select the mesh point that is down left of the face and press [D]. Turn several faces to be invisible when selecting more mesh points.



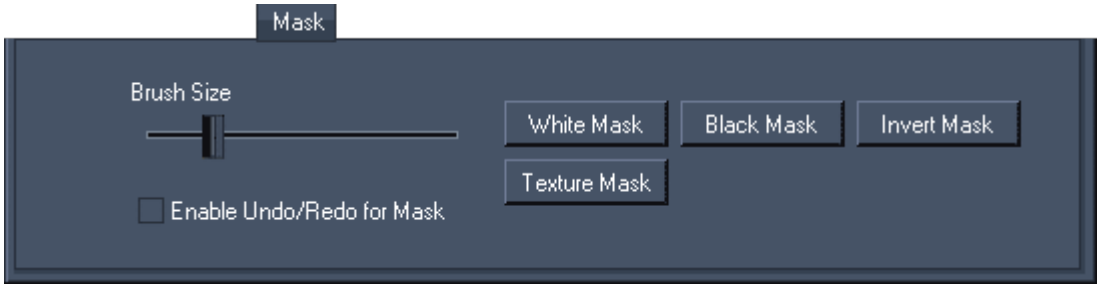
One mesh point is selected (red arrow is pointing at it).



After pressing the shortcut [D] the face top right of the selected mesh point turns invisible.

To reload the texture later on, select the same points and press [D] again.
To reload all hidden faces without selecting the according mesh points, use ‘Activate all Vertices' from the Edit Menu [2142].

9.11.2.3.4 Mask



The ScreenMasking Tool allows you to mask while painting onto the projected surface from within the Warper. After finishing the mask, export it as PNG file and use it on a high layer in Pandoras Box. This way the mask will perfectly match your warped surface. It is as well possible to open the PNG file with other graphical programs.

When changing to the ScreenMask tab, the grid disappears and you see a white plane instead. Everything what is white will turn transparent in the exported PNG file, i.e. it will not be visible and thus not hide underlying layers.

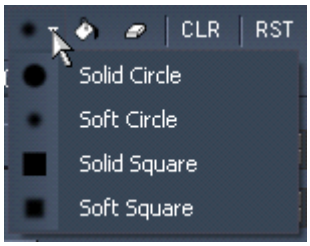
Now, select the Paint Mask Tool  from the Tools Bar [2149] and draw onto the background by holding the left mouse button clicked. Everything what is painted will be black, also in the exported PNG file. Thus it will cover underlying pixels when being exported to Pandoras Box.
The axis locks [2151] can be useful when wanting to draw a straight line.

If you want to remove parts of your paintings again, select the Erase Tool  from the Tools Bar and move your mouse with the left mouse button clicked.

Please note that due to the fact that each object within the project is applied with the same texture, all objects will as well show the same mask. If you would like to create individual masks for each mesh or 3D object, please save the project after finishing the first mask and apply a new mask to all objects to start with the second mask.

Per default the undo function [2149] does not cover the changes done in the mask as this consumes much memory space. However, if you like to use the undo option, tick the check box "Enable Undo/Redo for Mask".

Brush type and size



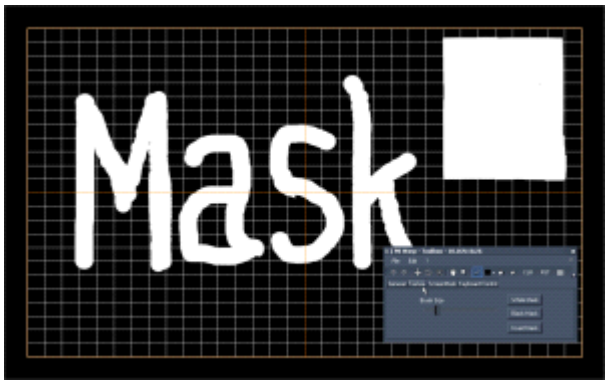
Choose a brush from the drop-down list in the Tools Bar.

Define the Brush Size by moving the slider left and right in the Mask tab.

Mask Buttons

Click the "White Mask" button to get a plain white mask (all parts are transparent). Paint the parts that should be masked with the Painting Tool  .
Click the "Black Mask" button to get a plain black mask (all parts will be masked). Erase the parts that should NOT be masked with the Erase Tool  .
Click the "Texture Mask" button to use the normal texture as a mask background. You can set the texture in the Texture tab [2155]. According to the brightness of the texture's pixels, the exported mask will be transparent, partly transparent or nontransparent.
Click the "Invert Mask" button to invert the mask. An example could be the right image (b) as an inverted mask from the left one (a).

a) Black parts painted on white background



b) White parts erased from a black background

Using the mask in Pandoras Box

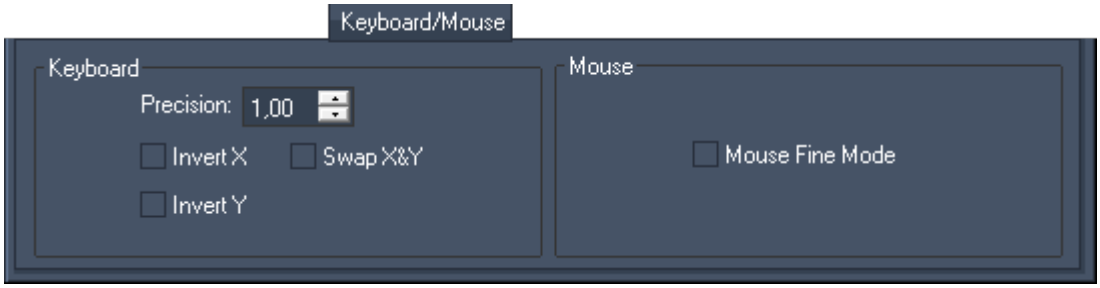
First of all the mask needs to be exported. Click on File > Save mask, browse to the location where you want to store your mask-file and name it. A PNG file will be created.
Use the mask on a layer in Pandoras Box that lies in front of other layers. It is a good practice to use the very last layer (before the Camera layers) and to rename this layer.

a) The mask from the example above is used in Pandoras Box on top of the content layer.

b) The inverted mask from the example above is used in Pandoras Box on top of the content layer.

The great benefit of this masking tool is the live painting on the projected surface, because this mask will exactly fit to your warping file.

9.11.2.3.5 Keyboard / Mouse



Keyboard

Use the keyboard control settings to adjust the keyboard behavior for editing the mesh.

PRECISION
The precision value ranges from 0,01 to 99, whereas the default value is 1 which means that one keystroke results in a movement of 1 pixel. When doing a rough Grid / FFD adjustment start with less precision (> 1) to save time. Later on the fine adjustment can be done with a higher precision (< 1) if needed.

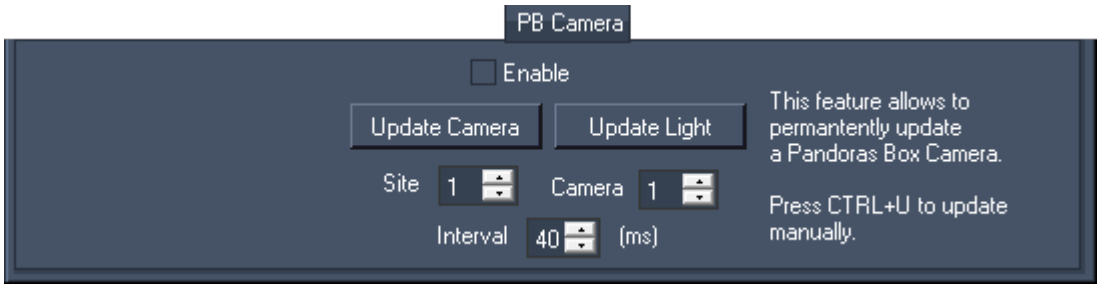
INVERT X AND Y
When warping a rear projection and / or the picture is upside down, inverting X and / or Y movement can help you adjusting the mesh with the keyboard.

Mouse

Use the mouse control settings to adjust the mouse behaviour for positioning the camera.

Tick the check box "Mouse Fine Mode" if you like to shift the camera position with more precision. You will see the result in the Camera / View tab [2159].

9.11.2.3.6 PB Camera



The tab is hidden in the basic mode. To activate the advanced mode, open the "Edit" menu and choose "Advanced Mode". This is only available in the Warper 3D edition.

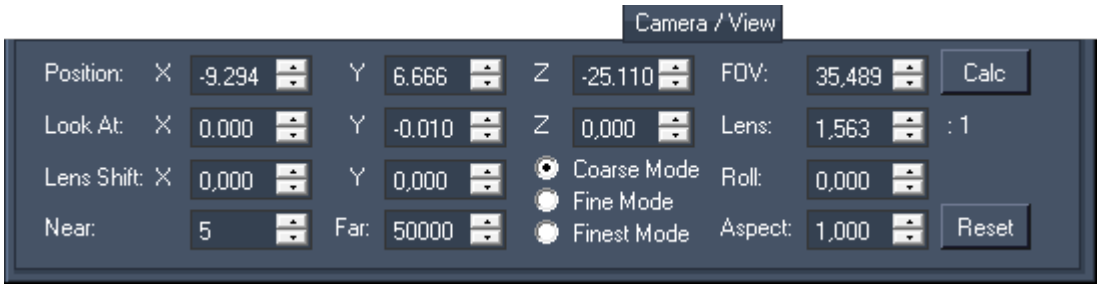
The "PB Camera" tab enables you to transfer values from a camera set up in the Warper to a Camera [675], Output [682] or Light Layer [666] in Pandoras Box.

First of all, choose the camera from the Warper from which you would like to transfer data. Once you have picked a camera via the drop-down list [2148] at the top right of the Toolbox, make sure that the settings in the tab "Camera / View" [2159] are correct. Now set up the connection to a Pandoras Box Master system (PB) via Edit > Network Settings [2147].

Click the "Enable" check box to send values permanently from the Camera / View tab [2159] to the Output or Camera layer in PB. The site and layer ID can be set up with the numeric boxes. The interval can be adjusted with the numeric box at the bottom.

If you do not want to update the layer permanently you may click the button "Update Camera" instead. This will transfer the current values only once. As an alternative, you may press [Ctrl + U].
If you like to transfer the values to a Light Layer, click the button "Update Light".

9.11.2.3.7 Camera / View



The "Camera / View" tab is hidden in the basic mode. To activate the advanced mode, open the "Edit" menu and choose "Advanced Mode". This is only available in the Warper 3D edition.

The tab enables you to adjust the position, orientation and lens settings of a camera in the Warper. This is needed when you do 3D Warping as described in the introductory chapter [2129]. A camera can be added and selected in the camera selection field [2148] at the top right of the Toolbox. Choosing a different camera will change the currently seen settings in this tab.
Please read the last chapter to learn how to transfer these settings to a camera in a Pandoras Box [2159] Master system. The next chapter covers a marker based camera calibration [2161].

Meaning of parameters and how to measure them for 3D Warping

Position, Look At, Roll
The six parameters for the X, Y, Z Position from the camera and its "Look At" target plus the roll parameter define where the camera is positioned, where it is looking and how it is orientated.
As said in the introductory chapter where 3D Warping is compared with 2D Warping, you need to adjust the camera's and the object's position and orientation as well as the lens setting. Regarding the position and orientation, there are two possible ways to do so: manual measurement or marker-based calibration [2161].
To measure distances and sizes manually, you should first define where the zero point of the coordinate system is; to simplify the process you can define that it lies on a certain corner of the object or projector. Based on that

zero point, you need the relative X,Y and Z distances to the screen (exactly where the object's pivot point[2132] lies) and projector lens (center of the lens). Then you need the size of the real object to convert the measured "number" into the unit that has been used for the size of the virtual object. If the virtual size is 2 GU, and the real size is 200cm, the conversion factor is 1 GU / 100 cm. Thus, you know, that a distance of 150cm is 1.5 GU (distance * conversion factor).
Now you can calculate the XYZ position of the camera and the object (corresponding with the Look At Target) as well as their orientation. The orientation from the object results from its XYZ angle. The orientation of the camera is the result of the difference between the position and the Look At target plus the Roll parameter

FOV, Lens

The parameters FOV and Lens express the same setting: the opening angle. When you enter the FOV, the Lens parameter is automatically calculated and vice versa.
There are two ways to find out the opening angle: manual measurement or iteration.
For manual measurement you need the width of the projected image plus the distance between the center points of the lens and the image (at the depth where the width was measured). These two numbers can be entered in the FOV calculator available with the Calc button. The measuring unit does not matter as long as it applies to both values. Click "Apply" to transfer the calculated value to the main tab.

Alternatively you can simply try out and approximate what the lens factor is. If you have a lens with a fixed zoom, use this lens factor and try out whether a slightly higher or lower number gives better results. (Even fixed zoom lenses, can slightly change due to mechanical vibrations.) If the lens has a variable zoom, start with the smallest lens factor and increase it slowly. It is clearly visible in the projected image whether you are approaching the correct factor or not.

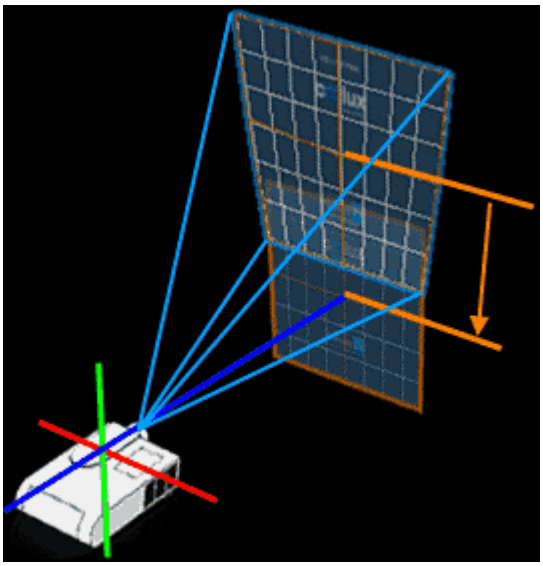
Aspect

The Aspect parameter expresses whether the cameras aspect mode is the same one as the output's aspect mode.

Near Plane, Far Plane

Near and Far plane define what part of the cameras cone is rendered. If an object lies outside of the range, it is not seen and rendered by the camera. In 99.9% of all cases the default values give good results and should be not be changed.

Lens Shift



The Lens Shift parameters define how much a projector is lens shifted. For 3D Warping[2129] (either measuring or marker-based) it is strongly recommended to reset the projector(s) lens shift to 0,0. However, some projectors have a build in lens shift that cannot be adjusted. You can see that easily by looking onto the lens if the projected image is in the center of the lens. If it is outside of the center, there is an active lens shift. Or, you can see it from the side: check whether the lens' center matches the center of the projected image. In the left image, one can clearly see that the projected blue outlines image is above the projector itself. If the projector had no lens shift applied, the image would look like the one with the orange outline.
In the case that the lens shift cannot be adjusted in the projector, use the lens shift parameters in the Warper. In practice you can measure the lens' center to the floor and mark it on your screen depending on the projectors angle, or you can use a laser. At the end you should be able to know where the unshifted image center should be.

Then adjust the lens shift so that the center of a default mesh is at the same height and width as the marked lens center. That could mean of course that parts of the mesh are outside of the projected image. This will be corrected after adjusting the position and orientation of the camera.

The Reset button resets all parameters. Please note, that there is no undo function!

Moving a camera using the mouse

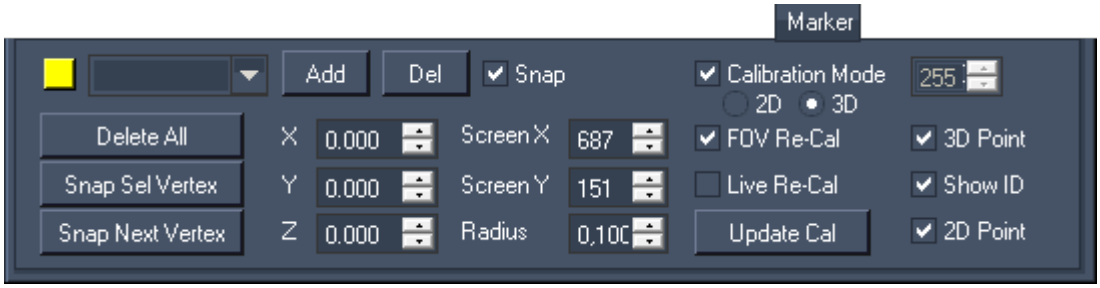
- Zoom: Scrolling with the mouse wheel influences the camera's Z position
- Rotation: Moving the mouse whilst holding the [Alt] key and the middle mouse button influences the camera's X and Y position. As all other parameters are not influenced this has an effect as if the camera rotates around a fixed point.
- Pan: Moving the mouse whilst holding the middle mouse button influences the camera's X and Y position and the "Look At" position at the same time.

For a finer adjustment, activate the option "Mouse Fine Mode" in the Keyboard / Mouse tab [2158].

Adjusting a camera using the numeric boxes

At the same time you may set up these and other settings, e.g. lens settings, with the numeric boxes in the tab. You may enter a number directly or increase / decrease it. For this you can either use the up / down buttons next to each field, or click once into the numeric box and scroll using the mouse. The in-, decreasing adjustment can be made more or less precise with the radio buttons: Coarse Mode, Fine Mode and Finest Mode.

9.11.2.3.8 Marker



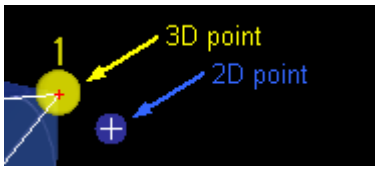
The "Marker" tab is hidden in the basic mode. To activate the advanced mode, open the "Edit" menu and choose "Advanced Mode". This is only available in the Warper 3D edition.

The tab enables you to add and position so called markers. As well view settings concerning the marker representation can be found here. There are more commands under Edit > Marker [2145].

Please read the previous chapters to learn what the Warper camera's parameters [2159] are and how to transfer them to a camera in Pandoras Box [2159].

General meaning of markers

Markers enable an automatic calibration of a camera's position and orientation (but not its lens settings!). A minimum of 4 markers is needed to adjust a camera in a 3D environment. The marker-based calibration is an alternative to manual measurement (of the projector's and screen's XYZ position) and data transfer to the parameters in the Camera tab. Both ways are attempts for a virtual 3D representation of the real 3D setup. Please keep in mind that this is not always necessary. Especially when working with only one camera or without 3D objects you can realize warping as well by straightening the projected image with a 2D mesh. This workflow is described in more detail in the introductory chapter [2129].



The calibration is based on the position of the markers. When creating one marker, a 3D point plus a 2D point are added to the project. At first, the two points share the same position but when you change the view by modifying the position/orientation of the camera you will see that there is a difference. The 3D point - per default a yellow circle with an ID - is "glued" to the object, thus it changes the position on the screen. The 2D point - a smaller blue circle with a white cross inside - has no reference to the object but stays on the same position of the screen as if painted on it. This explains the name "2D" it has no Z Position.
In short: Whilst the 3D point has its place on the virtual object, the 2D point needs to be placed on the same spot but looking at the projected image, i.e. the real model. After positioning the markers (placing the 3D and 2D points) the calibration finds the best camera position where both points share the same position again.

How to warp with markers

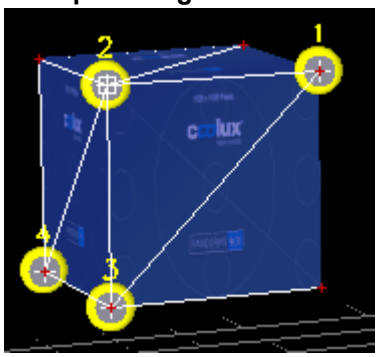
Before you start warping, you might want to setup an additional screen to have an better view of the Warper interface (as the projected interface is obviously distorted). You could either connect a second display and set the display driver to clone/duplicate the screen. Or you could connect another computer via VNC to the computer that is connected to the projector and that runs the Warper.

Import the 3D model from the screen / object you are projecting on. It must match the real screen / object as accurately as possible. However, if changes were applied to the real object after the virtual model has been created or if it simply does not fit, you can adjust the model by moving its vertices [2132]. If you are projecting on a two-dimensional screen you can create the object in form of a mesh in the Warper directly. Choose File > New and set a size according to the real screen's dimensions.

Open the "Edit" menu and activate the "Advanced Mode". Adjust the object size if needed (as described in the Tools Bar [2149]) and the view in the Warper (as described in the previous chapter [2159]) so that the interface looks very roughly like your view onto the object.

Now you can start adding markers. A minimum of 4 non-planar markers is needed when working with a 3D object. "Non-planar" means that one point must not lie on the plane that is defined through the other 3 points. When working with a 2D object, on the other hand, a minimum a 4 co-planar markers is needed. "Co-planar" means that all 4 points must share the same plane. In general the best results are achieved when placing the markers "far away" from each other.

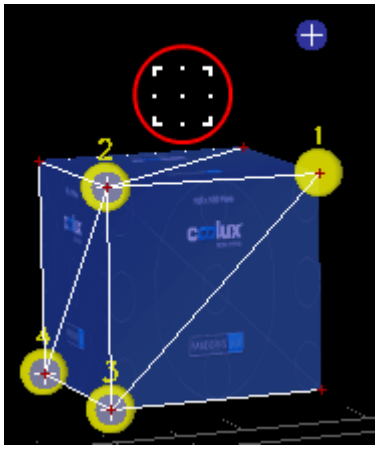
1. step: adding markers

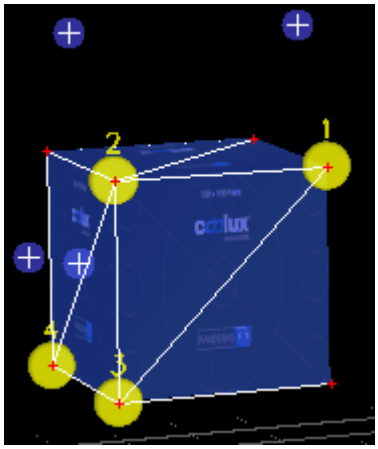


According to your view onto the object, decide where you want to place the 4 markers. You can do this in two ways.
a) Click on the "Add" button. The mouse cursor becomes a thick plus (+) icon when the mouse is within an object. Click onto a vertex to add the marker there. If you have many vertices and the marker did not snap to the right one, you can select the right vertex and click the button "Snap Sel Vertex"
b) Select a vertex by clicking on it directly or by selecting it through a selection rectangle. Now, choose "Edit > Marker > Add to selected vertex" to add the marker there.
You may modify the view, rotate or zoom, at any time. This is useful when working with an object with many vertices, as it can be tricky to pick a certain one. If you are done with the fourth marker, adjust the view so that you can see all markers again; if you like to relocate all 2D Points to the position of the 3D points, choose "Edit > Markers > Reset all 2D Points"

Now we have placed the markers, or rather the yellow 3D points of them. The next step is to position the blue 2D points.

2. step: dragging the 2D points





In order to get access to the 2D points, tick the check box "Calibration Mode". The view varies: the grid and 3D points are dimmed and the mouse cursor changes to a cube-like cursor as highlighted in the left image. If the texture distracts you, you can dim it with the number field to the right or toggle it off with "Edit > Marker > Show Calibration Overlay".
Underneath the check box there are two radio buttons "2D" and "3D".
- Pick "2D" if working with a two-dimensional object and having placed the four markers in a co-planar way.
- Pick "3D" if working with a three-dimensional object and having placed the four markers in a non-planar way, as seen in the example.

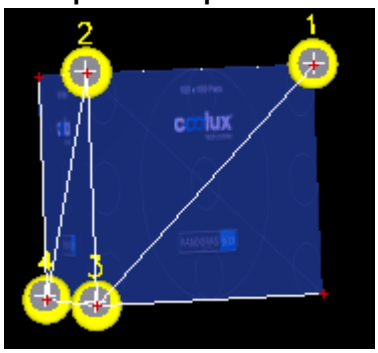
Again there are multiple ways how you can influence the position of the 2D points. Please note that the camera view must not be altered from now on.
a) Use the mouse, especially for the first rough setup. Click with the cube-cursor onto the first marker (i.e. the yellow 3D point) to select it. The blue 2D point jumps onto the cursor as soon as it moves a little and can now be dragged. Move the mouse until you see on your real screen that the 2D point is at the same spot as the 3D point is on the virtual model.

b) Use the shortcuts [Shift+ up/down/left/right arrow key] to move the selected marker. To select a marker you may either click on the according yellow 3D point (but do not move the mouse!) or select another marker by changing the drop-down list and them cli.

c) Use the number fields "Screen X" and "Screen Y", especially for fine-tuning.

If you have difficulties locating the cursor (and the attached 2D point) it could be helpful to activate the "Show Crosshair" option in the General tab [2151]. At the end all four 2D points are moved to the according spot from the real screen object. In other words, the 2D points have formed a specification for the camera position that needs to be found in the next step.

3. step: the Warper calculates a new camera position



According to the new position of the 2D points the camera can now be calculated. Click the button "Update Cal".
Now an algorithm runs over and over and returns the best possible position for the camera. From this position the camera sees the object just like the real projector "sees" the real screen. As a result the virtual object with its "glued" yellow 3D points is depicted in the camera view in such a way that the blue 2D points share the same position again.
If the result does not meet your expectations, please check the following options.

- Are the lens settings set up in the correct way?
Please select the Camera/ View tab [2159]. If you have a projector with a fixed lens shift or if you cannot shift it to 0,0 for any reasons, enter the lens shift (X and Y) in the according fields and run the calibration again by clicking "Update Cal". The algorithm now considers this offset in the calibration.
Please check as well whether the FOV accords with the lens factor of the projector lens. How to measure it is explained in the section "FOV / Lens". You do not necessarily need to click the "Update Cal" button. With an active Calibration Mode, you can tick the check box "FOV Re-Cal" and now, the calculation is done each time when a new FOV is entered. This way you can find the correct settings in just a couple of seconds.

- Is the 3D model correct?
Obviously the calculation depends on the accuracy of the 3D model. If changes need to done to the model you can select a vertex and move it to the correct position. You may transform a selection of multiple points as well.

- Are the 2D points positioned correctly?
Try to find a better position for the 2D points (using the mouse or the number fields as described above). You do not necessarily need to click the "Update Cal" button each time. With an active Calibration Mode, you can tick the check box "Live Re-Cal" and now, the calculation is done each time when a new position is entered. This way you can find the correct position in a very short time. Due to the cushion effect from the lens it's quite possible that the 2D points must be placed slightly outside the object.

- Have you tried adding more markers?
As said above the minimum marker number is 4. Adding more markers can give better results. When working with a 3D model, place additional markers outside of planes defined by three other markers as then a plane is overdetermined.

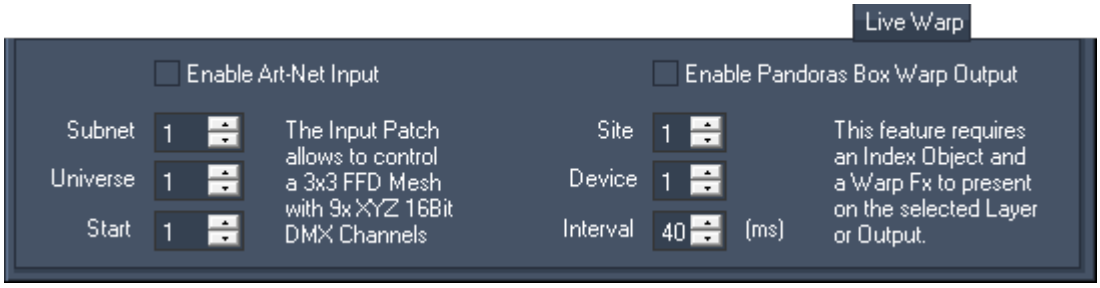
- Does the projector lens have a strong cushion effect?
If the projector distorts the lines due to a strong cushion effect you may apply a mesh to the output layer that evens it out. A good way to generate the needed mesh is to first apply all settings to the PB camera that were found with the current project. Then you may open a new project with a default mesh and use the Live Warping [2164] feature.

- Have you tried other options for the algorithm?
Under "Edit > Marker > Calibration Settings [2145]" there are options that adjust the algorithm.

Meaning of parameters, options and buttons

yellow color field	click into the field to open a dialog where you can pick another color for the 3D points and their ID
drop-down menu	allows to select another marker to adjust the according position of the 3D (parameters X Y Z) and 2D point (parameters Screen X and Y) as well as the 3D point's color and size (parameter radius)
button Add	turns the mouse cursor to a thick plus (+) icon and generates a marker when clicked
button Del	deletes the marker chosen in the drop-down menu; this command cannot be undone!
check box Snap	when a marker is added to the object with an activated Snap function, it will snap to the nearest vertex
button Delete All	deletes all markers; this command cannot be undone!
button Snap Sel Vertex	snaps the selected marker (via the drop-down menu) to the vertex that has been selected
button Snap Next Vertex	snaps the selected marker (via the drop-down menu) to the nearest vertex; if a marker is already located on a vertex it will not change its position
number fields X, Y, Z	position of the 3D point of a marker
number fields Screen X, Screen Y	position of the 2D point of a marker
number field Radius	size of the 3D point (in generic units)
check box Calibration Mode	activates the Calibration Mode (and its overlay), i.e. the mouse cursor changes to a cubic-like icon that allows to drag the blue 2D points; only with an activated Calibration Mode it is possible to activate the FOV Re-Cal and Live Re-Cal
radio buttons 2D / 3D	must be chosen before the calibration is started; - pick "2D" if working with a two-dimensional object and having placed the four markers in a co-planar way; - pick "3D" if working with a three-dimensional object and having placed the four markers in a non-planar way, as seen in the example above
check box FOV Re-Cal	enables to "interactively" change the FOV in the Camera / View tab; each time another value is entered, the calibration runs and returns the new camera position
check box Live Re-Cal	enables to "interactively" change the 2D points; each time a new value is entered and each time a 2D point is dragged, the calibration runs and returns the new camera position
button Update Cal	starts the calibration. An algorithm runs and estimates the best camera position depending on the position of the 2D points, as well as on the lens shift and FOV set up in the Camera / View tab
number field to the right	changes the opacity of the overlay shown when the Calibration Mode is active; the overlay can be toggled with "Edit > Marker > Show Calibration Overlay"
check box 3D Point	toggles the visibility of the yellow 3D point of a marker
check box Show ID	toggles the visibility of the identification number above a yellow 3D point of a marker
check box 2D Point	toggles the visibility of the blue 2D point of a marker

9.11.2.3.9 Live Warp



The "Live Warp" tab is hidden in the basic mode. To activate the advanced mode, open the "Edit" menu and choose "Advanced Mode". This is only available in the Warper 3D edition.

The tab enables you to setup a connection to another device in order to input and / or output data so that the Warper is controlled by another device and / or controls another device itself.

A guide that explains the Live Warping feature in more detail can be found here [2174].

Art-Net Input

The settings on the left side of the tab allow to activate the Art-Net input as well as to set the patch, i.e the Subnet, Universe and the starting number also known as the channel.

The general purpose of this feature is to be able to control a mesh from another device, may it be a lighting desk or Pandoras Box (outputting Art-Net). In detail, an FFD point can be moved along the X, Y and Z axis. Each of these 3 parameter has a resolution of 16bit. Hence, one FFD point listens to 6 Art-Net channels. In this manner, up to 9 FFD points can be remote controlled. An according device is available in Pandoras Box (tab "Device Types" > "DMX Fixtures" > "COOLUX" > "PB Warp 3x3.clib").

The first channel is applied to the FFD point in the upper left corner, or more precisely to the X parameter of it. Please see the following table for more information about the internal patch.

Ch 1,2 X / Ch 3,4 Y / Ch 5,6 Z — 1st FFD point	Ch 7,8 X / Ch 9,10 Y / Ch 11,12 Z — 2nd FFD point	Ch 13,14 X / Ch 15,16 Y / Ch 17,18 Z — 3rd FFD point
Ch 19,20 X / Ch 21,22 Y / Ch 23,24 Z — 4th FFD point	Ch 25,26 X / Ch 27,28 Y / Ch 29,30 Z — 5th FFD point	Ch 31,32 X / Ch 33,34 Y / Ch 35,36 Z — 6th FFD point
Ch 37,38 X / Ch 39,40 Y / Ch 41,42 Z — 7th FFD point	Ch 43,44 X / Ch 45,46 Y / Ch 47,48 Z — 8th FFD point	Ch 49,50 X / Ch 51,52 Y / Ch 53,54 Z — 9th FFD point

Please note that due to the Art-Net protocol the IP address needs to be set to 2.x.x.x and the subnet mask to 255.0.0.0. The FFD count of the mesh must not exceed 3x3 points.

It is possible to input Art-Net at the same time as the warp output is activated.

Warp Output

The settings on the right side of the tab allow to activate the Pandoras Box Warp Output and to set the according settings, that is the target Site ID, Device ID and interval (transmission rate).

This feature enables you to control an object on a Pandoras Box layer in real time from within the Warper or through another device. Or in other words, it makes it possible to warp live. This if of particular interest when projecting on a moving, deforming screen or when the projector is moving itself. Another application is described below.
In detail, the feature works like this that single or multiple mesh points can be moved in the Warper along the X, Y and Z axis (by hand or by an input device). According to the interval time, the changes done in the Warper are transmitted directly to an object that is part of a Video Layer [647] or Output [682] in Pandoras Box. The object needs to be a so called "index object". An index object can be exported from any object or mesh from within the Warper. The command is available in the File menu [2137]. Furthermore, a Warp FX is required on the layer. The effect actually receives the data and influences the index object accordingly. For more information and a step-by-step tutorial, please read the Live Warping guide [2164].

You can remote control Pandoras Box from one or more Warper applications. The Warper may run on the same computer or on another one connected via network. In the last case, please check the network settings [2147] in the Edit menu and make sure that the used network switch is capable to transmit the data in the set time.

As said above the application for the Live Warp feature include the projection with a moving screen or projector. Another application would be in a scenario where an object, a 2D or 3D object, is already applied to a Video Layer and thus deforms the according media file; or in a scenario where a camera is moved to another position. Under

certain circumstances it is needed to deform this result again. For example, if the projector's lens has a strong cushion effect. In that case, generate a mesh and export it as a index file. Apply it together with the Warp FX to the output layer. Now you can address this output layer in the tab "Live Warp" and whilst you are deforming the mesh in the Warper the output's object deforms as well and influenced all underlying Video Layers.

9.11.3 Warping Guide

This chapter includes step-by-step tutorials about 2D warping. Whilst 2D warping is based on deforming the 2D content in a way that it looks correct again, 3D warping achieves this by placing (3D) objects and the camera in 3D space. For more information please read the introductory chapter [2129].

The first tutorial explains, how to warp a 2D mesh [2166]. It shows how to decide for the best FFD and mesh count and how points can be moved.

Afterwards it is explained in detail how to export the mesh and use it in Pandoras Box [2168].

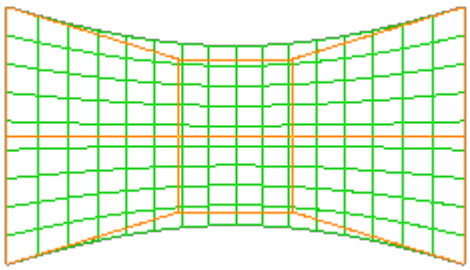
Then there is an tutorial showing how to work with associated meshes [2170].

The last one tells more about the live warping feature [2174].

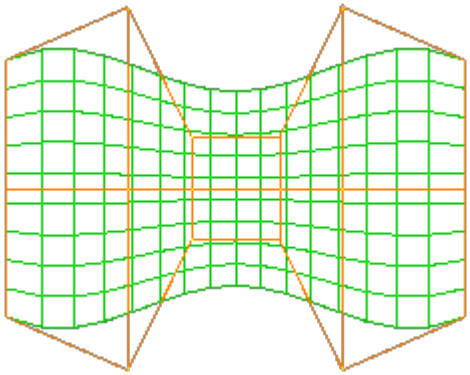
9.11.3.1 General Warping Workflow

Setting up FFD control points and mesh segments

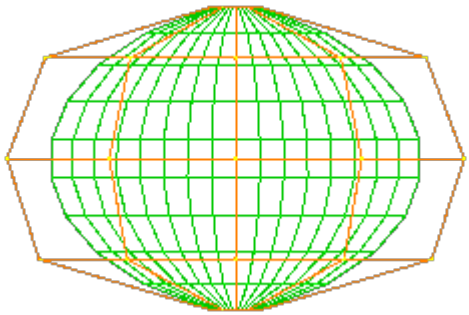
Before you start moving the control points you have to decide with how many FFD control points and mesh segments you should work. Please follow this link if you are not sure about the difference between FFD and mesh [2135]. The FFD count is set up via the File [2137] menu and the mesh count in the Mesh tab [2153]. The perfect amount depends on your screen surface and outline.
The more warping projects you have done, the faster you will be able to tell the best FFD and mesh count. If you are not sure in the beginning, simply make a guess and start warping. You will see quite fast, that you have picked too many or too less FFDs. Too many FFD points are not that bad, it might take more time during the FFD-phase but you could save time during the mesh-phase - at least if not way too many FFDs were picked. If you have picked too less FFD points you will notice that you will have to move mesh points at a very early stage of warping. This will definitely be more time-consuming than starting all over with a new mesh. In addition, it is easier to obtain a good quality mesh (as described below) with FFDs instead of offsetting single mesh points too far. If you like, you can save the current project and have a quick look whether a new mesh with more FFDs does give you better results and eases your work. This can be found out in a few seconds but save minutes or even hours.



Example 1, a 4x3 FFD
A simply bend screen will go well with only 3 vertical FFD control points. Horizontally there needs to be done more warping, thus 4 FFDs work better. The more smooth the outline has to be, the more horizontal mesh segments you should take. In this case there are 20 mesh segments.



Example 2, a 6x3 FFD
Curved screens that are more complex will be easier to handle if you increase the amount of FFD control points. In this case there are 6 horizontal control points and still 3 vertical ones.
The curved outline of the mesh is the result by only moving the FFD control points.



Example 3, a 5x5 FFD
Spherical screens are bend in all directions. They require to increase the amount of vertical FFD points as well. In the depicted example a mesh with an 5x5 FFD is shown. If the projector looks straight on the equator, the mesh deformation will be quite homogenous and the FFD should look similar to the example.

The amount of mesh segments can be altered as long as working only with the FFD control points. As soon as a mesh point is moved, the amount should not be changed any more.
In general, the amount of mesh segments depends on how exact the warp needs to be. Firstly this is a question how complex or detailed the screen is, including the outline, as shown in the above example with the simply bend screen. If the screen is quite flat itself but has a very detailed outline, it could be a faster solution to create a mask [2156] instead of increasing the mesh count.

Secondly, it is important whether you are projecting with single projectors only, or if several projectors overlap each other. Within the softedge area the pixels from both projectors must overlay each other perfectly. This requires a higher mesh count. As a rule of thumb, at least 7 mesh segments should lie within the overlap area.

By the way, it could be helpful to work with mesh segments that have the same height as width. If your projector has an aspect ratio of 16:9, you could set up a mesh count of 16 by 9 during the FFD-phase and increase it to 32 by 18 or even 48 by 27 before starting the mesh-phase.

Deforming the FFD and mesh

A good quality mesh refers to a mesh where the mesh lines are uniformly distributed on the screen. For example, if the screen is 2m wide, and there are 20 horizontal mesh segments, each segment should be 10cm wide. If this is not achieved sufficiently, and you project text that moves across the screen, it would scroll unevenly. Wherever there are smaller distances between mesh lines, the text would be smaller too. Wherever there are larger meshes, the text is enlarged.
If your content does not contain critical movements or visible geometrical forms, you can warp a little more rough. So before you start warping, or before you spend too much time within the last phase, check the content and decide how perfect the result really needs to be.

For some people it is quite hard to perceive equal distances. To fasten and ease the warping workflow, try to mark certain points on the screen. If you mark for example every 40cm with tape, it will be much easier to arrange the mesh equally. If you cannot tape on the screen, a rotatory laser can be helpful as well.
For the same reason it can be worth the time to create special test patterns. This is definitely recommended when projecting on complex geometries and the later on used content refers to the geometry.

Keeping this mind we can now start warping. The golden rule is always to warp as much as possible with the FFD, but not more than necessary. Or in other words: the **FFD is for the coarse adjustment and the mesh for fine-tuning**.

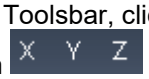
Whilst moving the FFD points, match the mesh outline (= content outline) as good as possible with your screen outline. At the same time keep an eye on the distance between the horizontal lines and between the vertical lines. As soon as you recognize that moving an FFD point helps within a small mesh area but "destroys" an higher number of other mesh areas it is better to finish with the FFD-phase. Decide for a final mesh count and move on to moving meshes. Here you will see that the better the FFD was adjusted, the less time needs to be spend for finishing.

When warping with overlapping meshes, you are done with warping as soon as all mesh lines overlay each other. This can be seen easily when both meshes have a different colors as the resulting color will be the sum. The closer the audience sees the projection, the more perfect the overlay needs to be.

In general it is possible to do a rough warp and start programming with it in Pandoras Box. Later on, when there is enough time or when it is sure the projector or screen will not move, you can load the warp project again and finalize it.

Different ways how to move FFD or mesh points

Depending on your preference you can work either using the mouse or the keyboard, or both. For more detailed information, please see the Tools bar chapter [2149].

	Mouse You might have to unhide the Toolbox by clicking the key [T]. [H] hides it again.	Keyboard You might have to unselect the Toolbox first by clicking into the Warper's main interface.
Switch the Edit Mode	In the Toolsbar, click on -Object  - FFD  - Mesh 	Use the shortcuts -[O] - [F] or - [E]
Select points	Select a single point by simply clicking on it. Select multiple points by holding down [Ctrl] additionally. Or, drag a selection box.	Navigate to another selected point with the [Up/Down/Left/Right arrow] key. Select multiple points by holding down [Ctrl] additionally.
Switch the Move Mode	In the Toolsbar, click on - Move  - Rotate  - Scale 	Use the shortcuts on the main keyboard - [1] - [2] - [3]
Toggle the Axis Locks (if necessary)	In the Toolsbar, click on the according axis button 	Use the shortcuts - [X] - [Y] - [Z]
Edit the selection	To move, rotate and scale, hold the right mouse button and move the mouse up/down or left/right.	To move, use [Shift + Up/Down/Left/Right arrow]

When working with keyboard shortcuts, you can adjust the precision of the arrow presses for positioning the Mesh / FFD points in the toolbox section "Keyboard / Mouse [2158]".

A third alternative to edit points is to use the dialog for relative transformation [2149]. Left-click on the according mode  to open the dialog. Now you can enter an exact factor for each axis and press "Apply".

The next tutorials explain how to export your mesh and use it in Pandoras Box [2168], how to work with associated meshes [2170] and how to use the live warping feature [2174].

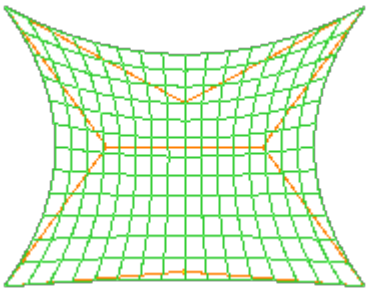
9.11.3.2 Using an Object in Pandoras Box

This chapter explains how a single mesh can be exported from the Warper and be used in Pandoras Box. In the next tutorial [2170] there is an example with a project containing associated meshes.

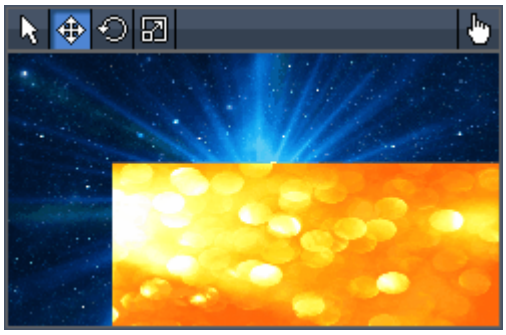
When you finished your mesh, please export your file in order to use it in Pandoras Box. Choose File menu [2137] > Export Modes > Export selected.
A small browser window opens. Browse to the directory to where you want to export your mesh as an X file.

It is highly recommended to always save the warping project too. Choose File menu [2137] > Save Settings. Whilst the exported X file contains only the mesh information, the WRP file saves all other settings and the FFD data. If the projector or a screen has moved, for whatever reason, or if you like to fine-tune / finalize your mesh, you can then simply load your warping project file (File > Load Settings) and make your changes.

Open your Pandoras Box application in Master mode. In the Assets tab [131] browse to the directory to where you exported your mesh. Drag and drop this X file into the Project tab [278]. You might need to spread the file. If you have created this file for a specific output (e.g. Output 2), assign it to the according Output layer [682]. Now all affected layers will be deformed in the same way. If you have created it for only one layer, assign it to an available Video layer [647]. Both scenarios are depicted in the example below.

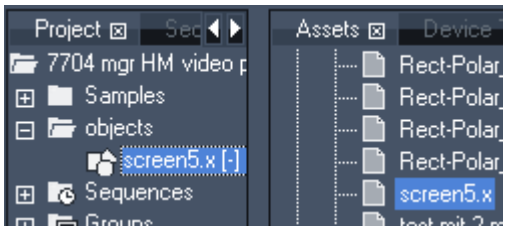


This grid is exported as screen5.x. File menu > Export Modes > Export selected

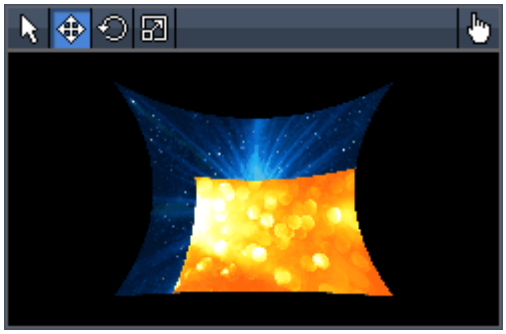


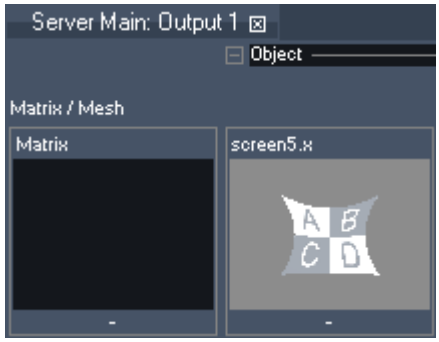
In Pandoras Box there are two layers in use - one for the blue background and one for the orange content. Currently there is no object assigned to any layer, neither to the content layers nor to the output.

The Preview tab is shown in the left image.

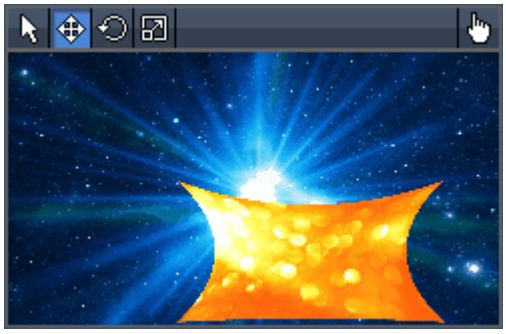


The saved object screen5.x is dragged from the Assets tab into the Project tab. Now it is part of the project.





The Preview shows the result when the object is assigned to the Output layer, everything is mapped onto the object ...

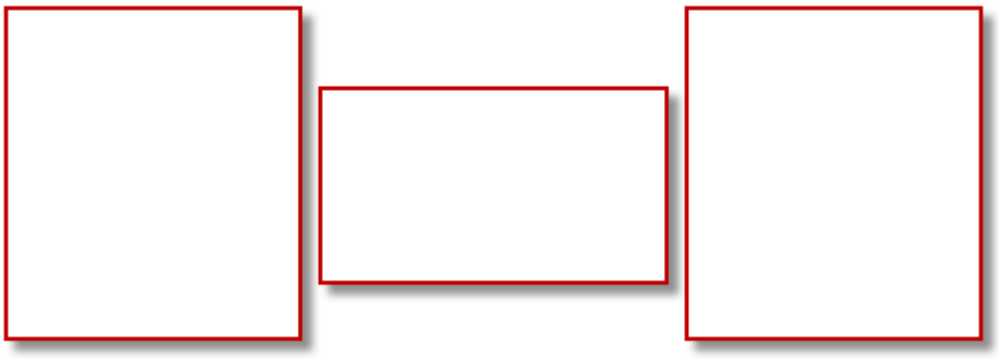


...whereas this would be the result if the object was assigned to the orange layer, and not to the output. As you can see, only the orange layer is deformed, the blue background keeps its original shape.

Please note that it is possible to combine these two ways: you might assign an object to a layer and to the output at the same time. Then, the image will be deformed twice.

9.11.3.3 Warping with Several Meshes

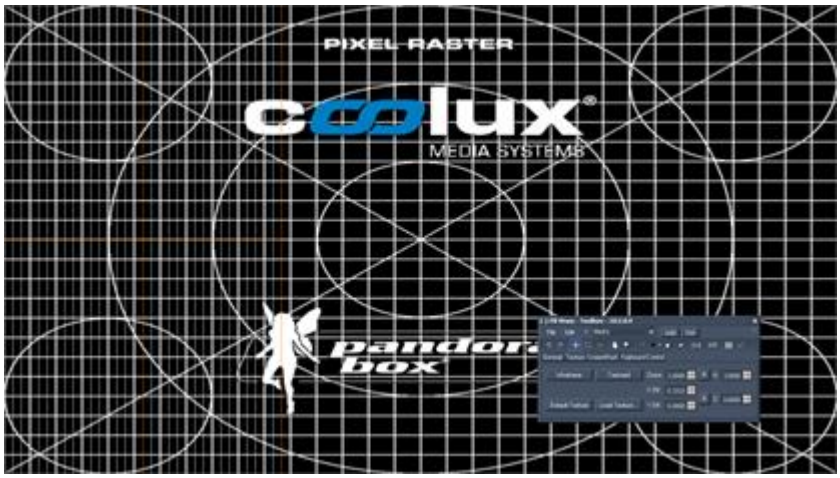
See here an example for a workflow if you have to project on a screen setup as shown in the image below.



The task is to project on this screen setup in this way:

The screen in the middle should show the whole content. Each of the two side screens should show only the a half of the content.

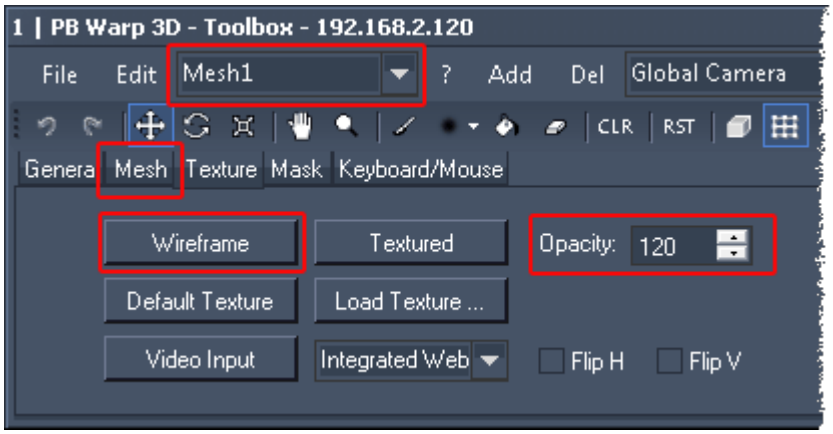
Important:
If your setup will contain overlapping meshes, please read the information about the Z-Order [2154] as well!



Start with a new warping file. As three meshes are needed, you could create additional meshes with the "Add" button in the mesh selection from the Menu bar [2147].

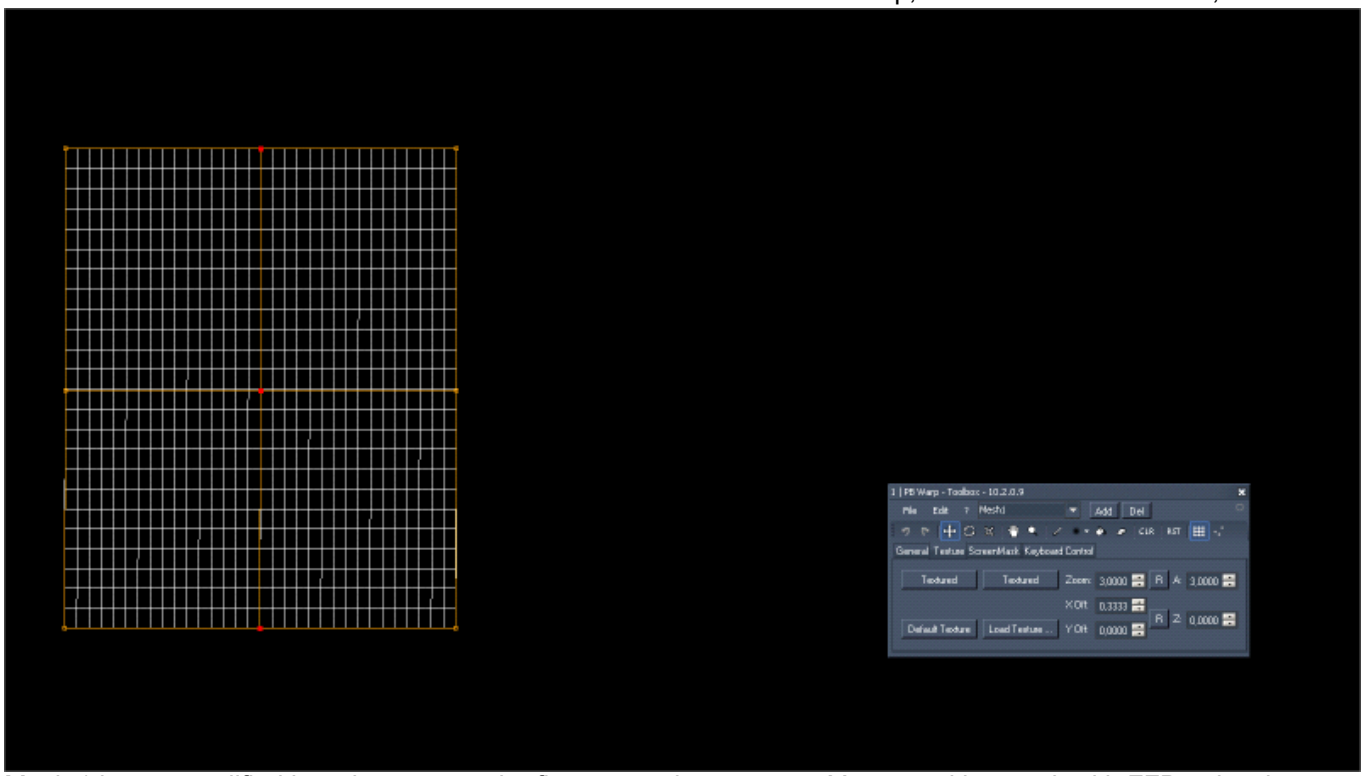
Alternatively, choose File Menu [2137] > New… > Triple.

The project contains three side by side meshes as seen in the left image. One texture is applied to all three meshes at the same time.

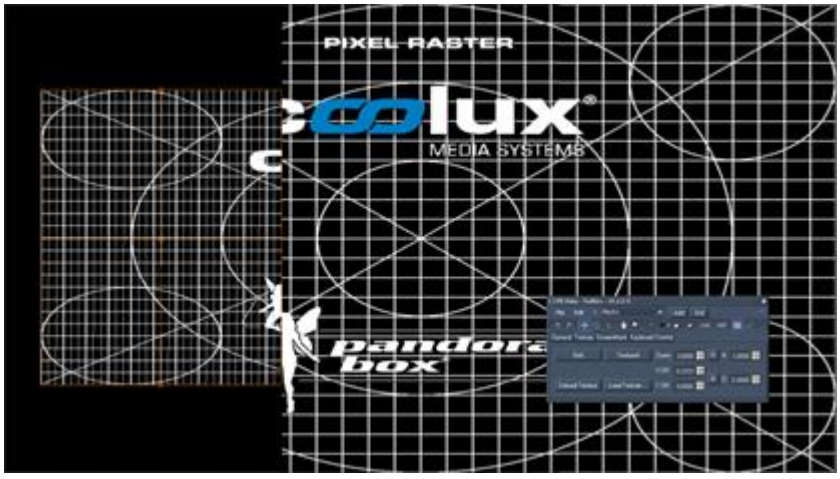


For an easier recognition of an individual mesh, you may alter the texture mode. Open the Texture tab [2155] and click "Wireframe". Or, lower the opacity to a value of 120. If you like to use another texture, click the button "Load Texture", but in this example we will keep the Pandoras Box test pattern.

Now, the FFD and mesh lines of the selected mesh are better visible. To change the selection, use the according drop-down menu in the Menu bar (the so called mesh selection [2147]). For the next step, select the leftmost mesh, Mesh1.



Mesh 1 is now modified in order to cover the first screen in our setup. You can either work with FFD points by moving or scaling a multi-selection of them, or you can scale the entire mesh whilst being in the object mode. The previous tutorial shows different ways how to do this.
First, adjust the outer FFD points so that the borders of your projected mesh exactly fit to the borders of the left screen. The mesh segments should all have the same size as well. This can be adjusted by moving the inner FFD points.



The size of the mesh is Ok, but as seen when switching back to the Textured View, it depicts only third of the content. The task was to show the left half of the content. This is done by influencing the so called texture map [2133]. Please have a look at the current settings in the Texture tab [2155].

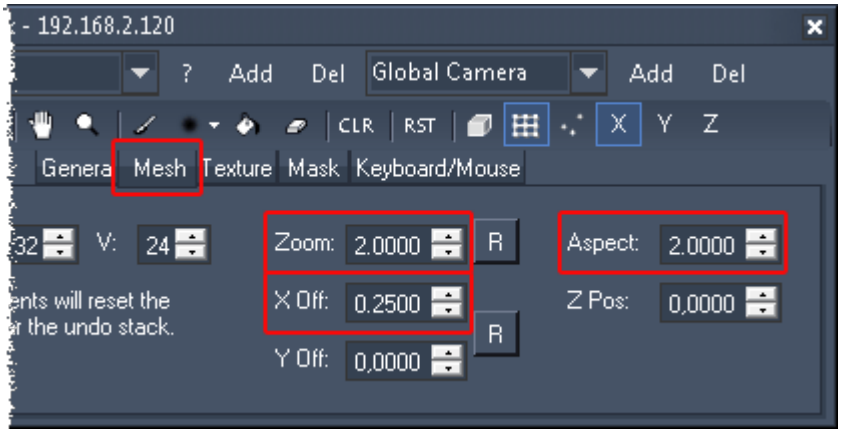
Up until now, the mesh's width is covered with a third of the texture. That means, the entire texture is zoomed with a factor of 3.

At the same time the height is not covered with only a third of the texture,

but the entire texture height. Setting the aspect ratio to 3 as well, zooms the height back in.

The last setting is the X Offset. For now, it is set to 0.3333 in order to shift the image 1/3 (of the image width) to the right side.

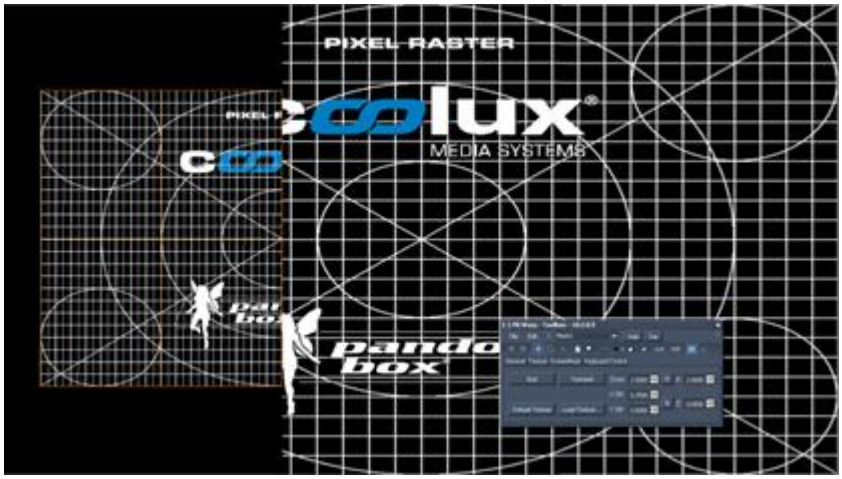
If you like you can reset the settings and try to find the solution for the task by yourself...



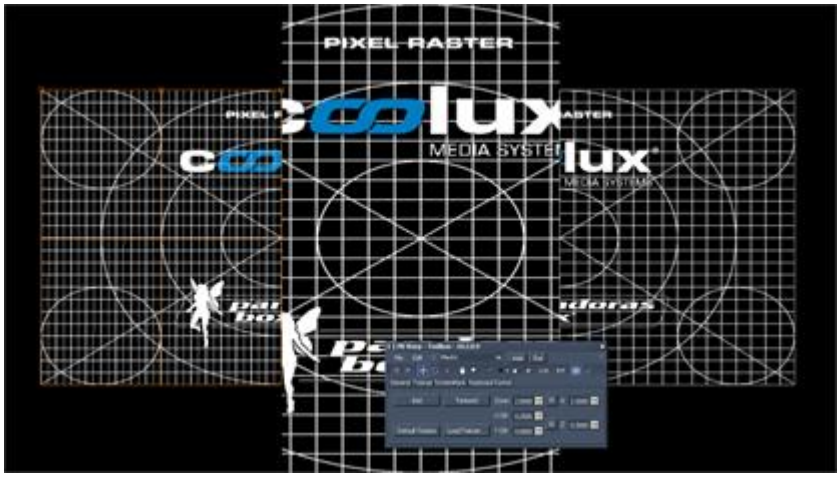
All three settings need to be adjusted to accomplish our task. We would like to:

- see half of the content's width => set the Zoom to 2
- see the whole height => set the aspect ratio to the same zoom factor of 2
- see the leftmost pixel row at the left object edge => set the X offset to 0.25

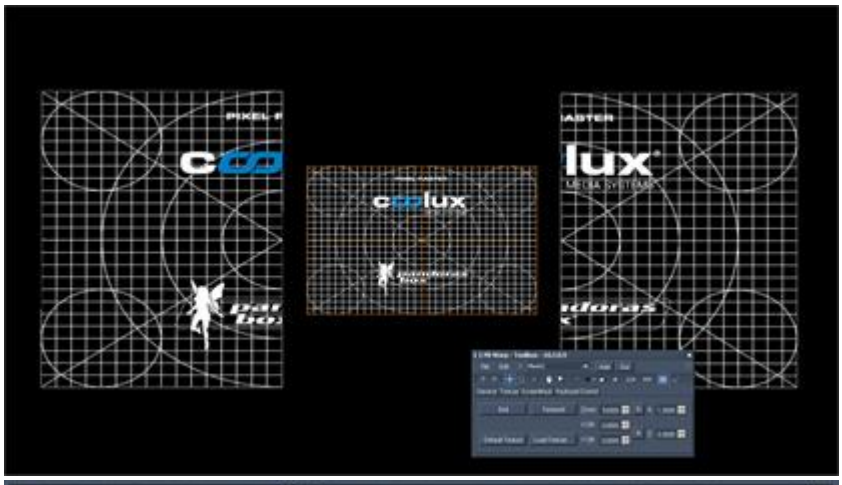
Why 0.25? We see half of the content on the mesh, in other words, one quarter is "behind" the left edge and the other quarter behind the right edge. So, to see the leftmost pixel we must shift the texture by 1/4 of the texture to the right side. One quarter is 0.25 and "to the right" requires a positive value.



Now the texture sits perfectly on the first mesh.

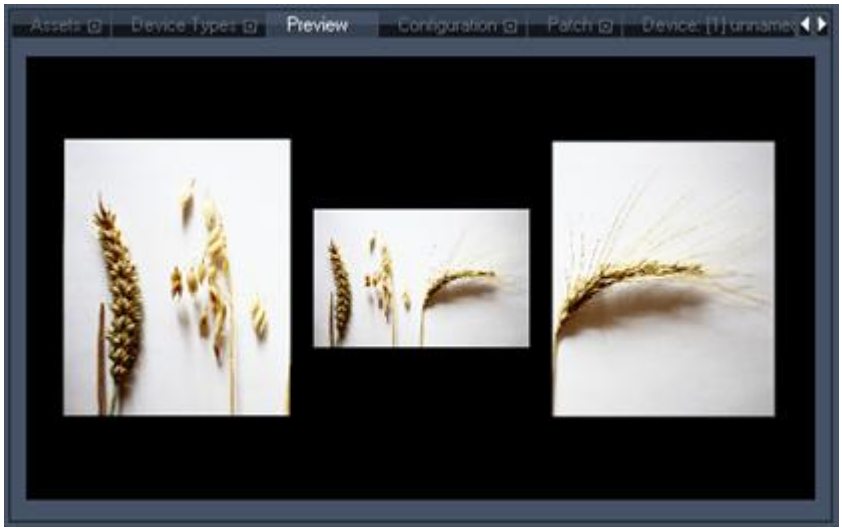


Repeat these steps for the right mesh covering the ride side screen. The result is seen left. If your result is different, check the texture mapping. As the texture needs to be shifted to the left, the X Offset should be negative: -0.25



Again, repeat the steps for the center mesh. First, transform the mesh according to the screen it is projected on. Then, adjust the texture mapping for the correct content. This time we would like to:

- see all of the content's width => set the Zoom to 1
- see the whole height => set the aspect ratio to the same zoom factor of 1
- see the leftmost pixel row at the left object edge => leave the X offset to 0

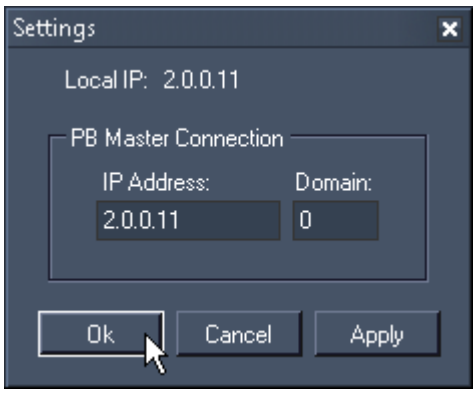


The last thing to do now, is to export the whole scene.
Go to File > Export All and choose the file location. All three meshes will be combined in the newly created X file. Use this object on the Output in Pandoras Box.

9.11.3.4 Live Warping

The live warping feature (previously known as dynamic warping) allows to have non-static objects on a layer. While projecting, the mesh can be influenced in real-time. Another feature - Morphing - allows to fade between two or three saved meshes.
Please note that those features are not available on former Pandoras Box Player Systems!

Controlling Pandoras Box via Warper

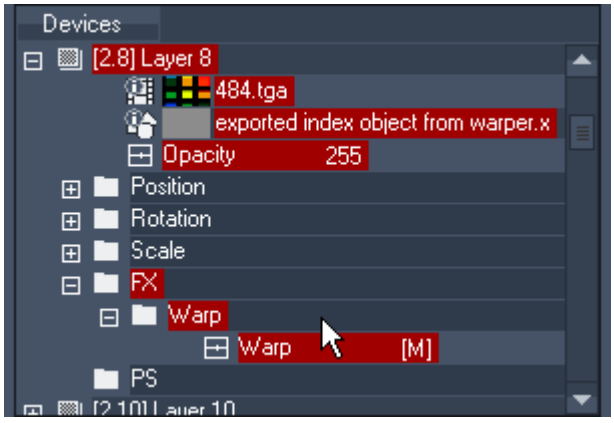


- In the Warper, set up the connection to the Pandoras Box (PB) Master system. To do this go to "Edit" > " Network Settings [2147]".

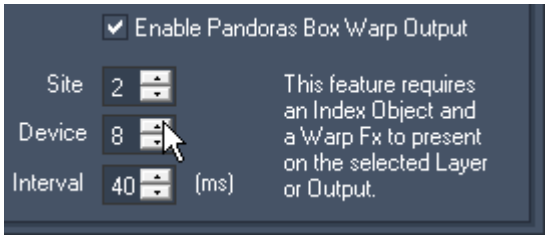
- For the next steps advanced features will be used, please go to "Edit" > "Advanced Mode" to toggle between the basic and advanced user interface where all features are visible in the menus and tabs.

- Set up the FFD and mesh count [2166] from your mesh you would like to use, or import an object.

- Export the selected mesh as an index object (as the previous mentioned step influences the object, the export has to be done each time you change the FFD and mesh count). To do this go to "File [2137]" > "Export Modes" > "Export Selected as Index Object". Save the X file. If the PB Master is connected and a project is opened, all exported objects will be automatically loaded into this project. If you start the project later drag and drop your saved object from the Asset tab [131] to the Project tab [278].



- In PB, choose which layer (or output) you want to warp live. First of all assign a media file to it and make sure the opacity is on.
Then go to the Project tab and assign the exported object as a mesh to the layer. The layer will now be stretched as if it was one line.
Go to the "Aeon FX [129]" tab > Folder "Warp" > drag the effect "Warp" onto the layer. Note that this effect has no parameters, you will not see it in the Device Control itself. Only when opening the FX parameter in the Device Tree [173] you see the effect. The effect cannot be turned off like other effects, the only way to deactivate it, is to delete it entirely.

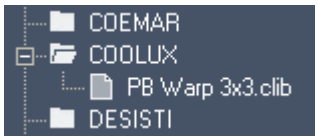


- In the Warper, go to the tab "Live Warp [2164]", tick the check box "Enable Pandoras Box Warp Output" and set up the site and device whereto you assigned the object and the effect in PB.

- Now, as soon as you edit the FFD or mesh in the Warper the layer in PB will not look like one line anymore. From now on it is influenced in real-time according to the mesh in the Warper.

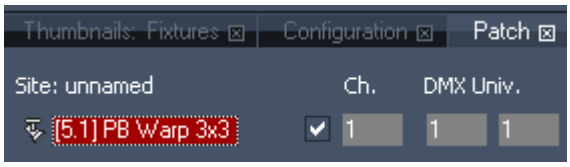
Controlling Warper via Pandoras Box / another Art-Net Device

- As this feature is based on an Art-Net communication make sure that Pandoras Box and the Warper are set to an IP address 2.x.x.x and Subnet Mask 255.0.0.0.



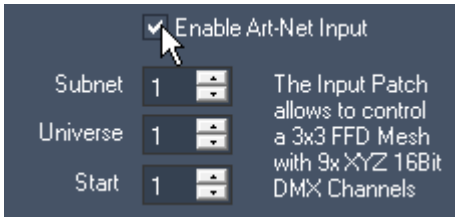
- In Pandoras Box go to the "Device Types" tab [183] and open "DMX Fixtures" > "COOLUX"

- and drag the "PB Warp 3x3.clib" into the Device Tree.



- Now select the warp device and go to the "Patch" tab [228] to patch it, for example to Channel 1 on Subnet 1 and Universe 1. You can enter the Subnet/Universe separately after deactivating the option "Universe as Single Number"

- Then, go to the "Configuration" tab [139], open the section "Remote Control Protocols" and press the button Art-Net "Activate Output".



- In the Warper, set up a mesh with an FFD count of 3 x 3 and a mesh count of your choice.

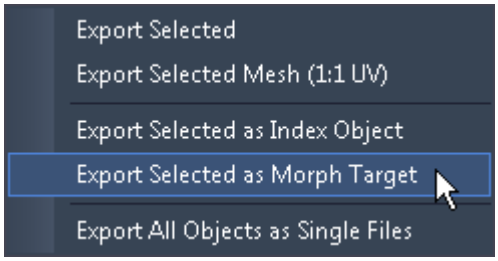
- Go to the "Live Warp [2164]" tab, tick the check box "Enable Art-Net Input" and patch the Channel as done in the previous step.

- Now, as soon as you change the parameters within the warp device in PB, the FFD points and therefore the mesh are influenced.

- Instead of the warp device in PB you can use any Art-Net sending device to control the FFD in the Warper.

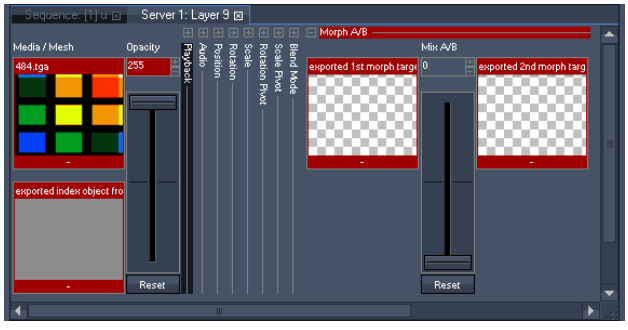
- Of course this feature can be combined with the above mentioned feature and the Warp FX. If you do this you are limited to a 3 x 3 FFD.

Morphing between several objects



- In the Warper, first of all open the "Edit [2142]" menu and activate the "Advanced Mode". Then set up the FFD and mesh count [2166]. Export the mesh as an Index Object. This has to be done only once, as long as you do not change the FFD and mesh count anymore.

- Then deform the mesh as you wish to have it when starting the morph process.



- In PB, choose which layer (or output) you want to morph between the meshes. First of all assign a media file to it and make sure the opacity is on.
Then go to the Project tab and assign the exported object as a mesh to the layer.

- Go to the "Aeon FX [129]" tab > Folder "Geometry" > drag the effect "Morph A-B" (or "Morph A-B-C)" onto the layer. Then, again, go to the Project tab and assign the exported morph targets as effect medias.

- When you have a morph sequence, import it as an image sequence [95] in Pandoras Box or convert it to an AVI video file, e.g. using the Image Converter [2069] tool or to a lossless video format from the coolux codec [102].
- Go to the "Aeon FX [129]" tab > Folder "Warp" > drag the effect "Warp Target [644]" onto the layer.
- Assign the morph image sequence or video to the effect.

9.11.4 Keyboard Shortcuts

[F1] – Help
[T] – Show Tools
[H] – Hide Tools

[S] – Select Mode
[M] – Move Mode ([1] Pos, [2] Rot, [3] Scale)
[F] – FFD Mode
[E] – Edit Points Mode

[G] – Toggle Grid / Textured View
[Space] – Play/Pause Video Texture

[D] – Toggle Face Visibility

[Esc] – Clear Selection
[Ctrl + A] – Select All
[R] – Reset Offset of Selected Points

[Up] – Selection Up
[Down] – Selection Down
[Right] – Selection Right
[Left] – Selection Left

[Shift + Up] – Move Selection Up
[Shift + Down] – Move Selection Down
[Shift + Right] – Move Selection Right
[Shift + Left] – Move Selection Left

[Ctrl + Up] – Multi-select Up
[Ctrl + Down] – Multi-select Down
[Ctrl + Right] – Multi-select Right
[Ctrl + Left] – Multi-select Left

9.12 Third Party Software

This chapter describes shortly how to use gmax, 3ds Max and Blender when working with 3D models. These third party applications allow creating 3D objects that can be used in the Warper or in Pandoras Box. If you are using a different 3D program please have a look at the end of this topic.

File formats supported by Pandoras Box and the Warper

Objects for Pandoras Box [100] or the Warper [2138] can be stored as a 3DS, FBX, OBJ or X file.

gmax, 3ds Max and Blender

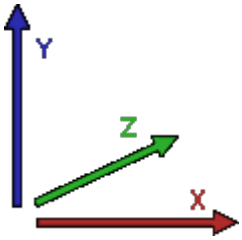
This topic focuses on three programs. It is only a brief introduction into 3D modeling, more documentation and tutorials might be found using various internet sources.
The 3D web resource page for 3D models www.turbosquid.com offers a light version of 3ds Max for personal use that is called gmax. gmax unlike 3ds Max will allow you to learn the most basic modeling and texturing concepts of 3D objects and exporting to X files with a 3rd party exporter.
The last program mentioned in this topic is the free and open-source 3D application Blender.

- Object Creation with gmax [2178]
- Exporting Objects from 3ds Max [2197]
- Object Creation with Blender [2198]

The chapter covering the Warper [2129] might be of interest as well.

Basic information you need to know when creating objects for Pandoras Box and the Warper

The chapter covering the Warper [2129] includes common 3D modeling term definitions [2131] that are quite useful. For example it includes explanations of the coordinate system used in Pandoras Box and the Warper as well as what units they are based on. UV Mapping is described too. This is a brief summary, as well of interest for those who use other programs than the ones described here:

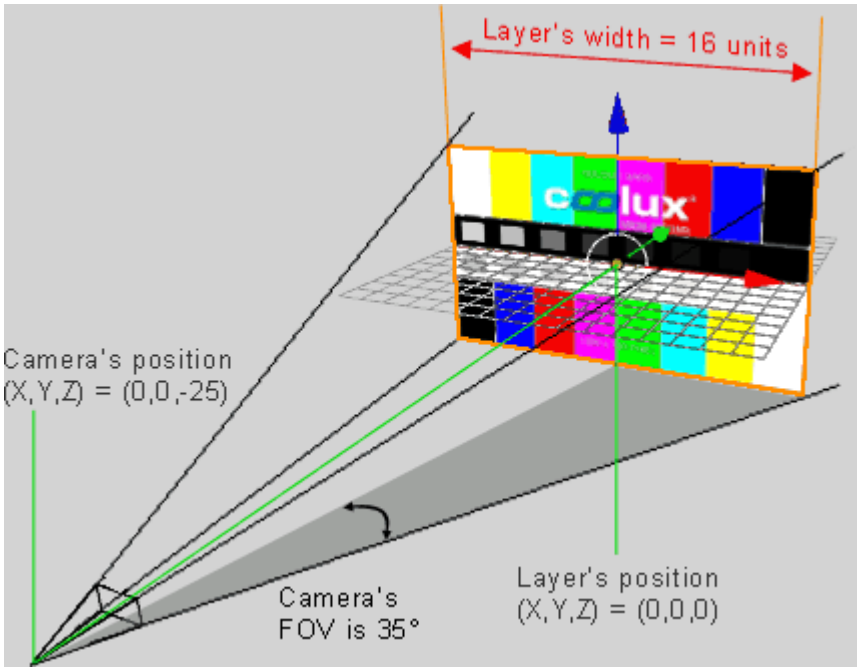


First of all, Pandoras Box' 3D space is based on a left-handed system. The units used are generic units defining that a screen's width is always 16 generic units. The camera's default position is (X,Y,Z) = (0,0,-25). The FOV is set to 35,489 degree (or 56,251 mm). If a planar object is positioned at (0,0,0) and is 16 generic units wide, it fills exactly the fullscreen size. The height is calculated by the display's aspect ratio. For example, if working with an output set to 1920x1080px, the planar object should have a size of 16x9 units, if working with an output set to 1024x768px, the planar object should have a size of 16x12 units.

When exporting objects, make sure a UVW mapping is applied.

For gmax, we have created Native Files that consist of different planes (for different aspect ratios) and a camera. Again, If you would export the planar object as it is to an X File and load it into Pandoras Box, the shape of a layer would not change, since this plane is a reference file to match the fullscreen scaling of a layer object. The camera helps us with a fixed view to see the boundaries of our output and allows us to create shapes that cross the boundaries.
When you use the files for warping, please make sure to always work within the camera view when shaping the grid. You may use the front view to work outside of the screen boundaries, but the camera view is a static reference for Pandoras Box.
If you have already started with your object in gmax and want to know how it would in Pandoras Box with the default settings applied, simply adjust the gmax camera in the above described way.

If using 3ds Max or Blender you can easily create this setup yourself.



9.12.1 gmax

As explained in the introducing chapter [2177], gmax is a freeware 3D modeling program for personal use that allows you to do create 3D objects and to export them as X files (with a 3rd party exporter) in order to use them in the Warper or in Pandoras Box.

Gmax Software Setup [2178]
Overview object creation and export with Gmax [2179]
Screen warping with Gmax [2189]
Keyboard Shortcuts [2197]

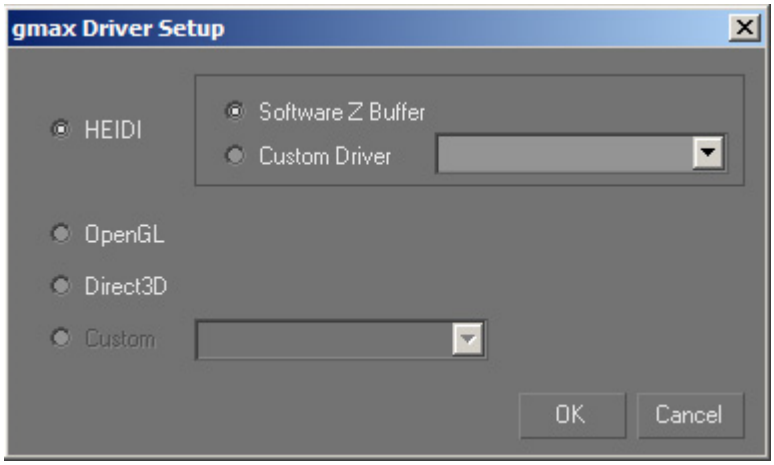
9.12.1.1 Software Setup

You may download gmax from the web resource page for 3D models www.turbosquid.com.

We have programmed some additional files that can be downloaded from the Download-Center on our website in order to improve the object creation and warping workflow. These files include:
- changes in the user interface that allow going into real fullscreen without any borders
- a default file including planes that stand for fullscreen objects with different aspect ratios as well as a camera with correct position and lens settings
- an exporter plugin to be able to save an object as an X file

After installing gmax, you may either download these files separately from the Download-Center and copy them into the according plugins,ui and scenes folder or simply run the gmax_installer that contains all needed files and copies them to the correct directory automatically. Please read the following topics for more detailed information.

Once gmax is installed and the plugin files are added to gmax, you may start the application. If you start gmax for the first time, you will be asked for your registration key that you can obtain by registering at www.turbosquid.com. Once the key is entered, the application will ask, which 3D graphics hardware acceleration to choose for gmax. Choose "HEIDI" and "Software Z Buffer".



The next chapter will give an overview about object creation and export [2179].

9.12.1.2 Object Creation and Export Using gmax

When creating a 3D object in gmax for Pandoras Box, it is helpful to use one of the PB Native Files. When exporting the Native File straight forward and importing it in Pandoras Box, it will fit exactly the fullscreen area as it is based on the coordinate system and camera settings used in Pandoras Box. Please make yourself familiar with the basics described in the introducing chapter [2177].

There are different Native Files, which one you need depends on two facts:
1) Do you create the object for Pandoras Box version 4.1 or version 4.5 and higher?
From Pandoras Box version 4.5 on the 3D perspective has been adjusted to match any output aspect ratio. For this reason 3D objects created for PB version 4.1 won´t exactly match when being used with PB version 4.5.

2) Do you want to use the (recommended) PB Exporter or the gmax Internal Aircraft Flightsim Exporter?
Generally, an exporter converts the gmax object to an X file to be imported in Pandoras Box. The exporters have different ways to convert units, thus the original file must have a different size.

The following table explains which Native File is needed and includes download links to our Download-Center.

PB Version	Exporter	Name of correct native file	Aspect Ratio
4.1	gmax International Flightsim Aircraft Exporter	PB Native 4to3 Scale	4:3
		PB Native 16to9 Scale	16:9
4.5 and higher		PB Native v4.5	4:3, 16:9, 16:10, 5:4
	PB Exporter (included in gmax_installer)	PB Native v4.5 PB Exporter	4:3, 16:9, 16:10, 5:4

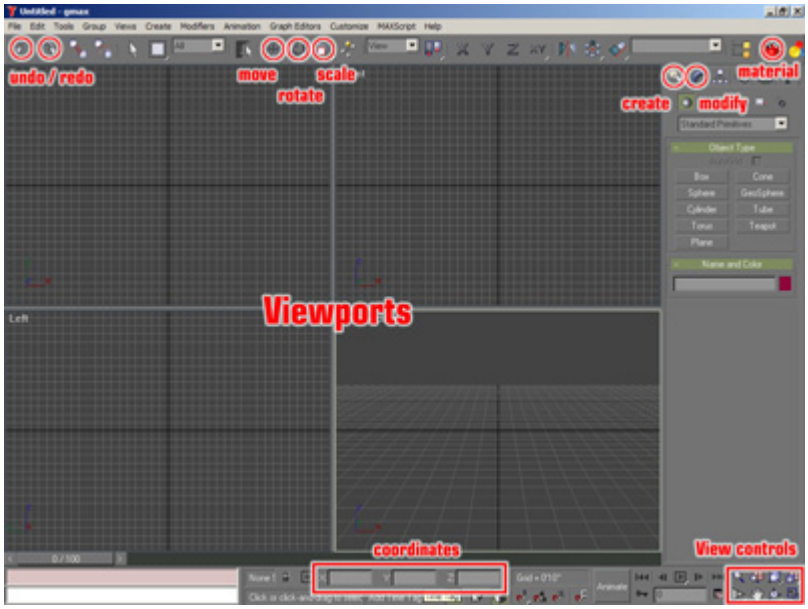
The different native files can be downloaded separately from the Download-Center to be copied into the "scenes" subfolder in the gmax installation folder. The files for the gmax International Flightsim Aircraft Exporter have to be copied to the "plugins" folder.
Alternatively, you may run the gmax_installer that contains the following files which are copied to the correct directory automatically:
- both exporters, the gmax International Flightsim Aircraft Exporter and the PB Exporter
- files that are needed to be able to go into real fullscreen
- the "PB Native v4.5 PB Exporter" file

See here further explanations for
- General object creation with Gmax [2180]: a tutorial showing the basic steps through a simple object creation and mapping process
- Object creation and export for PB Vers. 4.1 [2184]: shows how to export an object for PB version 4.1
- Object creation and export up from PB Vers. 4.5. [2185]: shows two ways how to export an object for PB version 4.5

- Warping with gmax [2189]: a tutorial showing more advanced steps in gmax and how to warp
- gmax keyboard shortcuts [2197]

9.12.1.2.1 General Object Creation in gmax

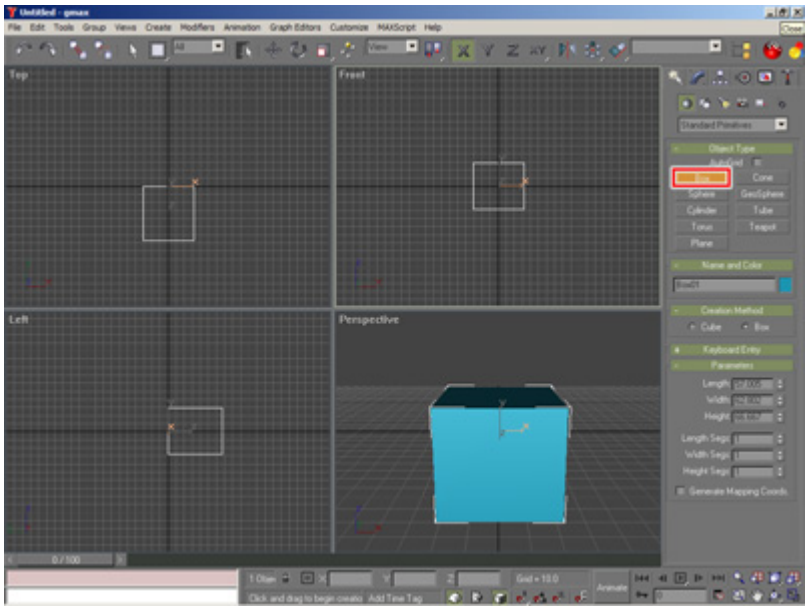
The following tutorial will show you the basic steps through a simple object creation and mapping process. Furthermore it explains the pattern to export to an x-file from gmax. This topic assumes that you are familiar with the basics described in the introducing chapter [2177].



All most important features to start with are outlined in red.

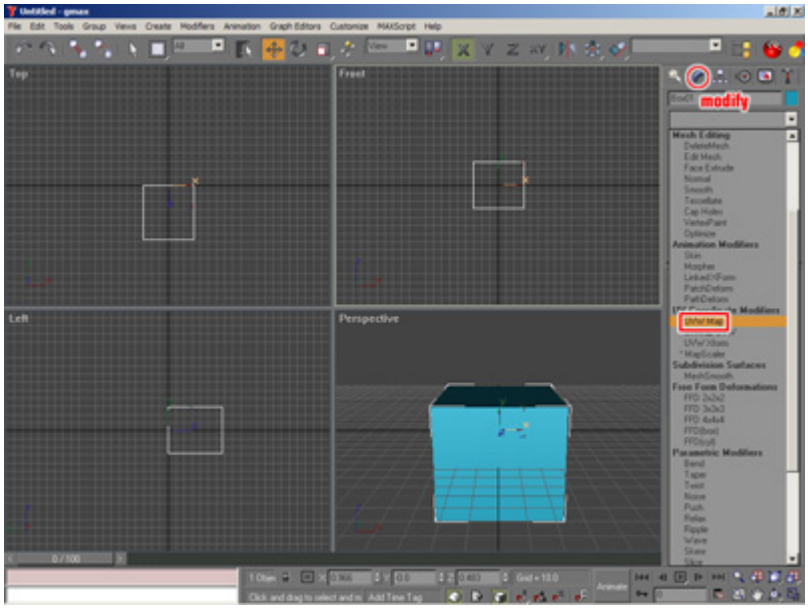
Creating a box object and applying a texture

1. Click on Box in the create tab.
2. Draw the Box with the mouse in the front view (upper right viewport), by left clicking and holding the mouse to define the front face size. Once the mouse left click is released, you can continue moving the mouse up and down to determine the depth of the box.

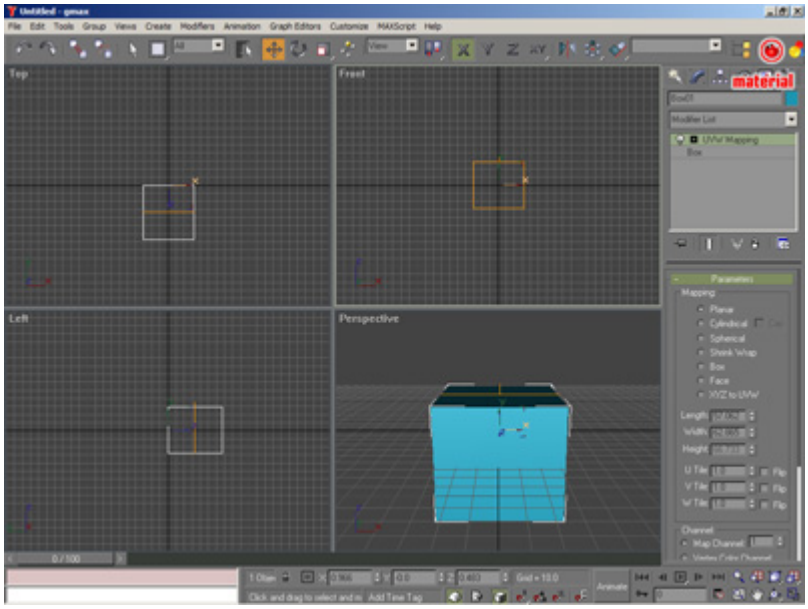


Once the box is created, you will need to assign the texture coordinates to the object in order to define how the texture should be displayed on the surfaces of the box.

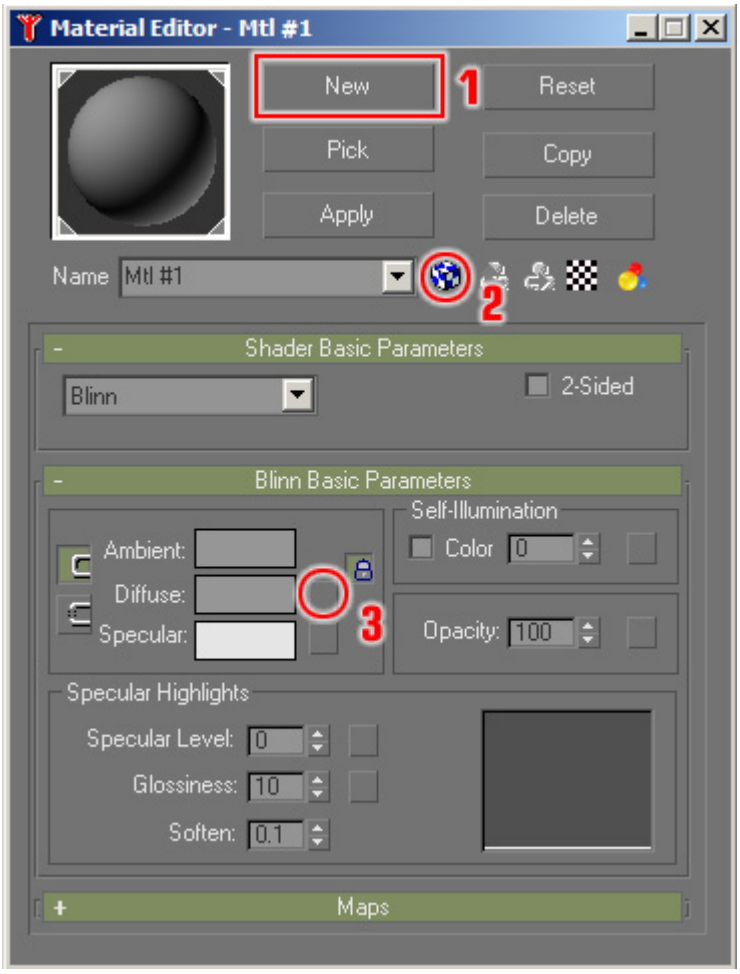
Therefore go to the Modify tab while the object is still selected (object is outlined white while selected) and apply a "UVW MAP" modifier from the pull down list to the object.



Once the "UVW MAP" modifier is applied to the object, you will see an orange outline that shows you how the texture is projected on the object.



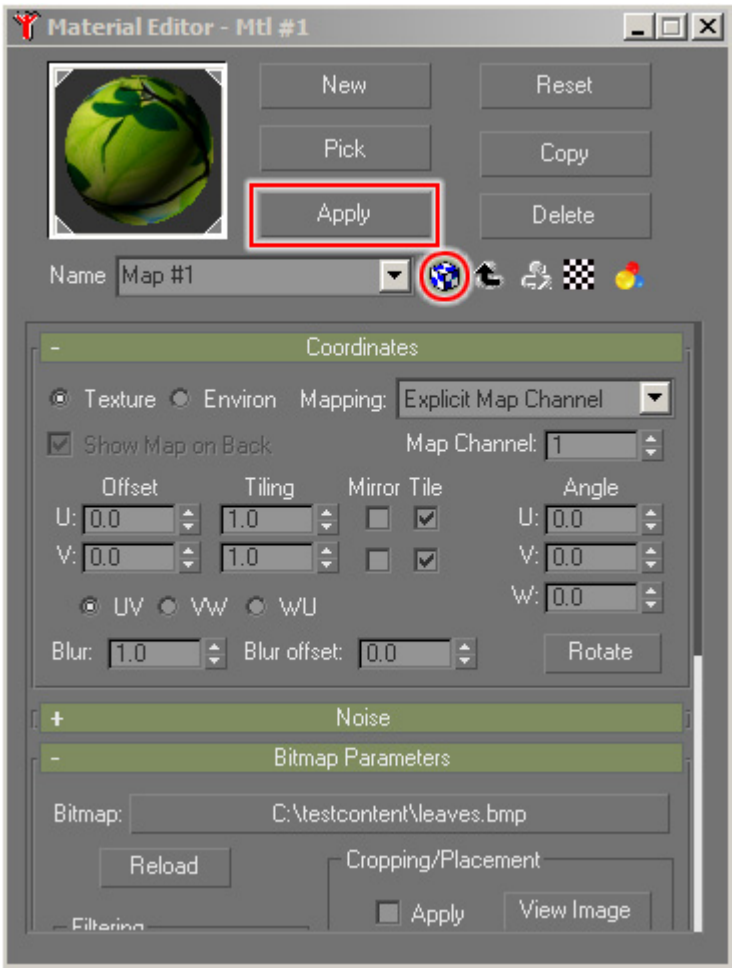
The last setup before exporting the X file you will have to apply a material with a diffuse channel to the object so that Pandoras Box can use this texture channel to map images on it. To create a new material click on the Material icon or use the keyboard shortcut "M" to open the Material Editor.



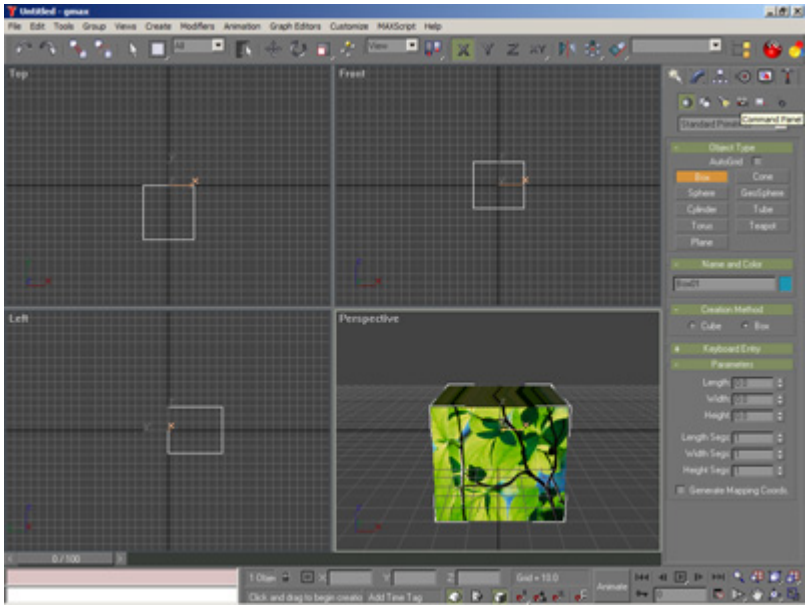
1. Create a new material
2. Toggle checked box to show texture in shaded viewport
3. Click on square box of the Diffuse channel

Choose bitmap from the dialog to assign a bitmap to the diffuse texture channel.

Next choose a bitmap from the opening file browser.



Now apply the material to the selected object by clicking "Apply" and you should be able to preview your mapping in the perspective viewport of gmax right away. If you do not see the texture in the viewport, the object was either not selected, the checked box not toggled or none of your views is in shaded preview ([F3]).



Exporting an object

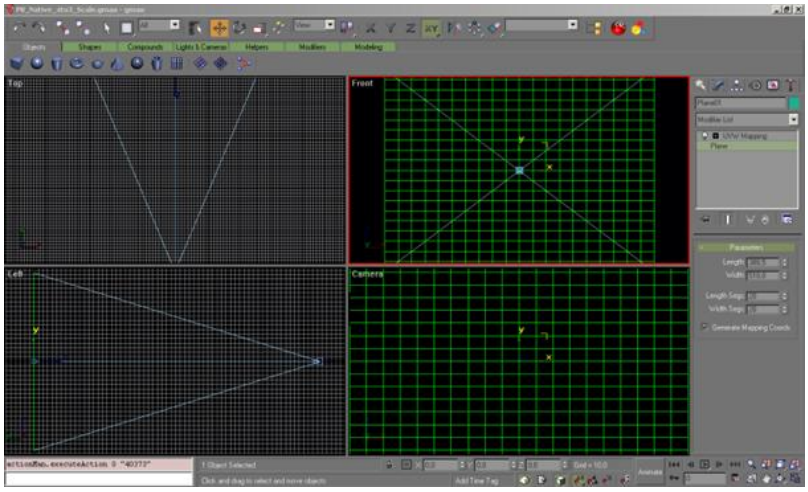
Before exporting the object for Pandoras Box you will have to make sure that it's placed correctly, since the 0,0,0 is relevant for the pivot point of an object once this is used in Pandoras Box.

To place the object right in the center, use the move tool from the main toolbar in gmax.

As last step you have to export your file as X file, for further information please see:
- Gmax Object Export for PB Vers. 4.1 [2184] or
- Gmax Object Export for PB Vers. 4.5 [2185].

9.12.1.2.2 Object Creation and Export for PB Version 4.1

As explained in the introducing chapter [2179] the Native File "PB Native 4to3 Scale" and the "gmax International Flightsim Aircraft Exporter" are needed. You may download the files separately and copy them to the "scenes" and "plugins" subfolder in the gmax installation folder. Alternatively, you may run the gmax_installer that contains all needed files and copies them to the correct directory automatically.
Please restart gmax after copying the plugin files. If running the fullscreen edition all commands are accessible within the context menu, please right-click into one of the four viewports to open it. Choose "File" > "Open" to load a file.



PB Native 4to3 Scale contains a flat planar object and a camera. If you would export the planar object as it is to an X File and load it into Pandoras Box, the shape of a layer would not change since this plane is a reference file to match the full screen scaling of a layer object.

The camera helps us with a fixed view to see the boundaries of our output and allows us to create shapes that cross the boundaries. Please make sure to always work within the camera view when shaping the grid. You may use the front view to work outside of the screen boundaries, but the camera view is a static reference for Pandoras Box.

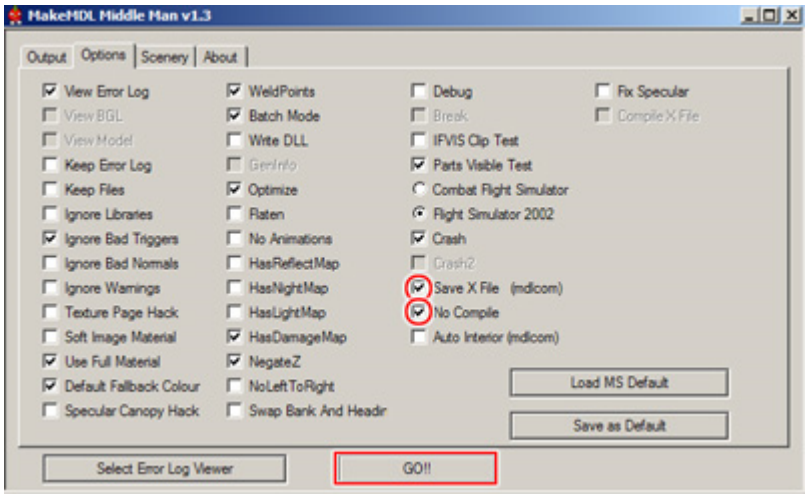
If you finished editing your object, you have to export it.

Exporting an object for PB version 4.1 using the Flightsim Aircraft Exporter

To export the object, make sure it is selected by clicking on it and then choose from the file menu "export selected". Export selected as "Flightsim Aircraft Object (*.MDL)".

Since the Flightsimulator uses different units a message box will pop up, simply click "OK" to continue.

In the next dialog that opens go to Options and check "xfile" and "no compile" and click on "GO".



You might save the options so that you don´t have to check this at every export by clicking "Save as Default".

The object is now properly exported as X file and can be loaded into Pandoras Box.

9.12.1.2.3 Object Creation and Export for PB Version 4.5

As explained in the introducing chapter [2179] you have two opportunities to export your 3D object for Pandoras Box version 4.5 (or higher). You may use the following exporters:
- the Flightsim Aircraft Exporter [2185] or
- the Pandoras Box Exporter Script [2187].

The difference between these two exporters is their handling of unit conversion. Via the flightsimulator plugin the scalings are always transformed to an inch based measuring system, the PB Exporter script transforms the units 1:1. Both exporters will give you the same result, but you have to use different PB Native Setup files, which gives you reference to the camera settings inside Pandoras Box.

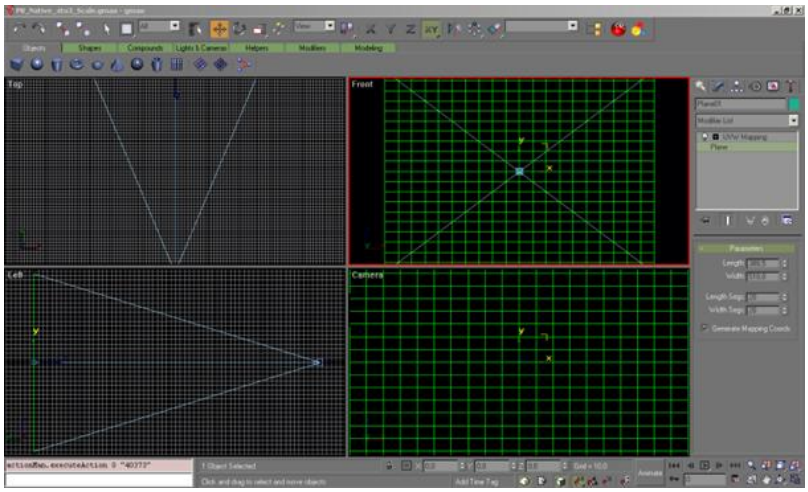
The flightsimulator plugin can be downloaded separately and be copied to the "plugins" subfolder in the gmax installation folder. Alternatively, you may run the gmax_installer that contains both exporters and other files needed and copies them to the correct directory automatically.

Please restart gmax after copying the plugin files.

9.12.1.2.3.1 Export Using Flightsim Aircraft Exporter Plugin

As explained in the introducing chapter [2179], when exporting with the "gmax International Flightsim Aircraft Exporter" for Pandoras Box version 4.5 and higher, the Native File "PB Native v4.5" is needed. Please run the gmax_installer that contains all needed files and copies them to the correct directory automatically or download the files separately before starting gmax.

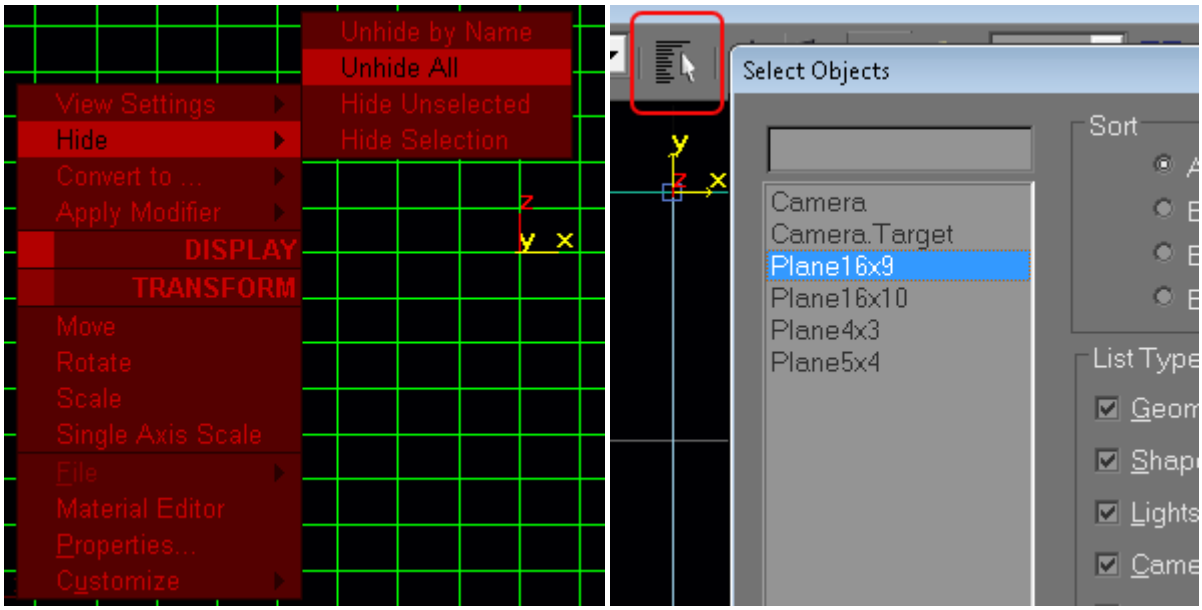
If running the fullscreen edition all commands are accessible within the context menu, please right-click into one of the four viewports to open it. Choose "File" > "Open" to load a file.



The PB Native 4.5 Setup File contains several items:
- four flat planar objects with different aspect ratios: 4:3 (visible) and 16:9, 16:10, 5:4 (hidden)
- a camera.

If you would export the 4:3 planar plane as it is to an X File and load it into Pandoras Box, the shape of a layer would not change since this plane is a reference file to match the full screen scaling of a layer object.

If you need another aspect ratio you want to fit your layer or camera to, like 16:9, 16:10 or 5:4, please do a right-click on the plane to open the context menu and choose Hide > Unhide All.
Then, via the Select by Name-Tool (found in the toolbar) you can choose another aspect ratio and confirm with clicking the "Select" Button. Open the context menu again by right-clicking and this time, choose Hide > Hide Unselected.



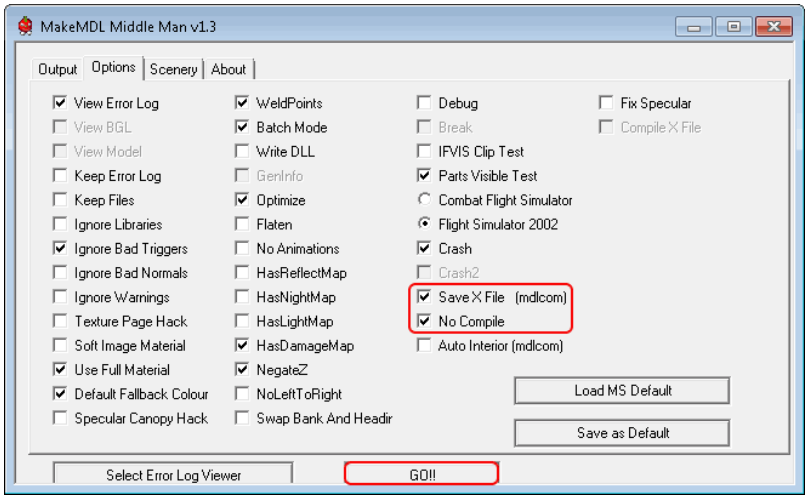
The camera helps us with a fixed view to see the boundaries of our output and allows us to create shapes that cross the boundaries. Please make sure to always work within the camera view when shaping the grid. You may use the front view to work outside of the screen boundaries, but the camera view is a static reference for Pandoras Box.

Exporting an object for PB version 4.5 using the Flightsim Aircraft Exporter

To export the object, make sure it is selected by clicking on it and then choose from the File > Export Selected from the right-click menu. Export selected as "Flightsim Aircraft Object (*.MDL)".

Since the Flightsimulator uses different units a message box will pop up, simply click "OK" to continue.

In the next dialog that opens go to Options and check "xfile" and "no compile" and click on "GO".



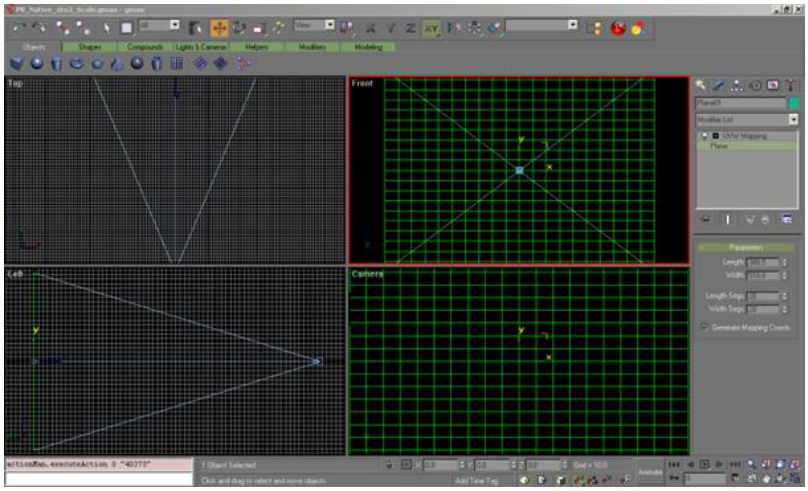
You might save the options so that you don´t have to check this at every export by clicking "Save as Default".

The object is now properly exported as X file and can be loaded into Pandoras Box.

9.12.1.2.3.2 Export Using the PB Exporter Script

As explained in the introducing chapter [2179], when exporting with the "PB Exporter" for Pandoras Box version 4.5 and higher, the Native File "PB Native v4.5 PB Exporter" is needed. Please run the gmax_installer that contains all needed files and copies them to the correct directory automatically before starting gmax.

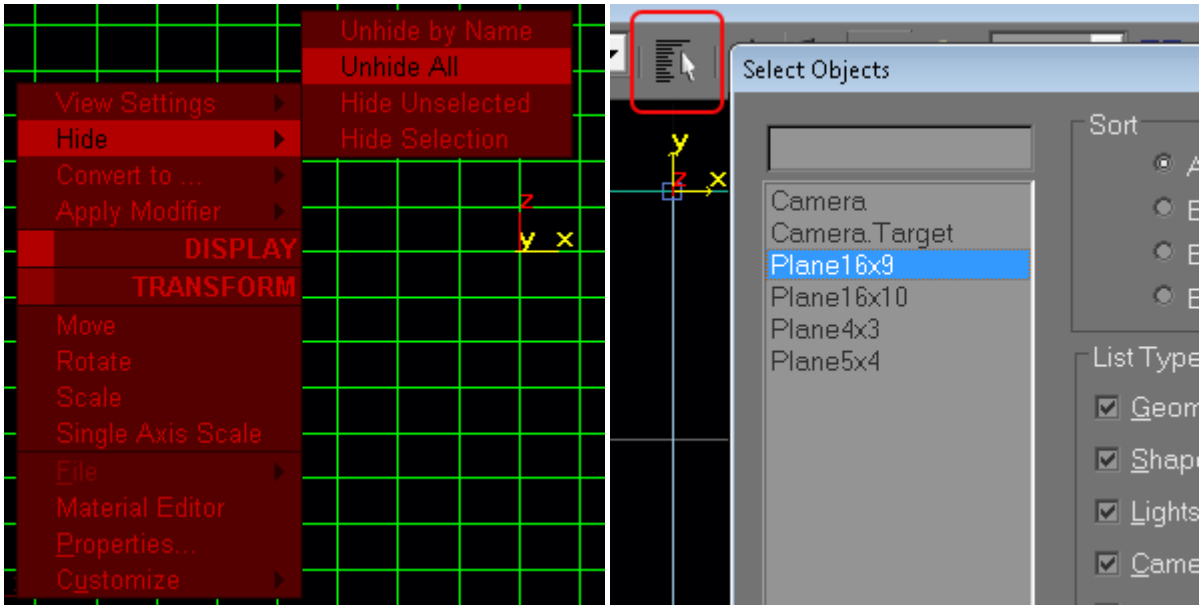
If running the fullscreen edition all commands are accessible within the context menu, please right-click into one of the four viewports to open it. Choose "File" > "Open" to load a file.



The PB Native 4.5 Setup File contains several items:
- four flat planar object, with different aspect ratios: 4:3 (visible) and 16:9, 16:10, 5:4 (hidden)
- a camera.

If you would export the 4:3 planar plane as it is to an X File and load it into Pandoras Box, the shape of a layer would not change since this plane is a reference file to match the full screen scaling of a layer object.

If you need another aspect ratio you want to fit your layer or camera to, like 16:9, 16:10 or 5:4, please do a right-click on the plane to open the context menu and choose Hide > Unhide All.
Then, via the Select by Name-Tool (found in the toolbar) you can choose another aspect ratio and confirm with clicking the "Select" Button. Open the context menu again by right-clicking and this time, choose Hide > Hide Unselected.

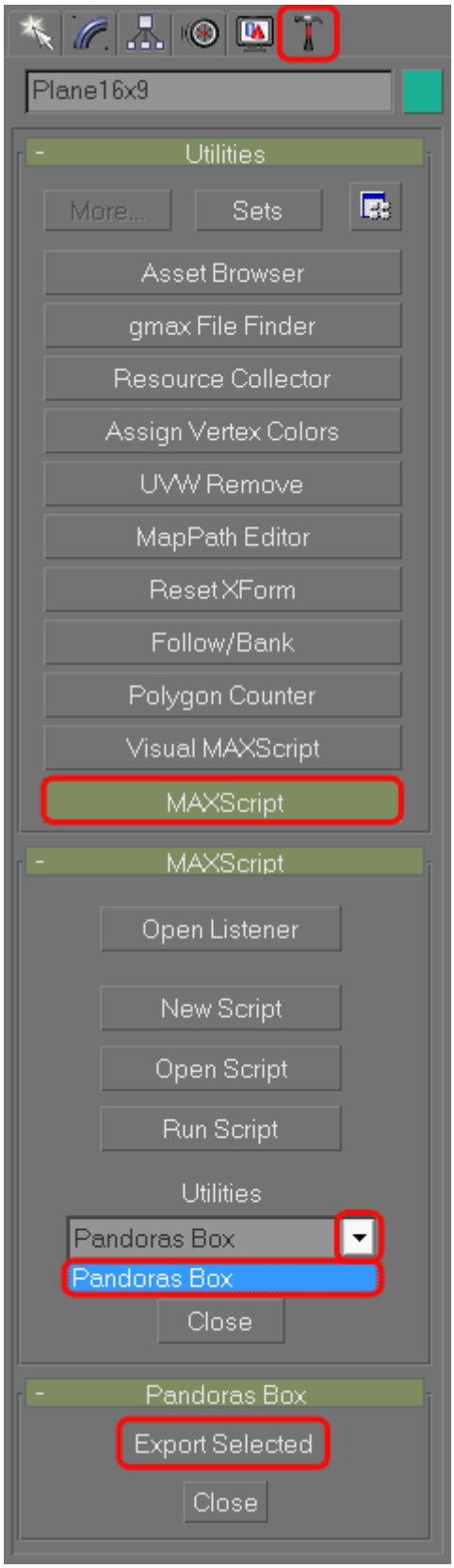


The camera helps us with a fixed view to see the boundaries of our output and allows us to create shapes that cross the boundaries. Please make sure to always work within the camera view when shaping the grid. You may use the front view to work outside of the screen boundaries, but the camera view is a static reference for Pandoras Box.

Exporting an object for PB version 4.5 using the PB Exporter

Before exporting the finished object via PB Exporter, please navigate to Hierarchy tab.
In the region "Adjust Transform" click on Reset: [Transform].

This rotates the axis inside Gmax to fit to Pandoras Box.



Please navigate to the Utilities Tab now.

Click on MAXScript, and you will get the following menu below the MAXScript button:

If you now unfold the Utilities list and choose the Pandoras Box Exporter again, below the list you will get the export option.

After you clicked [Export Selected], a Dialog opens and asks you to run the Pandoras Box Exporter. Please execute this exporter (PB GMAX X File Exporter.exe) that you downloaded.

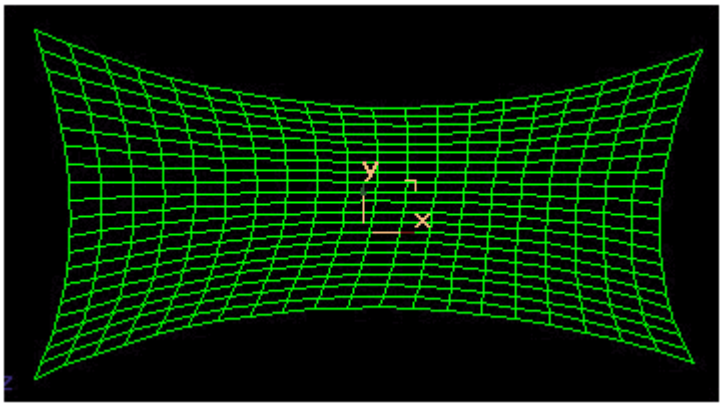
Choose a file location and your x-file will be created there.

9.12.1.3 Warping with gmax

Pandoras Box provides the ability to use shapes and objects for any display output to be rendered to. This tutorial is intended to show the basic steps for correcting curved screens. If you are interested in warping please have a look at the Warper [2129] too. It was developed by us to ease the warping process and provides specialized tools to optimize the workflow.

As gmax is a extensive and powerful 3D modeling tool we will only scratch the surface of 3D modeling in this tutorial.

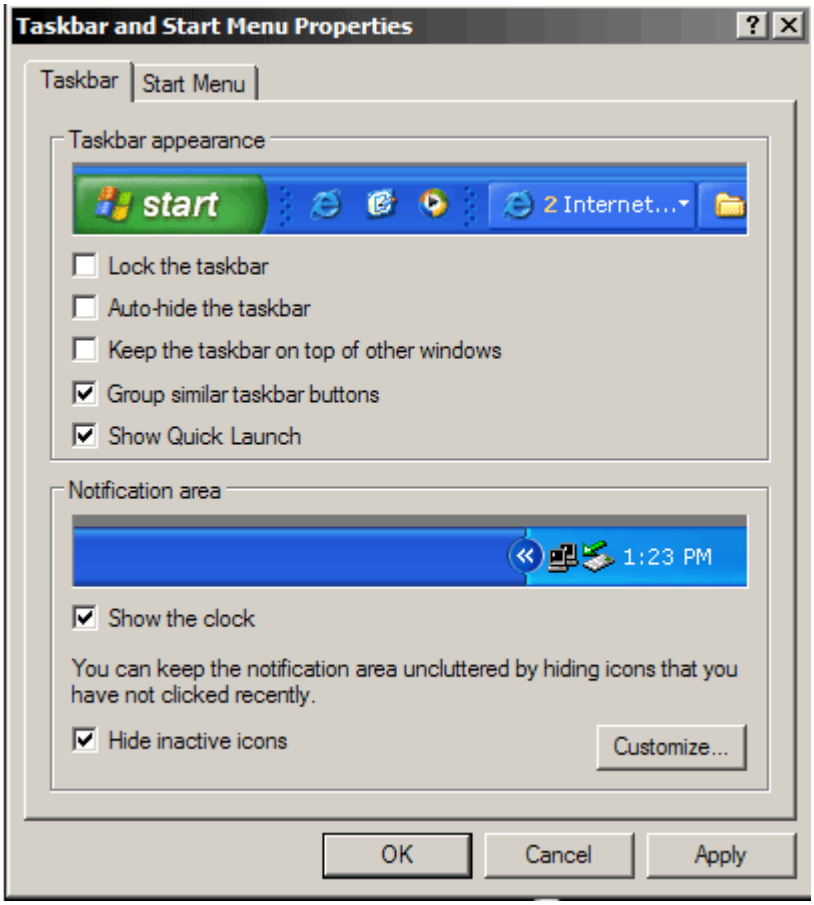
Geometric correction has to be set up on-site with the projector hooked up to your system and running gmax in full-screen.



First steps: Adjusting the view settings

In order to get started, make sure that gmax is installed properly on your system. If you need to work from a distance, use a remote desktop viewer like Real VNC to mirror the output of the projectors on your laptop or PC.

To gain a maximum warping area, make sure that the Windows taskbar is not set to be "on top of other windows" by right-clicking on the windows taskbar and choosing its properties dialog.



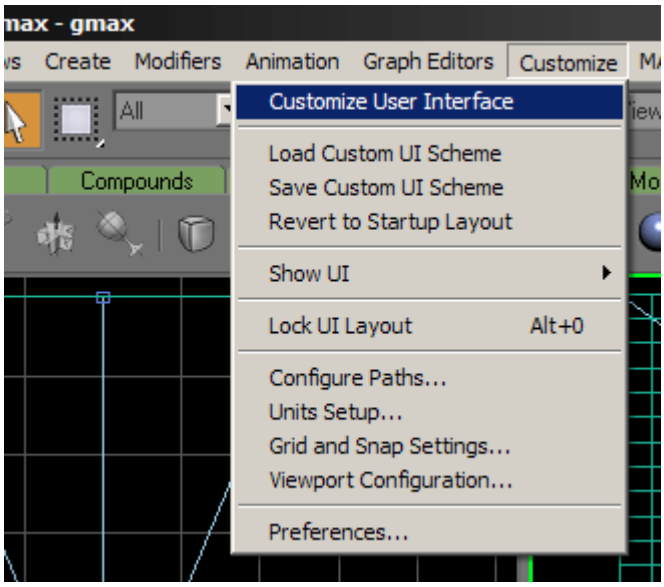
Start gmax.

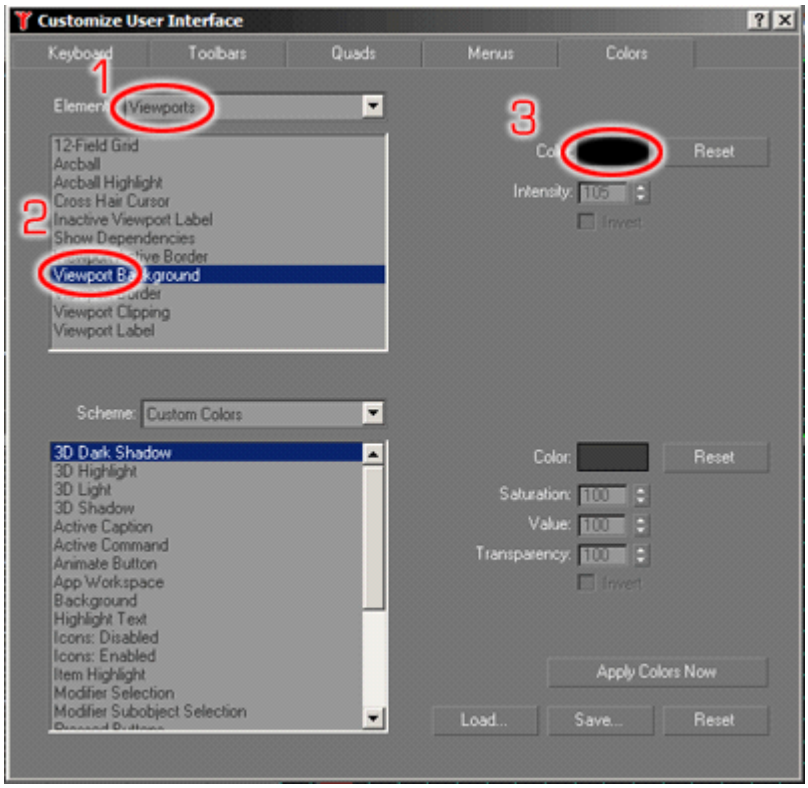
If you need to setup two outputs simultaneously, start gmax in **Desktop extended mode** once for each screen so that you can edit two warping grids at the same time.

Once gmax is started it is recommended to adjust the following settings for a better workflow during the screen shaping process:

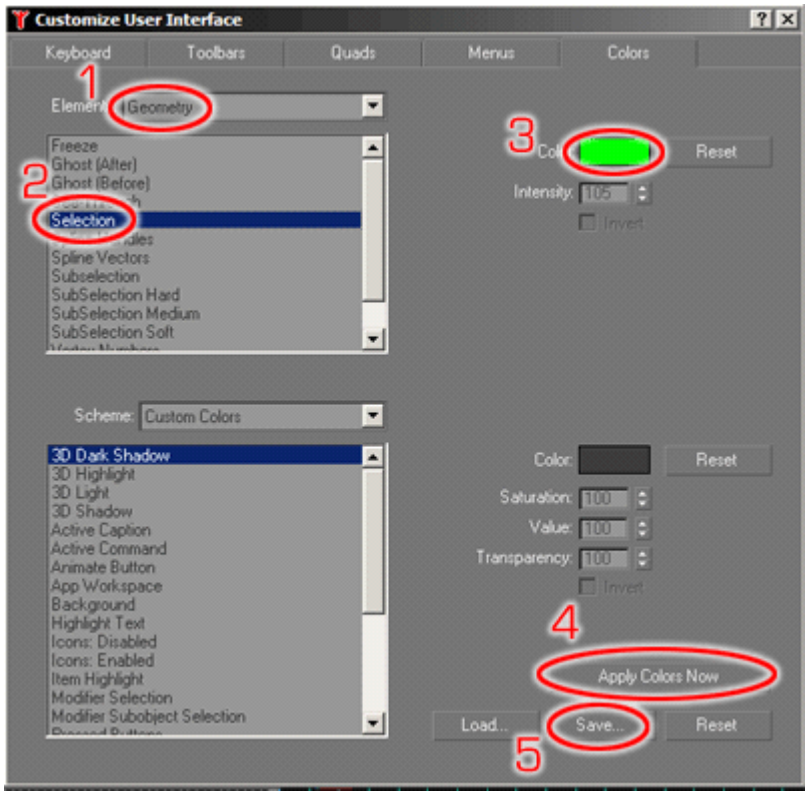
First of all we want to make sure that the background is all black and that all lines to be edited turn to primary green. This makes it much easier to see the grid in most lighting conditions.

To do this, click on Customize User Interface in the Main menu. Follow these steps:

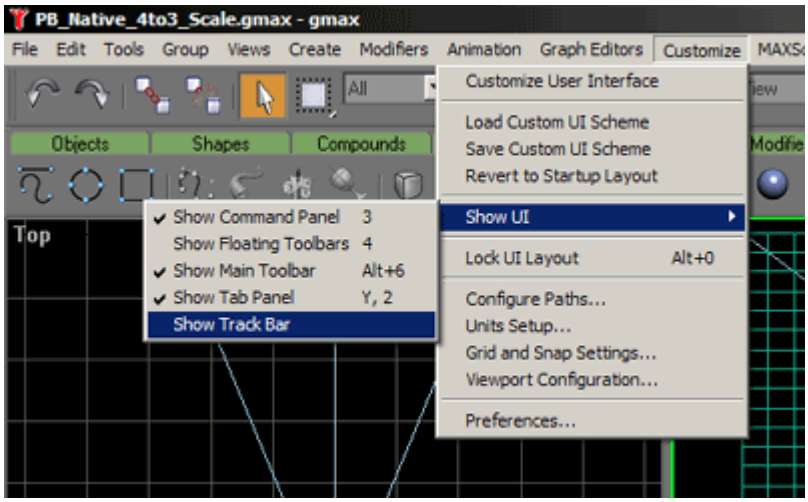




1. Select Viewports from the Elements pull-down menu
2. Choose Viewport Background
3. Change the colour in the colour dialog to black



1. Select Geometry from the Elements pull-down menu
2. Choose Selection
3. Change the colour in the colour dialog to primary green
4. Apply the new settings
5. Save these settings for the future in File dialog by overwriting the default colour file.



Last, make sure to hide the track bar from the main UI by u-checking "Show Track Bar".

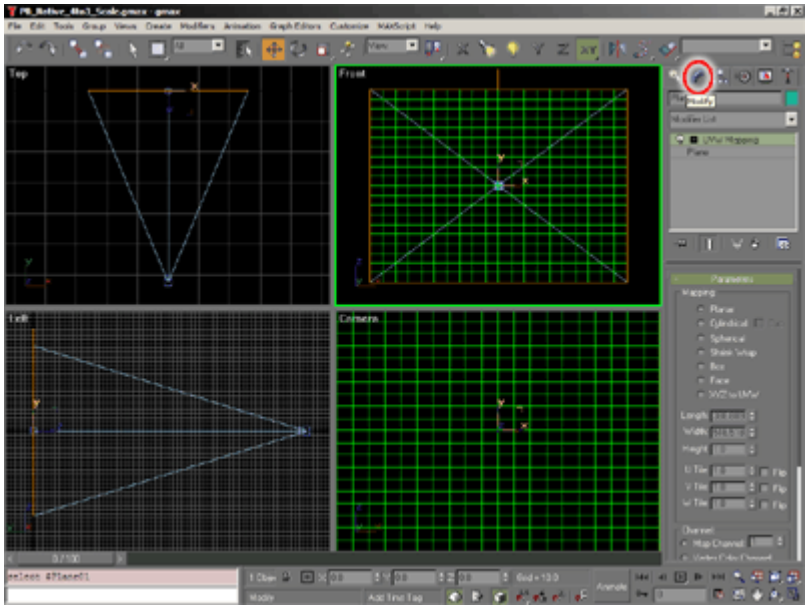
Now you are ready to start warping! Please always make sure that your gmax is set up this way, these settings are important for working in expert mode with gmax in order to use the full desktop.

How to warp using a plane

In this next section we will look at the basic deformation of a flat planar object. This tutorial is based on the steps explained in the previous chapters [2179], please be sure to have downloaded the according gmax PB Native File and Exporter.

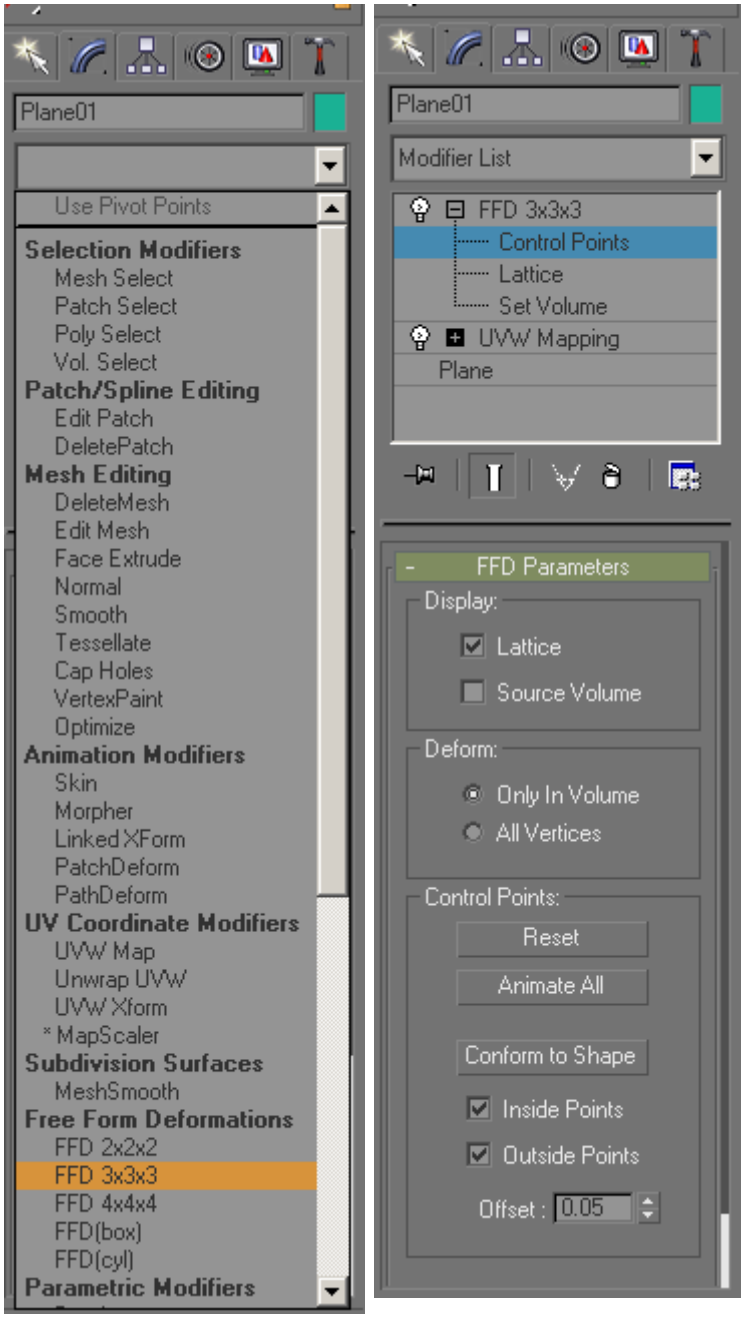
The most common task for screen shaping is a curved screen. To achieve this task, the planar object must be modified. The grid points have to be re-arranged in a way that the grid unwraps itself on the curved surface of your screen. To do this, all you need to do is to apply a tool that allows bending and shaping of planar objects.

In order to apply a new tool or a so called "modifier" in gmax, first select the planar object by clicking on it. Once it is selected it should turn primary green.

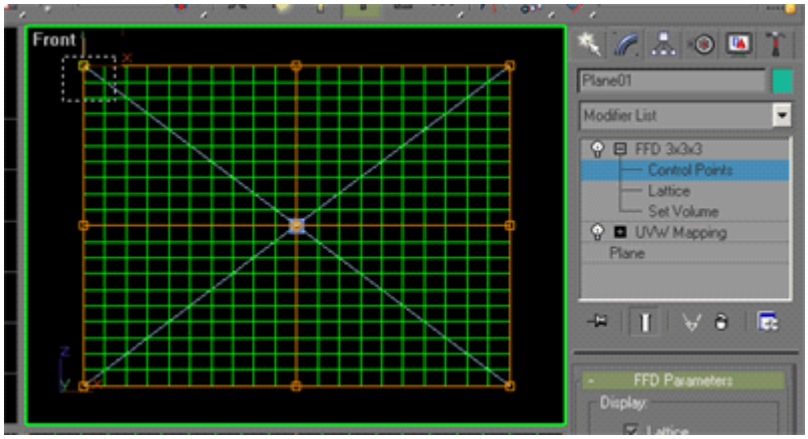


To apply a new modifier to the selected object you will need to switch to the modify mode by clicking on the "Modify" tab. Here you will see a hierarchy list of the history of the objects modifiers.

In order to apply a new one, select FFD 3x3x3 from the pull down menu.

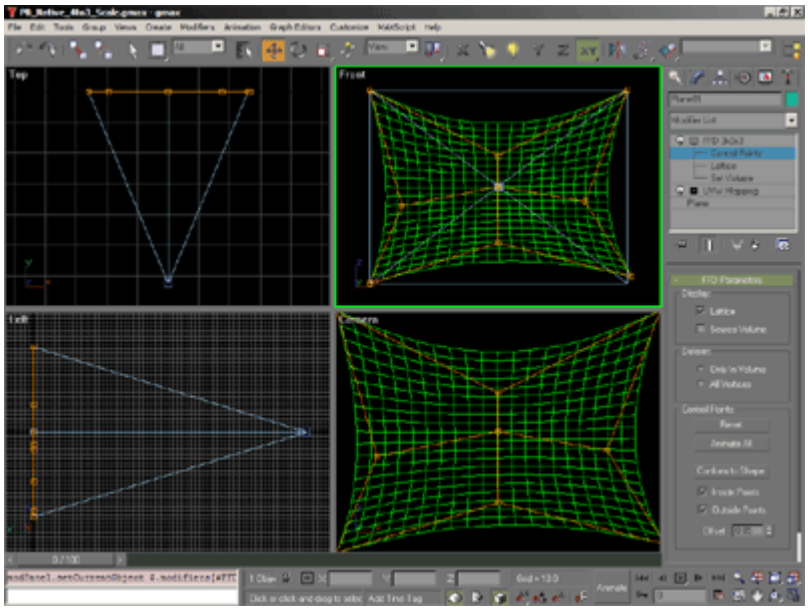


Then open the FFD 3x3x3 modifier in the stack and click on the Control Point level. Now you should see a 9 point orange control point grid surrounding the planar object.



Since gmax and all objects are always treated as 3D objects, the modifier works also in the 3rd dimension. Therefore there are 2 more control points underneath each of the 9 points . Since the object that we want to modify is totally flat we need to make sure to always catch all three control points by dragging a selection window over the control points.

Now feel free to select and move the control points around to get a feeling of the dynamic behavior of the underlying planar grid according to the control points positions.



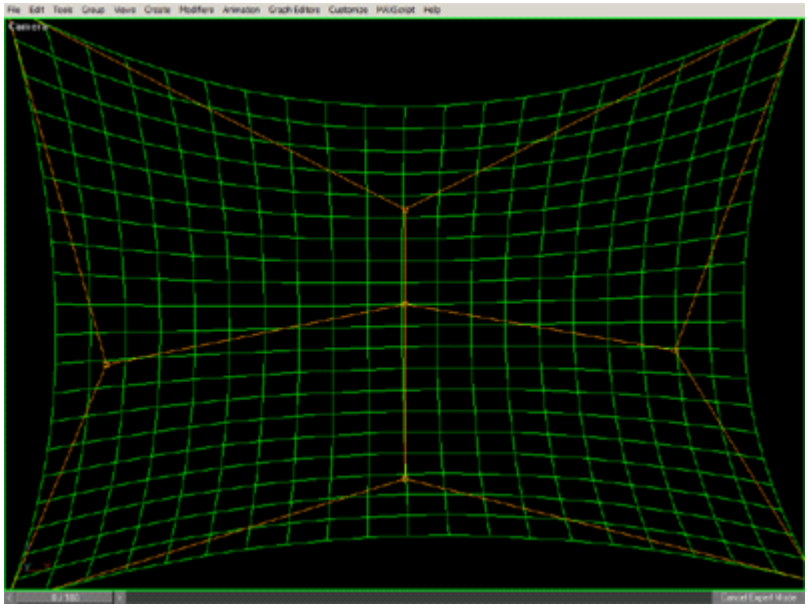
Please Note: Always make sure that none of your points moves towards the Z axis, all control points must stay at the same distance from the camera at all times. To control this, check your control points positions in the top and left view.

You might wonder how you can actually draw and shape this plane according to your screen, the answer is only a few clicks away.

In order to control the FFD´s Control points you are in the control point level of the selected objects modifier stack. Once you are in this mode you can right-click in the camera view and press [Ctrl + X] to toggle to the expert fullscreen mode of gmax. By pressing "W" on the keyboard you may toggle to the maximized camera view.

Now start modeling your planar object according to the physical shape of the screen.

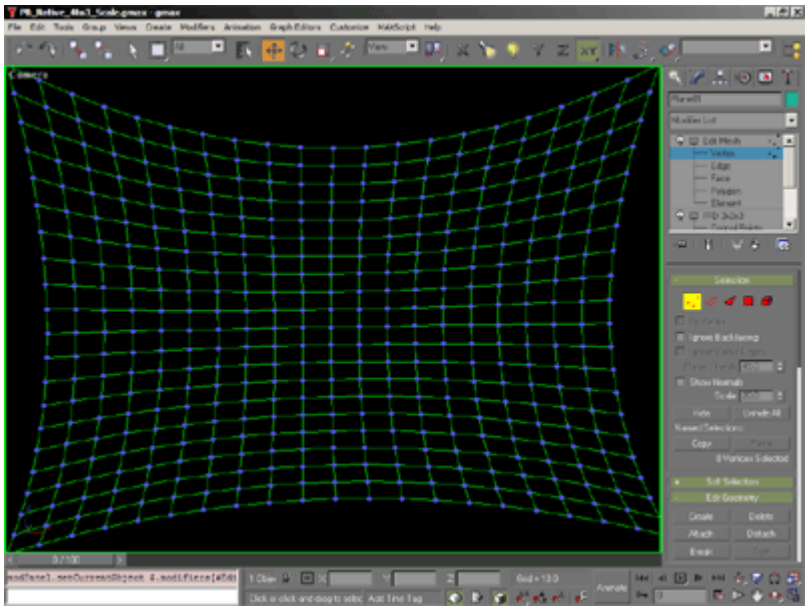
If you need to toggle to the front view while in expert mode, you may use the "F" key for front view, "L" for left view and "C" for the camera view.
The last gmax chapter includes all available shortcuts [2197].



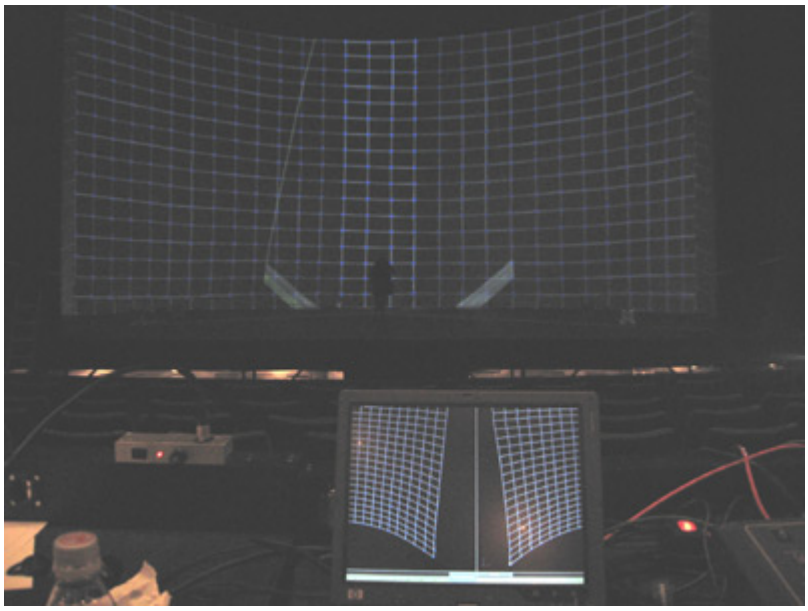
Please make sure that you make all adjustments to match your screens only according to the camera view, as this is your reference view for Pandoras Box output.

Once you're done with shaping the screen, you only have to export the selected object from the file menu and load the object into the output camera.

Anytime you need to get more detailed control over every grid point, explore the "edit mesh" modifier. Just add it to the stack and choose the Vertex level for in depth control of every segment of the shape.



This modifier will do a great job when you do Softedge setups with over-laying grids. The Edit Mesh modifier will allow you to get the overlapping vertices accurately on top of each other for a perfect shaped Softedge blend.



For this 180 degree screen, two gmax applications are edited at the same time. In the overlapping zone all vertices were touched up to match exactly the screen deformation.

Exporting an object

Please export your file as an X file. as described in the previous chapters:
- gmax Object Export for PB Vers. 4.1 [2184] or
- gmax Object Export for PB Vers. 4.5 [2185] (and higher)

9.12.1.4 gmax Keyboard Shortcuts

F3 Toggle textured view
F4 Toggle grid on textured view

F Front View
C Camera View
L Left View
R Right View
T Top View
P Perspective View

G Toggle Background Grid

Ctrl + X Toggle Fullscreen Mode

X Toggle XYZ Axis Control

9.12.2 3ds Max

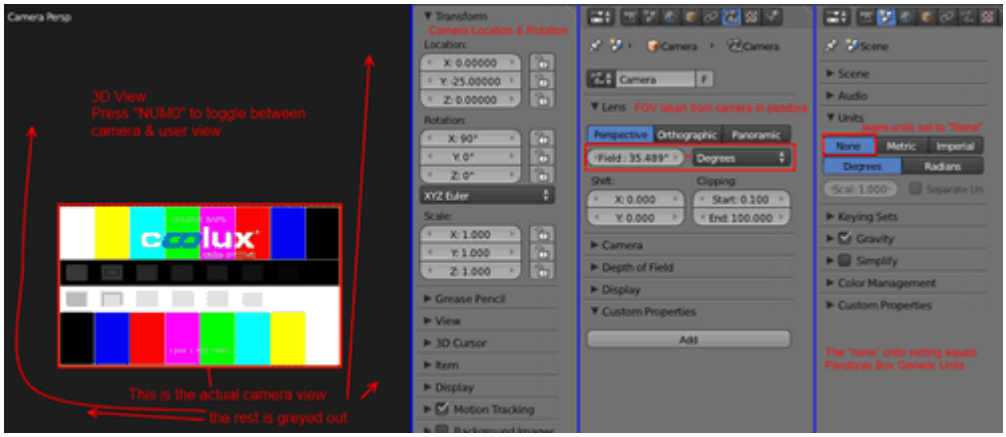
Please make yourself familiar with the basics described in the introducing chapter [2177] if you like using 3ds Max to create objects to be used in the Warper [2129] or in Pandoras Box [67]. Even more detailed information can be found in the Warper's chapter "General 3D Modeling Terms [2131]".

If you like to learn how to generally create an object [2180], please refer to the chapter in gmax as these programs are very similar.

9.12.3 Blender

Please make yourself familiar with the basics described in the introducing chapter [2177]. You need to know them when using Blender to create objects to be used in the Warper [2129] or in Pandoras Box [67]. Even more detailed information can be found in the Warper's chapter "General 3D Modeling Terms [2131]".

To set up generic units in Blender, usually you do not have to do anything. The "None" setting matches the generic unit already. In the same dialog you may adjust the camera to match Pandoras Box' camera.



If you like to learn how to generally create an object [2180] and how to apply an UVW Map, please refer to the chapters in gmax.

The next chapter describes how to create and export objects in Blender [2198].

9.12.3.1 Object Creation and Export Using Blender

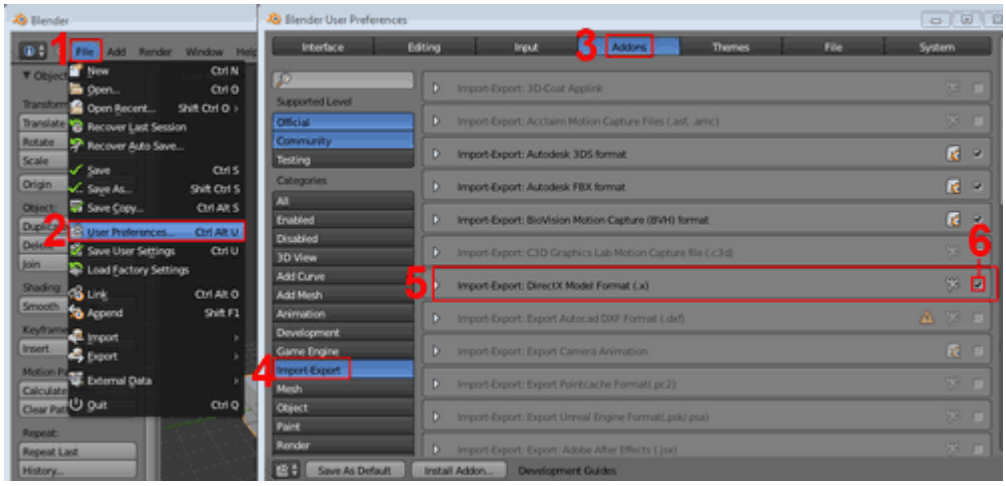
Creating objects in Blender

Object creation is very similar in every 3D program, please have a look at the gmax chapter [2178] as well.

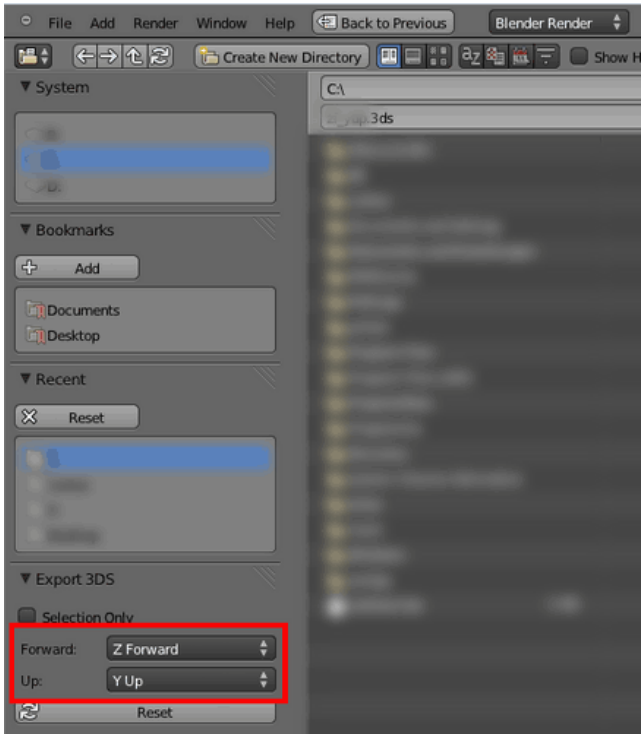
Exporting from Blender to Pandoras Box

If you are done with your object and are ready to export it, first of all, go into Object Mode and press [Ctrl + A]. Then select "Rotation & Scale". This will apply all your rotation and scaling. It might also be required to apply "Location". Now you can export either as X or 3DS.

To export directly to Pandoras Box simply choose "File->Export->Direct X (.x)". You might have to activate the plugin first (see image):



The Warper, may as well import files saved in the 3DS format. Choose "File->Export->3DStudio (.3ds)" to export a 3DS file. Refer to the following image for the correct export settings:



Christie
Pandoras Box

Index

- - 1 - -

12G-SDI Input Cards 1972

- - 3 - -

3ds Max
- Third Party Software 2197

3G-SDI Input Cards 1969

- - A - -

Abs, Filter Node, Widget Designer 1114
Accessories 1938
ACos, Filter Node, Widget Designer 1115
Active Values 127
ADAT Sound Card 1982
Add Layers, Using Widget Designer 1503, 2148
Add Relative, Filter Node, Widget Designer 1094
Add, Filter Node, Widget Designer 1116
Aeon FX 129
Airbar, Input Node, Widget Designer 979
AirScan
- Calibration, Widget Designer 1280
- General Settings, Widget Designer 1278
- Input Point Processing, Widget Designer 1279
- Mouse Control, Widget Designer 1283
- Point Tracking, Widget Designer 1281
- Widget Designer 1277

AirScan Hardware 1988
AirScan Multi-Point, Input Node, Widget Designer 980
AKAI APC40, Widget Designer 1488
Align Function, PB 124
Align Layers, Using Widget Designer
- Array 1504
- Circle 1505

Alpha channel in mpeg 2073
Analog Clock, Widget Designer 819
Angle To Point, Filter Node, Widget Designer 1095
Angle/Distance to X/Y, Filter Node, Widget Designer 1097
Animation, FX 339
Anti-Aliasing, FX 340
Art-Net
- Connection Manager, Widget Designer 1259
- Input Node, Widget Designer 953
- Output Node, Widget Designer 1179

Art-Net Monitor 2051
Art-Net Recorder, Button, Widget Designer 841
ASCII, Widget Designer 944
ASin, Filter Node, Widget Designer 1117
Assets 131
ATan, Filter Node, Widget Designer 1118
Audio
- Video Layer 650

Audio (Beta) , Input Node, Widget Designer 1038
Audio Card
- ADAT 1982
- MADI 2031

Audio Formats 92
Audio Layer
- Introduction 661

Audio Track
- Inpoint/Outpoint 664
- Media 662
- Pan 665
- Playback 663
- Volume 663

- - B - -

Backbuffer 320
Backup Menu 123
Bar Graph
- Output Node, Widget Designer 1226
- Widget Designer 852

Barco ImagePro2
- Configuration 1306
- Events 1307
- Members 1307

Barco MatrixPro2
- Configuration 1314
- Events 1315
- Members 1315

Behringer BCF 2000, Widget Designer 1489
Blacklevel Compensation 646
Blacklist, Widget Designer 1490
Blender
- Object creation and export 2198
- Third Party Software 2198

Blur, FX 342
Browser Inspector 198
Buttons, Widget Designer 821

- - C - -

Calibration Link 2000
- Input Node, Widget Designer 981
- Software 2008

Cam Wireframes, Preview Tab 249
Camera Inspector 218
Camera Layer
- Background Color 681
- Introduction 675

Camera Layer
Lens Shift 680
Projection Mode 676
Settings 680
Target 678
Viewpoint 677
Camera Point Tracker
Input Node, Widget Designer 1067
Camera Projection, FX 359
Camera Settings, Camera Point Tracker, Widget Designer 1292
Camera Tracker
Widget Designer 1291
Canvas Inspector 199
Canvas, Preview Tab 256
Ceiling, Filter Node, Widget Designer 1119
Changing Parameter Values
Device Control tab 171
Device Tree tab 177
CITP Desktop Streamer 2052
Client
Keyboard Shortcuts 317
Live Input Configuration 317
PB Client 316
Clip Inspector 206
Clip Summary Inspector 206
Clock , Input Node, Widget Designer 1041
Color
Light Layer 670
Color Effects - Crop, FX 370
Color Effects - Softborder, FX 383
Color Effects, FX 360
Color Picker
Input Node, Widget Designer 1076
Widget Designer 880
Color Picker, FX Parameter 324
Color Processing, Camera Point Tracker, Widget Designer 1294
Color Shift, FX 394
COM Port ASCII Stream, Output Node, Widget Designer 1180
COM Port, Output Node, Widget Designer 1181
COM Query String, Input Node, Widget Designer 957
COM Scripts, Widget Designer 1239
COM Stream, Input Node, Widget Designer 956
COM, Input Node, Widget Designer 954
Command List, Widget Designer 1520
Compact Player Hardware 1944
Compare Text, Filter Node, Widget Designer 1152
Compositing, FX 406
Composition Pass 320
Configuration, PB Tab 139
3D Hardware 151
ASIO Audio 166
Cache 147
Client Display 158
Controller Setup 154
Devices / Parameters 156
Local Preview 150
Network 147
Preview Display 145
Remote Control Protocols 148
Render Engine 162
Resources (Global) 155
Resources (User) 140
Sequence 142
SMPTE Link Timecode 152
Startup 150
Unit Management 160
Web Server 153
Configuration, WD 1305
Barco ImagePro2 1306
Barco MatrixPro2 1314
Event Listener 1353
Fader Extension 1354
Jog/Shuttle 1364
Lightware LW2 1378
Phidgets 1383
Phoenix 1319
PowerPoint Remote 1468
Spyder 1322
TCP Client 1471
Terra 1332
UDP Client 1475
UDP Server/Client 1479
Conformity 16
Connection Manager, Widget Designer 1258
Contains Text, Filter Node, Widget Designer 1153
Content
Audio Formats 92
Display and Content Formats 90
Image Formats 93
Image Sequence Formats 95
Introduction 90
Video Formats 97
Controller Boards 1991
Controller Setup, Configuration, PB 154
Copyright Pandoras Box 783
Cos, Filter Node, Widget Designer 1120
Count Down , Input Node, Widget Designer 1042
Counter , Input Node, Widget Designer 1043

Creating and Setting up Nodes, Widget Designer 937
Cropping, FX 412
Cue Control, Button, Widget Designer 831
Cue Generator, Using Widget Designer 1506
Cue Inspector 208
Cues
Tab 168
Curve Editor 169
Custom Script Button, Widget Designer 822
Custom Script, Input Node, Widget Designer 1077

- - D - -

Damping Predicition, Filter Node, Widget Designer 1102
Damping Timed, Filter Node, Widget Designer 1101
Date To Text, Filter Node, Widget Designer 1157
DEC, Widget Designer 944
Degree > Radians, Filter Node, Widget Designer 1121
Delta, Filter Node, Widget Designer 1104
Device Control
Tab 171
Device Control tab
Changing Parameter Values 171
Expand & Collapse Parameter Sections 172
Device Inspector 210
Device Settings, Multitouch, Widget Designer 905
Device Tree
Changing Parameter Values 177
Context menu 178
Device Selection 182
General information 173
Site, Layer and Parameter Structure 175
Device Types 183
Audio Track 661
Camera 675
DMX Devices 691
Light Layer 666
Notch Layer 671
Output 682
Pointer Layer 665
Serial Link Device 692
Venue Site 694
Video Layer 647
Widget Designer Device 696
Device Viewer 184
Devices and Layer Parameters 318
Digital Clock , Widget Designer 850
Digital Display
Output Node, Widget Designer 1227
Widget Designer 854
Displays , Widget Designer 851
Distort, FX 429
Divide, Filter Node, Widget Designer 1122
DMX Addressing 2083
DMX Devices
Introduction 691
DMX Input, Input Protocol 706
DMX Link 1999
DMX Link 8 2009
DMX Link In, Input Node, Widget Designer 958
DMX Link Out, Output Node, Widget Designer 1182
DMX Output, Output Protocol 721
DMX Sheet
Audio Track 708
Overview 708
Sequence Control 708
Video Layer 709
DMX Table
Audio Track 708
Overview 708
Sequence Control 708
Video Layer 709
DMX-Link, Connection Manager, Widget Designer 1260
Dome Master 2055
Drawing Canvas, Widget Designer 893
Drop Down List, Widget Designer 868
Dropdownlist , Input Node, Widget Designer 1078
DVI Input Cards 2014
DVI Processor
Create a new Timing 2024
Hardware 2018
Software 2019
DVI Processor, Quad Setup Menu 2017
Dynamic Trigger, Filter Node, Widget Designer 1105

- - E - -

EDID Link 2026
Hardware 2027
Software 2028
Edit Menu, PB 121
Editable Mesh 257
Effects
PB Tab 129
Video Layer Control 322
Electrical Information 16
Email Settings, Widget Designer 1490
Email, Input Node, Widget Designer 1068
Encoder
Input Node, Widget Designer 1079

Encoder
Widget Designer 877
Encoding
Encoder Extension 103
Encoder Extension Tab 185
Overview 100
Event Listener, WD 1353
Event Viewer, Widget Designer 871
Events
Input Node, Widget Designer 1069
Tools, Widget Designer 1483
Excel Reader , Input Node, Widget Designer 1035
Excel Writer, Output Node, Widget Designer 1206
External Control 706

- - F - -

Face Tracker, Widget Designer 1287
Face Tracking , Input Node, Widget Designer 1070
Factory Reset Tool (Acronis True Image) 2060
Fader
Controls, Widget Designer 874
Input Node, Widget Designer 1080
Output Node, Widget Designer 1228
Widget Designer 874
Fader Extension
Configuration, WD 1354
Events 1355
Hardware and Accessories 1993
Input Node, Widget Designer 983
Members 1356
File Inspector
Options in the tab 191
File Menu
Pandoras Box 119
Filter Nodes, Widget Designer 1092
FireFly Particles 185
Flipswitch, Button, Widget Designer 848
Floor, Filter Node, Widget Designer 1123
Folder Inspector 197
For Loop, Widget Designer 1890
Fractals, FX 443
Functions, Widget Designer 1897
FX
Adding, Editing and Removing FX 323
Copy / Paste FX 326
FX List 327
FX Order 325
Video Layer Control 322
FX Collections, FX 444
FX Explorer 129

- - G - -

Gauge
Widget Designer 856
Gauge (Angular Display)
Output Node, Widget Designer 1229
General Warping Questions 259
Generator, FX 445
Geometry, FX 447
Gizmo 250
Gizmo, Preview Tab 249
gmax
advanced steps 2189
general object creation in gmax 2180
Keyboard Shortcuts 2197
object creation and export 2179
Software Setup 2178
Third Party Software 2178
warping with gmax 2189
GPS , Input Node, Widget Designer 984
Gradients, FX 452
Graphic Settings 1948
Groups 189

- - H - -

Hardware 1938
HEX, Widget Designer 944
HTTP Listener, Widget Designer 1274

- - I - -

ID Tag Tracker, Input Node, Widget Designer 985
If Else, Widget Designer 1886
If, Filter Node, Widget Designer 1106
Image Adjust - Crop, FX 478
Image Adjust - Iris, FX 487
Image Adjust - Mediamask, FX 498
Image Adjust - Softborder, FX 505
Image Adjust, FX 464
Image Converter
Overview 2069
Settings 2075
Image Formats 93
Image Loader, Button, Widget Designer 833
Image Processing, Camera Point Tracker, Widget Designer 1294
Image Resource Manager, Widget Designer 1509
Image Sequence Formats 95
Image Sequence Inspector 199
Inpoint/Outpoint
Audio Track 664
Input / Output Control 706
Input and Output Cards 1947

Input Nodes, Widget Designer 952
Input Protocols 706
Inspector 190
Installation
Widget Designer 787
Intensity
Light Layer 668
Interaction Nodes, Widget Designer 1244, 1254
iPhone Remote , Input Node, Widget Designer 1008
iPhone WD Remote, Widget Designer 1275
IR ID Tracking 985

- - J - -

Jitter Reduce, Filter Node, Widget Designer 1111
Jog/Shuttle
Configuration, WD 1364
Events 1365
Hardware and Accessories 1991
Input Node, Widget Designer 1009
Members 1366
Jog/Shuttle Setup, Configuration, PB 154
Joystick , Input Node, Widget Designer 1010

- - K - -

Keyboard Input, Widget Designer 1483
Keyboard Shortcuts
gmax 2197
PB Client 317
PB Master 314
Warper 2176
Widget Designer 816
Keying, FX 512
Keystone
Output Layer 687
Kinect
Input Node, Widget Designer 1011
Widget Designer 1283
Kinesys , Input Node, Widget Designer 1012

- - L - -

Label
Input Node, Widget Designer 1082
Output Node, Widget Designer 1232
Widget Designer 888
Layer Inspector 211
Layer Picking 252
Layer UV Draw to Canvas, Interaction Node, Widget Designer 1245
LCD Menu 1942
Leading Zeros, Filter Node, Widget Designer 1158
Light Layer
Color 670
Intensity 668
Introduction 666
Media 667
Position 669
Settings 670
Target 669
Light Layer Inspector 214
Light Wireframes, Preview Tab 249
Light, FX 527
Lighting - Texture Projection, FX 538
Lightware LW2
Configuration 1378
Events 1379
Members 1379
List View, Widget Designer 890
Live Input
Configuration in PB Client 317
Input Cards 1947
Settings for StreamiX Desktop 728
Video Recording Tab 137
Live Inputs
PB Asset Tab 131
Log, Filter Node, Widget Designer 1124
Logitech Presenter R400, Widget Designer 1485
LUT (look up table), FX 544

- - M - -

Macros, Widget Designer 1897
MADI Sound Card 2031
Masking, FX 546
Math Filter Nodes, Filter Node, Widget Designer 1113
Matrix Patcher
Edit Menu 2079
File Menu 2078
Fixture Editor 2085
Fixture Library 2082
Fixture Navigation Tools 2083
Fixture Patch 2083
Overview 2077
Patch Settings 2085
Patching Guide 2089
Remapping Guide 2093
Toolbar 2081
User Interface 2077
Max, Filter Node, Widget Designer 1138
Media
Audio Track 662
Light Layer 667
Media Control Panel, Button, Widget Designer 828

Media Control , Input Node, Widget Designer 1083
Media Dongle, Media Encryption 222
Media Encryption
Decryption 226
Encryption 224
General Information 222
Policy Update 227
Media/Mesh
Video Layer 647
Menu Bar, PB 119
Mesh Editing Mode, Preview Tab 257
Midi In- & Output, Connection Manager, Widget Designer 1260
Midi In, Input Node, Widget Designer 959
Midi Input, Widget Designer 1485
Midi Message Editor, Widget Designer 1487
Midi Note Catch, Input Node, Widget Designer 960
Midi Note On/Off, Output Node, Widget Designer 1183
Midi Notes Editor, Widget Designer 1486
Midi Raw Message, Output Node, Widget Designer 1184
Midi Value, Output Node, Widget Designer 1185
Midi, Input Protocol 716
Min, Filter Node, Widget Designer 1139
Mirror, FX 555
Modbus
Master Configuration 1299
Master List 1304
Modbus FHPP Input 1013
Modbus in Widget Designer 1298
Modbus Input Node 1015
Modbus Output Node 1196
Patch Configuration 1300
Using Modbus 1302
Modulo, Filter Node, Widget Designer 1126
Motion Detector
Input Node, Widget Designer 1071
Widget Designer 1289
Mouse
Input Node, Widget Designer 1016
Output Node, Widget Designer 1197
Mouse Draw to Canvas, Interaction Node, Widget Designer 1249
Mouse Injection, Interaction Node, Widget Designer 1250
MSC Midi Show Control, Input Protocol 716
Multiply, Filter Node, Widget Designer 1127
MultiTouch Panel
Item , Input Node, Widget Designer 1084
Point , Input Node, Widget Designer 1086
Properties, Widget Designer 899
Widget Designer 899
Multi-User 234

- - N - -

NET Link 2000
Boards and Examples 2004
Hardware 2002
Input Node, Widget Designer 1016
Software 2008
Network
Configuration, PB 147
Network & Synchronization
Master / Client Remote Setup 67
Network Broadcasting Service, Widget Designer 1274
Network performance 2083
Node Commands, Widget Designer 1928
Nodes Tutorial, Widget Designer 939
Nodes, Widget Designer 936
Notch Layer 671
NVIDIA Cards
Changing the resolution 1949
Graphics Driver 1967
Setting up multiple displays 1952
Setting up the frame lock option 1961

- - O - -

Object Creation
Blender, Third Party Software 2198
gmax, Third Party Software 2179
Object Inspector 200
ODSL 30, Input Node, Widget Designer 1019
OnScreen Menu 2097
Opacity
Video Layer 649
Options, Widget Designer 1510
OSC, Input Node, Widget Designer 961
Output Inspector 220
Output Layer
Background Color 686
Introduction 682
Keystone 687
Matrix 683
Object 684
Pixel Warp 686
Render State 689
Softedge 688
Output Nodes, Widget Designer 1177
Output Pass 320
Output Protocols 721
Output Settings 1948
Overlay, FX 557

- - P - -

Pan
- Audio Track 665

Pandoras Box Tools, Widget Designer 1503

Panel Settings, Multitouch, Widget Designer 901

Panel, Widget Designer 892

Parameter Keyframe Inspector 209

Parameter List 1514

Parameter Names, Pandoras Box 1514

Particles 185

Patch DMX
- Audio Track 708
- Overview 708
- Sequence Control 708
- Video Layer 709

Patch for DMX
- Patch Tab 228
- Sequence 233
- Site 229

Pattern, FX 574

PB Automation 730

PB Device Control, Output Node, Widget Designer 1212

PB Menu 2097

PB Network Configuration, Widget Designer 1256

PB Sequence Control, Output Node, Widget Designer 1221

PB Sequence Seek, Output Node, Widget Designer 1222

PB Settings, Multitouch, Widget Designer 909

PB Text Parameter Control, Output Node, Widget Designer 1223

PB Text Unicode, Output Node, Widget Designer 1224

PB Text, Output Node, Widget Designer 1225

Percent, Filter Node, Widget Designer 1129

Phidgets
- Configuration 1383
- Servo output node, WD 1201

Phoenix
- Configuration 1319
- Events 1320
- Members 1321

Picture Box, Widget Designer 911

Ping , Input Node, Widget Designer 1047

Pivot Handles 250

Pivot Point, Preview Tab 249

Playback
- Audio Track 663
- Video Layer 649

Playlist
- Inspector 202
- Tab, PB 239
- Widget Designer 913

Point Tracker, Camera Point Tracker, Widget Designer 1296

Pointer Inspector 213

Pointer Layer
- Introduction 665

Polar > Rectangular, Filter Node, Widget Designer 1140

Ports Used by PB and WD 781

Position
- Light Layer 669
- Video Layer 651

Power X^Y, Filter Node, Widget Designer 1130

PowerPoint Display, Widget Designer 860

PowerPoint Remote
- Configuration 1468
- Events 1470
- Members 1471

Preset View 241

Preview 243
- Camera View 245
- Canvas 256
- Gizmo 249
- Handles 249
- Including Sites into the Preview 244
- Interactive Preview 250
- Layer Picking 252
- Mesh Editing 257
- Navigation 248
- Output View 245
- Patch View 245
- Picking Mode 252
- Using Pivot Handles 250
- Wireframes 249

Product Overview 62

Product Structure 62

Programming Statements, Widget Designer 1886

Project Tab
- Context Menu 279
- Groups 281
- Media Files 278
- Patch Templates 290
- Presets 283
- Sequences 281
- Tabs 288
- Views 287

Projector Calibration Manager, Widget Designer 1496

Projector Control , Input Node, Widget Designer 1024

Projector Recalibration, Widget Designer 1496

PSN MA Tracking Server, Connection Manager, Widget Designer 1262

- - R - -

Radians>Degree, Filter Node, Widget Designer 1131
Random , Input Node, Widget Designer 1048
Range Asymmetric, Filter Node, Widget Designer 1143
Range, Filter Node, Widget Designer 1147
Recalibration, Widget Designer 1496
Regex Compare, Filter Node, Widget Designer 1158
Regex Text, Filter Node, Widget Designer 1160
Relay Output Link
- Output Node, Widget Designer 1202

ReMap, FX 580
Remote Touch, Input Node, Widget Designer 963
Remoting, Widget Designer 1273
Render Target 320
Rotation
- Video Layer 652

Round, Filter Node, Widget Designer 1132
RSS
- Input Node, Widget Designer 1072
- Tools, Widget Designer 1493

- - S - -

Scale
- Video Layer 653

Script
- Output Node, Widget Designer 1207
- Overview, Widget Designer 1511
- Timer, Button, Widget Designer 826

Script List, Widget Designer 1520
Script Nodes, Widget Designer 1237
Scripting, Widget Designer 1511
Scroller, Widget Designer 844
SDI Input Cards
- 12G-SDI 1972
- 3G-SDI 1969

SDI Processor
- Hardware 2018
- Software 2019

SDI Processor, Quad Setup Menu 2017
SDK 730
- Datatypes and Structures 742
- Examples for connecting, sending and receiving values 738
- Function Reference 745
- Getting Started 732
- Other Languages 735
- Using JavaScript 734
- Using Visual Basic and C# 733

Sensor Link 2033
- Analog Inputs 2035
- Configuration Tool and Widget Designer 2037
- Connecting 2034
- Encoder Inputs 2035
- Input Node, Widget Designer 1026
- Stand-Alone Software 2039
- Technical Features 2033
- User Interface 2037

Sensor Link, Switch Inputs S1-S8 2035
Sequence
- Button Bar and Context Menu 293
- Configuration tab, PB 142
- Enforcing Default Values 300
- Introduction 292
- Programming 296

Sequence Control 303
Sequence Inspector 204
Sequence Logger, Using Widget Designer 1507
Sequence Patch 233
Serial COM Connections
- Connection Manager, Widget Designer 1269

Serial Input, Widget Designer 1277
Serial Link 2044
- GPI, Input Node, Widget Designer 1028
- Input Protocol 719
- Output Node, Widget Designer 1189
- Output Protocol 722

Serial Link Device 692
Server Hardware 1939
Settings
- Light Layer 670

Shadows, FX 582
Shapes, FX 583
Sharpen, FX 607
Shortcut Editor, Widget Designer 1484
Shortcuts
- gmax 2197
- PB Client 317
- PB Master 314
- Warper 2176
- Widget Designer 816

Sin, Filter Node, Widget Designer 1133
Site Patch 229
SMPTE Link 2000
- Connection Manager, Widget Designer 1258
- Input Node, Widget Designer 965
- Input, Input Protocol 719
- Output, Output Protocol 723
- Timecode, Configuration, PB 152

SMS
- Input Node, Widget Designer 1073

SMS
Tools, Widget Designer 1494
Snapshot, Button, Widget Designer 839
Snapshot, FX 610
Softedge
FX 611
Output Layer 688
Softedge - Softedge P-Curve Alpha, FX 614
Softedge - Softedge P-Curve Gamma, FX 616
Software Development Kit 730
Sound Card
ADAT 1982
MADI 2031
Space Navigator, Input Node, Widget Designer 1029
Spark
Output Protocol 722
Overview 699
Splitter 2120
Edit Menu 2123
File Menu 2121
Toolbar 2125
Workflow 2127
Spyder
Configuration 1322
Members 1323
Output node, WD 1204
Sqrt, Filter Node, Widget Designer 1134
Startup Dialog, PB 118
Status Bar, PB 124
StreamiX Desktop 724
Desktop Region 726
General Information 728
Live Input Settings 728
Network Connection 726
Output Format 727
Transmission 727
User Interface 725
Strobe, FX 618
Stylize, FX 620
Sub Mesh Inspector 201
Subtract, Filter Node, Widget Designer 1135
Sum, Filter Node, Widget Designer 1136
Switch Statement, Widget Designer 1888
Syntax TCP- / UDP- / Serial Messages, Widget Designer 944

- - T - -

Tab Button Inspector 205
Tabs Menu, PB 122
Tabs Overview 125
Tan, Filter Node, Widget Designer 1137
Target
Light Layer 669
Taskmanager 305
TCP
Input Node, Widget Designer 968
Scripts, Widget Designer 1242
TCP Server, Remoting, Widget Designer 1274
TCP ASCII Stream
Input Node, Widget Designer 965
Output Node, Widget Designer 1191
TCP Client
Configuration 1471
Events 1472
Members 1473
TCP Connections
Connection Manager, Widget Designer 1265
TCP Message, Output Node, Widget Designer 1192
TCP or UDP via SDK, Pandoras Box, Input Protocols 721
TCP Query String, Input Node, Widget Designer 969
TCP/IP
Input Protocol 720
Output Protocol 723
TCP/UDP/COM Connections, Widget Designer 1272
Terra
Configuration 1332
Events 1333
Members 1335
Tutorial 1345
Text Combiner, Filter Node, Widget Designer 1165
Text Filter Nodes, Widget Designer 1150
Text Input Editor 307
Text Input, Widget Designer 926
Text Inspector 203
Text Reader, Input Node, Widget Designer 1036
Textbox
Input Node, Widget Designer 1088
Output Node, Widget Designer 1234
Widget Designer 923
Third Party Software
3ds Max 2197
Blender 2198
gmax 2178
Introduction 2177
Thumbnail Viewer 310
Timax, Input Node, Widget Designer 1030
Timecode, Widget Designer 928
Tools and Other Software 2051
Tools Menu
Pandoras Box 122

Tools, Widget Designer 1483
Touch Injection, Interaction Node, Widget Designer 1251
Touch Output Processing, Camera Point Tracker, Widget Designer 1297
Track Inspector 217
Tracking
Camera Tracking 1067
Face Tracking 1070
ID Tags 985
Motion 1071
TrackScan 1032
with lighting console 1005
TrackScan Serial Link, Input Node, Widget Designer 1031
TrackScan, Input Node, Widget Designer 1032
Tranparent information in mpeg 2073
Transition, FX 623
TUIO Transmitter, Connection Manager, Widget Designer 1261
Tutorial
Terra 1345

- - U - -

UDP
Input Node, Widget Designer 975
Listener, Remoting, Widget Designer 1274
Message, Output Node, Widget Designer 1194
Scripts , Widget Designer 1240
UDP ASCII Stream
Input Node, Widget Designer 972
Output Node, Widget Designer 1193
UDP Client
Configuration 1475
Events 1476
Members 1477
UDP Connections
Connection Manager, Widget Designer 1267
UDP Server/Client
Configuration 1479
Events 1480
Members 1480
User Interface
Layout 313
Matrix Patcher 2077
Menu Bar in PB 119
Pandoras Box 117
Sensor Link 2037
Startup Dialog in PB 118
Status Bar in PB 124
StreamiX Desktop 725
Tabs Overview 125
Warper 2136
Widget Designer 789
User Interface, Tabs
Active Values 127
Aeon FX 129
Assets 131
Configuration 139
Controller Setup 169
Cues 168
Device Control 171
Device Tree 173
Device Types 183
Device Viewer 184
Encoder Extension 185
FireFly Particles 185
FX Explorer 129
Groups 189
Inspector 190
Patch 228
Playlist 239
Presets 241
Preview 243
Project 278
Sequence 292
Sequence Control 303
Taskmanager 305
Text Input Editor 307
Thumbnail Viewer 310
Video Export 135
Video Recording 137
View Tab 310
Virtual Site 311
UV Mapping, FX 631

- - V - -

Value
Input Node, Widget Designer 1049
Output Node, Widget Designer 1208
Variable
Input Node, Widget Designer 1050
Output Node, Widget Designer 1209
Variable List, Widget Designer 1900
Venue Site 694
Video Export 135
Video Formats 97
Video Input Cards
12G-SDI 1972
3G-SDI 1969
DVI 2014
Video Input Display, Widget Designer 865
Video Layer
Audio 650
Blend Mode 657
Introduction 647
Media/Mesh 647
Opacity 649

Video Layer
Playback 649
Position 651
Rotation 652
Rotation Pivot 654
Scale 653
Scale Pivot 655
Video Layer Control
Effects 322
FX 322
Video Logger, Using Widget Designer 1507
Video Player Time, Output Node, Widget Designer 1235
Video Player Volume , Output Node, Widget Designer 1236
Video Player, Widget Designer 932
Video Processing Pipeline 320
Video Recording 137
Video Snapshot, Button, Widget Designer 837
View Tab 310
Virtual Site 311
VNC Panel, Widget Designer 894
Volume
Audio Track 663

- - W - -

Warp, FX 644
Warper
3D Modeling Terms 2131
Camera / View tab 2159
Cameras 2148
Dynamic Warping 2174
Edit Menu 2142
File Menu 2137
General tab 2151
Keyboard / Mouse tab 2158
Keyboard Shortcuts 2176
Live Warp 2174
Live Warp tab 2164
Marker tab 2161
Mask tab 2156
Menu Bar 2137
Mesh tab 2153
Meshes 2147
Overview 2129
PB Camera tab 2159
Tabs 2151
Texture tab 2155
Tools Bar 2149
User Interface 2136
Using an Object in PB 2168
Warping Guide 2166
Warping with several meshes 2170
Workflow 2166
Warping in PB, Theory 259
Warping in PB, Tutorial 264
WD Remote Touch, Connection Manager, Widget Designer 1260
Web Server
Pandoras Box 153
Widget Designer 1929
Webbrowser, Widget Designer 934
What's New 18
What's New in Widget Designer 47
Wheel
Input Node, Widget Designer 1091
Widget Designer 883
Widget Designer 786
Command List 1520
Edit 797
File 792
Installation 787
Keyboard Shortcuts 816
Nodes 936
Pages 805
PB Network Configuration 1256
Programming Statements 1886
Script Language 1511
Tools 1483
User Interface 789
View 799
Widgets 818
Widget Designer Device 696
Widgets, Widget Designer 818
WII, Input Node, Widget Designer 1034

- - X - -

XY Panel
Input Node, Widget Designer 1092
Widget Designer 896