Viz Artist

Version 3.11

Disclaimer

Vizrt provides this publication “as is” without warranty of any kind, either expressed or implied. This publication may contain technical inaccuracies or typographical errors. While every precaution has been taken in the preparation of this document to ensure that it contains accurate and up-to-date information, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained in this document. Vizrt’s policy is one of continual development, so the content of this document is periodically subject to be modified without notice. These changes will be incorporated in new editions of the publication. Vizrt may make improvements and/or changes in the product (s) and/or the program(s) described in this publication at any time. Vizrt may have patents or pending patent applications covering subject matters in this document. The furnishing of this document does not give you any license to these patents.

Technical Support

For technical support and the latest news of upgrades, documentation, and related products, visit the Vizrt web site at www.vizrt.com.

Created on

2018/08/27

Contents

1 Introduction 31
1.1 Related Documents 31
1.2 Customer Feedback and Suggestions 31
1.3 Customer Support Requests 32
1.3.1 Submitting a Support Request 33
1.3.2 Content of a Support Request 33
1.3.3 To Submit a Support Request 33
1.3.4 Attach Log Files to a Customer Request 35
2 Getting Started 36
2.1 Viz Artist/Engine Folders 36
2.1.1 Installation Folders 36
2.1.2 Data Folders 36
2.2 Viz Artist Startup and Close 37
2.2.1 To Start Viz Artist 37
2.2.2 To Add a Viz Artist Startup Option 38
2.2.3 To Log on a New User after Viz Artist has Started 40
2.2.4 Log on History 41
2.2.5 Viz Artist 64-bit or 32-bit Version 41
2.2.6 Viz Configuration 42
2.3 Viz Command Line Options 42
2.3.1 Systems with Two or More GPUs 44
3 Artist Interface Overview 46
3.1 Main Window 46
3.2 Dual Monitor Configuration 47
3.2.1 Normal 47
3.2.2 Mosaic Horizontal 47
3.2.3 Mosaic Vertical 48
3.2.4 Dual Screen 48
3.3 Main Menu 48
3.4 Main Menu Left 49
3.4.1 Server Button 49
3.4.2 Built Ins Button 49
3.4.3 Control Button 50
3.4.4 Views Button 50

3.4.5 Tree Button 50
3.4.6 Stage Button 50
3.4.7 Server/Stage Button 51
3.4.8 +Tree/Stage Button 51
3.5 Main Menu Right 51
3.5.1 Import 51
3.5.2 Archive 52
3.5.3 Deploy 52
3.5.4 Config 52
3.5.5 Post 52
3.5.6 On Air 52
3.5.7 Button Bar 52
4 Manage Items and Built Ins 54
4.1 Server Panel 54
4.2 Server Tree 55
4.2.1 Folders 55
4.2.2 Server Tree Menu Bar 56
4.2.3 Server Tree Context Menu 57
4.2.4 Bookmarked Folders 57
4.3 Item Panel 60
4.3.1 Item List 61
4.3.2 Item Context Menu 62
4.3.3 Content Switcher 65
4.3.4 Server Menu 67
4.3.5 Properties Panel 68
4.4 What are items 71
4.4.1 Scenes 72
4.4.2 Scenes for Context-enabled Shows 72
4.4.3 Geometries 74
4.4.4 Materials 74
4.4.5 Materials Advanced 75
4.4.6 Images 75
4.4.7 Fontstyles 76
4.4.8 Audio Clips 76
4.4.9 Video Clips 77
4.4.10 Substances 77
4.5 Working with Items 78

4.5.1 Common Item Procedures 78
4.5.2 Active Items 81
4.5.3 Linked Items 83
4.5.4 Referenced Items 83
4.5.5 Working with Audio (Clips) Items 84
4.5.6 Working with Fontstyle Items 86
4.5.7 Working with Geometry Items 87
4.5.8 Working with Image Items 87
4.5.9 Working with Material and Material Advanced Items 88
4.5.10 Working with Scene Items 89
4.5.11 Working with Substances 90
4.5.12 Working with Video Items 91
4.6 Container and Scene Properties 92
4.6.1 To Add Properties to an Empty Scene Tree 92
4.6.2 To Add Properties to a Container 93
4.6.3 To Multi Drop Properties as Sub Containers 94
4.6.4 To Multi Drop Properties on Containers 95
4.6.5 To Add Multiple Properties to Containers 95
4.6.6 To Add Properties to Multiple Containers 96
4.6.7 To Add a Scene plug-in to a Scene 97
4.7 Assign Keywords to Items 97
4.7.1 To Create Keywords for items 98
4.7.2 To Assign Keywords from the Search Panel 98
4.7.3 To Assign Keywords from the Keyword Panel 99
4.7.4 To Assign a Keyword While Importing 100
4.7.5 To Assign Keywords from the Property Editor 101
4.7.6 To Remove a Keyword from an item 102
4.7.7 To Delete a Keyword 104
4.8 Image Editor 105
4.8.1 General Image Information 106
4.8.2 Image Effects 106
4.8.3 Apply and Preview Effects 109
4.9 Fontstyle Editor 110
4.9.1 Modifiers 111
4.9.2 Working with Modifiers 112
4.9.3 Preview Menu 117
4.9.4 Fontstyle Viewer 118

4.10 Material Editor 119
4.10.1 To Create a New Material 119
4.10.2 Material Color Properties and Schemes 120
4.11 Item Search 124
4.11.1 Search Panel 125
4.11.2 Search Parameters Area 125
4.11.3 Search Menu 126
4.11.4 Property Search 126
4.11.5 Reference Search 128
4.11.6 Keyword Search 131
4.11.7 Directory Search 132
4.11.8 Combination Search 134
4.11.9 Search Result Panel 134
4.12 Free Text Search 138
4.12.1 Search Pane 138
4.12.2 Result Pane 139
4.12.3 Entries Pane 139
4.12.4 Advanced Free Text Search 140
4.12.5 Wildcard text description 140
4.13 Background Loading 141
4.13.1 Automatic Background Loading 141
4.13.2 Manual Background Loading 141
4.13.3 Background Loading Limitations 142
4.14 Built Ins 143
4.14.1 Content Tabs 144
4.14.2 Plug-in Context Menu 145
4.14.3 View and Search for plug-ins 146
4.14.4 Add Plug-ins to Containers 148
4.15 Substance Editor 150
4.15.1 Working with Substances 150
4.15.2 Substance Basic 151
4.15.3 Substance Settings 153
4.15.4 How to create Substance shaders with Substance Designer 154
4.15.5 Simple texture shader 156
4.15.6 Basic procedural shader 157
4.15.7 Complex procedural shader 158
4.15.8 Publishing shaders 158

4.15.9 Substances Performance 159
5 Scene Tree 161
5.1 Scene Tree Menu 162
5.1.1 Tree Button 163
5.1.2 Sort Button 164
5.1.3 Search Button 166
5.2 Favorites Bar 170
5.2.1 To add an item as a Favorite 171
5.2.2 To Remove an item as a Favorite 171
5.2.3 To Add a Favorite item to a Container 171
5.3 Containers 171
5.3.1 Container Names 172
5.3.2 Container Icons 172
5.3.3 Container Context Menu 174
5.3.4 Add Containers to a Scene 177
5.3.5 Select Multiple Containers 178
5.3.6 Copy Containers 178
5.3.7 Move Containers 179
5.3.8 Rename Containers 180
5.3.9 Edit Container Properties 181
5.3.10 Delete Containers 182
5.3.11 Merge Containers 182
5.3.12 Group Containers 185
5.3.13 Container Editor 186
5.4 Modify Container Properties 187
5.4.1 Multiple Selection Container Properties 187
5.4.2 Edit Properties in the Scene Editor 189
5.4.3 Plug-in Editor 190
5.4.4 Material Editor 190
5.4.5 Quick Edit a Material in the Scene Editor 193
5.4.6 Manipulate Container Properties 193
5.5 Text Editor 194
5.5.1 Text Editor/Scene Editor 195
5.5.2 Basic Tab 198
5.5.3 Shadow Tab 201
5.5.4 Clock Tab 202
5.5.5 Characters Tab 204

5.5.6 Special Tab 204
5.5.7 To Create a Linked Fontstyle 206
5.5.8 To Modify Text in the Text Editor 206
5.6 Geometry Editor 206
5.6.1 Geometry Editor Properties 207
5.6.2 To Open the Geometry Editor 208
5.7 Transformation Editor 208
5.7.1 Transformation Editor Properties 208
5.7.2 Transformation Principles 210
5.7.3 Multiple Select Transformation 210
5.7.4 Copy Transformation Properties to Another Container 211
5.7.5 Quick Editor 211
5.7.6 How to use the Quick Editor 212
5.8 External Control 214
5.8.1 Control Channels 215
5.8.2 Control Objects 220
6 Scene Management 222
6.1 Open a Scene 222
6.1.1 To Open a Scene in the Database 222
6.2 Scene Settings 223
6.2.1 Clipper Panel 223
6.2.2 Global Settings Panel 225
6.2.3 HDR (High Dynamic Range) Panel 230
6.2.4 Media Asset Panel 231
6.2.5 Plug-in Panel 246
6.2.6 Rendering Panel 248
6.2.7 Script Panel 251
6.2.8 Real Time Global Illumination 252
6.2.9 Screen Space Ambient Occlusion 265
6.2.10 Virtual Studio Panel 267
6.2.11 Viz Libero and Viz Arena Render Sequences 268
6.3 Scene Editor 270
6.3.1 Grid Tool-bar 271
6.3.2 Layer Manager 273
6.3.3 Performance Bar 276
6.3.4 Scene Editor Buttons 278
6.3.5 Snapshot 279

6.4 Working with the Scene Editor 281
6.4.1 Scene Editor Functions 281
6.4.2 Scene Editor Context Menu 285
6.5 Scene Editor Views 291
7 Media Assets 293
7.1 Media Asset Manager 293
7.1.1 Media Asset Overview 294
7.1.2 Media Assets Added as DVE 297
7.1.3 Media Assets Added as Texture 298
7.1.4 To Create a Name for a Media Asset 299
7.2 Media Asset Workflow 299
7.2.1 Media Assets as DVE 300
7.2.2 Media Assets as Texture 304
7.2.3 Media Assets as a Background or Foreground 307
7.3 Media Asset Channel Types 307
7.3.1 Channel Folder Media Assets 308
7.3.2 Clip Channel Media Asset 308
7.3.3 Container Folder Media Assets 309
7.3.4 GFX Channels 313
7.3.5 Image Channels 315
7.3.6 Live Video Media Asset 317
7.3.7 Stream Media Asset 320
7.3.8 Super Channels 324
7.4 Playback of Media Assets 330
7.4.1 Media Asset Application 331
7.5 Video Clips 331
7.5.1 Video Clip Storage 332
7.5.2 Playback of Clips with VBI 333
7.5.3 Video Clip Playback 334
7.5.4 Video Clip Playback Considerations 345
7.5.5 Transfer Clips From Viz One 356
7.6 Keying Mode 356
7.6.1 Linear Key 356
7.6.2 Chroma Key 357
7.6.3 M-zone Key 357
7.6.4 Libero Key 357
7.6.5 Keying Best Practices 357

7.6.6 Keying Mode Configuration 358
7.7 Seamless Input Channel Switcher 366
7.8 Supported Codecs 367
7.8.1 Matrox Codecs 367
7.8.2 QuickTime Codecs for the Viz Artist/Engine 64-bit Platform 367
7.8.3 PAL Codecs 369
7.8.4 NTSC Codecs 373
7.8.5 720p50 and 720p59.94 (60M) Codecs 377
7.8.6 1080i25 and 1080i29.97 (30M) Codecs 382
7.8.7 1080p50 and 1080p59.94 (60M) Codecs 386
7.8.8 2160p UHDTV Codecs 387
7.8.9 QuickTime Writer plug-in Codecs 388
7.8.10 DVS and Bluefish Codecs 389
7.9 Advanced Issues with Video Codecs 390
8 Lights 391
8.1 Types Of Light 391
8.1.1 Local Light 391
8.1.2 Spot Light 391
8.1.3 Infinite Light 392
8.2 Light Editor 392
8.2.1 Light Editor Left Panel 392
8.2.2 Light Editor Right Panel 393
8.2.3 Light Editor Procedures 395
8.3 Light Visualization 400
8.3.1 To Modify the Light Direction with Light Handles 402
8.4 Light Source Animation 403
8.5 Shadow Maps 403
8.5.1 About Shadow Maps 404
8.5.2 To Use Shadows in a Scene 405
9 Cameras 406
9.1 Camera Editor 406
9.1.1 Camera Editor Left Panel 407
9.1.2 Camera Editor Right Panel 408
9.2 Stereo Settings 419
9.3 Stereoscopy Best Practices 420
9.3.1 What to do 420
9.3.2 What not to do 421

9.4 Stereoscopic Output Using Shutter Glasses 421
9.4.1 Hardware Requirements 421
9.4.2 Software Requirements 422
9.4.3 To Set Up a 3D Working Environment 422
9.5 Change Camera Parameters in Orthogonal Views 423
9.5.1 To View the Predefined Orthogonal Views 423
9.5.2 To Change the Camera Views 424
9.6 Track Objects with a Camera 424
9.6.1 To Track an Object 425
9.7 Receive Tracking Data from a Real Camera 426
9.7.1 To Receive Tracking Data from a Real Camera 426
9.8 Copy Properties from one Camera to Another 426
9.8.1 To Copy Camera Properties 426
9.9 Camera Selection 427
9.9.1 Camera Selection Panel Properties 428
9.9.2 Camera Selection in the Scene Editor Views 430
9.10 Camera Animation 432
9.11 Advanced Lens Distortion 432
9.11.1 Important changes compared to previous versions 433
9.11.2 Distortion algorithms 434
9.11.3 Parameter providers 435
9.11.4 Configuration file flags 436
9.11.5 Lens compatibility mode 436
10 The Stage for Animation 437
10.1 Stage Tree Area 438
10.1.1 Stage Tree 438
10.1.2 Stage Tree Bar 439
10.1.3 Stage Tree Menu 440
10.2 Stage Editor 442
10.2.1 Stage Editor Menu 443
10.2.2 Stage Editor Bar 443
10.3 Time-line Editor 445
10.4 Time-line Marker 446
10.4.1 To Manually Move the Time-line Marker 447
10.4.2 To Move the Time-line Marker a Set Amount of Fields 447
10.4.3 To Move the Time-line Marker a Set Amount of Fields with Shortcuts 448
10.5 Artist Director Control Panel 448

10.5.1 Working with the Director Control Panel....449
10.6 Director Editor....450
10.6.1 Director Editor Properties....450
10.7 Master Clip....451
10.8 Actor Editor....452
10.8.1 Actor Editor for Integer Channels....452
10.9 Channel Editor....452
10.9.1 Common Channel Editor Properties....453
10.9.2 Position Channel Editor Properties....453
10.9.3 Action Channel Editor Properties....455
10.10 Dopesheet Editor....456
10.10.1 To Enable/Disable the Dopesheet Editor....457
10.10.2 To Enable the Dopesheet Editor Together with the Spline Editor....457
10.10.3 To Resize the Height or Width of the Dopesheet Editor....458
10.11 Spline Editor....458
10.11.1 Spline Editor Procedures....459
10.11.2 Editing Handles....460
10.11.3 Position in the Spline Editor....461
10.11.4 Rotation in the Spline Editor....461
10.11.5 Scale in the Spline Editor....461
10.11.6 Material in the Spline Editor....462
10.11.7 Additional Best Practices for the Spline Editor....462
10.12 Stage Object Editor....463
10.12.1 To Enable/Disable the Stage Object Editor....463
10.13 Key Frame Editors....463
10.13.1 Common Key Frame Editor Properties....464
10.13.2 Locked and Unlocked Key Frames....464
10.13.3 Action Key Frame Editor Properties....465
10.13.4 Position Key Frame Editor Properties....466
10.13.5 Rotation Key Frame Editor Properties....468
10.13.6 Scale Key Frame Editor Properties....469
10.13.7 Material Key Frame Editor Properties....470
10.13.8 Clip Key Frame Editor Properties....471
10.13.9 Adding Camera Events Using Key-frames....472
10.14 Event Editor....473
11 Create Animations....475
11.1 Directors....475

11.1.1 Directors in the Stage Tree....476
11.1.2 Directors in the Dopesheet Editor....477
11.1.3 Events in a Director....478
11.2 Actors....482
11.2.1 Actors in the Stage Tree....484
11.2.2 Actors in the Dopesheet Editor....486
11.3 Channels....486
11.3.1 Channels in the Stage Tree....487
11.3.2 Channels in the Dopesheet Editor....489
11.4 Action Channels....489
11.4.1 Action Channels in the Stage Tree....490
11.4.2 Action Channels in the Dopesheet Editor....491
11.5 Key Frames....491
11.5.1 To Add a Key Frame....492
11.5.2 To Select One or More Key Frames....492
11.5.3 To Move One or More Key Frames....493
11.5.4 To Move a Key Frame to a Specific Time....493
11.5.5 To Copy a Key Frame....493
11.5.6 To Rename a Key Frame....493
11.5.7 To Delete a Key Frame....494
11.5.8 Reverse the Order of Selected Key Frames....494
11.5.9 To Reverse the Order of Key Frames....494
11.6 Basic Animation Functions....495
11.6.1 To Play All Animations in a Director from the Start of the Time-line....495
11.6.2 To Play Multiple or All Animations in a Scene....495
11.6.3 To Play an Animation in Reverse....496
11.6.4 To Stop a Director....496
11.6.5 To Stop an Animation....497
11.6.6 To Stop all Animations....497
11.6.7 To Continue an Animation....497
11.6.8 To Jump to Start of an Animation....497
11.6.9 To Jump to the End of an Animation....497
11.6.10 To Delay Animations in a Director with Offsets....497
11.6.11 Scale Animations....497
11.6.12 To Scale Animations in a Director....498
11.6.13 To Scale a Region Within a Director....498
11.7 Create a Basic Animation....498

11.7.1 Animation Task 498
11.7.2 To Create the Basic Animation Scene 498
11.7.3 To Create a Basic Animation 499
11.7.4 To Create Key Frames with the Transformation Editor 500
11.8 Create an Advanced Animation 501
11.8.1 To Edit the First Key Frame 501
11.8.2 To Edit the Second Key Frame 502
11.8.3 To Create an Additional Key Frame 502
11.8.4 To Add a Stop Point 502
11.8.5 To Create Key Frames for Rotation 503
11.8.6 To Modify the Rotation Settings 503
11.8.7 To Run the Animations Independently 503
11.8.8 To Add and Animate a Material 503
11.9 Advanced Animation Functions 504
11.9.1 Animating Channels 505
11.9.2 Path Control 507
11.9.3 Spline Control 508
11.9.4 Animate Audio 510
11.9.5 Animate Cameras 511
11.9.6 Animate Light 512
11.9.7 Merging Animations 512
11.9.8 Multi Select Animation 512
11.10 Create an Over the Shoulder Scene 513
11.10.1 To Create a Scene 513
11.10.2 To Add an Image and Basic Geometries 513
11.10.3 To Add Materials 514
11.10.4 To Scale Geometries and Place them in Z Space 515
11.10.5 To Add Text 515
11.10.6 To Add Gradients 516
11.10.7 To Add Background items 516
11.10.8 To Add Flares 517
11.10.9 To Animate 518
11.10.10 To Animate Sub Containers 519
11.10.11 To Animate Single Containers 519
11.10.12 To Animate Single Objects 520
11.10.13 To Animate Out 522
11.10.14 To Fade out the Graphic with the Alpha plug-in 522
11.10.15 To Copy the Directors, Flip them Around, then Reverse the Animation 522

11.10.16 To Create a Separate Out Animation 523
11.10.17 To Add Stop Points 523
11.10.18 To Add Controls 524
11.11 Create a Stand-alone Scene 525
11.11.1 To Create a Scene 525
11.11.2 To Add a Background 526
11.11.3 To Add Text 526
11.11.4 To Create an In and Out Animation 528
11.11.5 To Add Stop Tags 530
11.11.6 To Add Key Functions to the Container 530
11.11.7 To Add Expose Properties 531
11.11.8 To Edit Multiple items with a Single Value 533
11.12 Create Transition Effects 533
11.12.1 To Create Dynamic Textures 534
11.12.2 To Create a Transition Scene 535
12 Import and Archive 537
12.1 Import of Files and Archives 537
12.1.1 Import Window 538
12.1.2 Import Menu 539
12.1.3 Supported Files for Import 539
12.1.4 Import Archives 543
12.1.5 Import Files 546
12.2 Archive of Graphical Resources 550
12.2.1 Archive Properties 551
12.2.2 Entry-Points Menu 551
12.2.3 To Create an Archive 553
12.3 Deploy items 554
12.3.1 Deploy Panel Overview 555
12.3.2 To Deploy Selected items 556
12.3.3 To Deploy Single items 556
12.4 Post Render Scenes 557
12.4.1 Post Render Screen 559
12.4.2 To Post Render a Scene 559
12.4.3 Post Render Templates 562
12.4.4 Post Rendering plug-ins 564
12.5 Placeholder names used for file-name expansion 568
13 Geometry plug-ins 570

13.1 Default 570
13.1.1 2D Line 571
13.1.2 2D Patch 572
13.1.3 2D Ribbon 575
13.1.4 Alpha Map 579
13.1.5 Arrow 580
13.1.6 Circle 581
13.1.7 Cog Wheel 582
13.1.8 Cone 583
13.1.9 Connector 584
13.1.10 Cube 585
13.1.11 Cycloid 586
13.1.12 Cylinder 587
13.1.13 Cylinder3 588
13.1.14 Dexter 589
13.1.15 DisplacementMap 603
13.1.16 Eclipse 604
13.1.17 Fade Rectangle 605
13.1.18 Filecard 606
13.1.19 Graph 607
13.1.20 Graph2D 608
13.1.21 Icosahedron 609
13.1.22 Image FX 610
13.1.23 Noggi 615
13.1.24 Pointer 616
13.1.25 Polygon 617
13.1.26 Rectangle 623
13.1.27 Ring 624
13.1.28 Roll 625
13.1.29 SoftClip Draw Pixels 626
13.1.30 Sphere 627
13.1.31 Spline Path 628
13.1.32 Spline Strip 629
13.1.33 Spring 631
13.1.34 Star 632
13.1.35 Torus 633
13.1.36 Trio Scroll 634
13.1.37 Wall 636

13.1.38 Wave 638
13.2 PixelFX plug-ins 639
13.2.1 Lens Flare Geometries 640
13.2.2 pxLensEnergyBolt 641
13.2.3 pxLensRays 641
13.3 Dynamics 642
13.3.1 Cloth 642
13.3.2 Cloth Flag 643
13.3.3 Flag 645
13.4 Primitives 646
13.4.1 N Quad 646
13.5 RealFX plug-ins 647
13.5.1 RFxSmoke 648
13.6 Ticker 652
13.6.1 Scroller 652
13.7 Visual Data Tools 655
13.7.1 Overview 655
13.7.2 Advanced Bar Chart Creation 656
13.7.3 Area Chart 662
13.7.4 Bar Chart 668
13.7.5 Line Chart 674
13.7.6 Pie Chart 680
13.7.7 Scatter Chart 685
13.7.8 Stock Chart 690
14 Container plug-ins 698
14.1 Arrange 698
14.1.1 Circle Arrange 698
14.1.2 Grid Arrange 700
14.1.3 Time Displacement 701
14.2 Container 703
14.2.1 BoundingBox 703
14.2.2 Cobra 703
14.2.3 Coco 704
14.2.4 Colin 705
14.2.5 Cora 706
14.2.6 Corena 708
14.2.7 Toggle 708

14.3 Container FX 712
14.3.1 Common Container FX Properties 713
14.3.2 CFX 2D Follow 714
14.3.3 CFX Alpha 715
14.3.4 CFX Arrange 715
14.3.5 CFX Color 716
14.3.6 CFX Explode 716
14.3.7 CFX Jitter Alpha 717
14.3.8 CFX Jitter Color 718
14.3.9 CFX Jitter Position 718
14.3.10 CFX Jitter Scale 719
14.3.11 CFX Plus Plus 720
14.3.12 CFX Rotate 720
14.3.13 CFX Scale 721
14.4 Control 722
14.4.1 Common Control plug-in Properties 723
14.4.2 Apply Shared Memory 723
14.4.3 Control Action 726
14.4.4 Control Action Table 726
14.4.5 Control Audio 727
14.4.6 Control Bars 728
14.4.7 Control Chart 728
14.4.8 Control Clip 729
14.4.9 Control Clock 730
14.4.10 Control Condition 730
14.4.11 Control Container 731
14.4.12 Control Data Action 732
14.4.13 Control Datapool 733
14.4.14 Control DP Object 733
14.4.15 Control FeedView 734
14.4.16 Control Geom 735
14.4.17 Control Hide in Range 735
14.4.18 Control Hide on Empty 736
14.4.19 Control Image 737
14.4.20 Control Key Frame 738
14.4.21 Control List 739
14.4.22 Control Map 741
14.4.23 Control Material 741

14.4.24 Control Multihop 742
14.4.25 Control Num 742
14.4.26 Control Object 743
14.4.27 Control Omo 748
14.4.28 Control Parameter 749
14.4.29 Control Payload 750
14.4.30 Control Pie 751
14.4.31 Control Scaling 752
14.4.32 Control Sign Container 753
14.4.33 Control SoftClip 753
14.4.34 Control Stoppoint 754
14.4.35 Control Targa Alpha 755
14.4.36 Control Targa Clip 755
14.4.37 Control Text 756
14.4.38 Control Video 757
14.4.39 Control World 758
14.4.40 Control Field Renamer 759
14.4.41 Placeholder 760
14.5 Default 760
14.5.1 Global Magnifier Controller 761
14.5.2 Screen2World 762
14.5.3 Trio Scroll item 762
14.6 Feed 764
14.6.1 Feed Activate 764
14.6.2 Hide in Range 765
14.6.3 Feed View 766
14.7 Global 767
14.7.1 Alpha 767
14.7.2 Audio 768
14.7.3 Clipper 772
14.7.4 Expert 774
14.7.5 Extrude 777
14.7.6 Glow 778
14.7.7 HDR 780
14.7.8 Key 781
14.7.9 Look-At 783
14.7.10 Mask Source and Mask Target 784
14.7.11 Lighting 788

14.7.12 Z-Sort 790
14.7.13 Projector Source and Projector Target 793
14.7.14 Script 794
14.7.15 Shadow Caster and Shadow Receiver 795
14.7.16 Synchronized Properties 798
14.7.17 Video Clip 799
14.7.18 Window Mask 800
14.8 Lineup 802
14.8.1 Tree Props 802
14.9 MultiTouch plug-ins 802
14.9.1 Mt2D Control plug-in 803
14.9.2 MtButton plug-in 806
14.9.3 MtNavigator plug-in 810
14.9.4 MtTelestrator plug-in 812
14.9.5 Plug-in Event and Notification System 814
14.9.6 Mt3D Control plug-in 815
14.10 PixelFX 818
14.10.1 pxLensMulti 818
14.11 Presenter 820
14.11.1 Bar 820
14.11.2 Bar Value 821
14.11.3 Bar Values 823
14.11.4 Pie Slice 824
14.11.5 Pie Values 825
14.12 pxColorWorks 827
14.12.1 PixelFX plug-ins 827
14.12.2 pxAddSubtract 828
14.12.3 pxBlackAndWhite 829
14.12.4 pxBrightContrast 829
14.12.5 pxColorMatch 829
14.12.6 pxGamma 830
14.12.7 pxHueRotate 830
14.12.8 pxMask 830
14.12.9 pxSaturation 831
14.12.10 pxStack 831
14.12.11 pxTint 833
14.13 RealFX 833

14.13.1 RealFX Architecture 834
14.13.2 RFxColliderSrc 834
14.13.3 RFxColliderTgt 835
14.13.4 RFxLatLong 836
14.13.5 RFxMagnet 836
14.13.6 RFxTurb 837
14.13.7 RFxVortex 837
14.14 Script plug-ins 838
14.15 Sounds 838
14.15.1 Text2Speech 838
14.16 SplineFX 840
14.16.1 2D Follow 841
14.17 TextFX 842
14.17.1 Common Text FX Properties 843
14.17.2 Convert Case 844
14.17.3 Mark Text 845
14.17.4 Text FX Alpha 847
14.17.5 Text FX Arrange 848
14.17.6 Text FX Color 850
14.17.7 Text FX Color Per Vertex 851
14.17.8 Text FX Emoticons 852
14.17.9 Text FX Explode 853
14.17.10 Text FX Jitter Alpha 855
14.17.11 Text FX Jitter Position 856
14.17.12 Text FX Jitter Scale 857
14.17.13 Text FX Plus Plus 858
14.17.14 Text FX Rotate 860
14.17.15 Text FX Scale 861
14.17.16 Text FX Size 862
14.17.17 Text FX Slide 863
14.17.18 Text FX Vertex Explode 864
14.17.19 Text FX Write 866
14.18 Texture 867
14.18.1 BrowserCEF 867
14.18.2 GeoGraffiti 875
14.18.3 Grabbit 877
14.18.4 GraffitiTex 885

14.18.5 Image Clip 888
14.18.6 Light Blur 891
14.18.7 MoViz 892
14.18.8 Noise 895
14.18.9 SoftClip 896
14.18.10 Tex Component 900
14.19 Ticker 901
14.19.1 Scroller Action 901
14.20 Time 903
14.20.1 Analog Watch 903
14.20.2 Clock Rotation 905
14.21 Tools 905
14.21.1 Advanced Counter 906
14.21.2 Autofollow 907
14.21.3 Autorotate 908
14.21.4 Bounding Actions 909
14.21.5 Cloner 910
14.21.6 Colorize 912
14.21.7 Counter 913
14.21.8 DVE Follow 913
14.21.9 Heartbeat 914
14.21.10 Hide On Empty 915
14.21.11 Image Link 916
14.21.12 Jack 917
14.21.13 Level Of Detail (LOD) 918
14.21.14 Magnify 919
14.21.15 Match It 920
14.21.16 Max Size 921
14.21.17 Max Size Lines 923
14.21.18 Object Zoom 923
14.21.19 Omo 924
14.21.20 Pablo 925
14.21.21 Parliament 926
14.21.22 Rotations Order 927
14.21.23 Slide Show 927
14.21.24 System Time 928
14.21.25 Temo 929
14.21.26 Text Auto Scale 930

14.21.27 TextBG 930
14.21.28 Text Link 932
14.21.29 Text Parameters 932
14.21.30 TransitionLayers 933
14.21.31 VCF Parameter 934
14.22 Transformation 935
14.22.1 Justifier 936
14.22.2 VertexBone and VertexSkin Plugin 937
14.23 Visual Data Tools 942
14.23.1 Area Stack 943
14.23.2 Bar Stack 944
14.23.3 Data Fit 946
14.23.4 Data Import 949
14.23.5 Data Label 952
14.23.6 Data Storage 955
14.23.7 Line Stack 957
15 Shader plug-ins 960
15.1 Default 960
15.2 Effects 960
15.2.1 Chroma Keyer 961
15.2.2 Fluid 962
15.2.3 Frame Mask 963
15.2.4 Image Mask 963
15.2.5 Soft Mask 964
15.2.6 Water Shader 965
15.3 Filter 965
15.3.1 Blend Image 966
15.3.2 Blur 967
15.3.3 Color Balance 967
15.3.4 Radial Blur 968
15.3.5 Sepia 969
15.3.6 Sharpen 969
15.4 Material 970
15.4.1 Anisotropic Light 970
15.4.2 Bump Map 971
15.4.3 Cartoon 972
15.4.4 Gooch 972

15.4.5 Normal Map 973
15.4.6 Simple Bump Map 973
15.5 PixelFX 974
15.5.1 PixelFXLenseFlare pxBCubic pxCCBase pxEqualize pxGradient pxInvert pxLensDistort pxMotionBlur pxNoise pxPixelate pxPosterize pxRecolor pxRipple pxSparkle pxTurbDissolve and pxTurbWipe pxTurbulence pxTwirl pxWaves 975
15.5.2 Common Properties 975
15.5.3 Most of the plug-in icons have mini icons that show certain properties of the particular plug-in: 975
15.5.4 PixelFXLenseFlare 976
15.5.5 pxBCubic 976
15.5.6 pxCCBase 977
15.5.7 pxEqualize 977
15.5.8 pxGradient 977
15.5.9 pxInvert 978
15.5.10 pxLensDistort 979
15.5.11 pxMotionBlur 979
15.5.12 pxNoise 979
15.5.13 pxPixelate 980
15.5.14 pxPosterize 980
15.5.15 pxRecolor 981
15.5.16 pxRipple 981
15.5.17 pxSparkle 981
15.5.18 pxTurbDissolve and pxTurbWipe 982
15.5.19 pxTurbulence 982
15.5.20 pxTwirl 983
15.5.21 pxWaves 983
15.6 RealFX 983
15.6.1 RFx2DSphere 985
15.6.2 RFxClouds 985
15.6.3 RFxFlame 985
15.7 RTT Advanced Materials 986
15.7.1 Cube Map 988
15.7.2 HDR 988
15.7.3 Normal Map 988
15.7.4 Fresnel 988
15.7.5 RTT 989
15.7.6 Anisotropic Light Shader 989
15.7.7 Brushed Metal Shader 990

15.7.8 Bump Optimized Shader 991
15.7.9 Bump Shader 994
15.7.10 Fabric Shader 995
15.7.11 Glass Shader 996
15.7.12 Gooch Shader 997
15.7.13 Lacquered Surfaces Shader 998
15.7.14 Metal Reflection Shader 999
15.7.15 Microstructure Shader 1000
15.7.16 Monitor Shader 1001
15.7.17 Velvet Shader 1002
15.8 Texture 1003
15.8.1 Drop Shadow 1003
15.8.2 Emboss 1004
15.8.3 MultiTexture 1004
15.8.4 Substance 1005
16 Scene plug-ins 1008
16.1 Default 1008
16.1.1 VCF 1008
16.2 Image 1009
16.2.1 Background Clip 1009
16.3 Libero 1011
16.3.1 EVSControl plug-in 1011
16.4 Lineup 1013
16.4.1 Tree Status 1013
16.5 MultiTouch plug-ins 1014
16.5.1 MtSensor plug-in 1014
16.6 Script plug-ins 1015
16.7 Texture 1016
16.7.1 Graffiti 1016
16.8 Tools 1017
16.8.1 Level Of Detail (LOD) Manager 1018
16.8.2 Scene Synchronized Properties 1019
17 On Air Mode 1020
17.1 Control Buttons 1021
17.2 Director Control Panel 1021
17.3 Viz Artist Performance 1022
17.3.1 Performance Analyzer 1022

17.3.2 Performance Editor 1023
17.4 On Air Information 1025
17.4.1 Basic Tab 1025
17.4.2 Clients Tab 1026
17.5 License Information 1027
17.6 Lens File Editor 1027
17.6.1 Lens File Editor Workspace 1028
17.6.2 Lens File Editor Contextual Menu 1029
17.6.3 To Create a New Lens 1030
18 Transition Logic 1031
18.1 Standalone versus Transition Logic Scene Design 1032
18.1.1 Take Lower Third 1032
18.1.2 Take OTS 1032
18.1.3 Take Out OTS 1032
18.1.4 Take Out Lower Third 1033
18.2 Toggle-Layer 1033
18.3 State Transition Animation 1034
18.3.1 Toggle layer states 1034
18.3.2 Defining States 1034
18.3.3 Identifying States 1035
18.4 Cross Animation 1038
18.5 Geometry Animation 1038
18.6 Master Scene 1039
18.7 Object Scene 1040
18.8 Tutorial 1041
18.8.1 Create a Project 1041
18.8.2 Create a Master Scene 1041
18.8.3 Create an Object Scene 1044
19 Scripting 1047
19.1 Script Editor 1047
19.1.1 Script Editor Properties 1048
19.1.2 Script Editor Search 1050
19.1.3 Script Editor Procedures 1050
19.2 Create and Run Scripts 1053
19.2.1 Example I 1053
19.2.2 Example II 1054
19.3 Create Script-based plug-ins 1055

19.3.1 To Create a Script-based plug-in 1056
19.3.2 To Edit a Script-based plug-in 1057
19.4 Control 3D Stereoscopic Clip Playback 1057
19.5 Program Examples 1058
19.5.1 Initializing Scripts 1058
19.5.2 Register Parameters 1059
19.5.3 Mouse Events I (Set) 1060
19.5.4 Mouse Events II (Start, Stop and Reverse) 1060
19.5.5 Mouse Events III (Grow and Shrink) 1061
19.5.6 Simple Gesture Recognition 1062
19.5.7 Tetris 1062
19.5.8 Calculate the Days of a Week and Month 1068
19.5.9 Update a Scene Using a Text File 1069
19.6 Event Pool 1071
19.6.1 Built-In Events 1071
19.6.2 Technical Notes 1072
19.6.3 Sample Scripts 1072
20 Shared Memory - SHM 1074
20.1 Data Sharing 1074
20.1.1 VizCommunication.Map 1075
20.1.2 Map Access Through a TCP or UDP Interface 1075
20.1.3 Map Access through the Command Interface 1075
20.1.4 Map Access from Within plug-ins 1075
20.1.5 Script Examples 1075
20.1.6 Set and retrieve data from the SHM through scripting 1077
20.2 External Data Input 1079
20.2.1 TCP Protocol 1080
20.2.2 UDP Protocol 1081
20.2.3 Plug-in API 1082
20.2.4 Command Interface 1082
20.3 Internal Data - Interactive Scene 1083
20.4 Synchronization 1084
20.4.1 TCP and UDP Synchronization 1085
20.4.2 External Control Synchronization 1085
20.4.3 Command Synchronization 1086
20.5 Snapshot 1087
21 Third Party Applications and Files 1088

21.1 Adobe After Effects 1088
21.1.1 System Requirements 1088
21.1.2 Importing Adobe After Effect Files 1089
21.1.3 Import Limitations 1090
21.2 CINEMA 4D 1091
21.2.1 System Requirements and Configuration 1091
21.2.2 CINEMA 4D Scenes for Import 1092
21.2.3 CINEMA 4D LiveLink 1096
21.2.4 CINEMA 4D Integration Limitations 1097
21.3 FBX Files 1099
21.3.1 Scene Graph Traversal 1100
21.3.2 Transformations 1100
21.3.3 Export FBX Files from 3ds Max and Maya 1102
21.3.4 Reimporting Maya FBX files 1107
21.4 TriCaster 1109
21.4.1 System Requirements 1109
21.4.2 NDI Protocol 1110
21.4.3 To Connect a Viz Engine to TriCaster 1110
21.4.4 To enable Audio pass through from Viz Engine to TriCaster 1110
21.4.5 To Set TriCaster Macros 1111
21.4.6 Viz Trio and Viz Engine for a TriCaster Production on the Same Laptop 1111
21.4.7 TriCaster NDI Support 1111
21.5 Ncam AR for Unreal Engine 4 1112
21.5.1 Requirements 1112
21.5.2 Installation 1113
21.5.3 Configuration 1113
21.5.4 Use Cases 1114
21.5.5 Scene Setup in Viz Artist 1116
21.5.6 Camera Control 1117
22 Keyboard and Mouse Shortcuts 1119
22.1 Application Controls and Shortcuts 1119
22.2 Integer and Float Controls 1120
22.3 Server Panel Shortcuts 1121
22.3.1 General Shortcuts 1121
22.3.2 General item Panel Shortcuts 1121
22.3.3 Server Panel Tree Shortcuts 1122
22.3.4 Built-in View Shortcuts 1123

22.4 Scene Tree Shortcuts 1123
22.4.1 General Shortcuts 1123
22.4.2 Basic View Controls and Shortcuts 1125
22.4.3 Sort View Shortcuts 1126
22.5 Scene Editor Shortcuts 1126
22.5.1 General Shortcuts 1127
22.5.2 Camera Controls and Shortcuts 1128
22.5.3 Orthogonal View Controls and Shortcuts 1129
22.5.4 Animation Controls and Shortcuts 1129
22.5.5 Text Editor 1130
22.6 Stage Shortcuts 1130
22.6.1 General 1130
22.6.2 Stage Tree 1131
22.6.3 Stage Views 1131
22.7 Import Shortcuts 1134
22.7.1 File Shortcuts 1134
22.7.2 Archive Shortcuts 1134
22.8 On Air Shortcuts 1135
22.9 Polygon plug-in Editor Shortcuts 1136
22.10 Script Editor Shortcuts 1137

This User Guide gives the information required to understand the Viz Artist User interface, and how to do certain tasks.

Throughout this guide, the term Viz refers to the complete software suite installed, and as a general reference for the following modes:

- Viz Artist
- Viz Engine (see the Viz Engine Administrator Guide)
- Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide)
  The available features and modes of the software depend on the license on the connected license hardware dongle. The different modes and the hardware dongle are detailed in this User Guide.

See Also

- **Viz Engine plug-in SDK reference:** Located in the **plugin** subdirectory of the Viz installation directory, this reference documentation contains information for developers of customized plug-ins for Viz.
- **Viz Artist Script reference:** Located in the Viz installation directory, this reference documentation contains information for Viz Artist scene script developers.

1 Introduction

This User Guide gives the information required to understand the Viz Artist User interface, and how to do certain tasks.

Throughout this guide, the term Viz refers to the complete software suite installed, and as a general reference for the following modes:

- Viz Artist
- Viz Engine (see the Viz Engine Administrator Guide)
- Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide) The available features and modes of the software depend on the license on the connected license hardware dongle. The different modes and the hardware dongle are detailed in this User Guide.

This section contains information on the following topics:

- Related Documents
- Customer Feedback and Suggestions
- Customer Support Requests

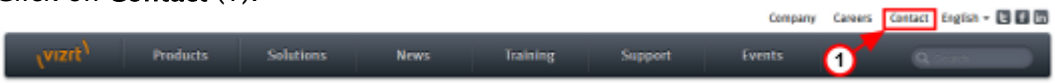
1.1 Related Documents

- Viz Engine Administrator Guide: Contains information on how to install and configure Viz Artist and Engine, and supported hardware.
- Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide): Contains detailed information on how to configure Viz Artist and Engine.
- **Viz Engine plug-in SDK reference:** Located in the **plugin** subdirectory of the Viz installation directory, this reference documentation contains information for developers of customized plug-ins for Viz.
- **Viz Artist Script reference:** Located in the Viz installation directory, this reference documentation contains information for Viz Artist scene script developers.

1.2 Customer Feedback And Suggestions

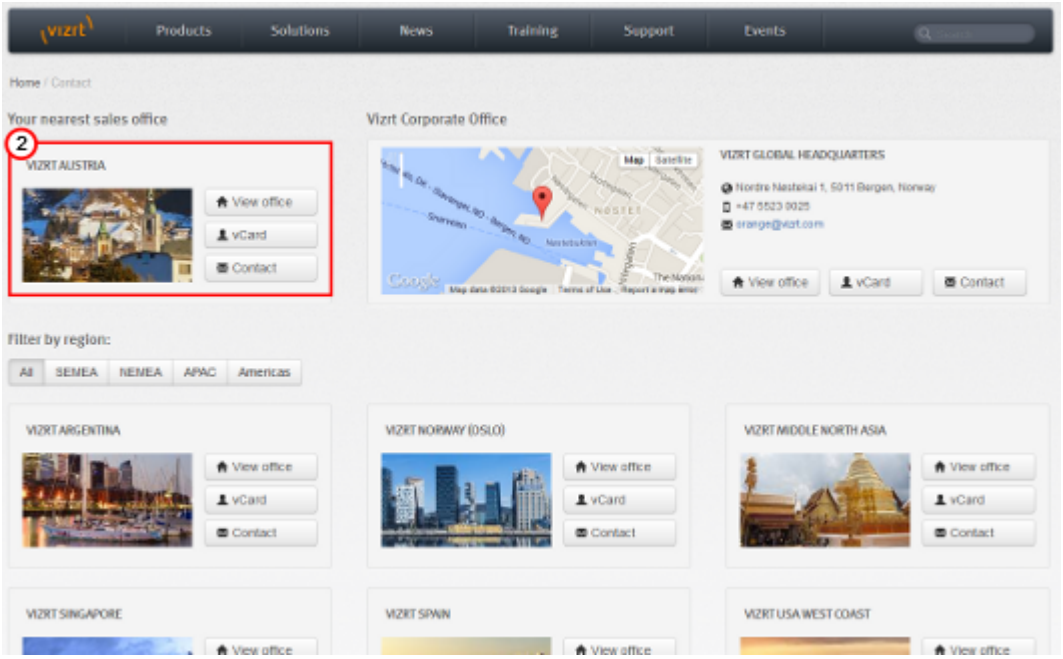
We encourage suggestions and feedback about our products and documentation. To give feedback and/or suggestions, find your local Vizrt customer support team at www.vizrt.com.

1. Go to www.vizrt.com.
2. Click on **Contact** (1).



3. The Vizrt office which is nearest to your location will be shown (2), or select from the list of Vizrt offices.

4. Click on 'Contact'.



5. Complete the required details in the window that opens.

> ⚠ **Note:** If the message is for Customer Support and there is a Support Contract in place, click on the **For support requests, please visit our support portal** link in the message window.

6. Click on **CONTACT ME**.
   A Vizrt representative will contact you as soon as possible.

See Also

- Customer Support Request

1.3 Customer Support Requests

Support Requests are supported by Vizrt for customers who have a valid Service Agreement in operation. Customers without a valid Service Agreement who would like to enter into one, should contact their regional sales representative (see Customer Feedback and Suggestions).

When submitting a Support Request relevant and correct information should be given to Vizrt Support, to make sure that Vizrt Support can give the quickest and best solution to your Support Request.

This section contains information on the following topics:

- Submitting a Support Request
- Content of a Support Request
- To Submit a Support Request
- Attach Log Files to a Customer Request

1.3.1 Submitting a Support Request

Before submitting a Support Request, make sure to read and check:

- The relevant User Guide or Guides.
- The release notes.
- That the system is configured correctly.
- That the specified hardware, tested and recommended versions are in use.
  Always refer to your Vizrt Service Level Agreement document.

When completing a Support Request, add as much information as possible.

1.3.2 Content of a Support Request

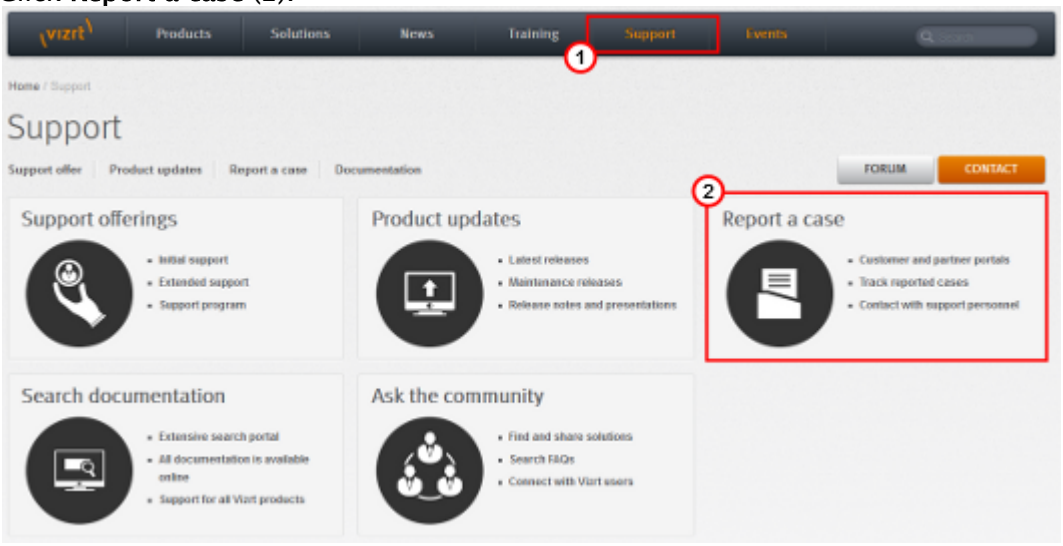
The report should contain information about these topics:

- **Problem Description:** Include a good description of the problem and how to reproduce it.
- **Screen Shots and Illustrations:** Use to simplify the message.
- **System Log Files:** Send the system log files (see Attach Log Files to a Customer Request).
- **Dump Files:** Send any dump files generated from a crash. These are located in the *<viz data folder>*.
- **Viz Configuration File:** Send the system configuration file, located in the *<viz data folder>*.
- **Software Configuration:** Add exact version of the software in use. If the software build number is available, include this.
- **Hardware Configuration:** Add exact versions of hardware in use.
- **System Setup:** Describe differences in the installation, if any, from the recommended setup.
- **System Network:** Add a description of how the network, bandwidth, routers, and switches are configured.
  Always refer to your Vizrt Service Level Agreement document.

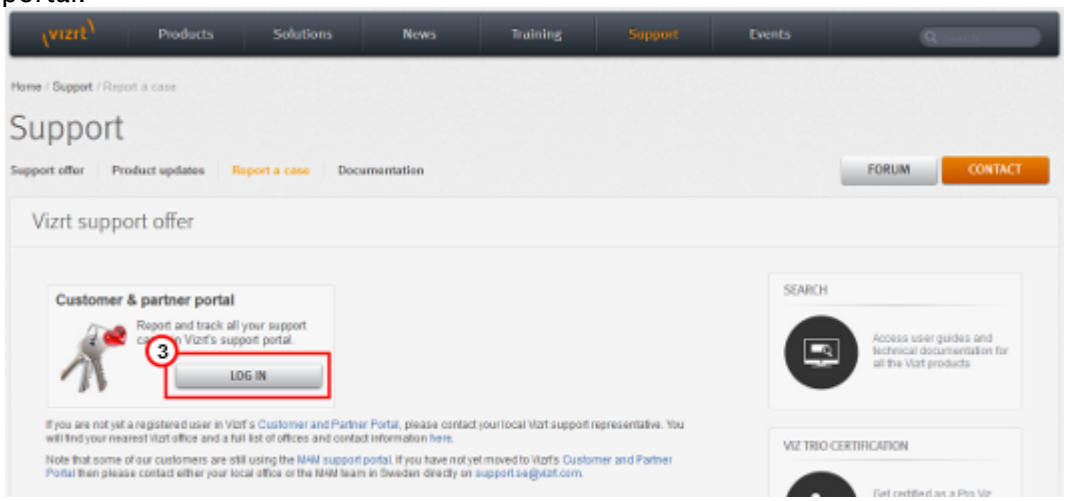
1.3.3 To Submit a Support Request

1. Go to www.vizrt.com.
2. Click **Support** (1).

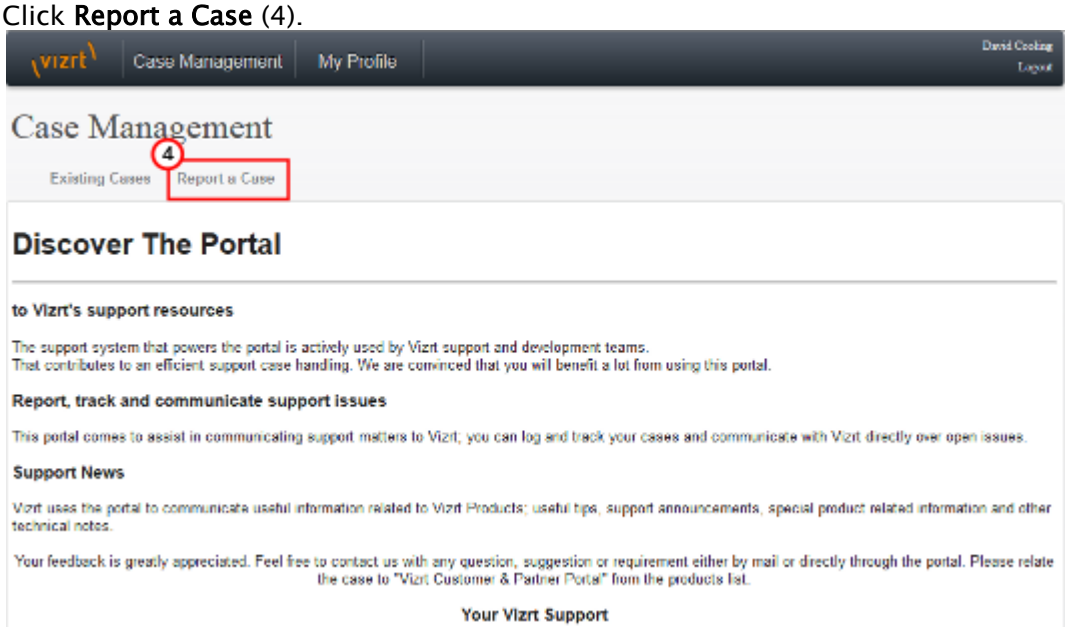
3. Click **Report a case** (2).



4. Click **LOG IN** (3), and provide the login credentials to access the Customer and Partner portal.



5. Click **Report a Case** (4).



6. In the online form, complete the required minimum information (indicated by a red asterisk):
   - **Contact:** Your name
   - **Account:** Your account
   - **Product:** The product the support request refers to.
   - **Case Type:** The type of support request required
7. Click **SAVE**.
8. In the saved Support Case that opens, complete the various text boxes and upload any required documents, files, etc. (see Content of a Support Request).

   ⚠ **Note:** The entered text or uploaded documents / files are automatically added.

   Log on to the Customer and Partner portal to track the status of open support tickets, or to add information and communicate about the case directly with the support team.

1.3.4 Attach Log Files to a Customer Request

Log files provide important information when dealing with an error situation or when diagnosing performance issues. Viz Artist/Engine writes various log files that can be valuable when solving issues. For more information about the log files available, see the **Viz Artist/Engine Log Files** page in the Configuring Viz section of the Viz Engine Administration Guide.

Consult your customer representative when deciding which logs files to attach. As a general rule: the more log files, the better. Always try to include log files for:

- The software/hardware configuration in use, as detailed as possible.
- The relevant Viz Artist/Engine log files.

2 Getting Started

This section describes the location of the installation files, how to start Viz Artist and logging on to a Graphic Hub, and how to close Viz Artist.

This section contains information on the following topics:

- Viz Artist/Engine Folders
- Viz Artist Startup and Close
- Viz Command Line Options

2.1 Viz Artist/Engine Folders

This section details the location of the Viz Artist/Engine default installation and data folders.

2.1.1 Installation Folders

The default installation folders for Viz Artist and Engine are:

Windows System	Viz Artist/Engine Platform	Installation Folder
Windows 32-bit	32-bit	C:\Program Files\vizrt\Viz3\
	64-bit	N/A
Windows 64-bit	32-bit	C:\Program Files (x86)\vizrt\Viz3\
	64-bit	C:\Program Files\vizrt\Viz3\ C:\Program Files (x86)\vizrt\Viz3Gui\

On 64-bit platforms, the user interface is installed in a 32-bit folder, as it is a 32-bit application. In this User Guide, any reference to the Viz Artist and Engine installation folder is replaced with the text *<viz install folder>*.

⚠ **Note:** Viz configuration files that already exist from a previous installation are automatically copied on install time if the installation folder did not change. If Viz Engine/Artist is installed in a different installation folder then Viz configuration files must be manually copied from the old installation folder to *<viz data folder>.*

2.1.2 Data Folders

Files which are created or modified by Viz Artist and Engine are located in `%ProgramData%\vizrt\viz3`. By default, this is `C:\ProgramData\vizrt\viz3`. This folder is referenced as *<viz data folder>* throughout this User Guide. The folder contents include:

- Viz configuration files,

- Script plug-ins,
- Crash dump files.
  Temporary files are located in `%TMP%\vizrt\viz3`, which usually resolves to `C:\Users\<user name>\AppData\Local\Temp\vizrt\Viz3`. This folder is referenced as *<viz temp folder>* throughout this User Guide.

Script plug-ins are located in the `<viz data folder>\ScriptPlugins` subdirectory.

> ⚠ **Note:** Script plug-in files that already exist from a previous installation are automatically copied on install time if the installation folder did not change. If Viz Engine/Artist is installed in a different installation folder, then Script plug-in files must be manually copied from the old installation folder to `<viz data folder>\ScriptPlugins`

See Also

- **User Account Control (UAC)** page in the **Software Configuration** section of the Viz Engine Administrator Guide.

2.2 Viz Artist Startup And Close

To start and run Viz Artist, a Graphic Hub must also be available. Viz Artist can also be started with certain options enabled or disabled, by passing selected Viz Command Line Options on start.

This section contains information on the following topics:

- To Start Viz Artist
- To Add a Viz Artist Startup Option
- To Log on a New User after Viz Artist has Started
- Log on History
- Viz Artist 64-bit or 32-bit Version
- Viz Configuration

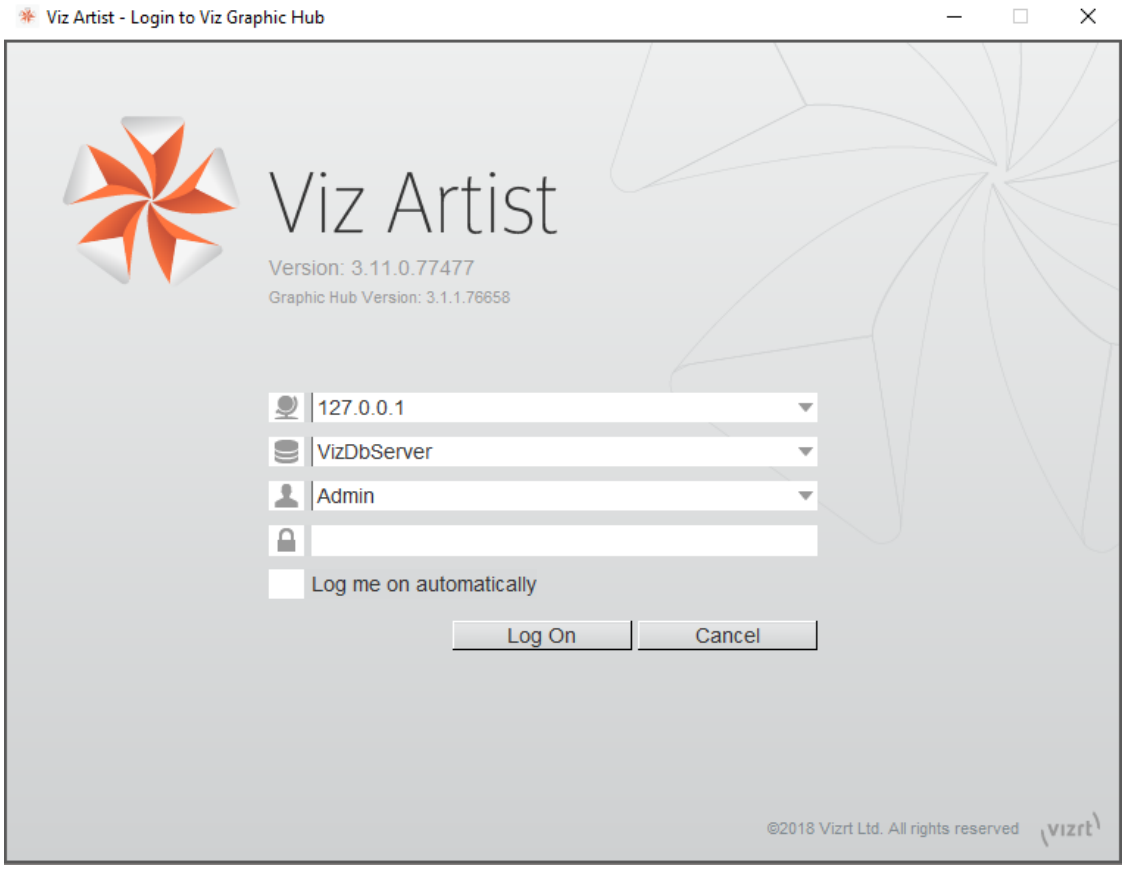
2.2.1 To Start Viz Artist

1. Start Viz Artist
2. In the Graphic Hub log on window, enter the required details:
   - : Select a Host from the drop-down list

     > ⚠ **Note:** In some circumstances, for instance if the server is located on a different subnet, the host will not be in the list. The host name must then be entered manually.

   - : Select a Graphic Hub from the drop-down list
   - : Select a User from the drop-down list

- : Enter the password for the selected user



- **Log me on automatically:** If automatic log on is enabled in Viz Configuration, checking this check box will activate it. When active, Viz Artist will log on to the selected Graphic Hub using the provided credentials on start.

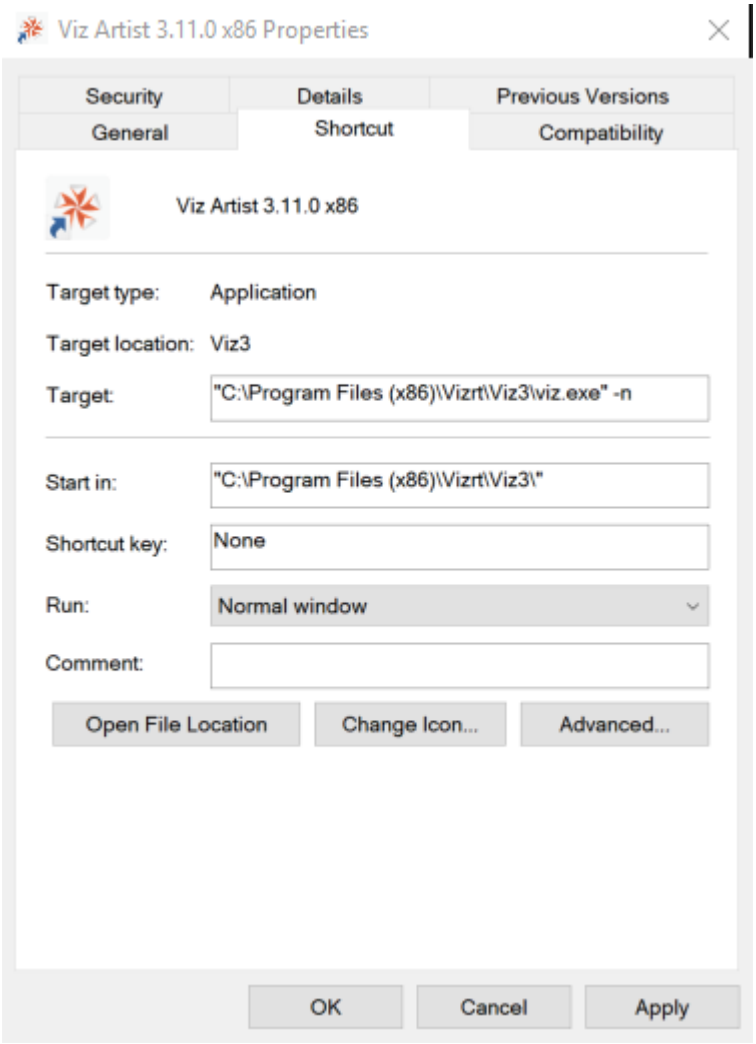
3. Click:
   - **Log on**: Log on to the selected Graphic Hub and start Viz Artist.
   - **Cancel:** Cancel log on and close the application.
     After logging on to a Graphic Hub, Viz Artist starts and opens the main screen (see Overview for an overview of the Viz Artist main screen).

The log on credentials are saved in the **Database** section of the Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide) where it can be modified as required.

2.2.2 To Add a Viz Artist Startup Option

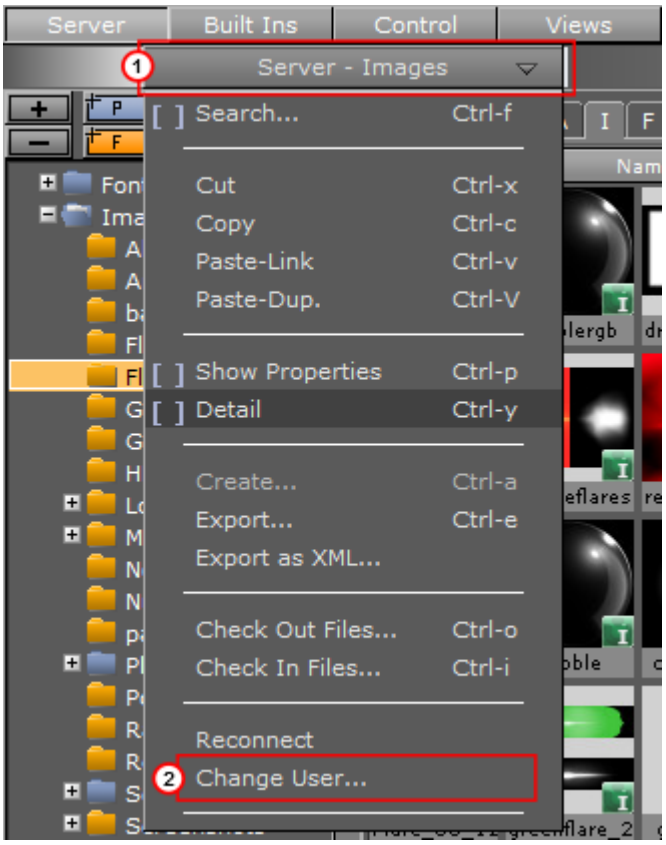
1. Open the Viz Artist Properties panel:
   - Right click the desktop icon and select **Properties** in the context menu , or
   - Right click the **Viz Artist** shortcut under the Windows **Start** menu and select **Properties** in the context menu.
2. In the Properties window, apply a command in the **Target** field (see Viz Command Line Options).

⚠ **Example:**“<viz install folder> viz.cmd” -n

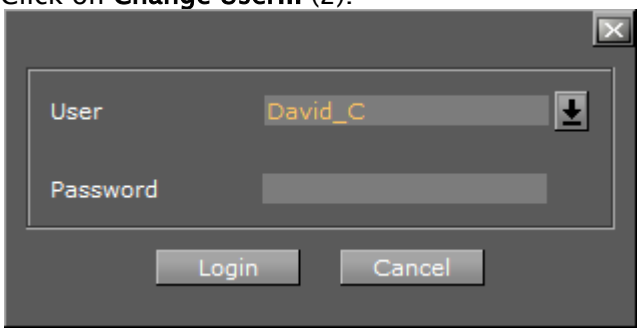


3. Click **OK**.
4. Start Viz Artist.

2.2.3 To Log on a New User after Viz Artist has Started



1. Click on **Server**.
2. Click on the Server Panel drop-down menu (1).
3. Click on **Change User...** (2).



4. In the **Change User** window, select a User from the drop down menu and enter the password in the **Password** field.
5. Click:
   - **Login** to change User, or
   - **Cancel** to cancel the procedure.

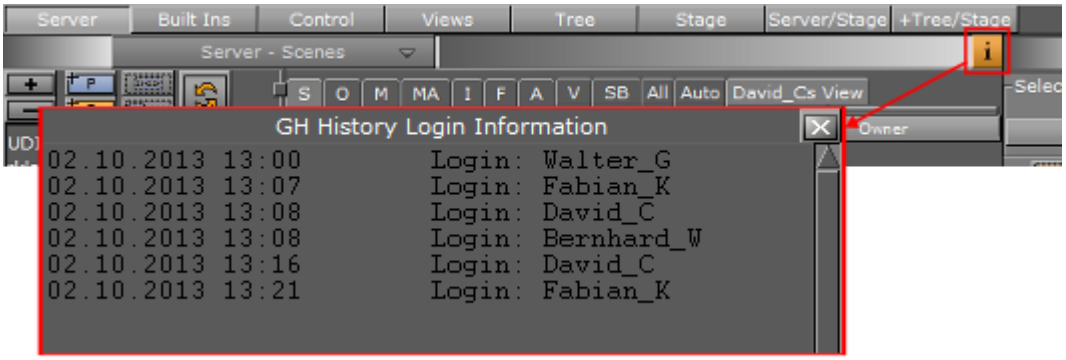
2.2.4 Log on History

Viz Artist creates a read only Graphic Hub log on history for each user who logs on to the current Viz Artist session.

When Viz Artist starts, the **startup** User is logged. If a new User logs on, this is recorded and the Graphic Hub History icon will change from gray to orange. This shows that another User, other than the **startup** User, has logged on to the current Viz Artist session.

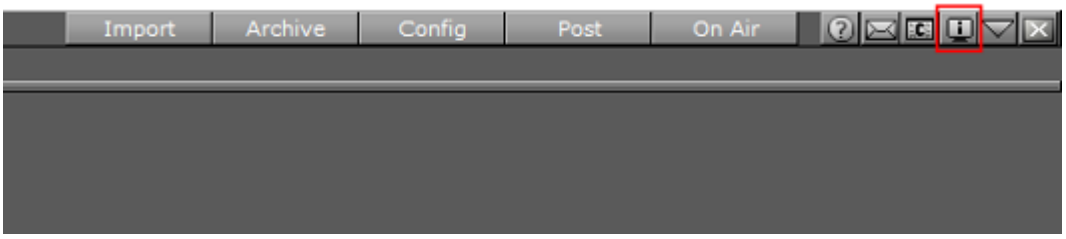
When the **startup** User logs on again, the Graphic Hub log on history icon color will turn back to gray.

View the log-on history by clicking the information icon. This opens the **GH History Login Information** window.

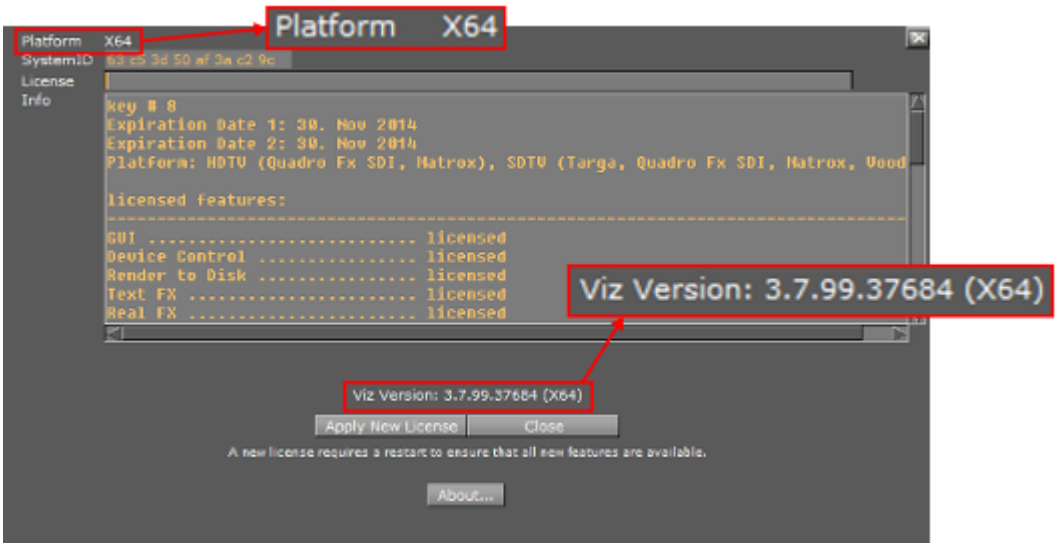


2.2.5 Viz Artist 64-bit or 32-bit Version

To check if the current installation is in 64-bit or 32-bit, click the **Show License Information** button:



In the upper left corner of the **License Information** window, **Platform** displays the current installation as either **Win32** or **X64**. The **Viz Version** field displays the version and build number, followed by the platform in parenthesis.



2.2.6 Viz Configuration

Viz is mainly configured by the Viz Configuration panel (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

All configuration settings are stored in a configuration file, found in the *<viz data folder>*. The name of the Viz configuration file includes the machine host name, to uniquely identify which machine Viz Artist/Engine is installed on.

⚠ **Example:** `Viz-<host name>-0-0.cfg`.

Any changes to the host name will affect the Viz Artist/Engine configuration. If the host name is changed, a new Viz configuration file is created with default configuration values. The old configuration file is not deleted, but left unused. If required, start Viz using the command line option `-g <configuration file>` to reassign the old Viz configuration file.

⚠ **IMPORTANT!** Folder, File and Path Naming Convention: Do not use the Hash tag (#) in any folder, file or path name. If used in Folder, File and Path Names, text after a Hash tag will be ignored.

See Also

- Viz Command Line Options
- **Configuring Viz** section of the Viz Engine Administrator Guide.

2.3 Viz Command Line Options

You can start Viz Artist and Viz Engine with various startup options. These command line parameters must be passed to the Viz executable file, `viz.exe`, and not to the batch file, `viz.cmd`. The batch file is only there for compatibility reasons. The table below details the most common Viz

Command Line Options. You can get a list of all available startup options, by running `viz.cmd -h` or `-?` from the command line.

Command	Description
-B <path>	Specify the path where Viz Engine stores its temporary data (see the **Viz Artist/ Engine Folders** page in the **Software Configuration** section of the Viz Engine Administrator Guide).
-c	Start in Viz Configuration mode (see also -u1, -u2, -u3).
-C	Start without a console.
-db user:pw@server/name-server:port	Specify which Graphic Hub to connect to on Viz Engine startup. `password` can be omitted from the argument, but `user`, `server`, `name-server` and `port` must always be provided. ⚠ **Example:** `viz.exe -db Guest:@VizDbServer/localhost:19396` In the example above, Viz Engine will connect to a Graphic Hub server called `VizDbServer` (⚠ case sensitive) on the name-server `localhost` at port `19396`, as the user `Guest` with no password.
-g <config file>	Start with a predefined Viz Config file. This allows a User to have more than one startup option.
-h, -?	Display the available commands.
-i	Enable pre-initialization of textures. Textures are generated on the graphics card immediately after loading an image.
-l	Specify a console title to distinguish Viz Engines in a Dual Channel setup (example: `-l <title>`).
-n -o <scene> -o <layer> <scene>	Start in Viz Engine mode (see also -u1, -u2, -u3). There is also an optional argument may give a scene that is loaded upon startup with the following syntax: **SCENE*scene_id** ... load scene_id into the main layer. **1 SCENE*scene_id** ... load scene_id into the main layer. **0 SCENE*scene_id** ... load scene_id into the back layer. **2 SCENE*scene_id** ... load scene_id into the front layer.
-P	Disable automatic mouse capture.

Command	Description
-t	Enables non-interactive mode for all occurrences except dongle issues and sound driver setup. The non interactive mode was introduced for use cases where there is no user and when there is no GUI one can interact with.
-T	Keep the Viz Engine Console always on top.
-u1, -u2, -u3 (up to a maximum of 24)	For systems with two or more graphic cards (i.e. Trio One Box / Dual Channel). Specify GPU1 (-u1), GPU2 (-u2), etc., to run Viz Artist on. See Systems with Two or More GPUs below.
-v [argument]	Enable verbose mode. This enables all possible information to be shown in the Viz Engine Console. The optional argument is a number composed by addition of: **1:** Produces verbose output to console. **2:** Adds a timestamp. **4:** Logs OpenGL. **16:** Logs 2D-texture messages. **32:** Logs Graphic Hub releated debug messages. **64:** Logs medium and high GL warnings.
-w	Start in Engine mode and show render windows (videowall mode).
-W	Disable restart on crash.
-x	Start without any GUI.
-X	Write extended dump file (full memory dump) in case of a program crash.
-y	Start in Artist mode. Viz Artist is started by Viz Engine after the Engine has started.
-Y <path>	Specify the path where Viz Engine stores its program data (see the **Viz Artist/ Engine Folders** page in the **Software Configuration** section of the Viz Engine Administrator Guide).

2.3.1 Systems with Two or More GPUs

It is possible to specify which GPU Viz Engine should run on, by providing the flags `-u1` and `-u2` for GPU1 and GPU2, respectively. A maximum of 24 GPUs are supported. This allows the system to run the Engine on one GPU while running Viz Config on another, for example:

```
<viz install folder>\viz.exe -c -u2
<viz install folder>\viz.exe -n -u1
```

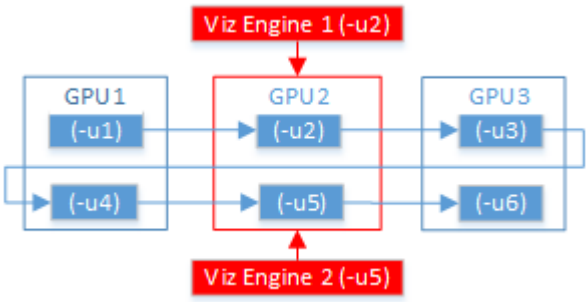
More than one Viz Engine can run on one GPU (multiple Viz Engines can not run on the GPU with the same port number), use the formula m%n to calculate which Viz Engine instance to use (Viz Engine defined by -u<m>). Note, however, that Viz Artist can only run on Viz Engine instance 1 (-u1). Only the control GPU should have a display connected and will be used for Artist/Preview.

Formula: m%n, where:

- m = -u1/-u2/etc.
- n = number of GPUs

For example, with two Viz Engines:

- System with two GPUs: Engines run on GPU2: Engine 1 (-u2) / Engine 2 (-u4)
- System with three GPUs: Engines run on GPU2: Engine 1 (-u2) / Engine 2 (-u5)



When either Trio One Box or Dual Channel Viz Artist versions are installed, GPUs are selected by default.

3 Artist Interface Overview

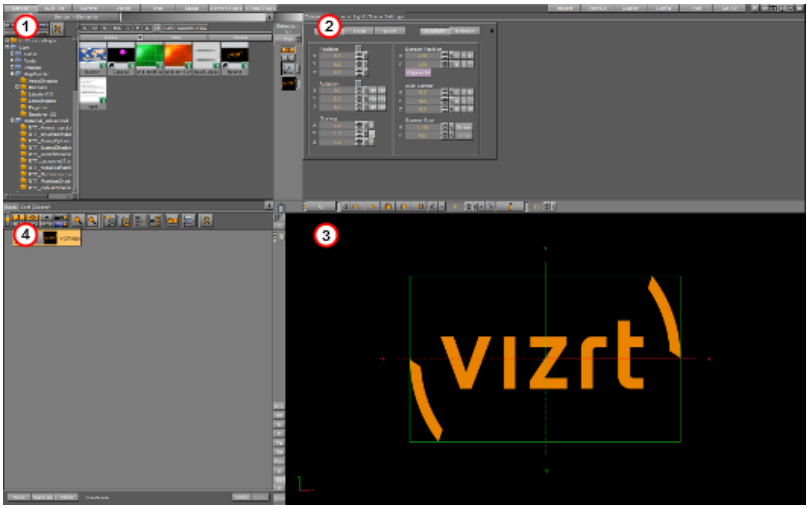
This section gives an overview of the Viz Artist user interface and its main panels.

This section contains information on the following topics:

- Main Window
- Dual Monitor Configuration
- Main Menu

3.1 Main Window

When you start Viz Artist, the main window opens after you have logged on.



At the top of the main window is the Main menu bar. Below the Main menu bar is the display area. The content of the display area changes according to selections made in the Main menu.

You can also run Viz Artist in a Dual Monitor Configuration, and resize the panel areas to accommodate your workflow.

At system start, the area is divided into four panels:

Fig Item	Panel	Description
1	Server Panel	From the **Server** panel, it is possible to control items such as Materials, Images, and Fontstyles, which are located in the Graphic Hub.
2	Properties	Properties for Containers, Cameras, Lights and Video Clip Playback Considerations are controlled from the **Property** panel, where size, position and other settings can be modified as required.

Fig Item	Panel	Description
3	Scene Editor	A visualization of the Scene is shown in the Scene Editor panel.
4	Scene Tree	Items in the Graphic Hub database can be added to the Scene Tree, which is the logical visualization of a Scene. The Scene Tree is built up of Containers, which can hold items such as Materials or plug-ins.

3.2 Dual Monitor Configuration

Viz Artist can be configured to work on two monitors. More information can be found in the **User Interface** page in the **Configuring Viz** section of the Viz Engine Administrator Guide. The Viz Artist user interface screen options are:

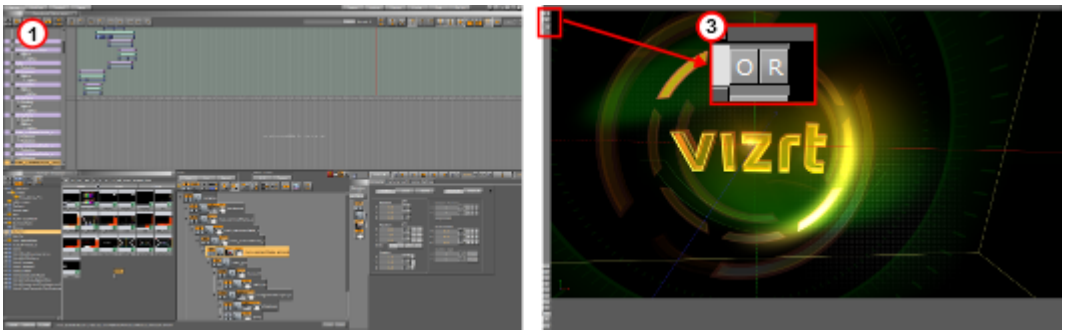
- Normal
- Mosaic Horizontal
- Mosaic Vertical
- Dual Screen

3.2.1 Normal

The Server, Properties, Scene Tree and Scene Editor panels are all displayed on the left monitor. This is the default view.

3.2.2 Mosaic Horizontal

The Server, Properties and Scene Tree panels are displayed on the left monitor, and the Scene Editor is displayed on the right monitor.



The monitor which displays the Scene Editor panel can be set to display either screen size or output aspect, by clicking the **O** or **R** buttons (3).

3.2.3 Mosaic Vertical

The Server, Properties and Scene Tree panels are displayed on the top monitor, and the Scene Editor panel is displayed on the bottom monitor.

The monitor which displays the Scene Editor panel can be set to display either screen size or output aspect, by clicking the **O** or **R** buttons (3).

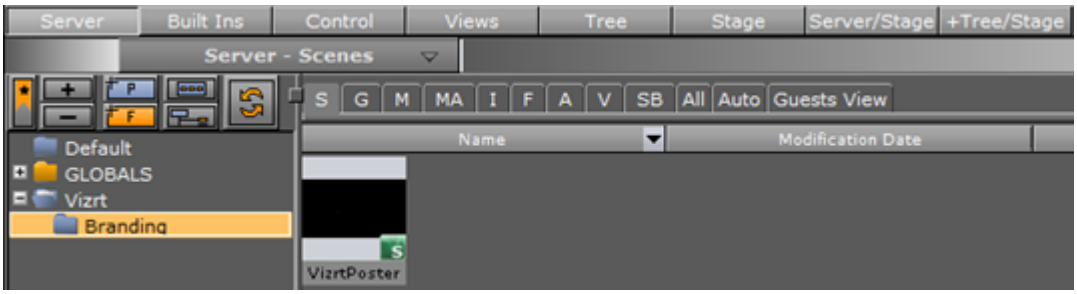
3.2.4 Dual Screen

The Scene Editor panel opens in a new window, and the items cannot be dragged to the Scene Editor panel.

3.3 Main Menu

Use the buttons in the Main menu to select what is displayed in the display area.

3.4 Main Menu Left



The menu's left hand side contains the following:

- Server Button
- Built Ins Button
- Control Button
- Views Button
- Tree Button
- Stage Button
- Server/Stage Button
- +Tree/Stage Button

3.4.1 Server Button

Use the Server Panel to control database items, such as scenes, materials and images. The items are common for all users connected to the database, and can be added to the scene. Server is the default view when Viz Artist starts.

Shows the following panels:

- Server (top left)
- Scene Tree (bottom left)
- Property (top right)
- Scene Editor (bottom right).

3.4.2 Built Ins Button

From the plug-ins area, it is possible to control the built-in plug-ins, such as Geometries, Container, Shader, Texture and Scene plug-ins. The features vary according to the licensed and installed plug-in packages.

Shows the following panels:

- Plug-ins (top left)
- Scene Tree (bottom left)
- Property (top right)
- Scene Editor (bottom right).

3.4.3 Control Button

Use the Control area to view, modify, or create parameters exposed in a Scene for modification in an External Control Application. The **Control Objects** panel shows the exposed parameters when a Control plug-in is used. In the **Control Channels** panel, parameters can be dragged from the Properties panel, and saved as exposed parameters.

Shows the following panels:

- Controls (top left)
- Scene Tree (bottom left)
- Property (top right)
- Scene Editor (bottom right)

3.4.4 Views Button

The Views area shows three configurable camera views, in addition to the one in the Scene Editor area. The additional camera views make it possible to monitor the scene from up to four different angles simultaneously.

Shows the following panels:

- Views (left)
- Property (top right)
- Scene Editor (bottom right)

3.4.5 Tree Button

The Tree view is similar to the Server Panel, excluding the database view. This option is practical to use when the scene tree is complex and holds many containers. If the scene tree is still not big enough in the regular Tree view, it is possible to click the Zoom Out button above the scene tree. This will shrink the size of all the containers, allowing for even more containers to show.

Shows the following panels:

- Scene Tree (left)
- Property (top right)
- Scene Editor (bottom right)

3.4.6 Stage Button

The Stage is used to control the animations in the scene. The Stage shows the time-line from animation start (left side) to stop (right side).

Shows the following panels:

- Stage (top)
- Scene Tree (bottom left)
- Scene Editor (bottom right)

3.4.7 Server/Stage Button

The Server/Stage view is similar to the regular Stage view, but here the scene tree has been replaced with the database content.

Shows the following panels:

- Stage (top)
- Server (bottom left)
- Scene Editor (bottom right)

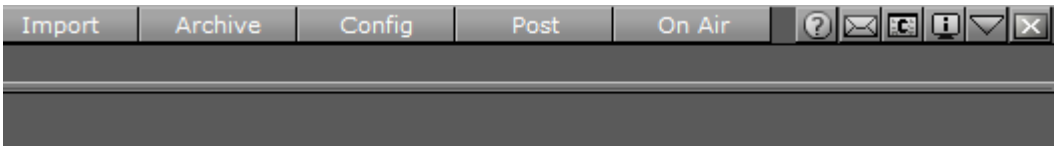
3.4.8 +Tree/Stage Button

The +Tree/Stage view is only available when using a screen resolution of 1600x1200 or higher, and shows both the database content and the scene tree.

Shows the following panels:

- Stage (top)
- Server (bottom left)
- Scene Tree (bottom center)
- Scene Editor (bottom right)

3.5 Main Menu Right



The menu's right hand side contains the following:

- Import
- Archive
- Deploy
- Config
- Post
- On Air
- Button Bar

3.5.1 Import

Click to open the Import panels. Use these panels to transfer archives and various files into the Graphic Hub database. For more information, see the Import and Archive chapter.

3.5.2 Archive

Click to open the Archive panels. Use these panels to create an archive of multiple database items into a single file, which is saved to the hard drive.

3.5.3 Deploy

If Deploy is enabled in the **Database** section of Viz Configuration, click to open the Deploy panels. Use these panels to copy items, with all their necessary data, from one database server to another (see Deploy items).

3.5.4 Config

Click to open the Viz Configuration panel. Use this panel to change system options and functionality (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

The content of the Viz Configuration panel changes with different platform types. An installation on a machine that contains a video board will run the software in Video mode, a machine with no video board will run the software in Standard PC mode. To find the platform type select the License Information button.

Viz Configuration can also be opened as a separate application from the desktop or Start menu.

> ⚠ **Note:** After any modifications in the Viz Configuration settings, the system must be restarted for the changes to take effect.

3.5.5 Post

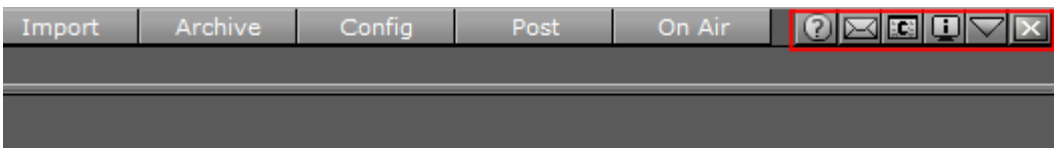
Opens the Post Rendering panel. From this panel it is possible to export image chains or Scene clips (see Post Render Scenes).

3.5.6 On Air

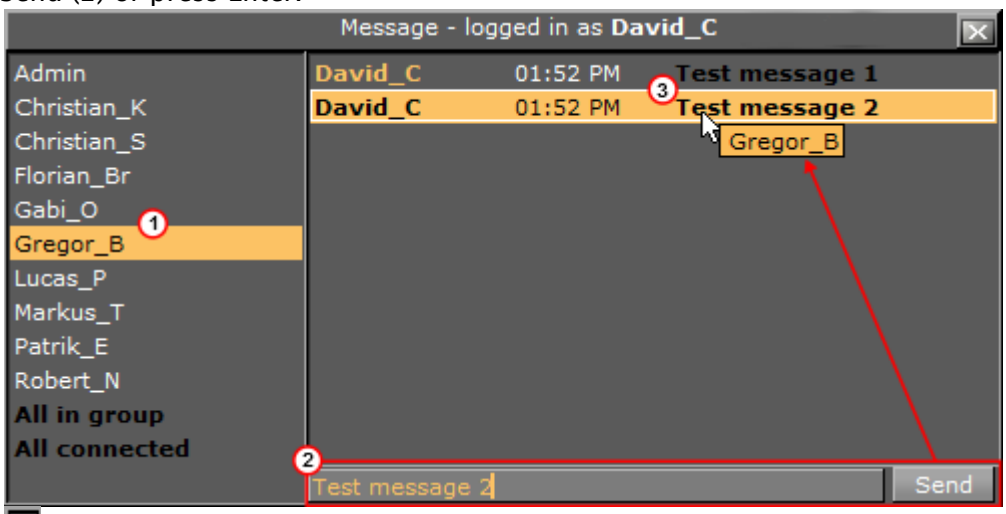
Opens the On Air panel. This panel switches the application mode from a 3D modeling tool to a plain rendering engine. The application waits for control commands through its communication port, or it can be controlled manually from the control buttons when Viz is set in On Air Mode.

By default, the port number is set to 6100. The port number and other communication settings can be modified in the **Communication** section of Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

3.5.7 Button Bar



- **Documentation:** Opens the Viz Artist User Guide.
- **Message Box:** Opens the Message window. This window shows server warnings, and allows for chat messages to be sent and received to and from other users connected to the same database (3). To send a message, click on a recipient (1), type a message and click on Send (2) or press Enter.



- Show Commands: Opens the Viz Engine Console window. This window shows the commands that are used for the communication between the user interface and the rendering Viz Engine. The console window is mainly used for debugging purposes.

> **Tip:** The Viz Engine Console window can be opened by pressing the keyboard shortcut `Alt+C`.

- **License Information:** Opens the License Information and About window. This window shows the status of the current license, with details such as license expiration date, platform type (Standard PC or Video), licensed plug-in packages, Viz version number, and so on. This is also where the license can be renewed. For details on how to renew the license, see the Viz Engine Administrator Guide.
  Click the **About...** button to view the About statement and see all legal notices.
- **Minimize:** Minimizes the Viz Artist window.
- **Close:** Opens the Quit or Restart menu.

4 Manage Items And Built Ins

To start Viz Artist, the user must log on to a Graphic Hub. The Graphic Hub is the database solution where all Viz Artist items are stored. Items can be either:

- Server Panel
- Server Tree
- Item Panel
- What are items
- Working with Items
- Container and Scene Properties
- Assign Keywords to Items
- Image Editor
- Fontstyle Editor
- Material Editor
- Item Search
- Free Text Search
- Background Loading
- Built Ins
- Substance Editor

The Graphic Hub database can be installed on the local machine for a single-user environment. It can also be installed on a centralized server for a multi-user environment, which provides shared access to multiple users simultaneously.

Individual items are stored in the Graphic Hub database as file objects, and are managed in terms of properties and *Universally Unique Identifiers*, commonly referred to as *UUIDs*.

Clients can work with and organize items through Viz Artist, in a logical folder structure provided by the Graphic Hub. Although each item can be listed and linked in more than one folder, it will be in the Graphic Hub only once.

4.1 Server Panel

The Server panel shows an overview of all available items in the Graphic Hub database.

The Server panel is divided into two sections:

- Server Tree (1): Shows the logical tree of folders in the Graphic Hub database.

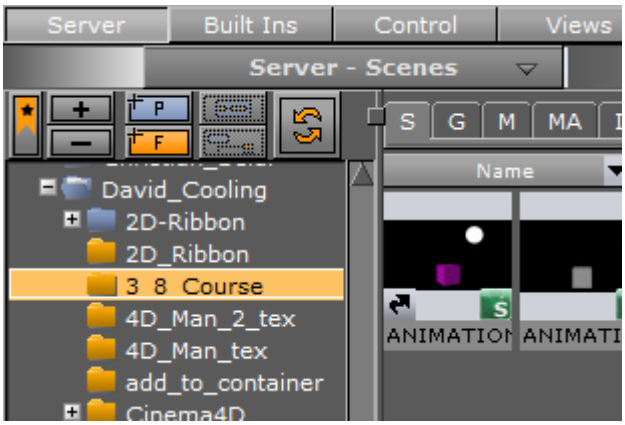
- Item Panel (2): Shows the items contained in the folder selected in the Server Tree.

  > **Tip:** To resize the two areas, drag the vertical separator.

  The Server panel is enabled when Server, Server/Stage or +Tree/Stage is selected from the Main Menu.

4.2 Server Tree

The Server Panel tree shows a structured storage of items.



Right-click on any folder in the Server Tree to open the Server Tree Context Menu.

This section contains the following topics:

- Folders
  - Folder Position in the Server Tree
  - Highlighted Folders
- Server Tree Menu Bar
- Server Tree Context Menu
- Bookmarked Folders

4.2.1 Folders

The Server Tree can contain two types of folders:

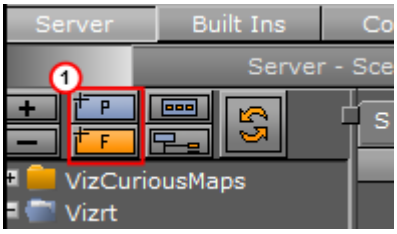
- Regular folders
- Project folders

The difference between a regular folder and a project folder is that by defining a working project, the tree can be collapsed so that only the currently selected project and its sub-folders and sub-projects' folders are shown.

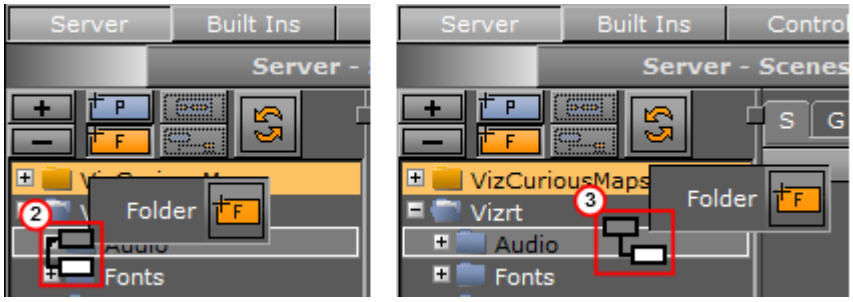
Both types of folders can be dragged around after being added to the Server Tree. Only Project folders can contain Sub-Project folders; however, sub-folders can be added to both regular folders and project folders.

Folder Position in the Server Tree

Drag and drop folders using the **Add New <Folder/Project>** button (1) to add new folders. The folder will be created at the location it is dropped in the Server Tree. Folders can be moved in the same way, by dragging an existing folder and dropping it in the desired location.



- Drop the **Add New <Folder/Project>** button on the left side of an existing folder to create a new folder at the same level as the selected folder (2).
- Drop the **Add New <Folder/Project>** button on the right side of an existing folder to create a sub-folder of the selected folder (3).



Highlighted Folders

The Folder Tree can be configured to highlight only the folders in the tree that contain a selected item type (see the **User Interface** page in the **Configuring Viz** section of the Viz Engine Administrator Guide). When Folder Type Highlighting is enabled, folders and sub-folders which do not contain any of the selected item type are dimmed.

4.2.2 Server Tree Menu Bar

The Server Tree menu bar consists of various buttons to create folders and modify the appearance of the Server Tree. It is located at the top left side of the Server Panel.

The Server Tree bar contains the following buttons:

-  **Bookmarks**: Shows only bookmarked folders (see Bookmarked Folders).
-  **Expand/Collapse**: Expands or collapses the currently selected branch and all its sub-branches.
-  **Add New Project/Folder**: Adds a new project () or folder () to the Server Tree.
-  **Restrict to Current Project Folder**: Restricts the Server Tree view to the selected project folder and sub-folders. This option is only available if a project is selected in the Server Tree.
-  **Show All Project Folders**: If the scene tree is restricted to one project folder, this expands the tree to show all folders.
-  **Refresh**: Refreshes the Server Tree and adds updates.

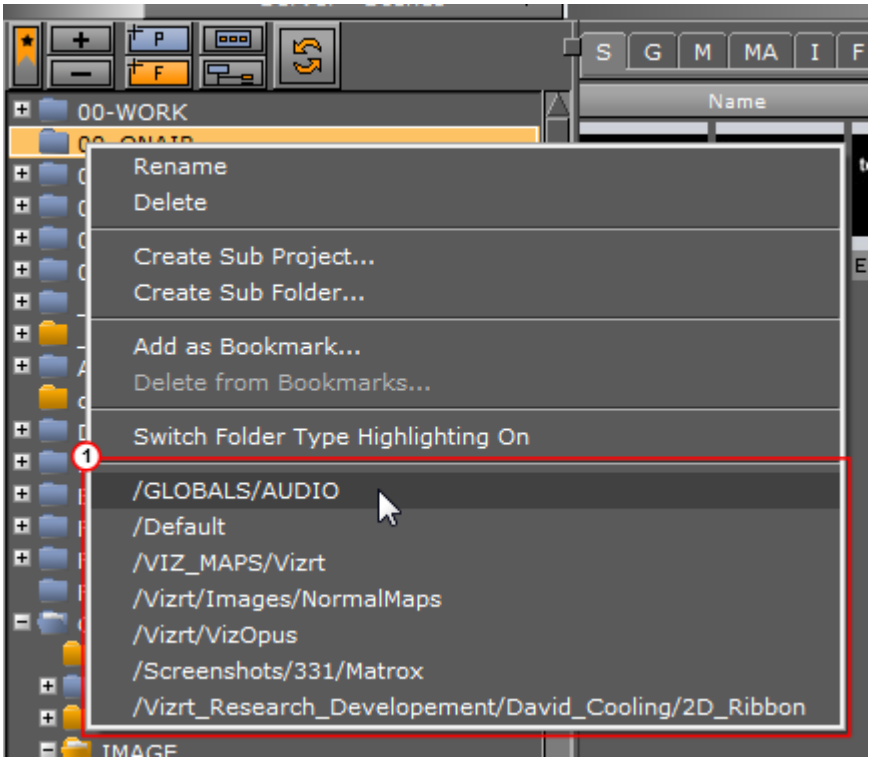
4.2.3 Server Tree Context Menu

Right-click a folder in the Server Tree to open the Server Tree context menu.

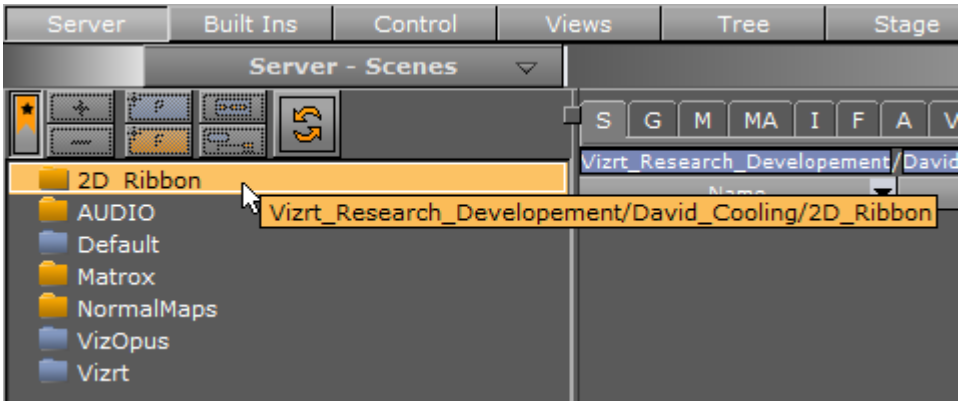
- **Rename**: Changes the name of the selected folder.
- **Delete**: Deletes the selected folder.
- **Create Sub Project...**: Creates a sub-project folder under the currently selected project folder. Only Project folders can contain Sub-Project folders.
- **Create Sub Folder...**: Creates a sub-folder under the currently selected folder. Sub-folders can be added to both regular folders and project folders.
- **Add as Bookmark.../Delete from Bookmarks...**: Adds or deletes the selected folder as a bookmark (see Bookmarked Folders).
- **Switch Folder Type Highlighting <On/Off>:** Enables or disables folder highlighting. When Folder Type Highlighting is enabled, folders and sub-folders which does not contain any of the selected item type are dimmed.
- **Bookmark(s):** If any bookmarks have been created, the bottom part of the menu contains a list of shortcuts to all currently bookmarked folders.

4.2.4 Bookmarked Folders

Folders can be bookmarked to speed up navigation to the most used folders. Bookmarked folders are added to the Server Tree context menu. Click a bookmark in the context menu list (1) to go to that folder in the Server Tree.



All Bookmarked folders can also be viewed together in a Bookmark View (see To View Bookmarked Folders):

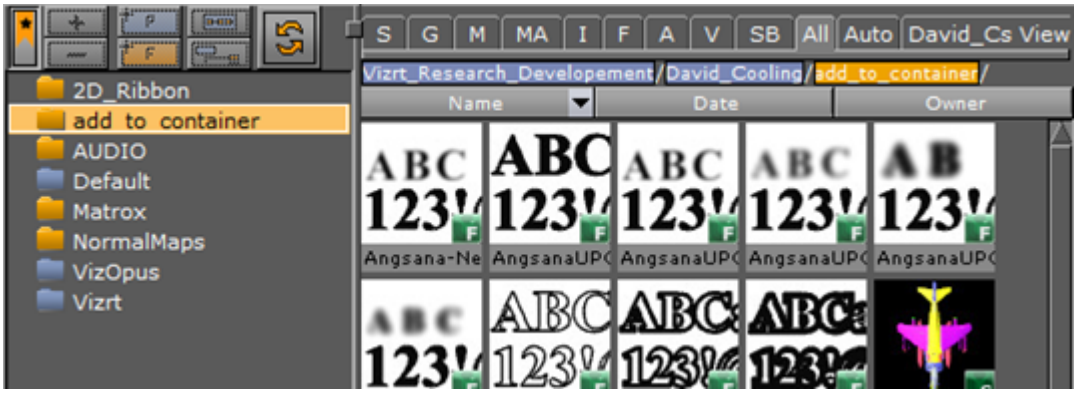


This section contains the following topics and procedures:

- Bookmark View
- To View Bookmarked Folders

Bookmark View

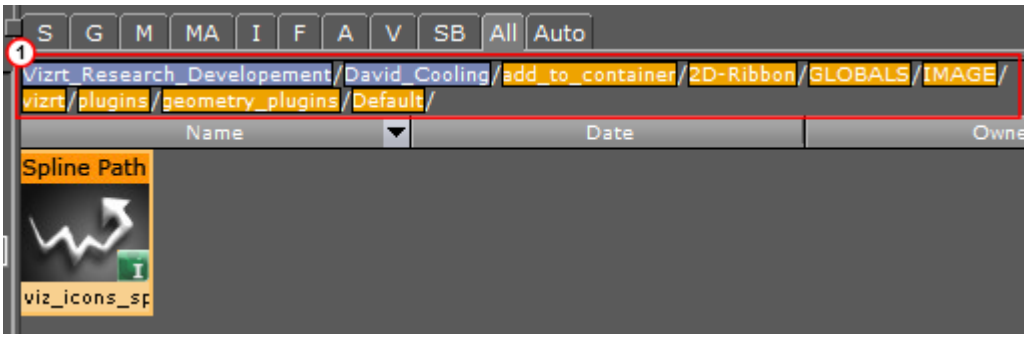
Bookmark View displays only the bookmarked folders.



To view the contents of a bookmarked folder, click on the folder. The path (1) and the contents of the folder are shown in the Item Panel. Right-clicking a folder or file in the item panel opens the Item Context Menu. In the Item Panel, double-click a folder to open and view any sub-folders and its content:

- If a sub-folder is bookmarked, it will be selected in the Bookmark View.
- If a sub-folder is not bookmarked, only the parent folder will be selected in the Bookmark View.
  Use the Content Switcher to view the required items.

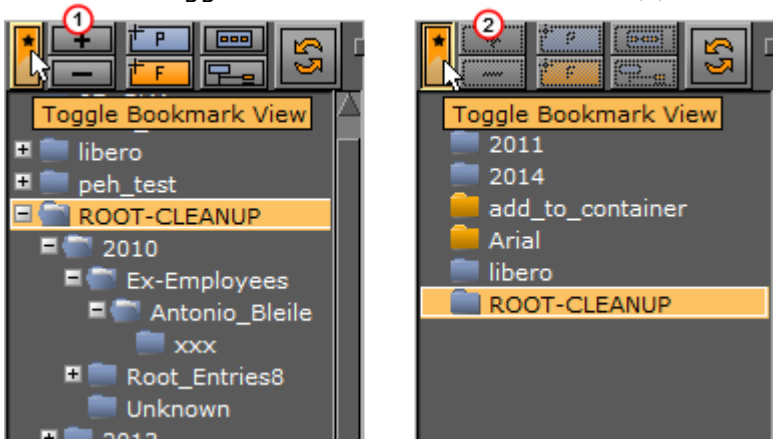
Clicking a link in the folder path (1) opens that folder. If the folder is bookmarked, it will be selected in the bookmark view. If not, the view will switch to Server View, and the folder will be selected.



- **Blue**: Project folders
- **Yellow**: Normal Folders

To View Bookmarked Folders

- Click  to toggle between the Server Tree view (1) and the Bookmark view (2).



If no bookmarks have been created, you will get a notification.

4.3 Item Panel

The Item Panel shows the items that are available in the folder selected in the Server Tree.

This section contains information on the following topics:

- Item List
  - Item Views
  - Item Properties
  - To Show the Properties Panel
- Item Context Menu
- Content Switcher

- Server Menu
- Properties Panel

4.3.1 Item List

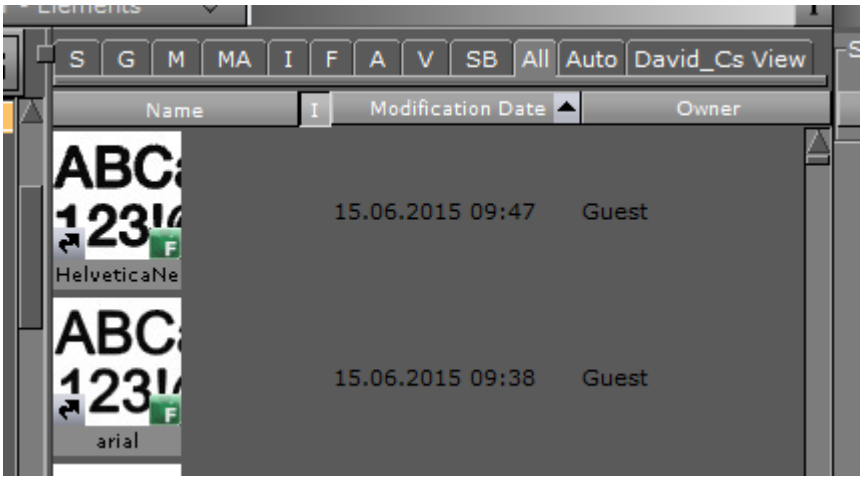
The item list displays all items in the selected folder.

This section contains information on the following topics and procedures:

Item Views

By default, items are displayed with a thumbnail and their name, alphabetically sorted. Right-click an item in the Item List to open the Item Context Menu. The tool-tip for thumbnails contains the complete item name.

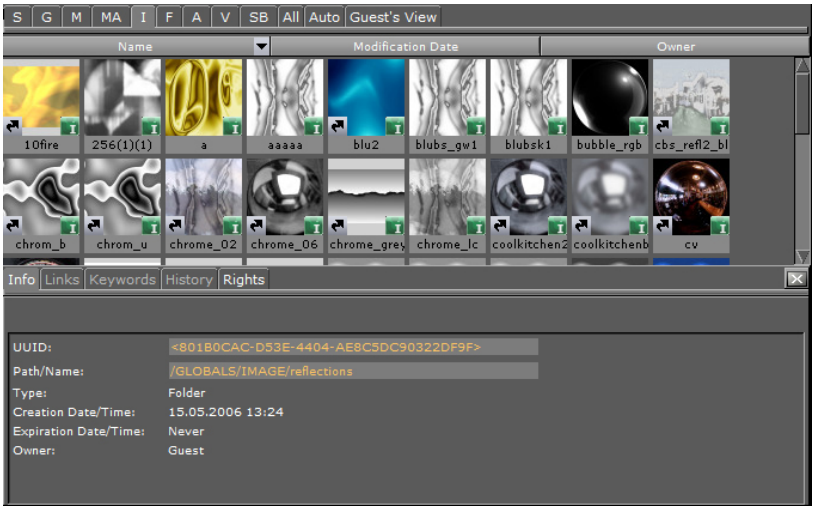
Items in the list can also be viewed with information about their owner and creation date. Click Detail in the Server Menu or press the keyboard shortcut Ctrl+Y while hovering the pointer over the server panel to enter **Detail View**.



When in Detail view, click the **I** in the items list header to switch between thumbnail and list view. When no icon is selected, the item name is displayed.

Item Properties

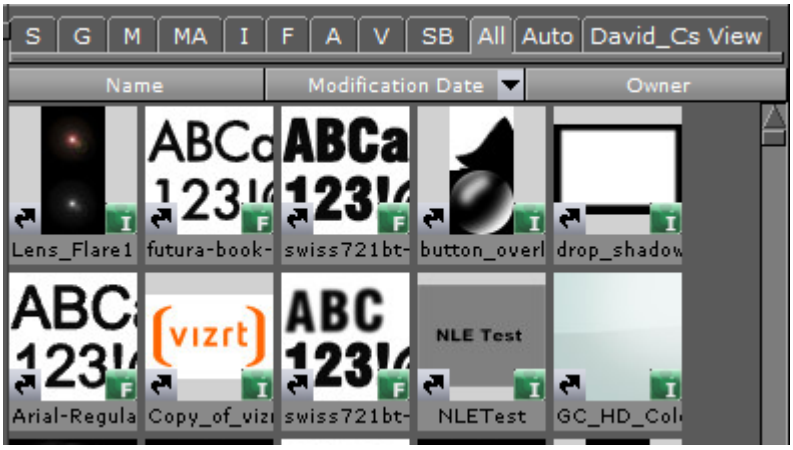
The Properties Panel displays detailed information about file, links, keywords, history and rights for the selected item.



To Show the Properties Panel

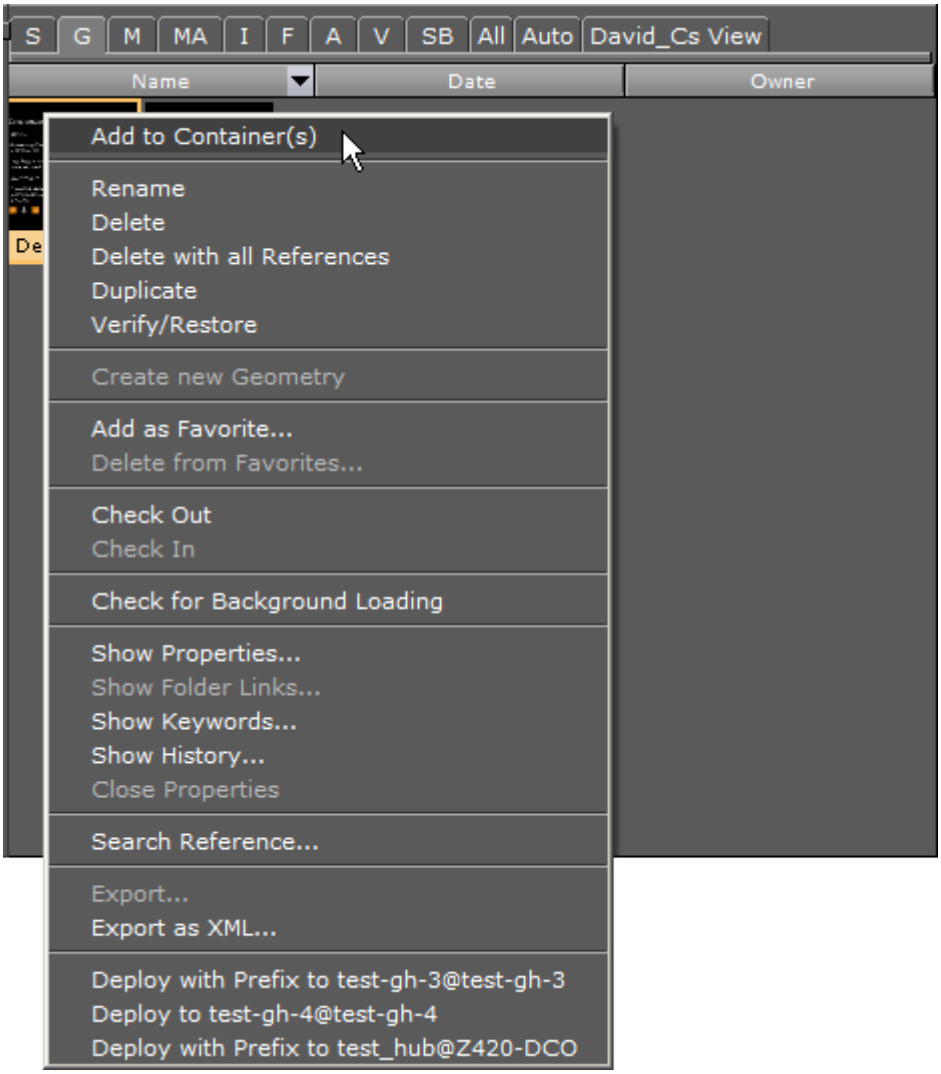
- Select **Show Properties** from the Server Menu, or
- Hold the pointer over the Server Panel and press the keyboard shortcut <Ctrl+P>, or
- Right-click an item in the Item List. In the menu that opens, select **Show Properties**...

The item list can be sorted by **Name**, **Date**, or **Owner**, by clicking the corresponding column. An arrow will indicate if the selected sorting is ascending or descending. By default, items are sorted alphabetically by name.



4.3.2 Item Context Menu

Right-click an item in the Item List to open its context menu, which contains the following options:

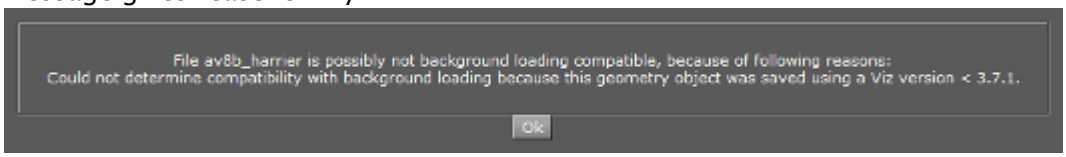


- **Add to Container(s)**: Add the selected item to the selected Container(s).

> ⚠ **Note:** Not available for Scene or Audio items.

- **Open <Scene name>:** Opens the selected Scene.
- **Rename**: Changes the name of the selected item.
- **Delete**: Deletes the selected item.
- **Delete with all references**: Deletes the selected item and all its referenced items. Referenced items are not deleted if they are referenced by other items.
- **Duplicate**: Creates an independent copy of the selected item. The copy will be placed in the same folder as the original item.
- **Verify/Restore:** Locates, verifies and attempts to correct a corrupted item.
- **Switch to Small/Large Icons:** Toggles between small or large item icons in the Item List.
- **Create New item**: Creates a new Scene or Material.
- **Add as Favorite...**: Adds the selected item to the Favorites Bar in the Scene Tree area. Scenes and Audio items cannot be added to favorites.

- **Delete from Favorites**: Deletes the selected item from the Favorites Bar.
- **Check Out**: Checks out the selected item, so that only the user that checks out the file can modify it. Other users can see the name of the user that has checked out the file from the Properties Panel.
- **Check In**: Checks in a previously checked out item, so that all users can modify it.
- **Check for Background Loading**: Checks if the selected item is suitable for Background Loading. A message box opens and shows if the item is suitable or not. If not suitable, the message gives reasons why.



- **Show Properties...:** Shows the Properties Panel at the bottom of the item panel, with the **File** Info tab selected.
- **Show Folder Links...:** Shows the Properties Panel at the bottom of the item panel, with the **Links** tab selected.

> ⚠ **Note:** The folder links option is only available for items that are linked.

- **Show Keywords...:** Shows the Properties Panel at the bottom of the item panel, with the **Keywords** tab selected.
- **Show History...:** Shows the Properties Panel at the bottom of the item panel, with the **History** tab selected.
- **Show Rights...:** Shows file attributes at the bottom of the item panel, with the **Rights** tab selected.
- **Close Properties**: Closes the Properties Panel.

> ⚠ **Note:** Close Properties will show Close Folder Links, Close Keywords, Close History or Close Rights if those are selected and shown in the Properties Panel.

- **Search Reference...:** Opens the **Search** window and searches the database for other items with references to the selected item.

> ⚠ **Note:** It is not possible to perform a reference search for materials, as materials are not referenced. Material information is held inside the Scene.

- **Export...**: Exports selected Image and Audio clip items. Applies only to Audio and Image items
- **Export XML...**: Exports an XML representation of a scene and other objects into an XML file. XML files can be used to compare two versions of a Scene using a file comparison utility.
- **Deploy...**: Deploys an item with all its necessary data from one server to another. For example, when a scene is deployed, all items referenced by the scene, such as images, geometries, etc., will also be deployed.

> ⚠ **Note:** Available when Deploy is enabled and a valid Deploy server is entered in **Deploy Properties** (see the **Database** page in the Viz Configuration section of the Viz Engine Administrator Guide).

4.3.3 Content Switcher

Folders can contain various types of items.

To view only a selected type of item, contained in a folder, click an item type tab from the Content Switcher (1). If the **All** tab is selected all items in the selected folder, regardless of type, will be shown.

- **S**: Scenes
- **G**: Geometries
- **M**: Materials
- **MA:** Materials Advanced
- **I**: Images
- **F**: Fontstyles
- **A**: Audio Clips
- **V**: Video Clips
- **SB**: Substances
- **All**: All item types
- **Auto**: Shows the item type with the most instances in the selected folder. For example, if there are five materials in a folder, but only one instance of each of the other item types, the materials will be shown in the Item List.
- **<Username> View**: Personalized view to enter and save free text searches for later use (see Free Text Search).

This section contains information on the following topics:

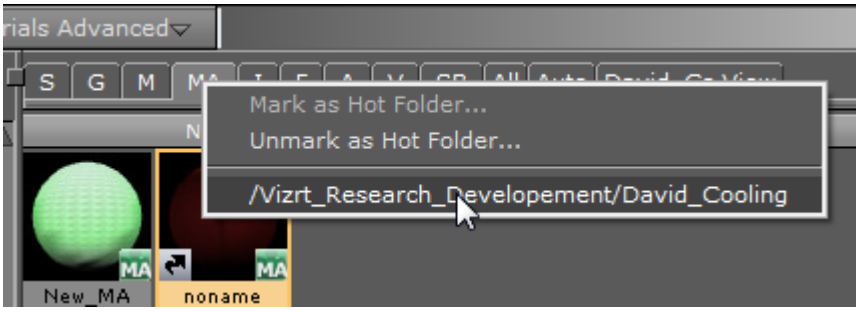
- Content Switcher Menu
- Item Type History
- To show the item type history list
- To Show the item type Last Used Folder

Content Switcher Menu

Right-click one of the tabs in the Content Switcher to open the Content Switcher menu. *Hot folders* for the selected item type is listed at the bottom of the menu. Hot folders are item specific favorite folders frequently used by the logged on user. Selecting a hot folder in the list will open

the corresponding folder and switch to the selected item type. Each item type has a separate list of Hot Folders.

The Content switcher menu contains the following options:



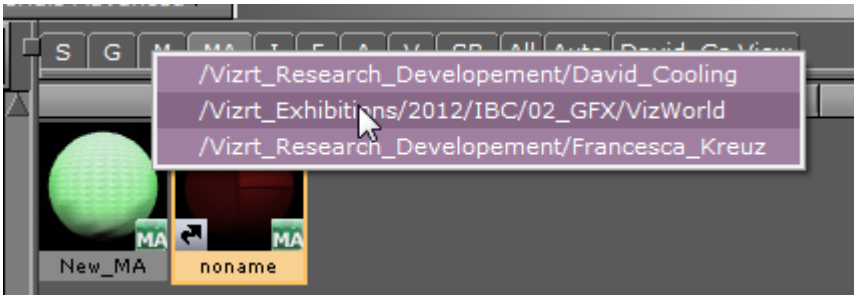
- **Mark as Hot Folder...**: Adds the currently selected folder to the list of favorites.
- **Unmark as Hot Folder...**: Removes the currently selected folder from the list of favorites.

Item Type History

The item type history shows an overview of previously used folders and is saved independently of the item type. Each item type has a separate history list, which can contain up to 20 hits.

> ⚠ **Note:** A folder is added to the History list when an item has been added to the folder, regardless of whether the item was actually used in a scene.

To show the item type history list



1. Hold the pointer over one of the tabs in the Content Switcher.
2. Press and hold <Ctrl>, and right-click.
3. Move through the list and click to show the selected folder.

> ⚠ **Note:** Empty folders do not show item type history. The item list is shown in last used order.

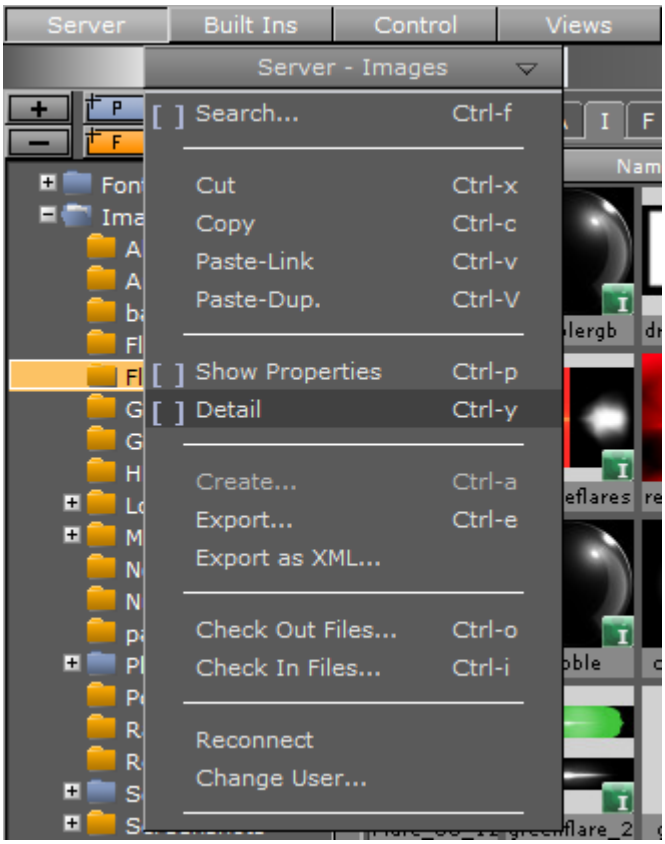
To Show the item type Last Used Folder

1. Hold the pointer over one of the tabs in the Content Switcher.

2. Press and hold <Ctrl>, and left-click.
3. The content of the last used folder for this item type will be shown.

4.3.4 Server Menu

At the top of the Server Panel is the Server menu.



- **Search**: Opens the Search window (see Item Search).
- **Cut**: Cuts the currently selected item, so that you can paste it in another folder.
- **Copy**: Copies the currently selected item, so that you can paste it in another folder.
- **Paste-Link**: Pastes a copied item as a linked file. Since the new item is linked, the item thumbnail will contain a Link icon in the lower left corner (see linked items).
- **Paste-Dup**: Pastes a copied item as an independent file.
- **Show Properties**: Shows the Properties Panel at the bottom of the item Panel, with the File Info tab selected.
- **Detail**: Shows additional information about owner and creation date is displayed. By default, items are presented in the item list with thumbnail and name.
- **Create**: Creates a new scene or material item (see Working with Items).
- **Export**: Exports an image or audio item.
- **Export XML**: Exports an XML representation of a Scene and other objects into a file. XML files can be used to compare two versions of a Scene using a file comparison utility.
- **Check Out Files:** Checks out the selected items, so that only the User who checked out the files can modify them.

> ⚠ **Note:** Other users can see the name of the user that has checked out the files from the Properties Panel.

- **Check In Files**: Checks in the selected items, so that all Users can modify them. This option is only available for items that have already been checked out.
- **Reconnect**: Re-establishes the connection to To start Viz Artist, the user must log on to a Graphic Hub. The Graphic Hub is the database solution where all Viz Artist items are stored.
- **Change User...**: Changes the current User.

4.3.5 Properties Panel

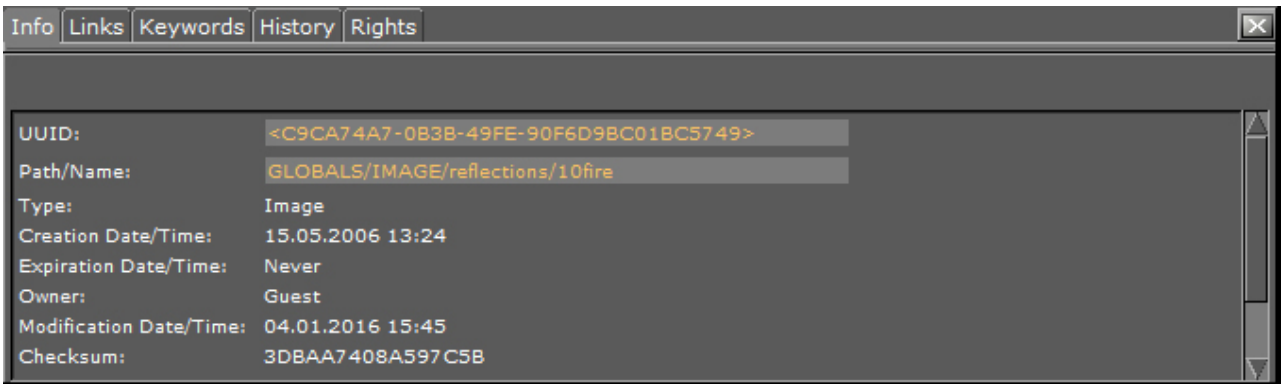
The Properties panel shows detailed information about the selected item in the Item List. It is shown below the item list.

This section contains the following topics and procedures:

- File Info Tab
- Links Tab
- Keywords Tab
- History Tab
- Rights Tab
- To Show the Properties Panel
- To Show the Properties Panel with a Selected Tab Open
- To Close the Properties Panel

File Info Tab

The File Info tab shows various information about the selected item:

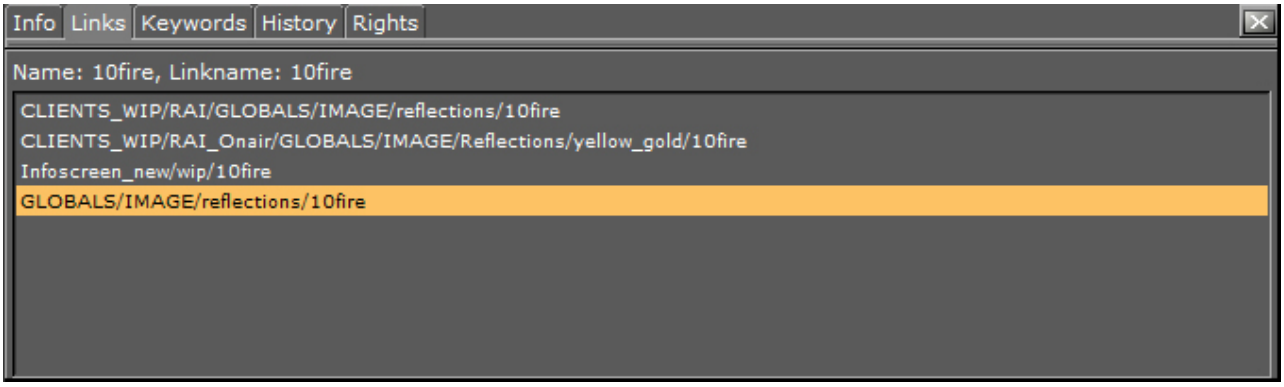


- **UUID**: Shows the universally unique item identifier.
- **Path/Name**: Shows the path to where the item is located and the name of the item.
- **Type**: Shows the item type, for example, Fontstyle, Image, or Material.
- **Creation Date/Time**: Shows the date and time when the item was created.
- **Expiration Date/Time**: Shows the date and time when the item expires, if applicable.
- **Owner**: Shows the name of the user that owns the item.
- **Modification Date/Time**: Shows the date and time when the item was last modified.

- **Checksum:** Shows the file's checksum in case you need to check its validity.
- **File size**: Shows the size of the item file.
- **Checked Out By**: Shows the name of the User who did the checkout operation if the item has been checked out.
- **Locked By**: Shows the name of the user that has performed the lock operation if the item has been locked.

Links Tab

The Links tab shows the links that are assigned to the selected item:



- **Path**: Shows the path to where the linked items are located.
- **Link Name**: Shows the names of the linked items.

> ⚠ **Tip:** Click one of the links to select the folder where the linked file is placed, and to show the item in the Item List.

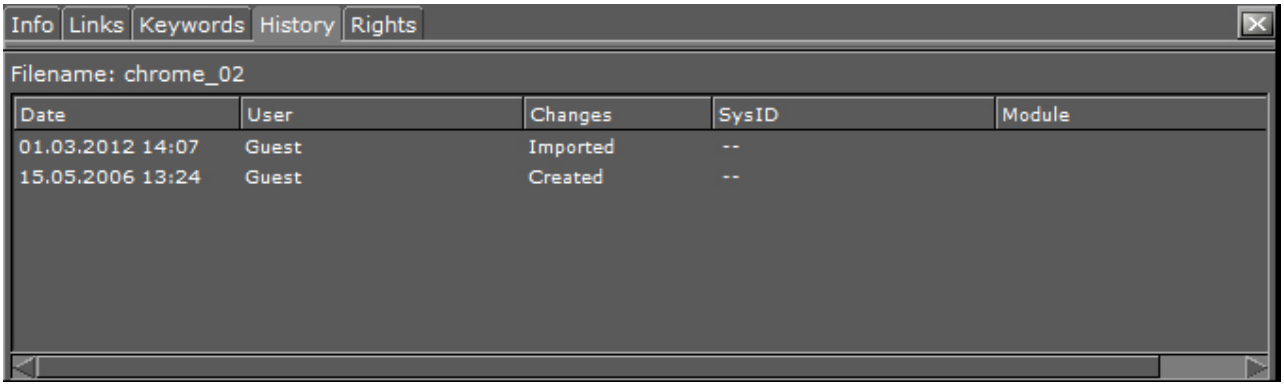
Keywords Tab

The Keywords tab shows the keywords that are assigned to the selected item. Keywords can be assigned, for example, when an item is imported to Viz Artist (see Assign Keywords to Items).

⚠ **Note:** If multiple items have been selected, and some keywords are grayed out, the gray keywords are assigned to at least one of the selected items, but not all.

History Tab

The History tab shows the history of the selected item:



- **Date:** Shows the date and time when the item was created or modified.
- **User:** Shows the name of the user that created or modified the item.
- **Changes:** Shows the operation that was performed on the item, for example, if the document was created.
- **SysID:** Shows the Viz Artist license number on the machine that performed the operation.
- **Module:** Shows the name of the application that changed the data.
- **Source:** Shows the source of the data. For example, the archive name for an imported object or the name of the Graphic Hub from where it was copied.

Rights Tab

The rights tab shows the rights of the selected file.

Rights can only be changed with the Graphic Hub Manager, here you can only see the applied rights of your file.

To Show the Properties Panel

- Select **Show Properties** from the Server Menu, or
- Hold the pointer over the Server Panel and press the keyboard shortcut <Ctrl+P>, or
- Right-click an item in the Item List. In the menu that opens, select **Show Properties**...

To Show the Properties Panel with a Selected Tab Open

- Right-click an item in the Item List. In the menu that opens, select either:
  - **Show Properties...**
  - **Show Folder Links...**
  - **Show Keywords...**
  - **Show History...**
  - **Show Rights...**

> ⚠ **Note:** The **Show Folder Links...** option is only available for items that are linked (see linked items).

To Close the Properties Panel

Do either of the following:

- Right-click in the Item List and select **Close Properties**, or
- Click the **Close** icon in the upper right corner of the Properties Panel, or
- Clear the **Show Properties** check box in the Server Menu.

4.4 What Are Items

Items are properties that are used to build up a Scene. Items can also be complete Scenes.

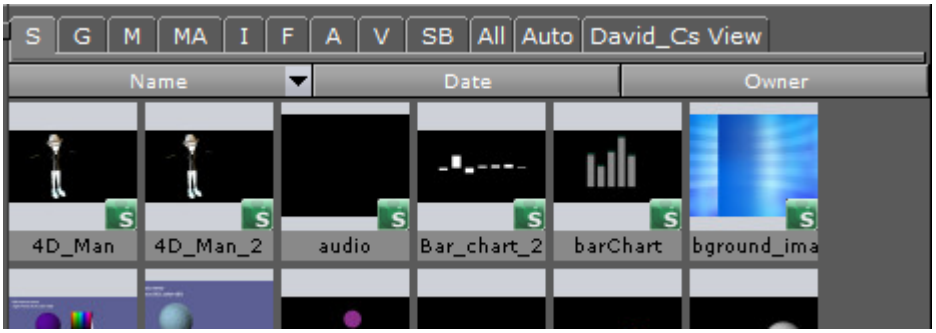
The Server Panel displays items available in the selected folder of the Server Tree.

This section contains information on the following topics:

- Scenes
- Scenes for Context-enabled Shows
- Geometries
- Materials
- Materials Advanced
- Images
- Fontstyles
- Audio Clips
- Video Clips
- Substances

4.4.1 Scenes

Scenes are root items that hold other items. Scenes are stored in the Graphic Hub database and are available through the Server Panel.



If a Scene is opened, the Scene Tree area shows the logical visualization of all the objects within the Scene. The Scene Tree consists of Containers that hold items and properties, organized in a hierarchy and grouped in logical divisions. The items and properties can be Geometries, Images, Materials, Fonts, transparency functions, key functions, and many more (see Item Panel and Built Ins).

> ⚠ **Note:** Viz supports the import of scenes from numerous third party file formats (see Advanced Issues with Video Codecs)

4.4.2 Scenes for Context-enabled Shows

A context-enabled show is a Viz Trio specific feature that allows the Trio operator to switch between concepts and variants within a show. For example, a News concept could be switched to a Sports concept, depending on the content being taken to air.

When designing context-enabled shows in Viz Artist, each scene must follow a set of naming conventions. All variants of a Scene within a specific concept will appear as one *template* in the Viz Trio **Template List**. Please refer to the Viz Trio User Guide for further information on Trio templates.

For a show to successfully use context-enabled Scenes in Viz Trio, the folders and Scene names must follow a naming convention that gives them properties and values for Viz Trio to recognize and use:

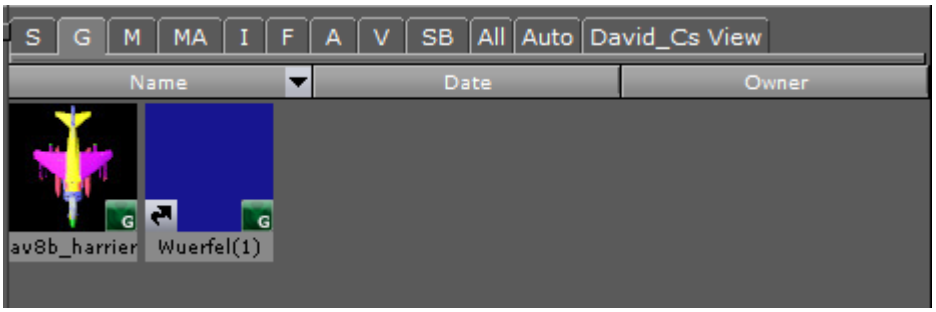
Type	Property		Name	Example
Concept (folder)	concept	=	`<conceptname>`	concept=`<conceptname>` concept=News concept=Sport
Context (folder) User-defined	<context>	=	`<contextname>`	`<context>=<contextname>` platform=HD platform=SD
Variant (file)	_variant	=	`<variantname>`	`<scene>`_variant=`<variantname>` chart_variant=Stock headlines_variant=rotation headlines_variant=simple weather_variant=3Days

In addition, to successfully create context-enabled show-designs, make sure that each scene variant is using the same containers, container names and plug-ins.

⚠ **IMPORTANT!** For the variants it is important to add an underscore "_" to separate the Scene name from the variant keyword.

4.4.3 Geometries

Geometries are stored in the Graphic Hub database and are available through the Server Panel.



Geometries can be created in Viz Artist or imported from an external modeling program (see Advanced Issues with Video Codecs).

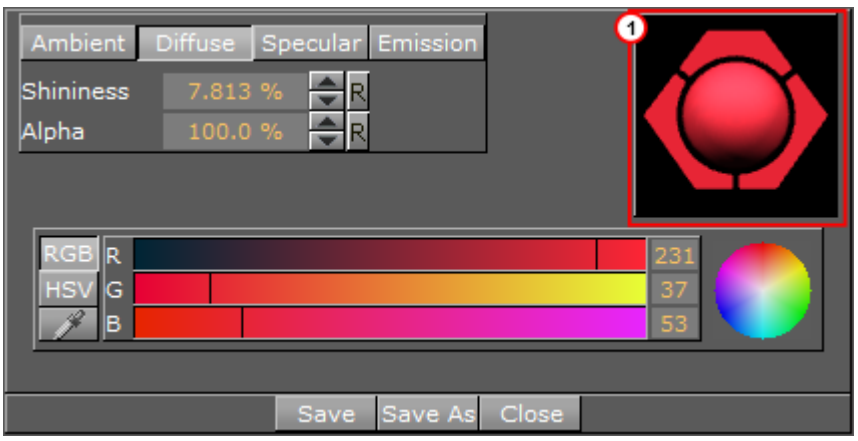
Viz Artist also has a set of Geometries that are built-in Geometry plug-ins. These geometries are common plug-ins that can be used in Scenes.

A Geometry can be added to a Scene as a standalone Container or as a Geometry property of a Container. Drag the object from the Item List (or Favorites Bar) to the Scene Tree, Scene Editor or add to a Container in the Scene Tree.

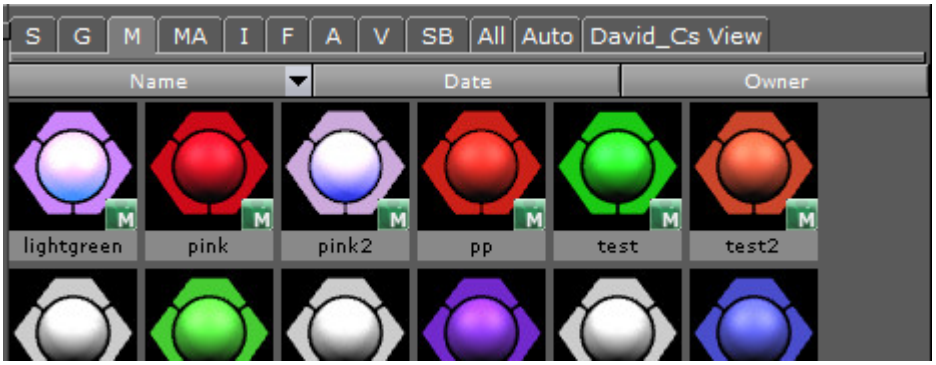
4.4.4 Materials

Materials hold colors, customized colors and light effects which can be applied to objects in a Scene.

The Materials icon (1) in the item panel shows the current state of the Material. The color for a lit object is shown on the sphere within the icon, while the three parts around the sphere show the plain color for an object that is not lit.



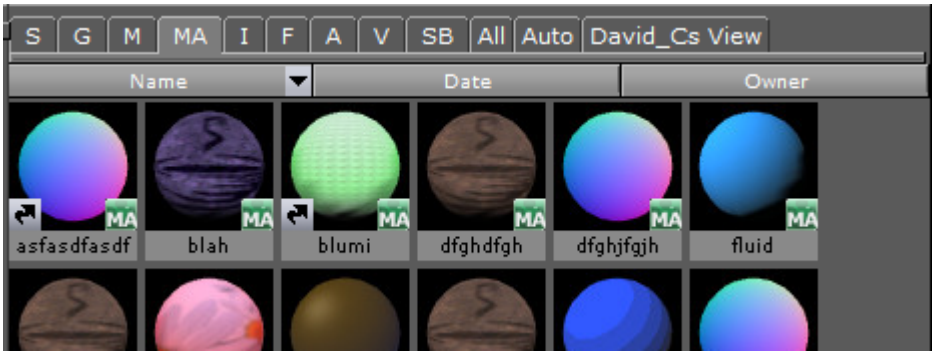
When a Material has been added to a Scene, the Scene-specific Material can be modified in the Material editor that opens in the Properties panel. Only the Scene specific Material will be modified, the original Material in the Graphic Hub database will not be affected (see Working with Material and Material Advanced Items).



4.4.5 Materials Advanced

An Material Advanced is an item that can combine a Shader plug-in, an Image and a Material item. A combination minimum of two items are required to create the Materials Advanced.

The Materials Advanced item is stored in the Graphic Hub database and are available in the Server Panel Item List.

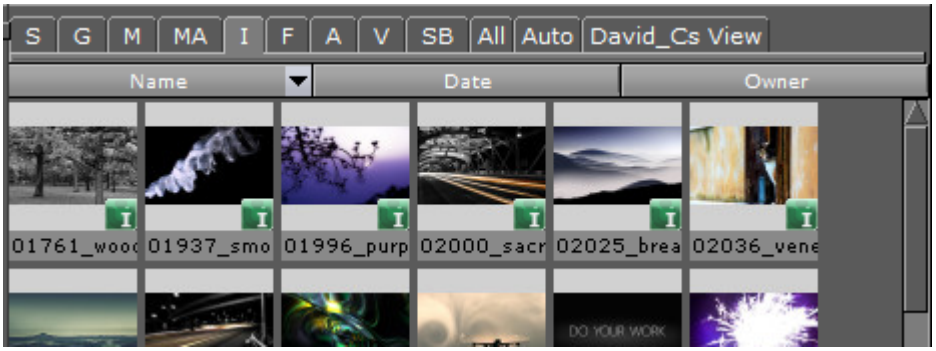


To add a Material Advanced as a Container item, drag the Material Advanced from the Item List (or Favorites Bar) to a Container in the Scene Tree.

4.4.6 Images

Viz Artist supports still images in formats TIF, BMP, SGI, JPG, TGA, PSD, XPM, HDR, PNG. The Images are stored in the GH database and are available through the Server Panel.

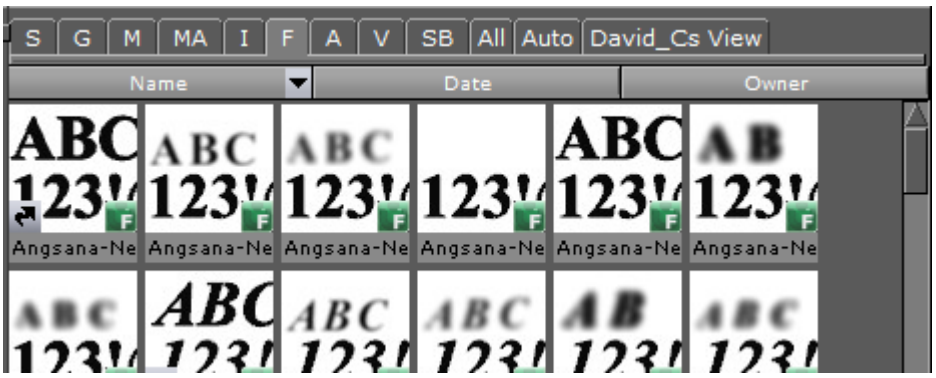
> ⚠ **Note:** The maximum size of an image that can be imported and used as a texture in Viz Artist, is 16k.



An Image can be added to a Scene as a standalone Container, or as a Container item. Drag the Image from the Item List (or Favorites Bar) to the Scene Tree, Scene Editor or add to a Container in the Scene Tree.

4.4.7 Fontstyles

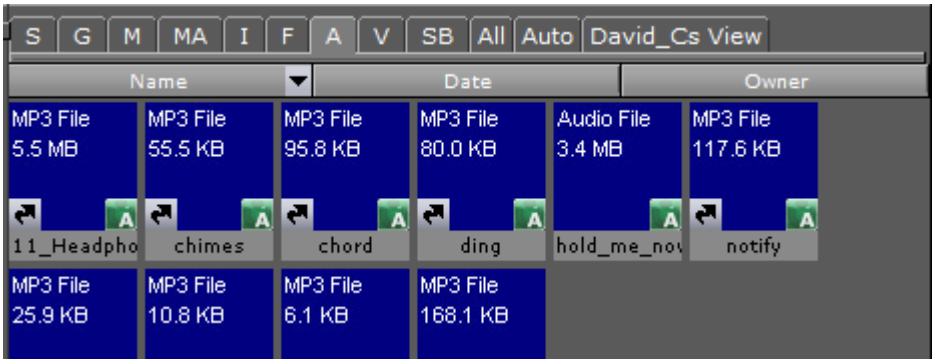
Fonts are complete character sets within a unique style. The Fonts are stored in the Graphic Hub database and are available through the Server Panel.



A Fontstyle can be added to a Scene as a standalone Container or as a Container item. Drag a Fontstyle from the Item List (or Favorites Bar) to the Scene Tree, Scene Editor or add to a Container in the Scene Tree.

4.4.8 Audio Clips

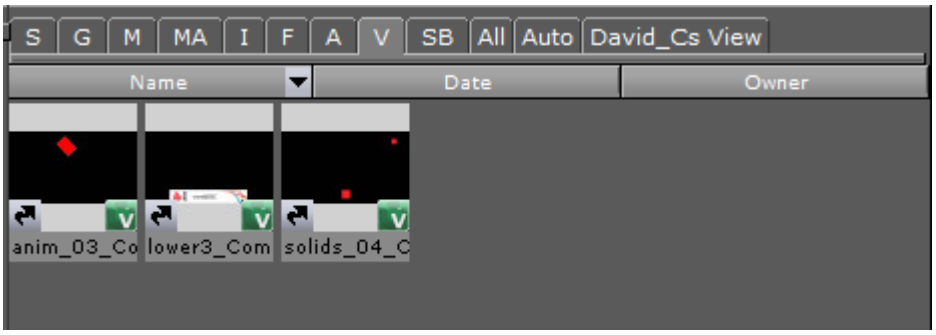
Viz Artist supports audio clips in formats WAV, MP3, and Vorbis (Ogg), which are stored in the GH database and are available through the Server Panel.



The audio clips can be added to Scenes and/or to the animation of a Scene.

4.4.9 Video Clips

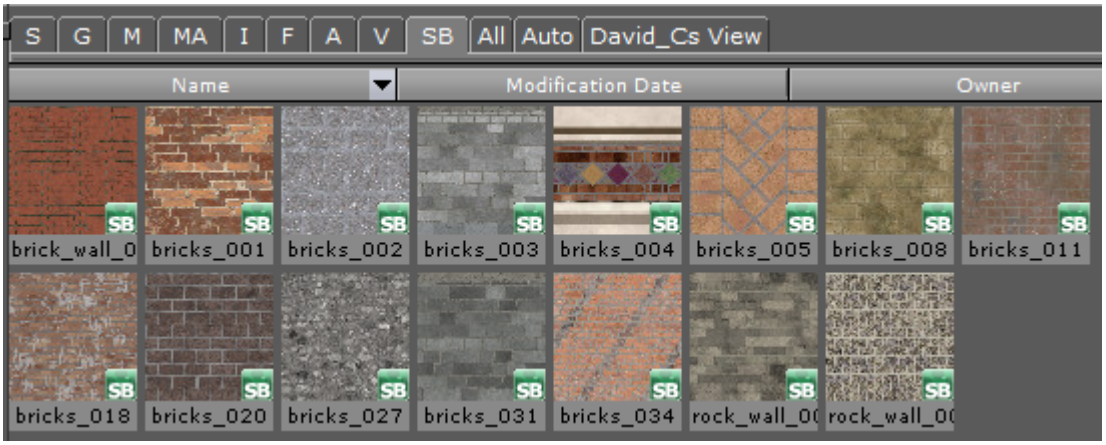
Video Clip items are video clips which are stored in the Graphic Hub and are available through the Server Panel.



Video Clip items are used with Clip Channel Media Assets, as the video source file.

4.4.10 Substances

Substances contain information and parameters for use with a Substance Shader.



To use a Substance on a Container, the Substance Shader must be used. The Substance Shader takes the substance as an input parameter.

⚠ **Note:** To import and use a Substance a license is required, issued by Vizrt.

4.5 Working With Items

This section contains information on the following topics:

- Common Item Procedures
- Active Items
- Linked Items
- Referenced Items

4.5.1 Common Item Procedures

These are procedures which are common to all item types.

This section contains information on the following topics:

- To View items in a Folder
- To Rename an item
- To Delete an item
- To Delete an item with all References
- To Create an item Copy in a New Folder
- To Create an item Copy in the Same Folder
- To Move an item to a New Folder
- To Create a Linked item

To View items in a Folder

1. Open the item panel from the Content Switcher:

Item	Tab
Scenes	S
Geometries	G
Materials	M
Materials Advance	MA
Images	I

Item	Tab
Fonts	F
Audio Clips	A
Video Clips	V
Substances	SB
All items	All
Shows the most common item type (mode)	Auto
User-constrained free text search (searches can be saved for later use)	<user>'s View

2. Select a folder in the Server Tree.

> ⚠ **Note:** Select **All** in the Content Switcher to view all item types in the folder selected.

To Rename an item

1. Select the item to be renamed.
2. Open the rename text field:
   - Right-click the item and select **Rename** from the context menu, or
   - Press <F2>.
3. In the text field that appears, type a new descriptive name.
4. Press <Enter>.

> ⚠ **Note:** Special characters, such as space, are not allowed in item names.

> ⚠ **Note:** If renaming a linked item, only the name of the selected item will change, not the linked one (see Linked Items).

To Delete an item

1. Select the item to be deleted.
2. Delete the item:
   - Drag the item to the trash can, or
   - Right-click the item and select **Delete** from the context menu, or
   - Press <Delete>.
3. Click on **Yes** to confirm.

> ⚠ **Tip:** To delete multiple items, select the items to be deleted and drag to the Trash Can.

> ⚠ **Note:** If deleting a linked item, only the link to the folder will be removed. The item itself will remain in the database (see Linked Items).

> ⚠ **Note:** Referenced Items cannot be deleted. Referenced items are used by other items.

To Delete an item with all References

Deletes the selected item and all its referenced items. Referenced Items are not deleted if they are referenced by other items.

> ⚠ **Note:** Referenced items are deleted if they are referenced to ONLY the selected item. If they are also referenced to others, then they will not be deleted.

1. Locate the item to be deleted.
2. Right-click the item and select **Delete with all references** from the context menu.
3. Click on **Yes** to confirm.

To Create an item Copy in a New Folder

Use this procedure to create an independent copy of an item(s) in a new folder:

> ⚠ **Note:** To select more than one item press and hold <Ctrl> and click on each required item or press and hold <Shift> to select a range or items.

1. Select the item(s) to be copied.
2. Press <Ctrl> and drag the item(s) to the target folder.
   or

1. Select the item(s) to be copied.
2. Select **Copy** from the Server Menu or <Ctrl+C>.
3. Open the folder where the target item(s) are to go.
4. Select **Paste-Dup** from the Server Menu or <Ctrl+Shift+V>

To Create an item Copy in the Same Folder

Use this procedure to create an independent copy of an item in the same folder:

1. Right-click the source item.
2. In the context menu that shows, select **Duplicate**.
3. In the dialog box that opens, type a descriptive name for the new item.

4. Click the **Ok** button.
   or

1. Select the item(s) to be copied.
2. Select **Copy** from the Server Menu or <Ctrl+C>.
3. Select **Paste-Dup.** from the Server Menu or <Ctrl+Shift+V>
4. In the dialog box that opens, type a descriptive name for the new item.
5. Click the **Ok** button.

To Move an item to a New Folder

> ⚠ **Note:** To select more than one item press and hold <Ctrl> and click on each required item.

- Drag the item or items from the Item List onto the target folder in the Server Tree.
  or

1. Select the source item(s).
2. Select **Cut** from the Server Menu, or <Ctrl+X>
3. Open the folder where the target item(s) are to go.
4. Select **Paste-Dup.** from the Server Menu, or <Ctrl+Shift+V>.

To Create a Linked item

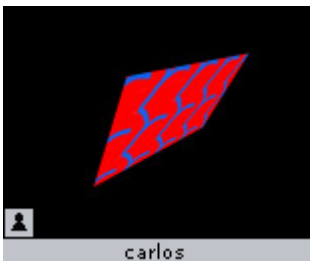
1. Select the source item.
2. Press **<Alt>** and drag the item to the target folder.
   or

1. Select the source item.
2. Select **Copy** from the Server Menu, or <Ctrl+C>
3. Select the folder where the target item is to go.
4. Select **Paste-Link** from the Server Menu, or <Ctrl+V>.

4.5.2 Active Items

Items that are currently in work are active items. Active items are locked or checked out.

Items with locked icon



An active item has a small icon in the lower left corner of the thumbnail in the Item List which changes to show different states:

- **Locked:** If a user connected to the database opens an item, the item will be locked for all other users and marked with the keyhole icon. A locked item means that the item can be opened and viewed by all users, but only modified or saved by the user that has locked it. The item will be unlocked when the user closes the item or disconnects from the database.
- **Checked Out:** If a user connected to the database checks out an item, the item will be locked for all other users and marked with the stop icon. A checked out item means that the item can be opened and viewed by all users, but only modified and saved by the user that has checked it out. To cancel the check out, the item must be checked in.

  > ⚠ **Note:** The item will stay checked out (locked) even if the user closes the item or disconnects from the database.

- **Locked and Checked Out:** For a locked and checked out item, the icon represents a combined keyhole and stop sign.

This section contains information on the following procedures:

- - To Lock an item
  - To Check Out items
  - To Check In items
  - To Lock and Check Out items

To Lock an item

- An item is automatically locked when a user opens the item. It is not possible to perform this operation manually.

To Check Out items

1. Select the item or items.
2. Check out the items:
   - Right-click an item and click on **Check Out** in the context menu (one item only), or
   - Hold the pointer over the Server Panel and use the keyboard shortcut <Ctrl+O>, or
   - Select **Check Out Files...** from the Server Menu.

To Check In items

1. Select the item or items.
2. Check in the items:
   - Right-click the item and click **Check In** from the context menu (one item only), or
   - Press the keyboard shortcut <Ctrl+I> while holding the pointer over the Server Panel, or
   - Select **Check In Files...** from the Server Menu.

To Lock and Check Out items

- Items are automatically locked when a user opens the item. You cannot lock items manually.

4.5.3 Linked Items

Every item in the database is shown in one or more folders.

A linked item has a small link icon attached to the lower left corner of the thumbnail in the Item List. A linked item will show, logically, in multiple folders, although only one item is physically saved in the Graphic Hub database.

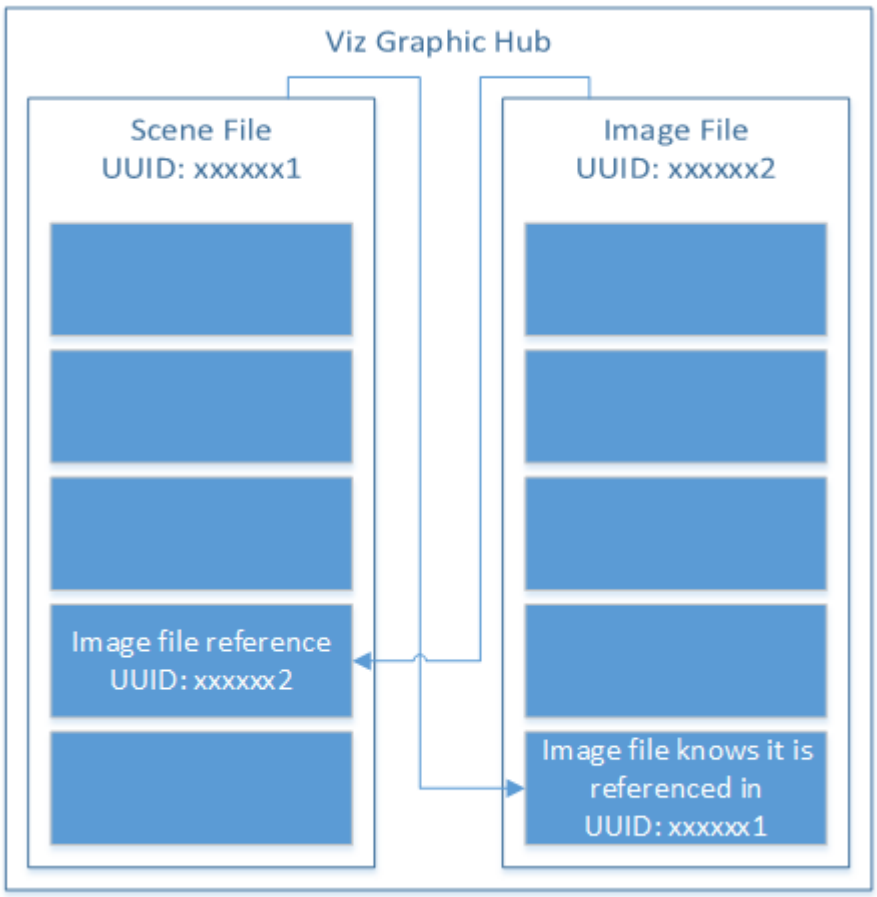
To view the UUID of the linked item, open the Properties Panel and select the File Info tab. The UUID of the one saved item is the same for all linked items, while information such as path and name may vary.

If modifying the item in one of the folders, for example cropping an image, all the linked items will reflect these changes.

To create a linked item see To Create a Linked item.

4.5.4 Referenced Items

A referenced item is an item which is connected to and used by another item, by a reference only.



Some items are, or can be, made up of other items, i.e. Geometries. A Geometry item can be made up of other items. These other items are not physically part of the Geometry item, but are used by reference. Because the Geometry item uses another item as part of its output, any changes to the referenced item will result in changes the Geometry item itself. The item which is referenced cannot be modified without a warning message. If the referenced item is modified then the item which is referencing that item will be modified too - it will look and act differently. A referenced item cannot be deleted. The item which holds the reference has to be deleted before the referenced item can be deleted from the Graphic Hub. An item can reference many other items.

4.5.5 Working with Audio (Clips) Items

This section contains information on the following procedures:

- To Add an Audio Clip to a Scene (Context Menu)
- To Add an Audio Clip to a Scene
- To Import an Audio Clip
- To Export an Audio clip

To Add an Audio Clip to a Scene (Context Menu)

1. Select the target Container.
2. Go to *Container plug-ins -> Global* and find the **Audio** plug-in.

3. Right click the plug-in and select **Add Audio Clip**.
   The Audio plug-in is added to the selected Container, and the Container will show that an animation was created.
   
4. Click on the **Stage** Button.
5. Highlight the **Audio Clip** in the Stage tree.
6. Click on the **Add key Frame** icon (1).
   
7. In the Key Frame panel add an Audio Clip. There are two ways to add an Audio Clip:
   a. Click **DB Clip** (1).
   b. Click on **Server/Stage** in the Main Menu.
   c. Select and drag an Audio Clip from the Server Tree panel to the **Clip drop box** (2).
   or
   d. Click **File** (1).
   e. In the **Clip** line (3), click the  icon to select an Audio clip to use.
   

To Add an Audio Clip to a Scene

1. Add the **Audio** plug-in (*Container plug-ins -> Global*) to a Container (see Built Ins).
2. In the Main Menu, click on **Stage**.
3. Drag the Container to the Stage Tree Area.
   
4. In the Stage Tree right-click the Container Actor (1)
5. Click *Audio > Clip* (2) (the Container will show that an animation was created).
   
6. Click on the Audio Clip Channel (3) in the Stage Editor, to open the Key Frame panel (if not open).
   
7. In the Key Frame panel add an Audio Clip. There are two ways to add an Audio Clip:
   a. Click **DB Clip** (4).
   b. Click on **Server/Stage** in the Main Menu.
   c. Select and drag an Audio Clip from the Server Tree panel to the **Clip drop box** (5).
   or
   d. Click **File** (4).
   e. In the **Clip** line (6), click the  icon to select an Audio clip to use.
   

To Import an Audio Clip

- For information about how to import an audio clip (or an archive) from a hard disk or a network share to the database see Import and Archive.

To Export an Audio clip

1. Open the Select Export Directory panel:
   - Click on an Audio clip or Audio clips and select **Export...** from the Server Menu, or
   - Right-click the Audio clip to be exported. In the menu that opens, select **Export**...
   

- In the Select Export Directory panel, select the folder where the Audio clips should be exported to.

2. Click the **Ok** button. The audio clip will be exported to the selected folder, with its original name.

4.5.6 Working with Fontstyle Items

This section contains information on the following procedures:

- To Import a Font
- To Edit a Fontstyle
- To Create a Fontstyle
- To Show/Hide Blur or Outline Fontstyles

To Import a Font

- For information about how to import a font from a hard disk or network share to the database see Import of Files and Archives.

To Edit a Fontstyle

1. Double-click the Fontstyle to open the Fontstyle Editor.
2. If the Fontstyle is referenced in more than one Scene, you must confirm that you want to change it.

3. In the Fontstyle Editor define the new settings.
4. Click the **Save** button.

To Create a Fontstyle

New Fontstyles are based on a font which already exists.

1. Double-click the Fontstyle to open the Fontstyle Editor.
2. If the Fontstyle is referenced in more than one Scene, you must confirm if you want to change it.

3. In the Fontstyle Editor, define the new settings.
4. Click the **Save As** button.
5. In the **Save As...** panel, type a descriptive name for the new Fontstyle in the text box.
6. Click the **Ok** button.

To Show/Hide Blur or Outline Fontstyles

- Right-click a Fontstyle and select either:
  - **Show/Hide "Blur" Fonts**, or
  - **Show/Hide "Outline" Fonts**

4.5.7 Working with Geometry Items

You can add geometries to a Scene as a standalone Container or as the Geometry property of a Container.

4.5.8 Working with Image Items

The maximum resolution of an image that can be imported and used as a texture in Viz Artist, is 16k.

This section contains information on the following procedures:

- To Add an Image to the Graphic Hub Database
- To Edit an Image
- To Export Images

To Add an Image to the Graphic Hub Database

- **Import Images**: Images can be imported from a hard disk or a network share to the GH database (see Advanced Issues with Video Codecs).
- **Create a Snapshot**: Images can be created by taking a Snapshot of the Scene Editor.
- **Image Editor:** In the Server Tree open an Image in the Image Editor (double-click the Image), modify (if required) and select **Save as...**.

To Edit an Image

1. Double-click the image to open the Image Editor.
2. In the Image Editor that opens, define the new settings.
3. Click the **Save** button.

To Export Images

You can export one or multiple images. Exporting RGB images with alpha creates RGBA image files.

1. Select the image(s) to be exported.
2. Open the **Select Export Directory** panel and select the folder where the image should be exported to
   - Select **Export...** from the Server Menu, or (when only one image is selected)
   - Right-click the image to be exported. In the menu that opens, select **Export**...

   
3. Select an image format for export.

> ⚠ **Note:** The exported image will retain its original name.

4.5.9 Working with Material and Material Advanced Items

This section contains information on the following:

- Working with Material Items
- To Edit a Material Advanced

Working with Material Items

When a Material has been added to a Container, the Container specific Material can be modified in a Material Editor that opens in the Properties Panel. Only the Container specific Material will be modified, the original Material in the Graphic Hub database will not be affected.

To create a new Material in the Graphic Hub database, see Material Editor.

To Create a Material Advanced

1. From the Item List, drag a Material and/or an Image to a Container in the Scene Tree.
2. From the Built Ins drag a Shader to the same Container in the Scene Tree (if required).

   > ⚠ **Note:** A combination of at least two items (Material, Shader or Image) must be applied to create a Material Advanced.

3. Right-click the Container in the Scene Tree.
4. Click **Merge to Material Advanced**.
   The Material, Shader, and Image icons in the Container will then be merged into one combined Material Advanced icon.
5. Select the project or folder where the Material Advanced should be placed (1).
6. Drag the Material Advanced icon from the Container in the Scene Tree to the Item List (2).
7. In the dialog box that opens, type a descriptive name for the Material Advanced.

   > ⚠ **Note:** Special characters, such as space, are not allowed in item names.

8. Click the **Ok** button.

To Edit a Material Advanced

1. Select the project or folder where the existing Material Advanced is placed.
2. Drag the Material Advanced to a Container in the Scene Tree.
3. Right-click the Material Advanced icon on the Container.
4. In the context menu click **Split Material Advanced**. The Material Advanced icon on the Container will then be replaced by separate icons for any Material, Shader or Image that the Material Advanced consists of.
5. Apply new items or modify the Material, Shader or Image settings.
6. Right-click the Container in the Scene Tree.

7. In the context menu, click **Merge to Material Advanced**. The Material, Shader and Image icons in the Container will merged into one combined Material Advanced icon.
8. Select the project or folder where the Material Advanced should be placed (1).

9. Drag the Material Advanced icon from the Container in the Scene Tree to the Item List (2).

10. In the dialog box that opens, type a descriptive name for the Material Advanced.

> ⚠ **Note:** Special characters, such as space, are not allowed in item names.

11. Click the **Ok** button.

4.5.10 Working with Scene Items

To Create a New Scene

This will create an empty new Scene. Add containers as required in the scene tree.

1. Open the Scene tab (see To View items in a Folder).
2. Select the folder where the new Scene should go.
3. Open the Create New Scene panel:
   - Right-click in the Item List. and select **Create new Scene**, or
   - Open the Server Menu and select **Create...**, or
   - Hold the pointer over the Server Panel and press <Ctrl+A>.
4. In the Create New Scene panel enter a text string for the new Scene.
5. Click on **OK**.


To Import a Scene

- For information about how to import a Scene (or an Archive) from a hard disk or network share to the database see Import and Archive.

To Save a Scene

- Click the **Save** button in the lower left corner of the window, or
- Hold the pointer over the Scene Tree area and press the keyboard shortcut <Ctrl+S>.

To Save a Scene with a New Name

1. Click the **Save As** button in the lower left corner of the window.
2. In the **Scene Save As...** dialog box that opens, type a descriptive name in the **Scene Name** text box.
3. Click the **Ok** button to confirm.

Scene Save as... panel Context Menu

Right-click in the **Scene Save as** panel to open its context menu. Select from:

- **Rename:** Renames the selected folder
- **Delete:** Deletes the selected folder
- **Create Sub Project:** Creates a new sub Project folder in the selected Project folder
- **Create Sub Folder:** Creates a new sub folder in the selected folder
- **Add to Bookmark:** Adds selected folder to Bookmarks
- **Delete from Bookmark:** Deletes selected folder from Bookmarks
- **Switch Folder Highlighting OFF/ON:** Sets highlighting of folders to ON or OFF. For example, if a folder (and sub-folders) contain a Scene it will be highlighted when the Scene tab is selected. This feature can also be set in **User Interface** of the Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide).
- **<Bookmark(s)>:** Selects a bookmarked folder

To Undo/redo Operations in a Scene

- Click the **Undo/Redo** button in the lower right corner of the Scene Tree panel, or
- Hold the pointer over the Scene Tree panel and press the keyboard shortcut <Ctrl+Z/Y>.

> ⚠ **Tip:** Holding the pointer over the **Undo/Redo** buttons will show a text in the status bar describing the action that is about to be performed. Also, operations can be undone/redone although the scene has been saved. However, this is not possible after the scene has been closed and reopened.

4.5.11 Working with Substances

This section details how to work with the Substance Shader.

Substances are usually created in Substance Designer and imported to be used within Viz Engine.

Substances use two different file extensions, `.sbs` and `.sbsar`:

- `.sbs` files are editable substance "source files". Use Substance Designer with these files.
- `.sbsar` files are performance optimized, published substances. You can import these files to Viz Artist. You cannot edit these files. To create a `.sbsar` file, first create a Shader with Substance Designer, and then publish this shader.

After importing a substance to Viz Artist, they can be used within the Substance Shader. Drag and drop the substance shader (Built Ins/Shader/Texture) onto a geometry. Then locate your imported substance assets in Viz Graphic Hub and drag it onto the Substance Field.

For details about the various options, please refer to the Plugin Description.

⚠ **Tip:** The import and use of Substances in Viz requires a separate license. You obtain this license from your local Vizrt representative.

4.5.12 Working with Video Items

There are several types of Media Assets that allow input sources to scenes as video clips, live video, video stream, graphics channels, image channels or super channels to be added.

Channel

Channel Folder Media Assets represent the rendered background and foreground graphics on the DVE Compositor. There are two channels, Background and Foreground, that can be used simultaneously.

Clip Channel

Clip Channels are used for displaying video clips. There are 16 Clip Channels available that can be used simultaneously.

GFX Channel

Graphics (GFX) Channels allow a designer to compose several independent scenes into a new scene. There are 16 Graphic Channels available that can be used simultaneously.

Image Channel

Image Channels are mainly used in conjunction with Viz Multiplay for creating video wall layouts, but can also be used to add images to a 2D environment in DVE mode. There are 16 Image Channels available that can be used simultaneously.

Live

Live Channels are used to display Live Video feeds. There are eight Live Channels available that can be used simultaneously.

Stream

Stream Channels are for IP-based live Video streams. There are 16 Stream Channels available that can be used simultaneously.

Super Channel

Super Channels provide a generic tool to create and group transitions between various types of assets in Viz Artist. There are eight Super Channels available that can be used simultaneously.

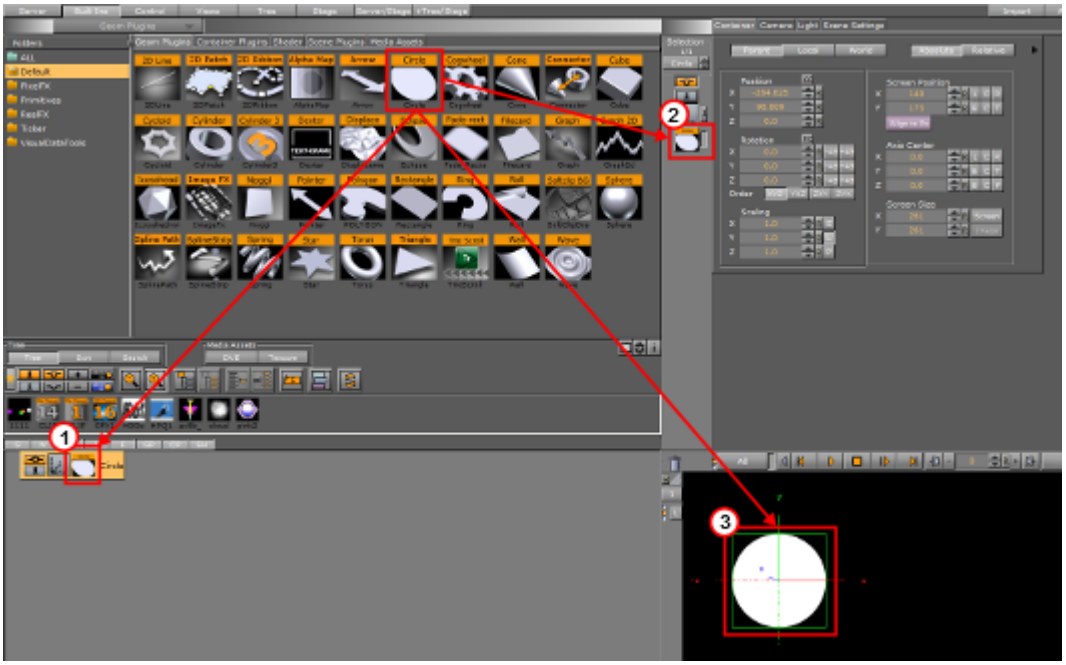
See Also

- Media Assets

4.6 Container And Scene Properties

This section details how to add **Items** and **Built Ins** to a Container, multiple Containers and Scenes.

Most properties can be dragged to a target Container, or Containers, in the Scene Tree (1), the Properties Panel (2) or the Scene Editor (3). Each property also has a context menu where a property can also be added to a Container, multiple Containers and Scenes.



This section containers the following procedures:

- To Add Properties to an Empty Scene Tree
- To Add Properties to a Container
- To Multi Drop Properties as Sub Containers
- To Multi Drop Properties on Containers
- To Add Multiple Properties to Containers
- To Add Properties to Multiple Containers
- To Add a Scene plug-in to a Scene

4.6.1 To Add Properties to an Empty Scene Tree

These properties can be added to an empty Scene Tree:

Items	Built In
· Geometries · Images · Fontstyles	· Geometries · Media Assets

1. Drag a property to either:
   - The Scene Tree (creates a new Container)
   - The Scene Editor (creates a new Container in the Scene Tree)

> ⚠ **Tip:** If multiple Geometry or Image items are dragged from the **Item List** to the **Scene Tree**, a Container for each selected item is created.

4.6.2 To Add Properties to a Container

These properties can be added to a Container:

Items	Built In
· Geometries · Materials · Material Advanced · Images · Fontstyles	· Geometry plug-ins · Container plug-ins · Shaders · Media Assets

Only one Geometry plug-in, one Shader or one Media Asset can be applied to a single Container. If a new Geometry plug-in, Shader or Media Asset is added the first one will be removed from the Container.

> ⚠ **Tip:** If a **Shader**, **Media Asset** or a few **Container plug-ins** are dragged to an object in the Scene Editor, they will be added to the Container which contains the Object. The **Media Asset** will be added as a Texture. If a Container plug-in can be added this way, the object becomes active when the Container plug-in is moved over the object.

1. Select a Container.
2. Right-click a property and select **Add to Container(s)**. Media Assets give the option to **Add <Media Asset> as DVE** or as **Texture.**

> ⚠ **Note:** If no Container is selected the property is added to the top container in the Scene Tree.

or

- Drag the property to a Container

> ⚠ **Note:** Media Assets are added as Texture to the Container.

Material items only:

1. Right-click a Container
2. Click on **Add Material...**

4.6.3 To Multi Drop Properties as Sub Containers

These properties can be multi dropped and create Sub Containers:

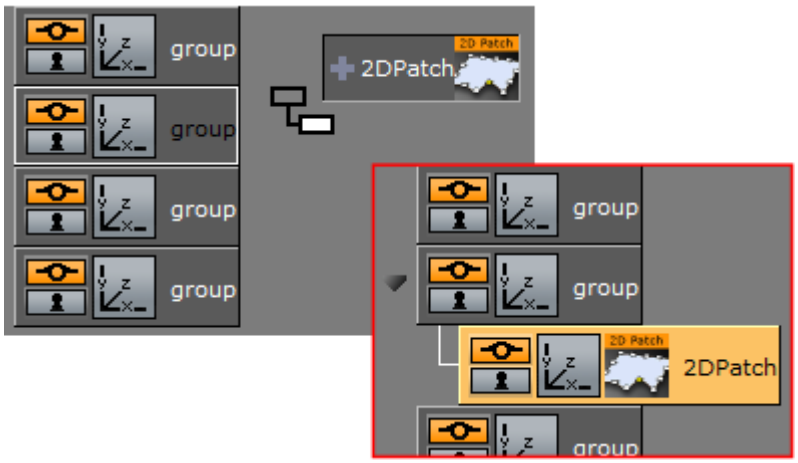
Items	Built In
· Geometries · Images · Fontstyles	· Geometry plug-ins · Media Assets

1. Drag a selected property to the Scene Tree (aligned with a Container).
2. Hold the left-click and right-click to add the property to a new Sub-Container of the highlighted Container.
3. Keep the left-click and repeat as required.

> ⚠ **Tip:** Also works if multiple Geometries are selected.

> ⚠ **Note:** When the last property to be added has been dropped, release the right-click while the cursor is still aligned the target Container. If the right-click is released when over or aligned with another Container, another (not required) Container or Sub-Container, and Property will be added.

> ⚠ **Note:** A **Media Asset** is added as a Texture.



4.6.4 To Multi Drop Properties on Containers

These properties can be multi dropped on Containers:

Items	Built In
· Materials · Material Advanced · Image · Fontstyles	· Geometries · Container plug-ins · Shaders · Media Assets

1. Drag a property over a Container.

   ⚠ **Tip:** These properties can be dragged to the Scene Tree and aligned with a Container to add them: Materials, Materials Advanced, Container plug-ins and Shaders.

2. Hold the left-click, and right-click to add the property to the highlighted Container.
3. Keep the left-click and repeat as required.

   ⚠ **Note:** When the last property to be added has been dropped, release the right-click while the cursor is still over the target Container. If the right-click is released when over or aligned with another Container, another (not required) Container or Sub-Container, and Property will be added.

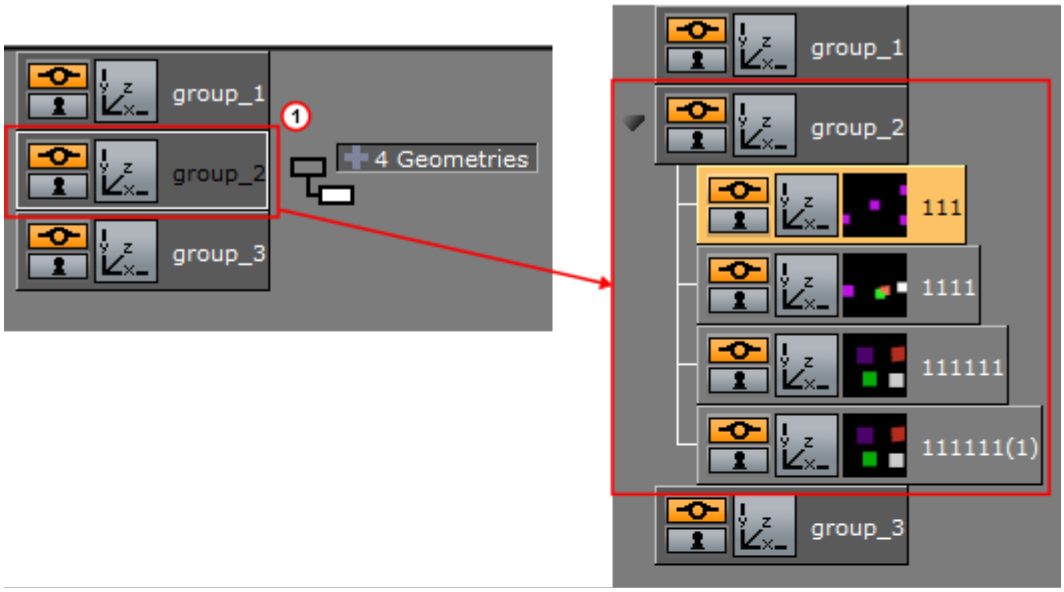
4.6.5 To Add Multiple Properties to Containers

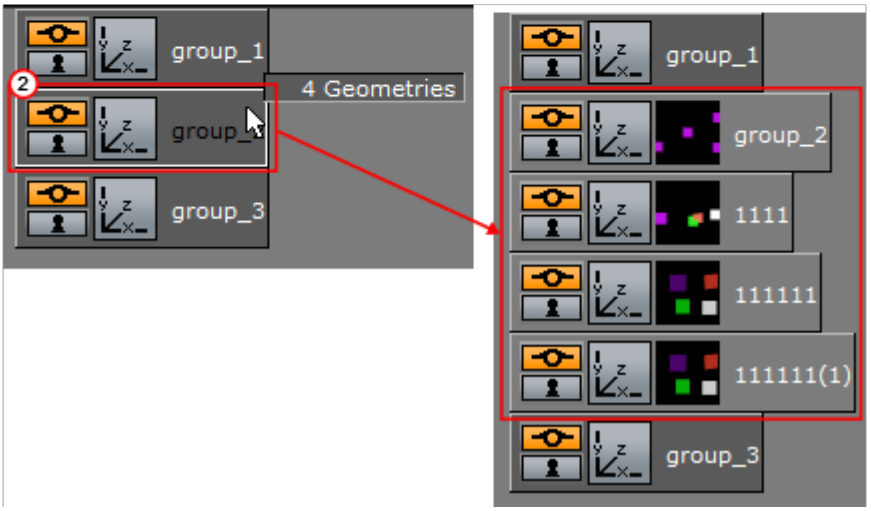
These properties can be Multi Added to a Container:

Items	Built In
· Geometries · Images	None

1. Multi select the required properties.
2. Drag the selected properties to the Scene Tree or a Container:
   - **In the Scene Tree** (1(aligned with a Container)): All selected properties are placed in the Scene Tree as sub-containers of the highlighted Container, or

- **Over a Container** (2). The first selected property is added to the highlighted Container and the other selected properties create new Containers, at the same level



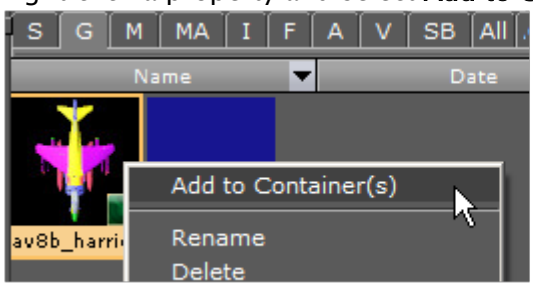


4.6.6 To Add Properties to Multiple Containers

These properties can be added to Multiple Containers:

Items	Built In
· Geometries · Materials · Material Advanced · Image · Fontstyles	· Geometries · Container plug-ins · Shaders

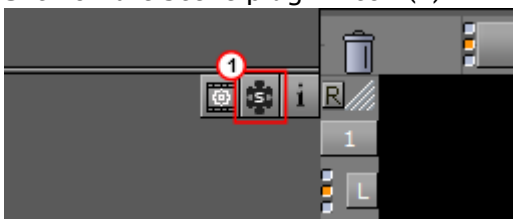
1. Multi select the required Containers.
2. Add a property:
   - Right-click a property and select **Add to Container(s)**,



   or
   - Drag a property to one of the multi selected Containers.

4.6.7 To Add a Scene plug-in to a Scene

1. Open the plug-in tab in Video Clip Playback Considerations:
   - Click on *Scene Settings -> plug-in*, or
   - Click on the Scene plug-in icon (1)



2. Drag a Scene plug-in icon to the Scene drop area.
   or

1. Right-click the Scene plug-in icon to open the plug-in Context Menu.
2. Click on **Add to Scene**...

   If a Scene plug-in has been applied to a Scene, the Scene plug-in icon will show orange  .

4.7 Assign Keywords To Items

Keywords are used to tag items in the Graphic Hub database. New keywords can be added in the Import Window, and when files are imported to the Graphic Hub database. Keywords can be assigned to an item, up to 20 keywords may be assigned to one item.

There are four ways to assign keywords. Each method starts from a different workflow point.

This section contains the following procedures:

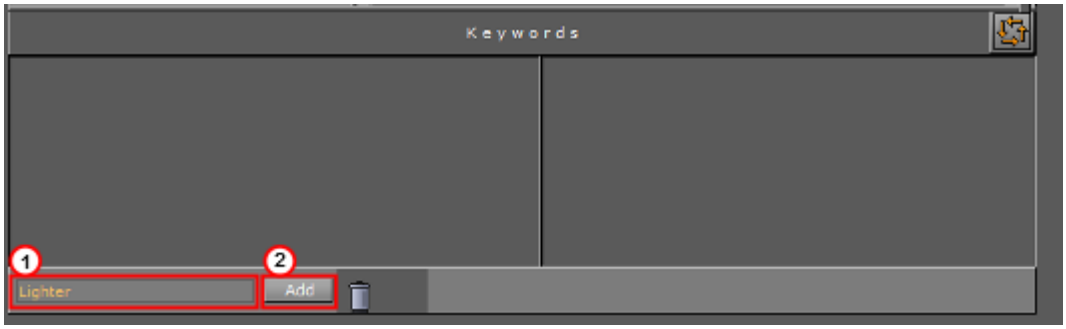
- To Create Keywords for items
- To Assign Keywords from the Search Panel
- To Assign Keywords from the Keyword Panel
- To Assign a Keyword While Importing
- To Assign Keywords from the Property Editor
- To Remove a Keyword from an item

- To Delete a Keyword

4.7.1 To Create Keywords for items

1. Select **Import** from the Main Menu.

> ⚠ **Note:** The Keywords panel shows when either Fonts, Images, Geometries, Scenes or Audio is selected to import in the **Import** Menu.



> ⚠ **IMPORTANT!** In the Keywords panel, type a descriptive keyword in the text box (1).

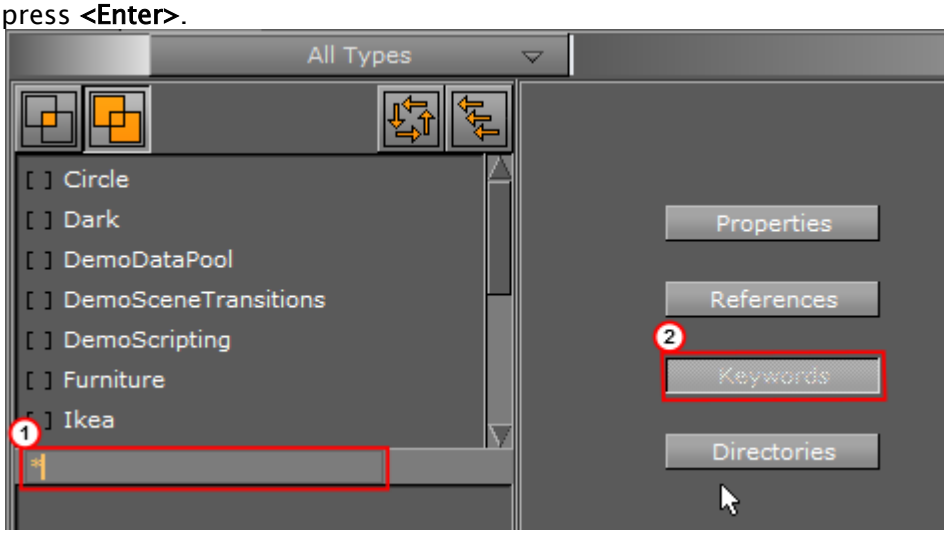
2. Click the **Add** button (2).

> ⚠ **Note:** The new keyword is saved and will show in the Keyword list.

4.7.2 To Assign Keywords from the Search Panel

1. Select **Server** from the Main Menu.
2. Open the Search panel:
   - In the Server menu, click **Search**, or
   - Hold the pointer over the Server Panel and press the keyboard shortcut <Ctrl+F>.
3. In the Search panel that opens, click Keywords (2).
4. Search for a Keyword or Keywords (1):
   - To search for all Keywords type * (an asterisk) in the text box, and press **<Enter>**, or

- To search for a specific Keyword type the name of the Keyword in the text box, and press **<Enter>**.



> **IMPORTANT!** Drag one or more Keywords from the Search panel to an item or selected items in the Item List.

5. Click **Yes** to confirm.

> **Note:** To view assigned Keywords on an item, right-click an item and click on **Show Keywords...** from the context menu.

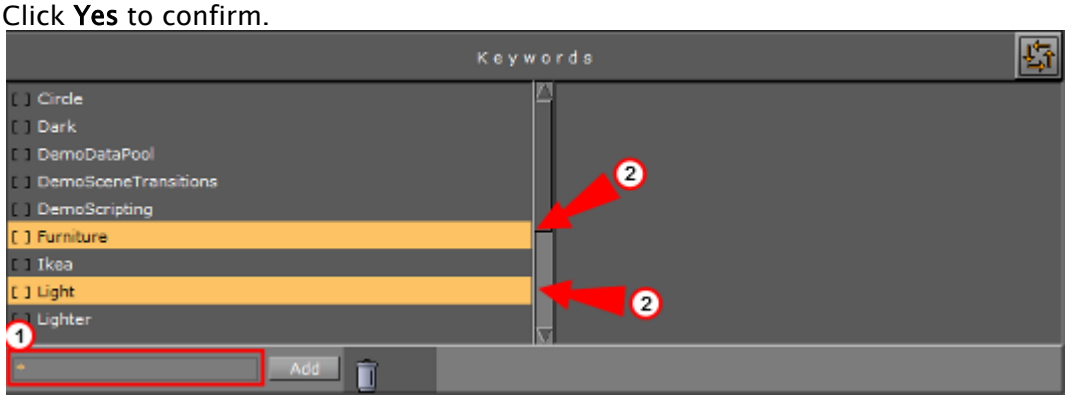
> **Tip:** Keywords can be assigned to multiple items at once. To do so, first select the items in the Item List, and then select the keywords in the Search window. Next, drag the keywords onto the items. In the dialog box that opens, confirm the assign operation.

4.7.3 To Assign Keywords from the Keyword Panel

> **Note:** This procedure assumes that there is at least one item in the Server Panel.

1. Select **Import** from the Main Menu.
2. In the Keywords panel search for a Keyword or Keywords (1):
   - To search for all Keywords, type * (asterisk) in the text box, and press **<Enter>**, or
   - To search for a specific Keyword, type the name of the Keyword in the text box, and press **<Enter>**.
3. Select a Keyword(s) (2).
4. Drag the selected keywords to the item or selected items in the Server Panel.

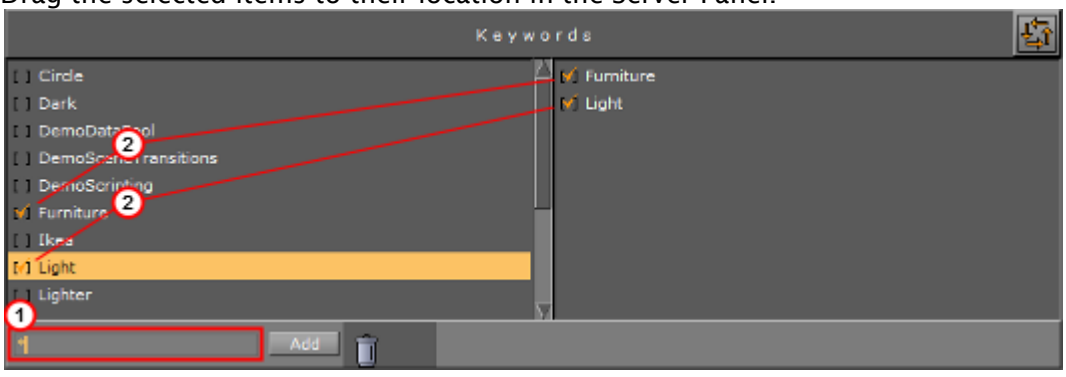
5. Click **Yes** to confirm.



> **Tip:** If the list of Keywords, in the left panel, is long, scroll and tick the required Keywords so that they show in the right panel. From the generated Keyword list in the right panel, highlight them all and drag to the item.

4.7.4 To Assign a Keyword While Importing

1. Select **Import** from the Main Menu.
2. In the Import panel, select the file or files to be imported.
3. Search for a Keyword or Keywords (1):
   - To search for all Keywords, type * (asterisk) in the text box, and press **<Enter>**, or
   - To search for a specific Keyword, type the name of the Keyword in the text box, and press **<Enter>**.
4. Select a Keyword or Keywords (2):
   - Click in the [ ] (to the left of each Keyword) to select a Keyword (the selected Keywords show in the right side panel), or
   - Click on one or more keywords and drag to the right side panel.
5. Drag the selected items to their location in the Server Panel.

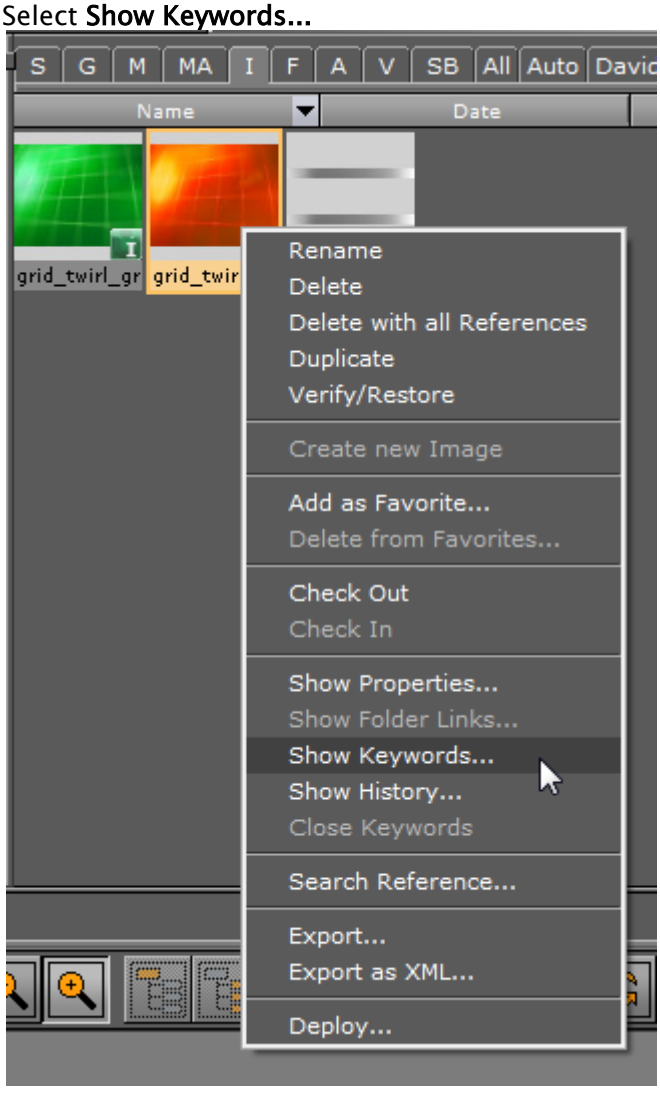


> **Note:** To view assigned Keywords on an item, right-click an item, in the Server Panel, and click on **Show Keywords...** from the context menu.

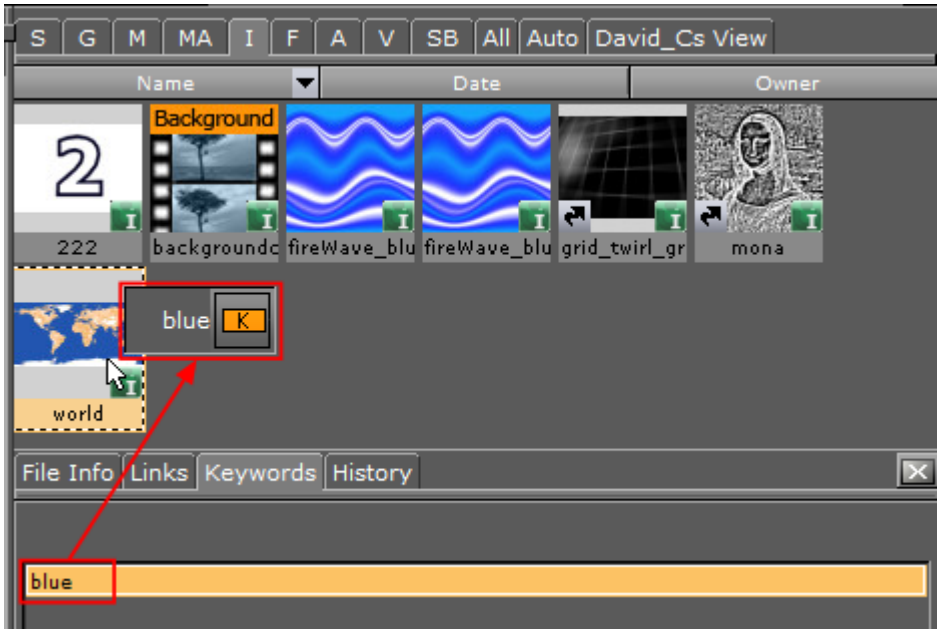
4.7.5 To Assign Keywords from the Property Editor

⚠ **Note:** The source and destination item must be in the same folder.

1. In the Server Panel, right-click the source item.
2. Select **Show Keywords...**

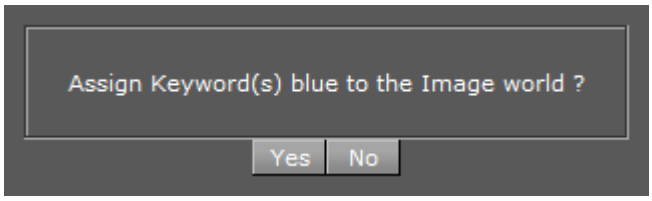


⚠ **IMPORTANT!** In the Keywords tab click the required Keywords, and drag to the required item in the Server Panel.



Note: When more than one source item is selected, and they all have a common keyword, only that common keyword will show in the Keywords panel.

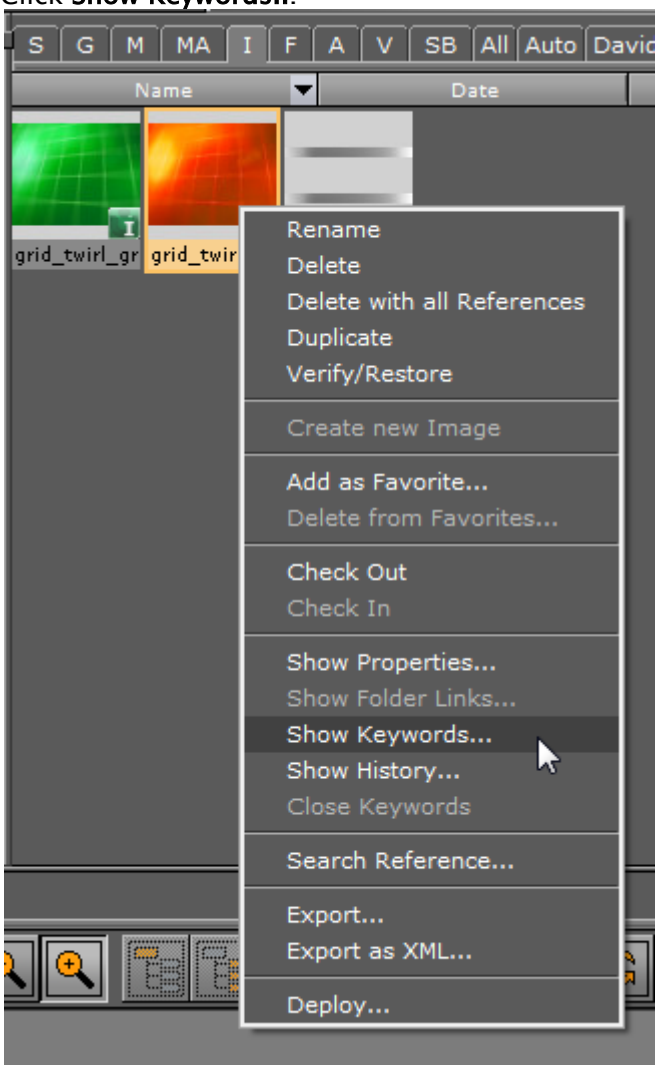
IMPORTANT! Click Yes to confirm.



4.7.6 To Remove a Keyword from an item

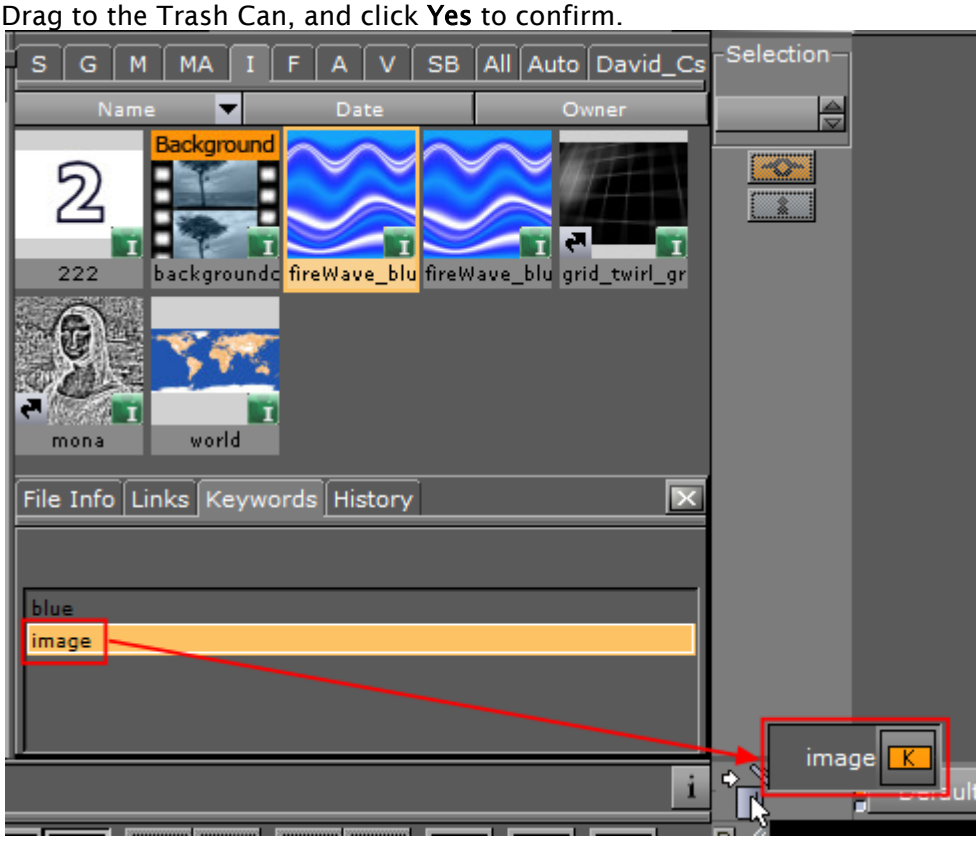
1. In the Server Panel, right-click the item with the Keyword to be deleted.

2. Click **Show Keywords...**



⚠ **IMPORTANT!** Click the Keyword(s) to be deleted.

3. Drag to the Trash Can, and click **Yes** to confirm.



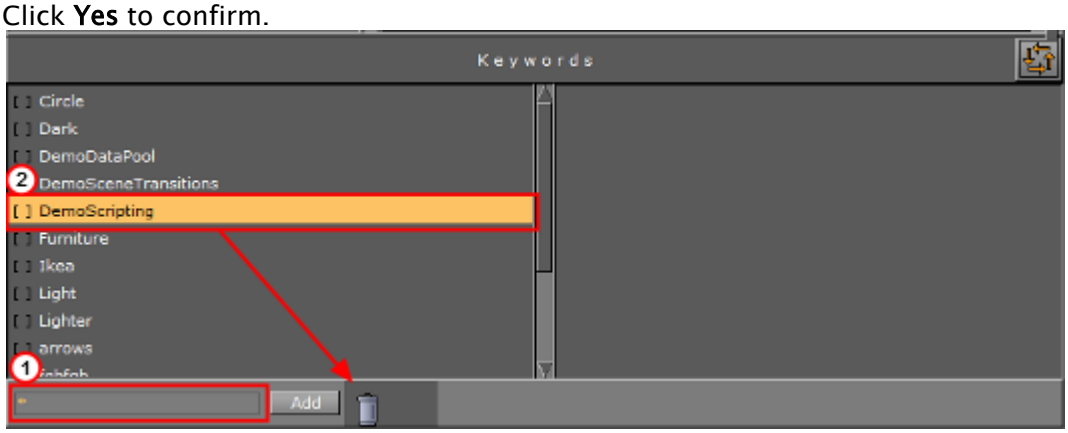
4.7.7 To Delete a Keyword

1. Select **Import** from the Main Menu.

   > ⚠ **Note:** The Keywords panel is not available if **Archives to Import** is selected in the Import file drop-down menu.

2. In the Keywords panel search for a Keyword(s) (1):
   - To search for all Keywords, type * (an asterisk) in the text box, and press <Enter>, or
   - To search for a specific Keyword, type the name of the Keyword in the text box, and press <Enter>.
3. Select a Keyword(s) (2), and drag to the Trash Can.

4. Click **Yes** to confirm.

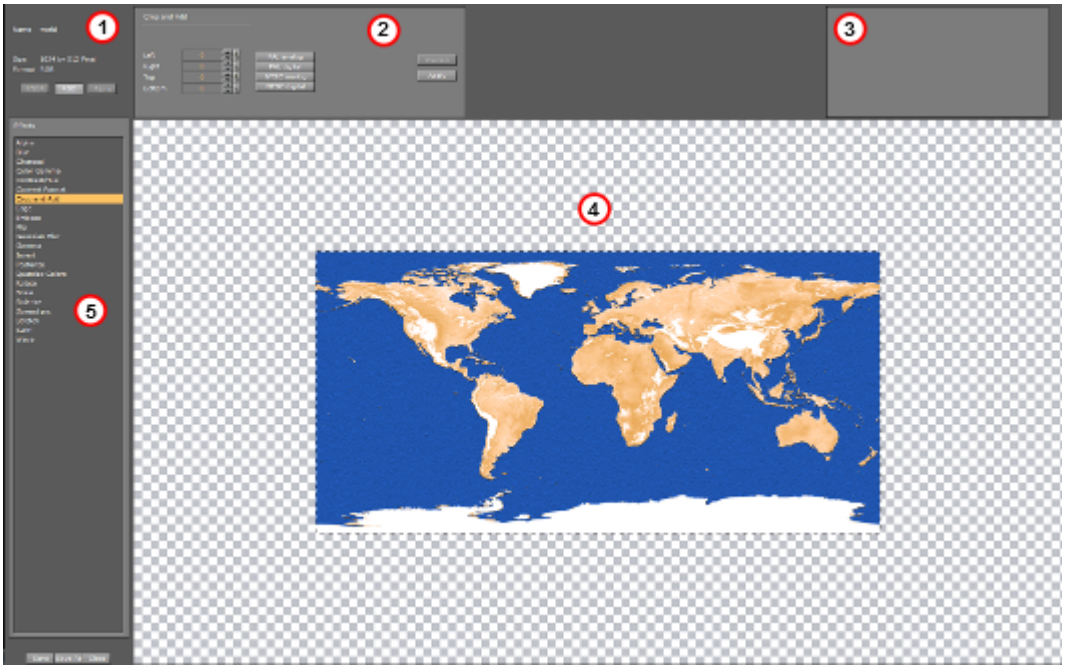


> **Note:** Keywords can only be removed from the Graphic Hub if they are not assigned to any files.

4.8 Image Editor

The Image Editor makes it possible to modify Image items in the Graphic Hub database. Individual editors are also available for Materials (Material Editor) and Fontstyles (Fontstyle Editor).

The modifications that can be done in the Image editor are similar to the ones in other image editing software.



The Image Editor consists of:

- Image General Information (1)
- Effect Modifications (2)
- Effect Preview (3)

- Main Image Area (4)
- Effects (5)

Double-click the image in the Server Panel to open the image in the Image editor. If the image is referenced in a Scene, a dialog box will show to confirm the action or not.

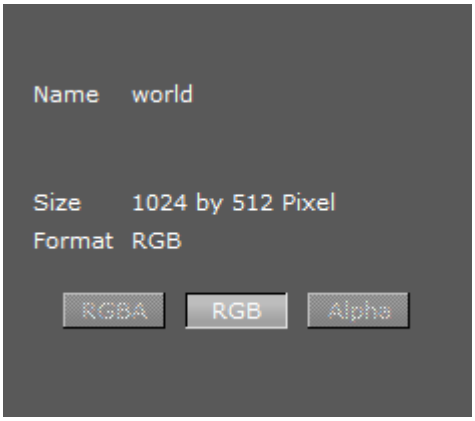
⚠ **Note:** Items that are opened in an editor, are locked in the database.

This section contains information on the following topics:

- General Image Information
- Image Effects
- Apply and Preview Effects
  - To Apply an Effect to an Image

4.8.1 General Image Information

The upper left corner of the Image editor shows general information about the image:

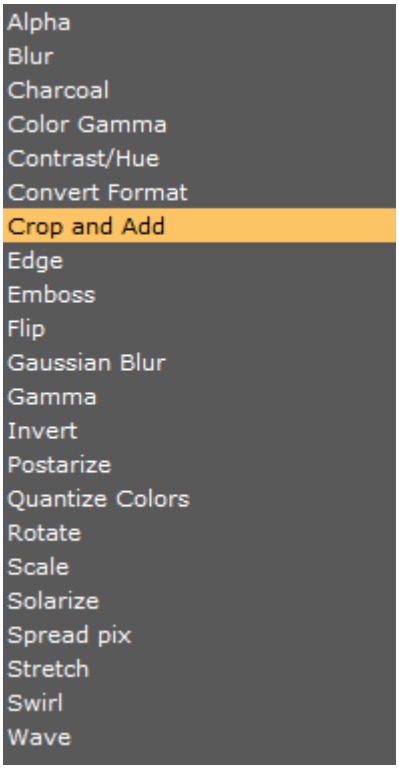


- **Name:** Shows the image name.
- **Size:** Shows the image size in pixels.
- **Format:** Shows the image format (RGBA, RGB, or Alpha).
  Click the format buttons to view the image various formats, if available.

⚠ **Note:** All images are treated as RGB or alpha format. An imported image that has alpha can be viewed as either RGB or alpha. Modifications to an image are automatically applied to the RGB or alpha information.

4.8.2 Image Effects

The following effects can be applied to an image. To view the effect properties and see changes made, see Apply and Preview Effects:

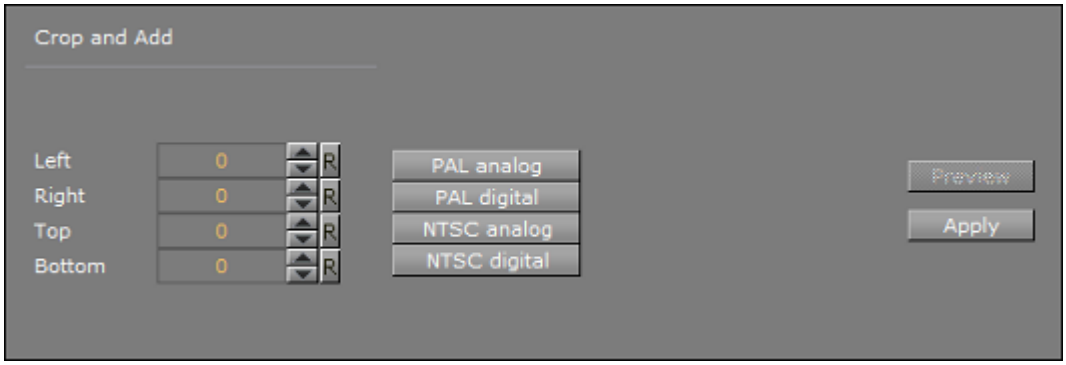


- **Alpha**: Makes the image transparent.

  ⚠ **Note:** This option only applies for alpha format images.

- **Blur**: Creates a blurred image.
- **Charcoal**: Creates a posterized, smudged effect by redrawing the image. Major edges are boldly drawn, while mid tones are sketched using a diagonal stroke.
  - **Radius:** The smaller the radius the more details will be seen.
  - **Sigma:** Affects the sharpness of the image.
- **Color Gamma**: Adjusts the color tones of the image.
- **Contrast/Hue**: Adjusts the brightness, saturation, and hue settings for the image.
- **Convert Format**: Changes the format of the image to one of the following:
  - **RGB**
  - **RGBA**
  - **Alpha**
  - **Luminance**
  - **Luminance Alpha**

    ⚠ **Note:** The current format is grayed out.



- **Crop and Add**: The image can be cropped according to the values in the input boxes. The values can be entered manually, or the standard TV format buttons can be used.
  - **Left**: Crops the image the specified number of pixels from the left side of the image border.
  - **Right**: Crops the image the specified number of pixels from the right side of the image border.
  - **Top**: Crops the image the specified number of pixels from the top of the image border.
  - **Bottom**: Crops the image the specified number of pixels from the bottom of the image border.
  - **PAL Analog**: Crops the image to a resolution of 768x576 pixels.
  - **PAL Digital**: Crops or adds the image to a resolution of 720x576 pixels.
  - **NTSC Analog**: Crops or adds the image to a resolution of 640x486 pixels.
  - **NTSC Digital**: Crops or adds the image to a resolution of 720x486 pixels.
- **Edge:** The Edge filter searches for borders between different colors and detects contours of objects:
  - **Radius:** A low value results in black, high-contrasted image with thin edges. A high value results in thick edges with low contrast and many colors in dark areas.
- **Emboss**: Makes the color at a given location of the filtered image correspond to the rate of color change at that location in the original image.
  - **Angle:** Defines the direction of the embossment.
  - **Value:** Defines amount of effect.
- **Flip:**
  - **Flip Vertical**: Flips the image upside down. To flip it back to its original settings, click the button again.
  - **Flip Horizontal**: Flips the image horizontally as if seen in reverse/through a mirror. To flip it back to its original settings, click the button again.
- **Gaussian Blur**: The visual effect of this blurring technique is a smooth blur resembling that of viewing the image through a translucent screen. It adds low frequency detail and can produce a hazy effect to the image.
  - **Range:** Affects the blur spread (not much visible since same effect may be achieved by configuring 'value' parameter).
  - **Sigma:** Amount of blur.
- **Gamma**: Gamma measures the brightness of mid-tone values produced by a device (often a monitor). A higher gamma value yields an overall darker image.

- **Invert**: Inverts the colors of an image. When inverting an image, the brightness value of each pixel in the channels is converted to the inverse value on the 256-step color values scale.

  ⚠ **Example:** A pixel in a positive image with a value of 255 is changed to 0, and a pixel with a value of 5 to 250.

- **Postarize**: Affects saturation of the image.
  - **Level (%):** Affects intensity. At high values, image becomes completely black and white.
- **Quantize Colors**: Reduces the number of colors in the image.
- **Rotate**: Rotates the image according to the specified number of degrees.
- **Scale**: Scales the image based on either size (percent) or width/height (pixels).
- **Solarize**: Blends a negative and positive image, similar to exposing a photographic print briefly to light during production.
- **Spread Pix:** Creates a dissolving effect for the image pixels.
- **Stretch**: Stretches the image based on either width/height in pixels or percent.
- **Swirl**: Swirls the image more sharply in the center than at the edges. Specifying the degree produces a swirl pattern.
- **Wave**: Creates an undulating pattern on the image, like ripples on the surface of a pond.
  - **Amplitude**: Defines the height of the waves.
  - **Wavelength**: Defines the width of the waves, the distance from one wave crest to the next.

4.8.3 Apply and Preview Effects

Various effects can be selected from the Image Effects. The selected effect can then be modified in the Effect Modification area (1) and the result viewed in the main image area, or previewed (2).

The Effect Modification area has two common buttons, which apply to most effect settings:

- **Preview**: Shows a preview of the changes in the image preview area (2).
- **Apply**: Applies the changes to the image.

  ⚠ **Note:** Some effect modifications apply directly to the image in the main image area.

To Apply an Effect to an Image

1. Select an effect from the Image Effects list.
2. Make modifications to the effect as required in the Effect Modification area. In most cases use the Preview area to view the effect on the image.
3. Add other effects as required.
4. Click Save or Save as...

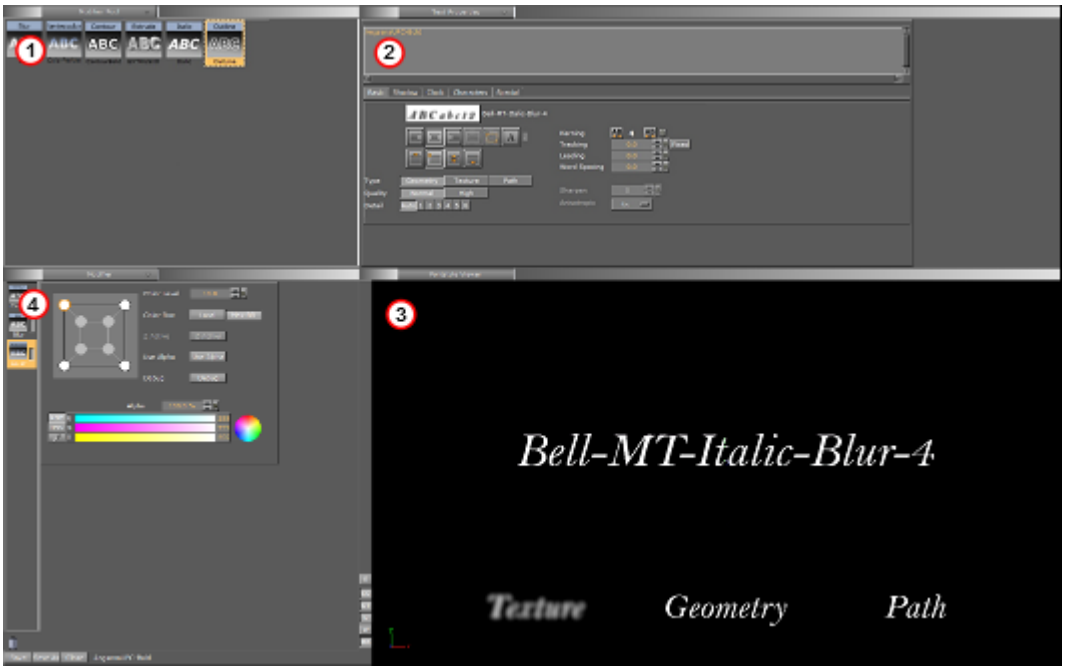
4.9 Fontstyle Editor

The Fontstyle Editor makes it possible to modify Fontstyle items in the Graphic Hub database. Individual editors are also available for Materials (Material Editor) and Images (Image Editor).

Viz Artist uses Fontstyles to show text in Scenes. When a Font is imported into the Graphic Hub database, Viz Artist imports the base font and creates up to eight different Fontstyles:

- One Fontstyle which looks like the base font (1)
- Up to four Fontstyles in different blur levels (2)
- Up to three Fontstyles using only the outline of the base font (3)
  The number of blur levels and outlines for each Fontstyle can be defined in the **Font/Text Options** section of Viz Configuration (see the Viz Engine Administrator Guide). The Outline and Blurred Fontstyles can be hidden or shown in the Server Panel. Fontstyles can be modified through the use of Modifiers in the Fontstyle Editor. A Fontstyle can be modified to reflect a corporate identity, for example.

As Viz Artist uses an open interface for the plug-ins used as modifiers, almost any modifications can be applied to a Fontstyle. New Fontstyles can be created based on existing Fontstyles through the **Save As...** operation.



The Fontstyle Editor consists of:

- Available Modifiers (1)
- Properties of previewed text (2)
- Fontstyle Viewer: Preview of the modifications (3)
- Modifier parameters (4) (see Working with Modifiers)

> ⚠ **Note:** The properties help to illustrate how the modifications will look like in a Scene. Changes made on the right side of the editor, for example kerning and material, will not be saved with the Fontstyle. However, there is one exception from this rule. If the kerning between two fixed characters are modified, this is saved with the Fontstyle. To create a kerning between two fixed characters, use the keyboard shortcuts <Alt+Left Arrow>/<Alt+Right Arrow>.

Double-click the Fontstyle in the Server Panel to open it in the Fontstyle Editor. If a Fontstyle is referenced in a Scene, a dialog box will show to confirm the action or not.

> ⚠ **Note:** An item that is opened in an editor, is locked in the database.

This section contains information on the following topics:

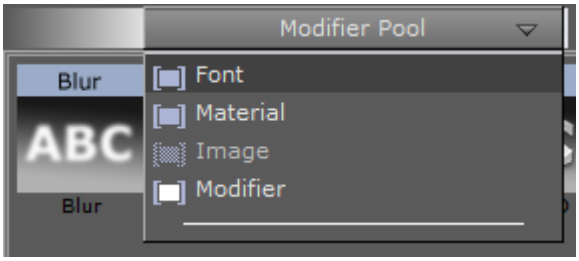
- Modifiers
- Working with Modifiers
- Preview Menu
- Fontstyle Viewer

4.9.1 Modifiers

Modifiers are used to change either the texture or the geometry of a Fontstyle, or both. A fixed set of modifiers come with a standard Viz Artist installation, but the system provides an open connection, so that additional modifiers can be obtained or created and added as plug-ins.

A modifier that is applied to a Fontstyle is automatically enabled. An enabled modifier implies that modifications will be reflected in the preview area. When adding multiple modifiers, the overview in the preview area can get confusing, and it may be necessary to disable one or more modifiers.

Available Modifiers

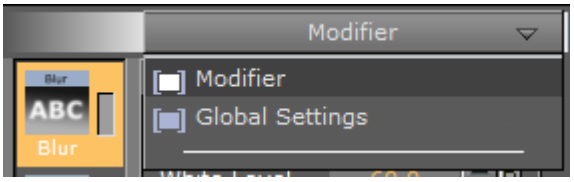


The available modifiers selected in from the Pool menu, which is positioned at the top left of the Fontstyle Editor.

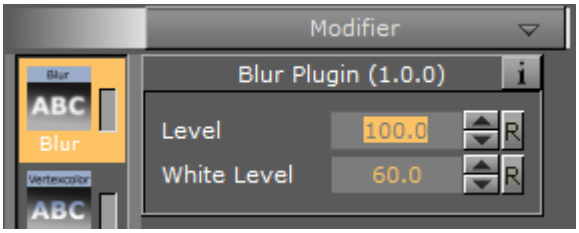
- **Font**: Shows all the Fontstyles in the database.

- **Material**: Shows all the Materials in the database.
- **Image**: Not available.
- **Modifier**: Shows all available modifiers that can be applied to the Fontstyle.

Modifier Parameters

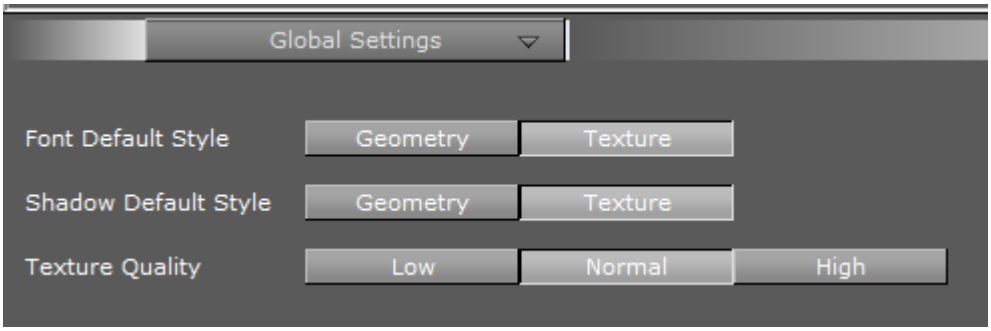


The Modifier parameters menu is positioned in the middle left part of the Fontstyle Editor.



- **Modifier**: Shows the properties for the modifier that is selected.

Global settings in Fontstyle Editor



- **Global Settings:** Shows various properties for the Fontstyle.
  - **Font Default Style**: Defines the font default style. Options are Geometry or Texture.
  - **Shadow Default Style:** Defines the shadow default style. Options are Geometry or Texture.
  - **Texture Quality**: Defines the quality of the texture. Options are Low, Normal, or High.

> ⚠ **Note:** The texture quality cannot be changed in a Text Editor.

4.9.2 Working with Modifiers

This section contains information on the following procedures and topics:

- To Add a Modifier to a Fontstyle

- To Edit a Modifier
- To Enable/Disable a Modifier
- To Remove a Modifier from a Fontstyle
- Blur
- Color Per Vertex
- Contour Bold
- Extrude
- Italic
- Outline

To Add a Modifier to a Fontstyle

1. Select **Modifier** from the Available Modifiers.
2. Drag the modifier from the list of available modifiers (top left) to the Modifier bar (bottom left, above the Trash Can).
3. Click the **Save** button.

To Edit a Modifier

1. Select the modifier in the Modifier bar (bottom left, above the Trash Can).
2. Define the values for the modifier.
3. Click the **Save** button.

To Enable/Disable a Modifier

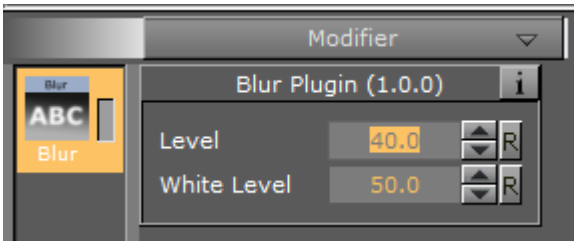
- Click the property switch next to the modifier icon in the Modifier bar.

To Remove a Modifier from a Fontstyle

1. Drag the modifier from the Modifier bar to the Trash Can.
2. Click the **Save** button.

Blur

Use the Blur modifier to blur a Fontstyle in texture mode.



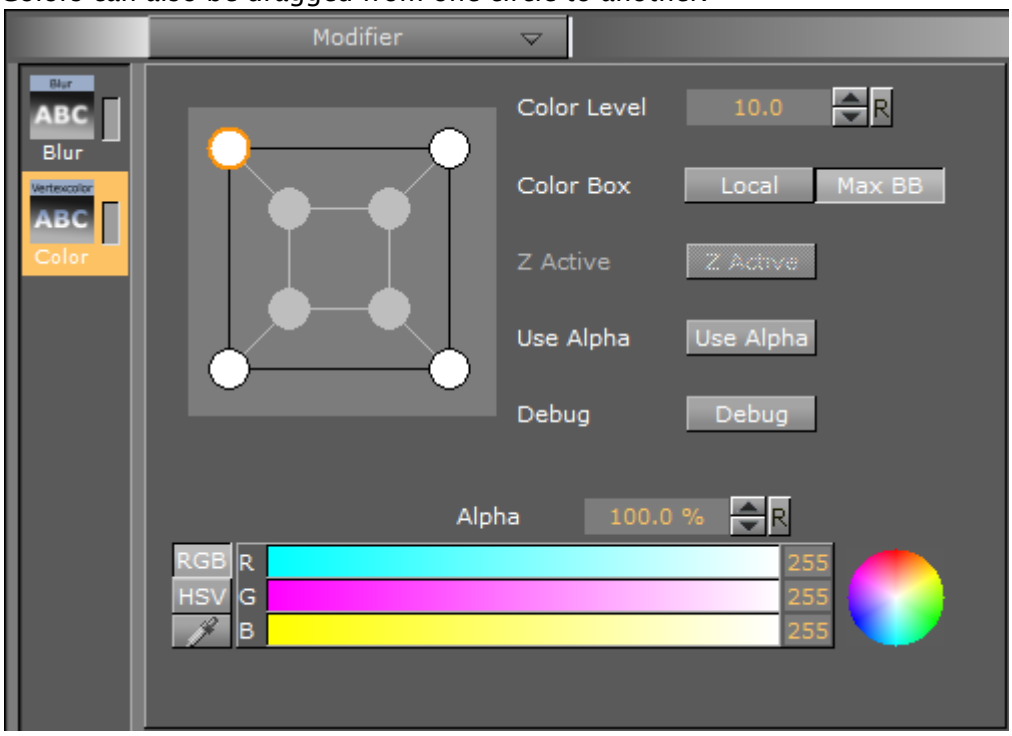
- **Level:** Sets the blur level
- **White Level:** Sets the white level in the blur effect

Color Per Vertex

Use the Color Per Vertex modifier to change the color of Fontstyles by applying it to the vertices.

The circles in the Color Per Vertex modifier represent the four corners of the Fontstyle text:

- Click a circle to edit its color values.
- Click on a line that connects two circles to select both circles.
- Hold <Ctrl> and click to select multiple circles.
- Colors can also be dragged from one circle to another.

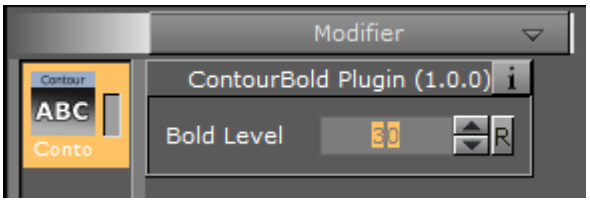


- **Color Level:** Defines how much color should be applied.
- **Color Box:**
  - **Local**: Applies the color to the vertices of every single character.
  - **Max BB**: Applies the color to the vertices of the bounding box.

- **Z Active**: If the Extrude modifier is used in addition to Color Per Vertex, the colors can be changed both in the front and back. If no such modifier is applied, only the colors in the front can be edited.
- **Use Alpha:** Shows where the color should include. For example if Alpha is enabled and the bottom right alpha is set to 0 the characters will fade out the bottom right.
- **Debug:** Draws the actual color points to see and understand how color changes effect the characters.

Contour Bold

Use the Contour Bold modifier to change the bold level of Fontstyles, which makes it show thicker or thinner.

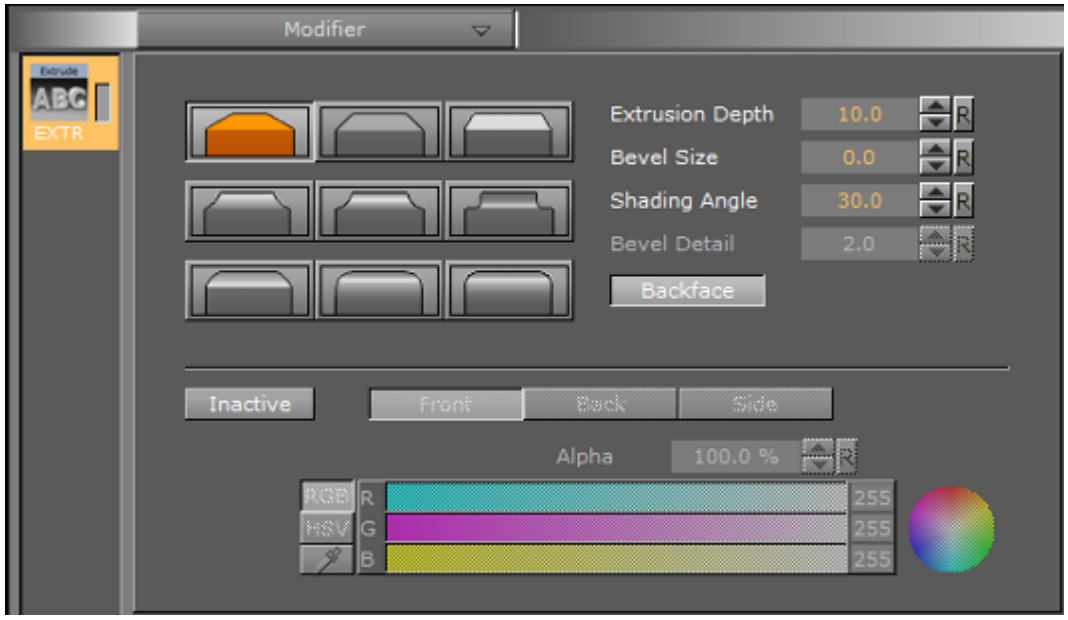


- **Bold Level:** Defines the bold level of a Fontstyle. The higher the value, the bolder the Fontstyle

Extrude

Use the Extrude modifier to extrude a 2D Fontstyle in the Z-axis, resulting in a 3D Fontstyle with front, back, and sides.

An extrusion can be explained simply as sweeping the 2D Fontstyle through space along its Z-axis. The sweeping path the outline of the Fontstyle follows during this process is used to create a surface.



The nine icons marked with different bevel types are used to set the type of bevel on the Fontstyle.

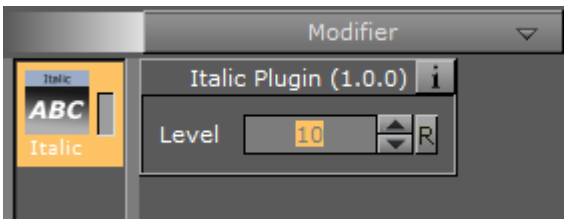
- **Extrusion Depth**: Defines the depth of the extrusion along the Z-axis.
- **Bevel Size**: Defines the size of the bevel.
- **Shading Angle**: Defines the angle of the shading function. To remove unevenness, increase the shading angle.
- **Bevel Detail**: Defines the degree of bevel detail. The lower the value, the more detailed the bevel is constructed and the smaller tilings it is constructed from.

> ⚠ **Note:** The Bevel Detail option is only available for certain bevel types.

- **Backface**: Enables or disables visualization of the back face.
- **Active/Inactive**: If colors are set to Active, the front, back, and side colors of the Fontstyle can be applied.

Italic

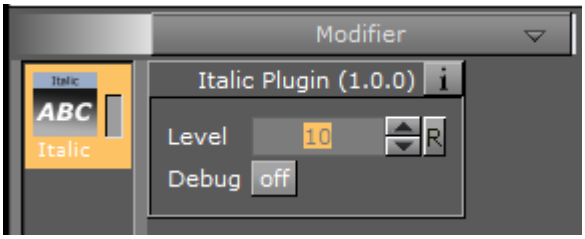
Use the Italic modifier to create a cursive appearance of a Fontstyle.



- **Level:** Defines the cursive level of a Fontstyle. The higher the value, the more the text is tilted.

Outline

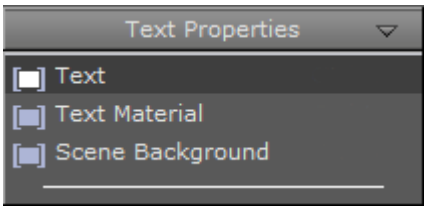
Use the Outline modifier to create an outline of a Fontstyle.



- **Outline Level**: Defines the width of the outline level. The higher the value, the bolder the outline of a Fontstyle.
- **Debug:** Draws the actual color points to see and understand how color changes effect the characters.

4.9.3 Preview Menu

The Preview menu is positioned at the top right of the Fontstyle Editor.



⚠ **Note:** Only the applied modifiers will be saved with the Fontstyle, not the settings described here. These settings are only used to give an overview of how the Fontstyle will look like when being used.

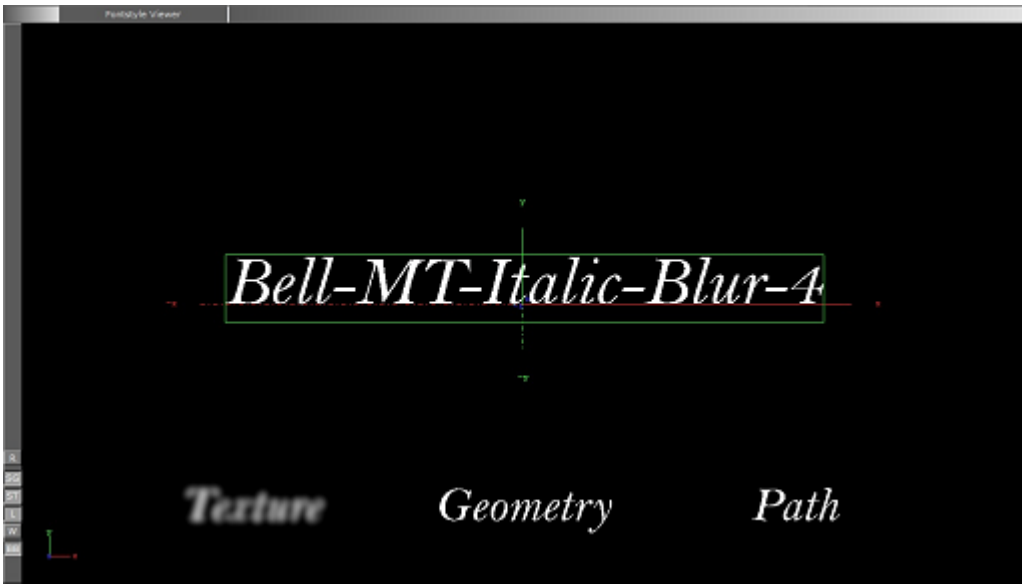
- **Text Properties**: To preview how the Fontstyle will look like with different settings, for example kerning, orientation, and content, select Text Properties from the Preview menu and modify the properties.
- **Text Material**: To preview how a Material would look if used on the Fontstyle, select Text Material from the Preview menu. In the Material Editor that opens, select a color.

> ⚠ **Tip:** To use a Material already from the database, select **Material** from the Available Modifiers in the Pool menu, select a Material and drag to the color icon in the Material Editor of the Fontstyle.

- **Scene Background**: To preview how a background color would look like if used together with the Fontstyle, select Scene Background from the Preview menu, and then select a color. The scene background feature provides a better contrast, especially for drop shadows.

4.9.4 Fontstyle Viewer

To visualize how the Fontstyle will look like when being used in Scenes, the Fontstyle Viewer shows a preview of the Fontstyle.



Additional to the customizable text in the Fontstyle viewer, the words Texture, Geometry and Path will be shown. These words show the modifications performed to Texture, Geometry and/or Path, applied by the modifiers, without reflecting the changes in the properties. For example, if kerning is applied to the customizable text, this will not be reflected in the Texture, Geometry or Path texts.

The customizable text reflects either the Texture, Geometry or Path text type, which is selected in the Global Settings, available from the Modifier Parameters.

Fontstyle Viewer Menu

Right-click the Fontstyle Viewer to open a menu, similar to the menu in the Scene Editor (see Working with the Scene Editor).

Fontstyle Viewer Bar

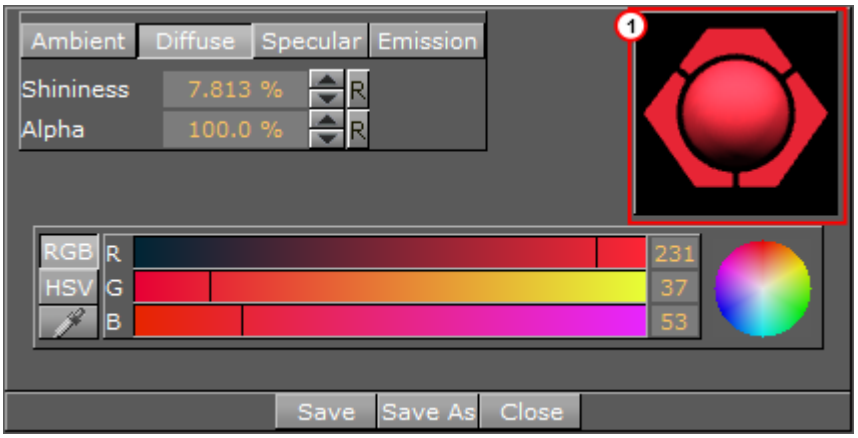
At the left side of the Fontstyle Viewer is the Fontstyle Viewer bar. The bar contains the following icons:

- R : Resets the position, rotation, and scaling for the three items in the Fontstyle viewer.
- SG : Shows/hides text with geometry style.
- ST : Shows/hides text with texture style.
- L : Enables/disables additional light.
- W : Switches on/off wire frame mode.
- BB : Switches on/off the bounding box visualization.

4.10 Material Editor

The Material editor is used to create and modify materials with customized colors and Lights effects.

The icon in the upper right corner of the Material Editor (1) shows the current state of the Material. The color for a lit object is shown on the sphere within the icon, while the three parts around the sphere show the plain color for an object that is not lit. For information about the various effects and types of lights that can be applied to a material, see Light Editor.



When a Material has been added to a Container, the Container-specific Material can be modified in the Material Editor that opens in the Properties panel. Only the Container-specific Material will be modified, the original Material in the Graphic Hub database is not affected (see Working with Material and Material Advanced Items).

This section contains information on the following topics:

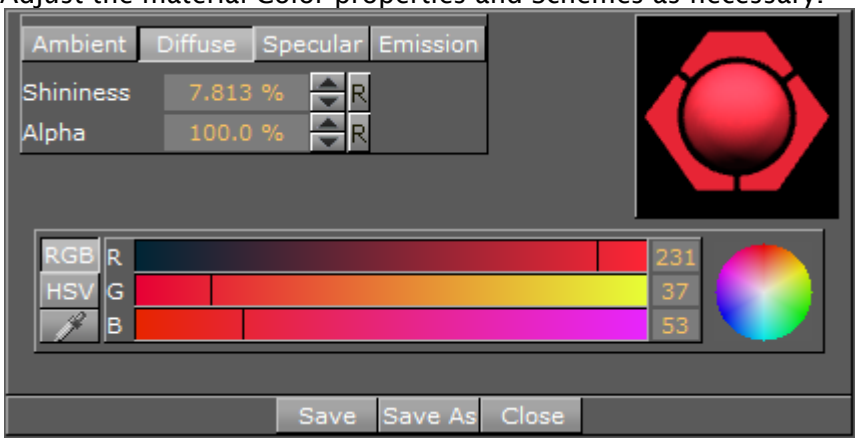
- To Create a New Material
- Material Color Properties and Schemes

4.10.1 To Create a New Material

1. Click **Server**.
2. Click the **M** (Material) tab.
3. Right-click and select **Create new Material**.
4. Enter a name for the new Material

5. Click **OK**.

6. In the Server panel, double-click the new Material to open the Material Editor.
7. Adjust the material Color properties and Schemes as necessary.



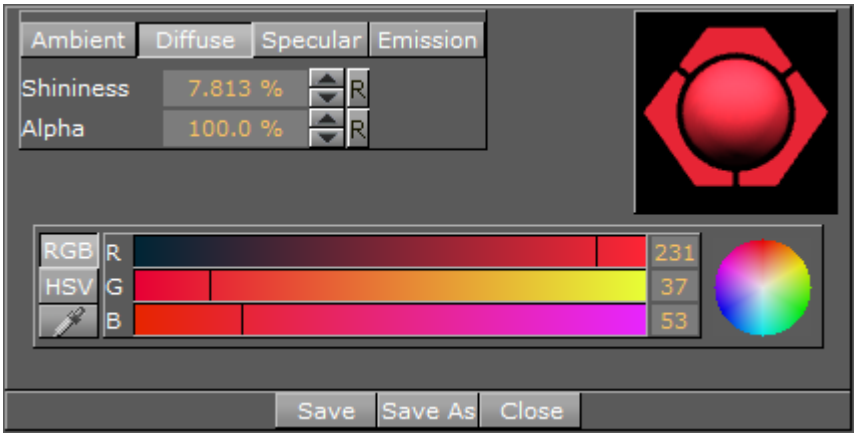
8. Save any changes:
   - **Save**: Saves the material in the Graphic Hub.
   - **Save As**: Creates a new Material in the Graphic Hub. In the Save As dialog box that opens, type a name for the Material and select a destination folder.
   - **Close**: Closes the Material editor without saving the changes.

4.10.2 Material Color Properties and Schemes

This section contains information on the following topics:

- Material Color Properties
- Material Base Color
- Material Enabled and Disabled
- To Change a Material Color

Material Color Properties

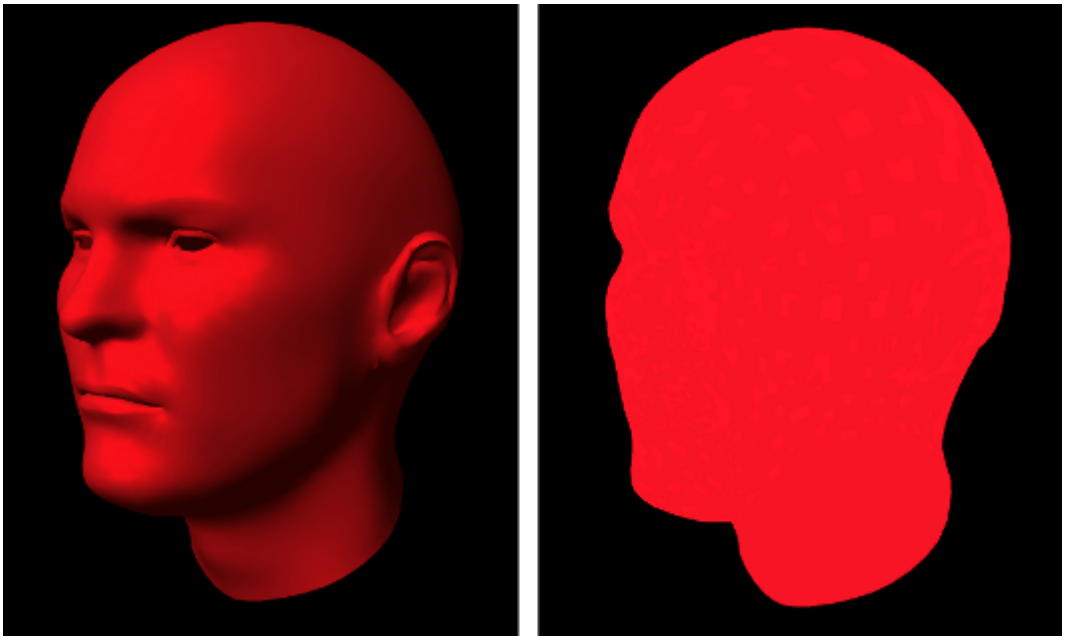


- **Shininess**: The shininess defines how spread out the highlighted area is. The lower the value, the more spread out the shine; the higher the value, the more focused the shine on a single point. However, if Specular is black (all values 0) then shininess does not have any visible effect.
- **Alpha**: The alpha value can be set directly on a material to adjust the level of transparency. Viz Artist uses schemes to describe the colors of a material.
- **Ambient**: The area which is not directly lit by any light source. Ambient light surrounds the object and the color looks the same on every side. Since the light does not come from any direction, but instead from “all” directions, the color/light spreads evenly over the object.
- **Diffuse**: The object’s “core color” observed under “normal” lightening conditions. Shows the color of an object when illuminated by a directional light source. It spreads evenly over the object. Light normally has some level of diffusion.
- **Specular**: Shows the color that shows when light comes from a particular direction and bounces back from the surface in a mirrored direction. The specular color is in fact the direct reflection of a directional light source. This is the typical effect of shiny metal and plastic. Specular color can be understood as a kind of shininess color.
- **Emission**: Emissive color is produced by an object’s natural glow. The object itself is the light source illuminating the color. The emissive color is unaffected by any other light sources and does not produce any light for other objects in the scene.

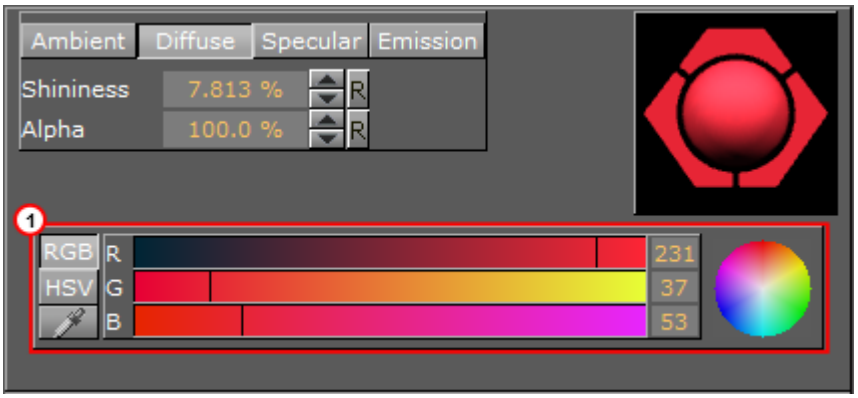
Material Base Color

If the Material icon in the Property area (1) is disabled it causes the object’s base color to be shown. This is a color without any shading and it is not influenced by light, which means that it is visible even if there are no active light sources. Its initial color is created through a conversion of the material RGB components, but the values can be modified afterwards.

Material Enabled and Disabled



To Change a Material Color



1. In the color selection area (1) select RGB or HSV.
2. In the color selection area, click a color bar, or the color circle, and drag to change the color.

> ⚠ **Tip:** For RGB only: Right-click on the color sliders and drag to lock all three sliders, so that all values can be changed at once.

or

1. Click on the eye dropper icon in the color field.
2. Move the cursor to pick a color from anywhere on the screen.
3. Click to accept the new color.

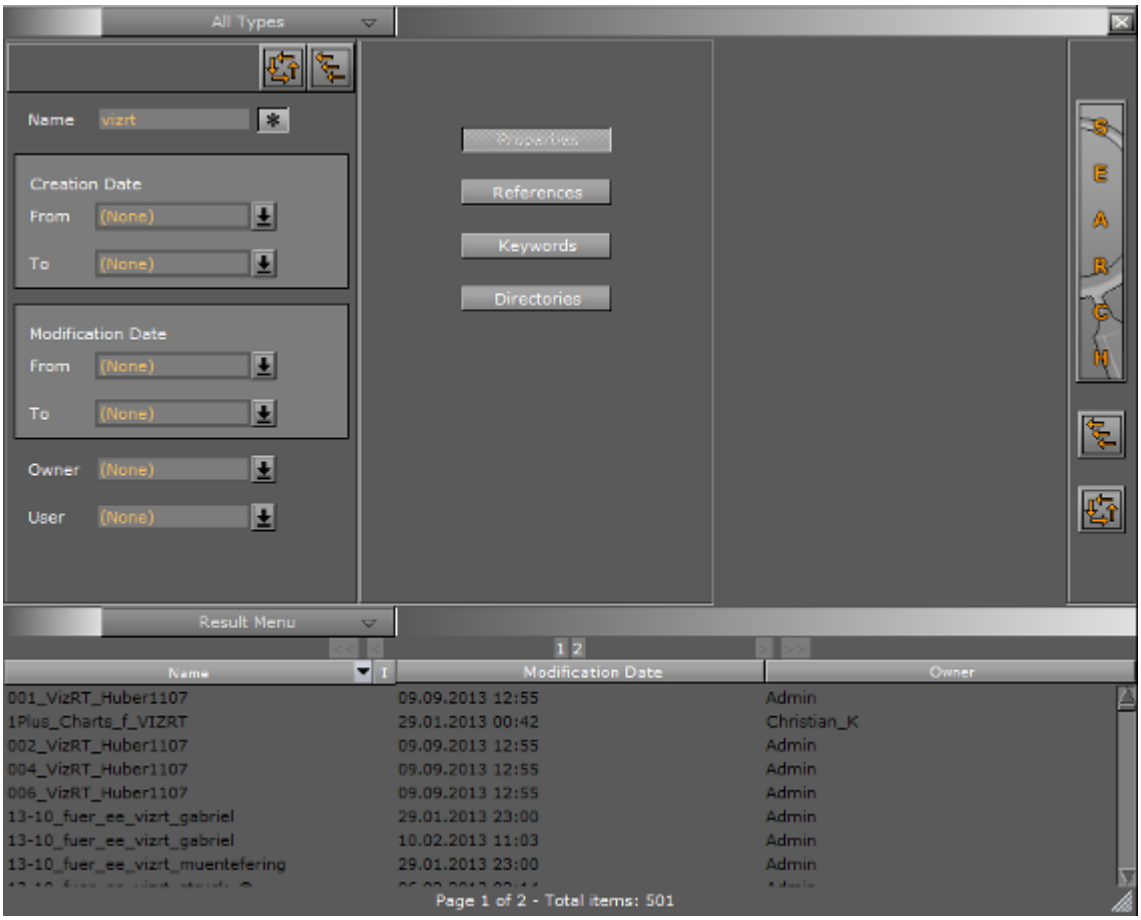
or

- Click a numeric input box and enter a color numeric value.

> ⚠ **Tip:** Click and hold in the input box and drag left or right to adjust the value.

> ⚠ **Tip:** Click in the input box and use the up and down arrow keys to adjust the value by a factor of 1. Press and hold the <Alt> key and use the up and down arrow keys to adjust the value by a factor of 10.

4.11 Item Search



Items can be searched based on various criteria. Four Search editors are located in the upper part of the Search window. Searches can be based on four different criteria:

- Property Search
- Reference Search
- Keyword Search
- Directory Search

> ⚠ **Tip:** Up to three search criteria can be combined in one search request. The search will be performed using a logical AND of the selected criteria.

- Combination Search

For information on keywords, see Assign Keywords to Items.

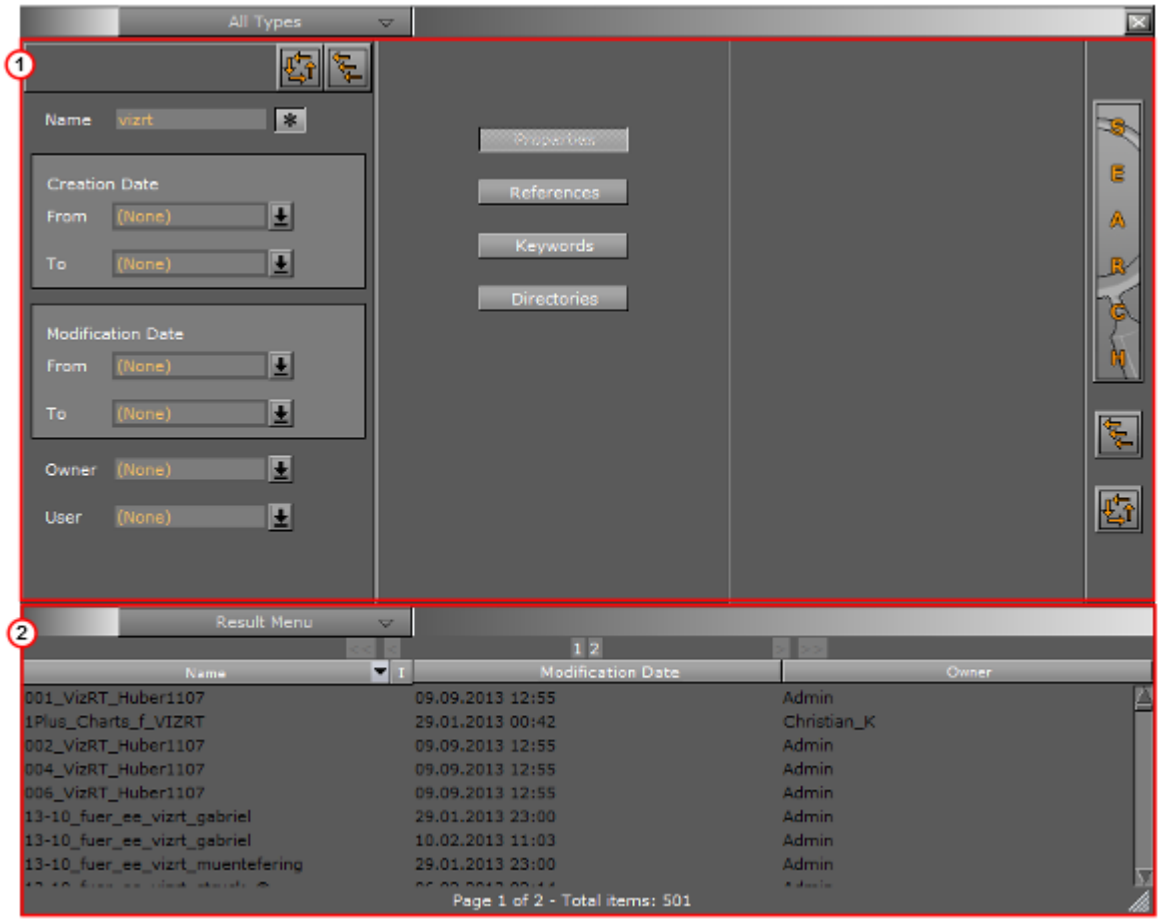
This section contains information on the following topics:

- Search Panel
- Search Parameters Area
- Search Menu

- Property Search
- Reference Search
- Keyword Search
- Directory Search
- Combination Search
- Search Result Panel

4.11.1 Search Panel

To open the Search panel, switch Server view and select **Search** from main menu, or click `Ctrl+F` while hovering the mouse over the Server Panel. The search panel will stay in the foreground until it is closed.

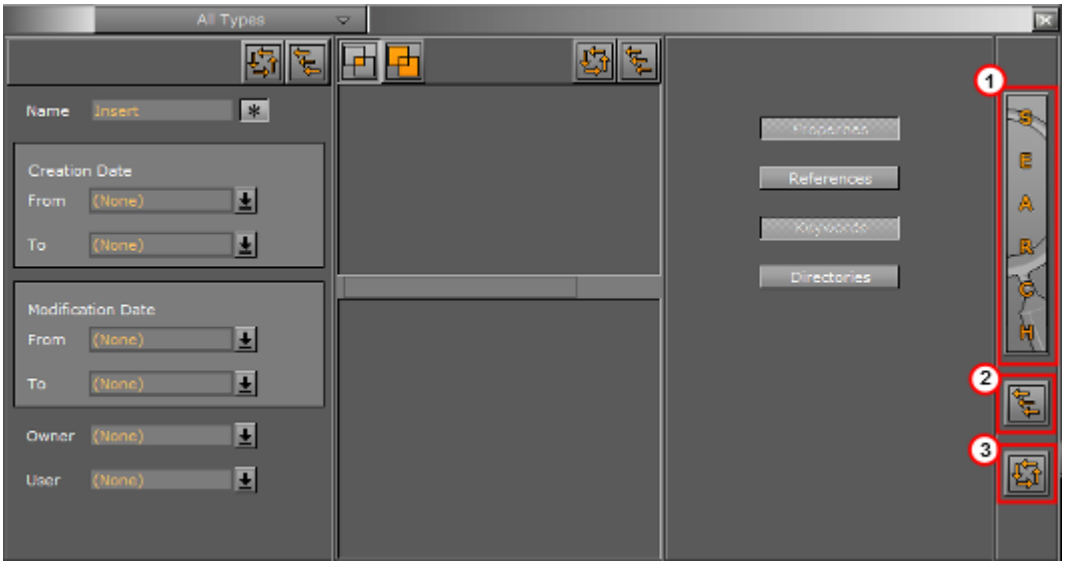


The Search window is divided in two areas:

- Search Parameters Area (1)
- Search Result Panel (2)

4.11.2 Search Parameters Area

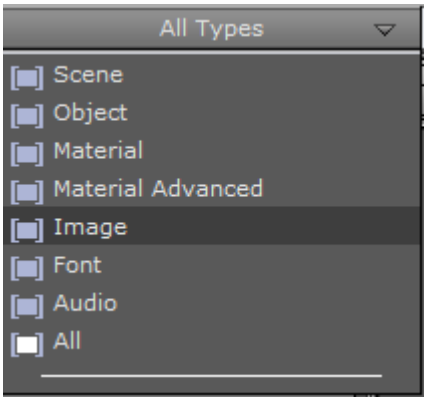
The Search parameters area is positioned at the top of the Search panel.



- **Search (1)**: Starts a search.
- **Close All Editors (2)**: Closes all search editors.
- **New Search (3)**: Clears all settings to start a new search.

4.11.3 Search Menu

The Search menu is positioned at the top of the Search parameters area. Select an item type from the menu to narrow the item search.



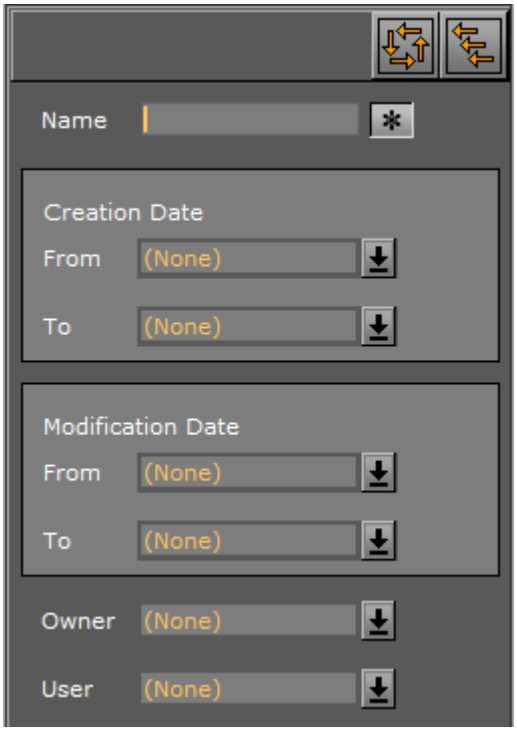
If, for example, 'Image' is selected, only search hits that are Images will be listed.

Clicking 'All' on the menu will not narrow down the search, but show hits from all item types.

'All' is selected by default.

4.11.4 Property Search

This editor searches the database for items based on the property values entered.



- **Properties Reset:** Clears the settings in the Properties editor to start a new search.
- **Close Property Search:** Closes the Properties editor.
- **Name**: Searches for items with a specified name.
  - Click the asterisk to search with an asterisk wildcard at the start and end of the text string. With the asterisk disabled, only the exact search string is used.
- **Creation Date**: Searches for items that were created in a specified time period.
- **Modification Date**: Searches for items that were created in a specified time period.

> ⚠ **Note:** The creation and modification dates can be set to; Today, 1 day, 1 week, 2 weeks, 1 month, or Older.

- **Owner**: Searches for items that were created by a specified user.
- **User**: Searches for items that were last modified by a specified user.

Wildcard Text Description

If the full Name is not known then text Wildcards can be used to make a search:

- ***: Match any characters. For example '*item'
- **?**: Match any single character. For example 'El?me?t'
- **[ ]**: Match a range or a list of characters:
  - 'Elem[b-g]nt', or
  - 'Elem[shje]nt'.

4.11.5 Reference Search

Any item can reference to any other item in the Graphic Hub. To find out where any given Graphic Hub item is used, use search by Reference.

Search by reference can be used to search for:

- Which objects use the selected object(s)
- Which objects are used in the selected object(s)

This section contains information on the following topics and procedures:

- Reference Search Menu
- To Search for Referenced items on one or more items
- To Search all References on one item (Context Menu)
- To Delete an item from a Search by Reference

Reference Search Menu

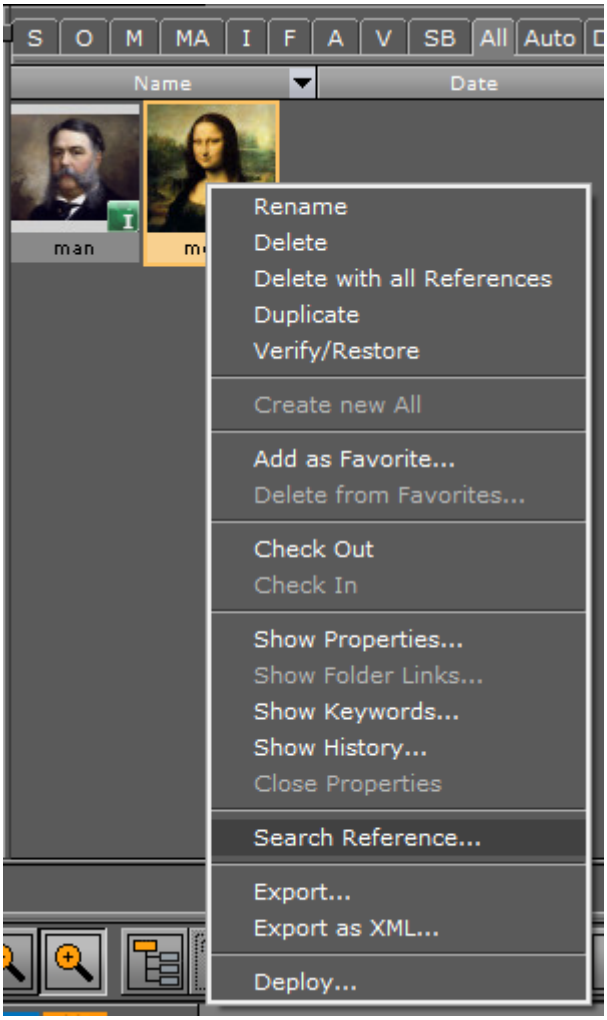
Note: Multiple items can be added to the References editor.

-  Searches for items which reference to the selected items, for example, Images, Geometries, Fonts, etc.
-  Searches for items which are referenced by the selected items, for example, Images, Geometries, Fonts, etc.
-  Performs a logical AND search of the items that are added. The result must be in all searched items.
-  Performs a logical OR search of the items that are added. The result must be at least one of the searched items.
-  Clears the settings in the References editor, so that a new search can be performed.
-  Closes the References editor.

To Search for Referenced items on one or more items

1. Open the Search panel:
   - Click on **Search...** in the Server Panel drop down menu, or
   - With the mouse pointer over the Server Panel, press <Ctrl+F>
2. Click on **References**.
3. Drag one or more items from the Server Panel to the search area.
4. Specify the search criteria:
   - Search only for references common to all items 
   - Search for references in all items 
5. Select to search by:
   - Which objects use the selected object(s) 
   - Which objects are used in the selected object(s) 
6. Click on the Search button (3).

To Search all References on one item (Context Menu)

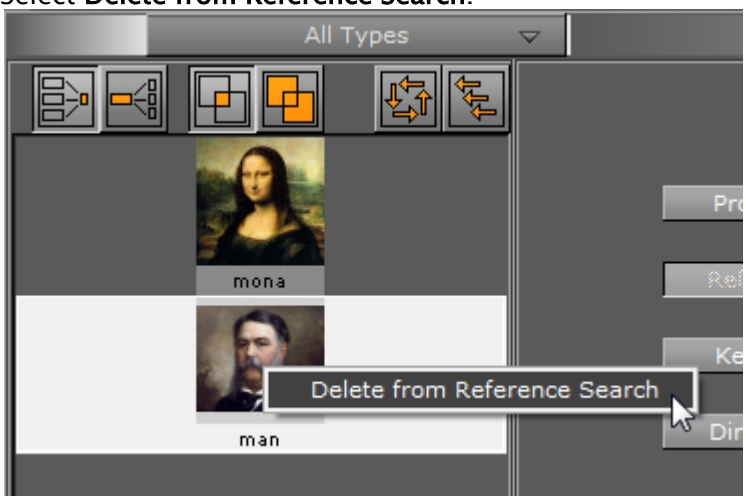


1. Click on an item.
2. Right-click the item to open its context menu.
3. Click **Search Reference...**
4. The Search Panel opens and shows any items which reference the selected item.

To Delete an item from a Search by Reference

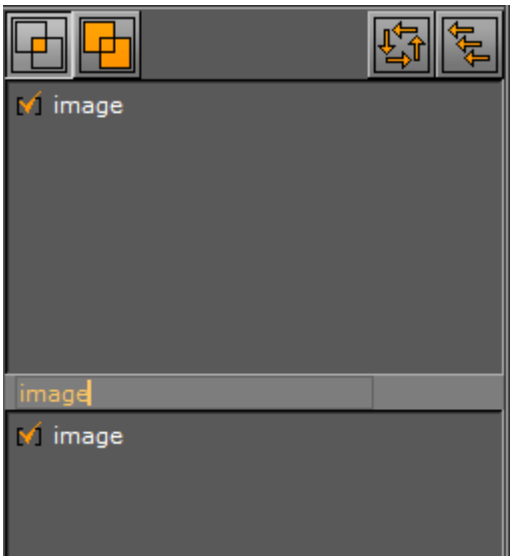
1. In the Search Panel right-click the item to be removed.

2. Select **Delete from Reference Search**.



4.11.6 Keyword Search

This editor searches the database for items based on keywords.



-  : Does a logical AND search of the keywords that are added. The result must be in all searched items.
-  : Does a logical OR search of the keywords that are added. The result must be in at least one of the searched items.
-  : Clears the settings in the Keywords editor, to starts a new search.
-  : Closes the Keywords editor.

> **Tip:** Keywords can be assigned to items from the Search window. To do so, select the keywords by highlighting them, and then drag them onto an item in the Server Panel. For more information, see Working with Items.

To Search for a Keyword

1. Type the keyword in the input box and press <Enter>.

2. Add the required Keyword to the bottom panel:
   - Click and drag the required keyword to the bottom panel, or
   - Click inside the brackets ([ ]), to the left of the required keyword.
3. Click the Search button. The keyword search is based on the keywords that have been moved into the bottom panel.

4.11.7 Directory Search

The directory search editor narrows down a search to a selected folder.

-  : Expands the currently selected branch and all its sub-branches.
-  : Collapses the currently selected branch and all its sub-branches.
-  : Refreshes the tree.
-  : Searches for items in the selected folder, and recursively in all sub-folders. When disabled, the search is performed in the selected folder only.
-  : Closes the Directories editor.

To Search for items in Directories

1. Click **Directories**.
2. Select a folder from the tree.
3. Click **Search**.

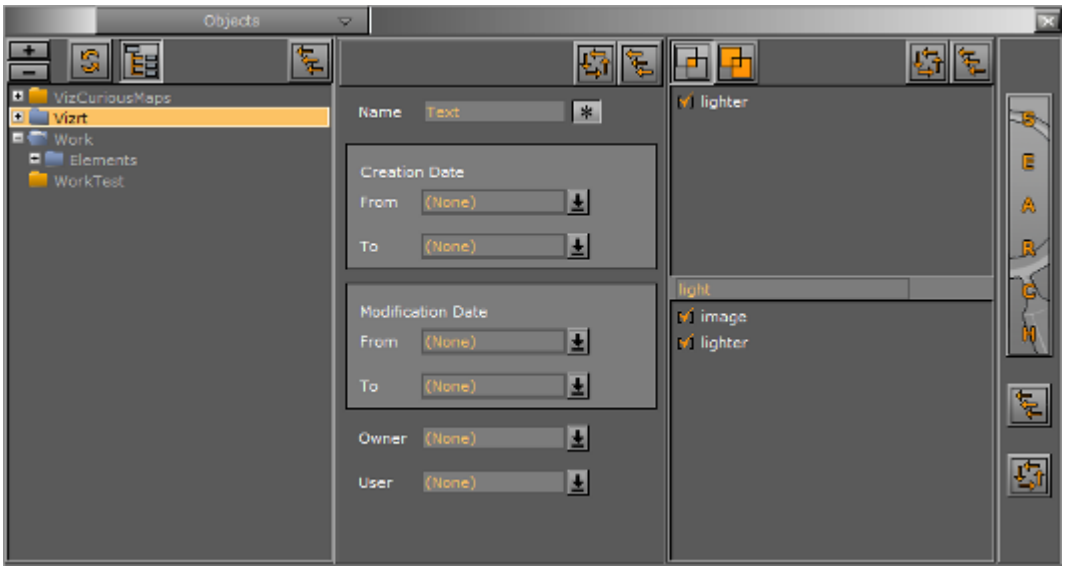
> ⚠ **Note:** If no other search criteria has been entered, when the Search button is clicked the search will show all items in the selected folder (including sub-folders).

4.11.8 Combination Search

With a Combination search, searches can be further refined. A combination search uses up to three search parameters to search the Graphic Hub database. Select from:

- Property Search
- Reference Search
- Keyword Search
- Directory Search

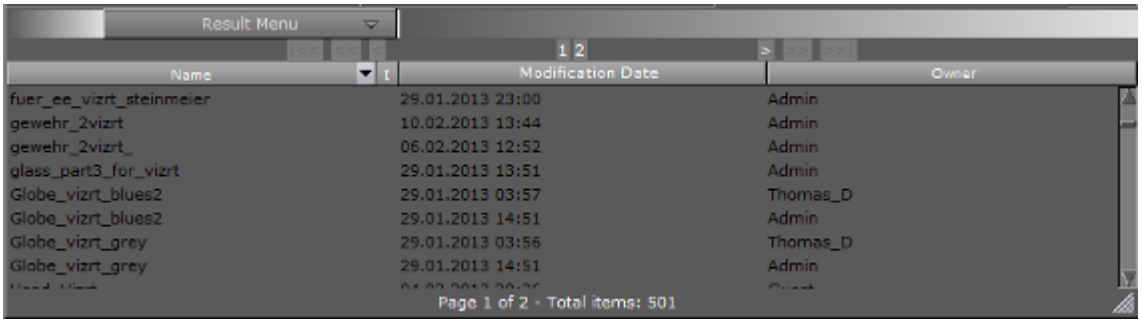
  Click on each required search parameter in the Search Panel.



In the example above, the search will be for files that contain the word **Text** which is in the **Vizrt** project folder and sub-folders, and also has a keywords **image** and **lighter** assigned.

4.11.9 Search Result Panel

The Search Result panel shows all search results.



⚠ **Note:** A double-click an item in the Result area will also select the item in the Server Panel.

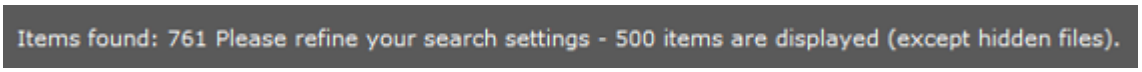
This section contains the following topics and procedures:

- Search Results
- Search Results Navigation
- Order Search Results
- To Use Searched items
- To Copy items
- To Create a Folder Link
- Search Result Menu

Search Results

The amount of search results to show is set in the **Database -> Global** section of Viz configuration. The default value is 500, and it can be changed to a specific number or set to All. Note that setting to **All** can produce a large number of search results, which may affect performance. The Search Results panel displays up to 500 search results per page.

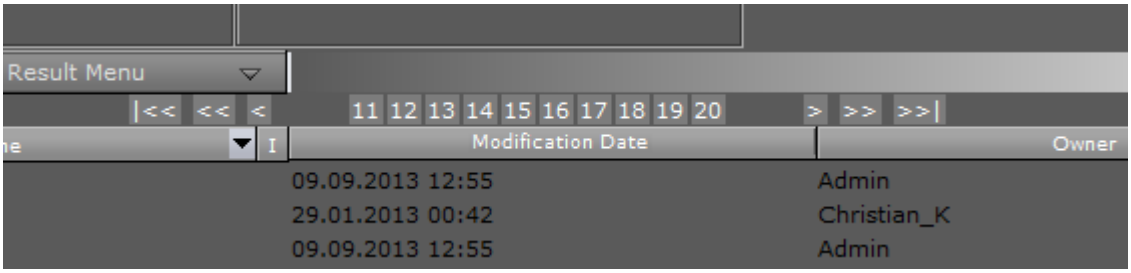
If a set amount of results is configured to show (but not **All**), this message will show:



In this example, only 500 items are displayed in the results panel (500 per page). If there is a large amount of search results, the search results panel will show a text list of results. The number of **Items Found** can differ from the shown number of files. This is because hidden files , such as base fonts, are also counted in the number of results. However, hidden files are not shown.

Search Results Navigation

When a search result is more than 500, pages are created for each set of 500 results. Use the navigation buttons to go backward and forward through the pages as required:

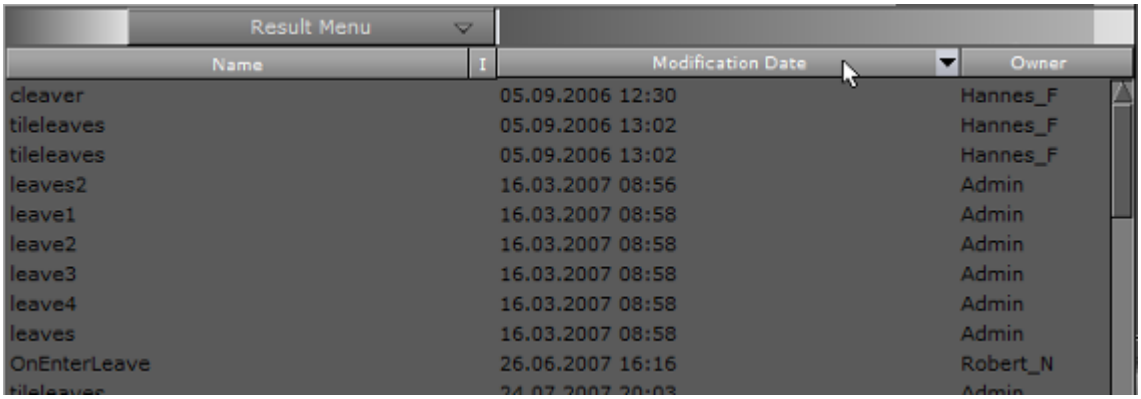


- 1 2 3 4 5 6 7 8 9 10 : Click on a number to go to that page
- < > : Go back or forward one page
- << >> : Go back or forward 10 pages (maximum 10 pages shown at any time)
- |<< >>| : Go to the first or last page

Order Search Results

Use the column header bars to order the search contents as required. Click on each column header bar to order its results ascending or descending. The small arrow shows which way the search results are ordered.

The middle column can be set to show the **Modification Date** or the **Creation Date** (see **Global** in **Database** of **Viz Configuration** (see the Viz Engine Administrator Guide)).



To Use Searched items

⚠ **Note:** To select more than one item press and hold <Ctrl> and click on each required item.

Click on an item(s) in the Results panel and drag directly to various areas where the item can be used, i.e.:

- The Scene Tree, or
- The Favorites Bar , or
- The Scene Editor, or
- The Container Editor

⚠ **IMPORTANT!** Be careful when moving a linked item, as the item will be moved from the first folder found to the new destination.

To Copy items

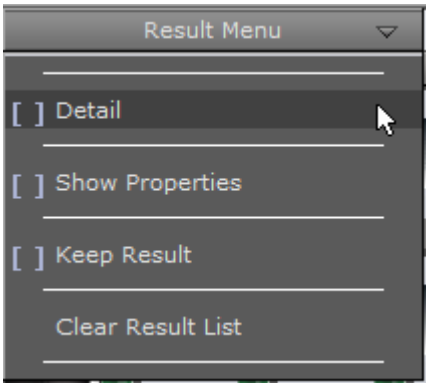
1. Click on an item(s) in the Results panel
2. Press <Ctrl> and drag the item(s) from the Result area to the target folder in the Server Panel.

To Create a Folder Link

1. Select an item in the Results panel.
2. Press <Alt> and drag the item(s) from the Result area to the target folder in the Server Panel.

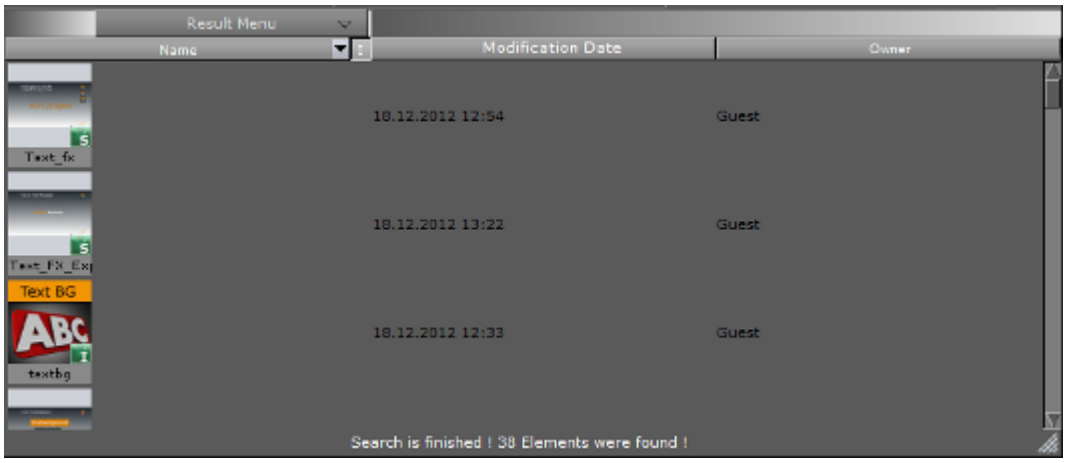
Search Result Menu

The Result area menu is positioned at the top of the Result area and it defines the appearance of the search results.



- **Detail**: By default, items are presented in the Result list with thumbnail and item name. A vertical list shows, with the additional information of date of creation and user name of item owner.

  > ⚠ **Note:** Click on I to enable or disable thumbnails.



- **Show Properties**: Shows the Properties Panel at the right side of the Result area, with the File Info tab selected.
- **Keep Result**: Keeps the current results in the Result list, while a new search is done. The results from the new search will be added to the list.
- **Clear Result List**: Clears the Result list.

-
  - Search Results
  - Search Results Navigation
  - Order Search Results
  - To Use Searched items
  - To Copy items
  - To Create a Folder Link
  - Search Result Menu

4.12 Free Text Search

Free Text Search allows for searching for any objects in the Graphic Hub using an arbitrary Search String.

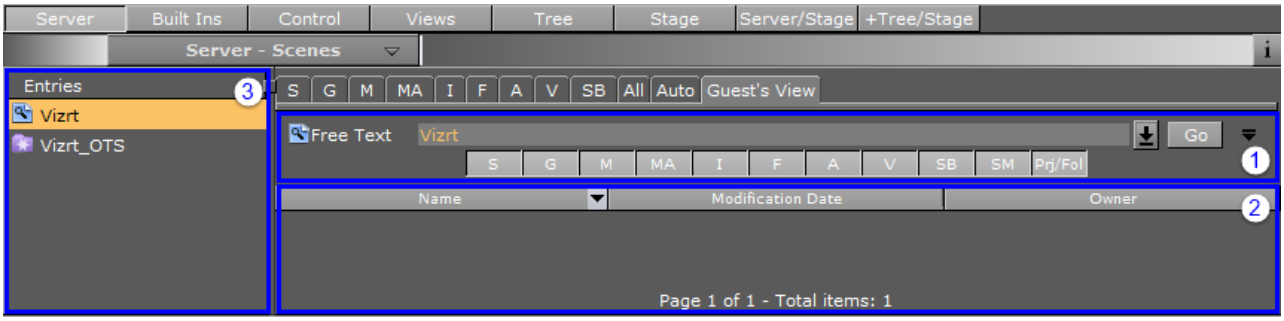
- Search Pane
- Result Pane
- Entries Pane
- Advanced Free Text Search
- Wildcard text description

The result is a set of database objects, called Files, that match the search string. The search string is matched against most properties of a database object. These properties are:

- File names
- Folder and project names
- File owners
- File UUID (if enabled in the Graphic Hub)
- File type (Geometry, Fontstyle, Images, etc.)
- Assigned keywords

If, for example, the Search String matches the owner of a scene, then that scene will be part of the search result.

The Free Text Search consist of three panes:



1. Search view to specify the search.
2. Result view to display the search results.
3. Entries view to store searches and results for later reuse.

4.12.1 Search Pane

Enter a search string into the **Free Text** field and press the **Go** button to start the search. The **Expand** button shows or hides the type selection buttons.

These buttons allow for filtering the result for specific File types.

Options	Options (continued)
· **S**: Scenes · **G**: Geometries · **M**: Materials · **MA**: Materials Advanced · **I**: Images	· **F**: Fonts · **A**: Audio · **V**: Video Clips · **SB**: Substances · **SM**: Shared Memory Map · **Prj/Fol**: Projects and Folders

The Type Selection buttons can also be quick selected. Click on a button with:

- **<Ctrl+Click>:** Select all.
- **<Shift+Click>:** Deselect all.
- **<Alt+Click>:** Only select the clicked button.

To view previous searches, click the arrow to the right of the **Free Text** field to view all free text searches since Viz Artist was launched.

Searching for UUIDs and searching with wildcards is described in Advanced Free Text Search and Wildcard text description, respectively.

4.12.2 Result Pane

The Result view lists the Files found by the search.

- Files can be dragged directly to where they are needed.
- Double-click on folders to view sub-folders and contents.
- Right-click an object and select **Jump to Server View** to display the file in its home directory.

Use the column header bars to order the search contents as required. Click on each column header bar to order its results ascending or descending. The small arrow shows which way the search results are ordered.

The middle column can be set to show the **Modification Date** or the **Creation Date** (see **Global** in **Database** of **Viz Configuration** in the Viz Engine Administrator Guide).

4.12.3 Entries Pane

In the Entries view Search Strings can be saved for reuse and Search results can be saved in File Sets. File Sets are a collection of objects saved under an assigned name. File Sets are unique to each user, on each instance of the Graphic Hub. Placing objects and folders in a File Set does not move them from their location in the Graphic Hub, it just makes them all accessible from the Free Text tab.

You can save a Free Text search string by draging the **Free Text** icon into the **Entries** pane. To reuse a saved Free Text search string, click on the saved search in the **Entries** pane.

You can also store search results as a new File Set, by selecting the required Files from the search result and dragging them into the **Entries** pane. To add Files to an existing File Set, select all required Files from a search result, and drag them into the target File Set in the **Entries** pane.

Right click in the **Entries** pane to bring up a context menu, from where you can rename, delete, or add new, empty File Sets.

4.12.4 Advanced Free Text Search

Additional search options can be specified in the Free Text search string for advanced search. The free text search string syntax is very similar to the that of web search engines.

Example 1:

```
zebra scene -project -folder
```

Example 2:

```
giraffe -image -geometry
```

In example 1, the search is conducted on any file names, file owners, UUIDs, file types, and keywords that include the string `zebra` and `scene`. This search *excludes* `projects` and `folders`, plus any other files that include the words project and folder.

In example 2, the search is conducted on any file names, file owners, UUIDs, file types and keywords that include the string `giraffe`. This search *excludes* `images` and `geometries`, plus any other files that include the words image and geometry.

Note: When connected to a Graphic Hub of version 3 or higher, any File Type in the search string is ignored. Instead, the File Type is determined by the **Type Selection** buttons.

4.12.5 Wildcard text description

If the full Name is not known then text Wildcards can be used to make a search:

- `*`: Match any characters. For example: `Element`
- `?`: Match any single character. For example: `El?me?t`
- `[ ]`: Match a range or a list of characters. For example:
  - `Elem[b-g]nt`
  - `Elem[shje]nt`
- `< >` or `{ }`: Match a specific project, folder or file:
  - Searching for a specific UUID, for example `<6E6D3467-0A37-4D35-B80683B3A929772D>` or `{6E6D3467-0A37-4D35-B80683B3A929772D}`, will return the specific UUID.
  - Searching for a partial UUID, for example `<6E6D`, will also include the underlying UUID.

4.13 Background Loading

The background loading of Geometries, Materials and Images is supported in Viz Artist (with limitations). Background Loading can either be an automatic or a manual operation.

When background loading Geometries, the render method (See the **Render Options** page in the **Configuring Viz** section of the Viz Engine Administrator Guide) must be set to **VBO**.

This section contains the following topics:

- Automatic Background Loading
- Manual Background Loading
- Background Loading Limitations

4.13.1 Automatic Background Loading

When set to perform automatic background loading, Viz Engine automatically uses background loading whenever an image, and/or an object, is loaded. There are some limitations to this behavior, which are detailed in Background Loading Limitations.

Automatic Background Loading Configuration

In the Viz Config file, the option **Background_Loading_Level** is available to set up automatic background loading.

Available values for Background_Loading_Level are:

Value	Description
BGL_INACTIVE	Disables automatic background loading
BGL_IMAGES	Enables automatic background loading for images
BGL_OBJECTS_AND_IMAGES	Enables automatic background loading for images, geometries and materials

4.13.2 Manual Background Loading

Manual background loading can be done on an on demand basis. To do this commands are available to use Container objects:

Command	Description
GEOM SETBGL	Load (and set) a geometry in the background. The usage of the command is exactly the same as the existing GEOM SET command.

Command	Description
IMAGE SETBGL	Load (and set) an image in the background. The usage of the command is exactly the same as the existing IMAGE SET command.
MATERIAL SETBGL	Load (and set) a material in the background. The usage of the command is exactly the same as the existing MATERIAL SET command.

Manual Background Loading Configuration

To use manual background loading, the **Background_Loading_Level** configuration flag in the configuration file must be set to **BGL_INACTIVE**.

4.13.3 Background Loading Limitations

This section details the limitations of background loading.

- Scenes
- Compressed and Un-compressed Images
- Objects
- Function Plug-ins
- Image Combining
- Black/Transparent Flashing Image Textures
- Textures Not Visible on SDI Output with Background Loading Enabled

Scenes

Background loading of whole Viz Artist Scenes is not supported.

Compressed and Un-compressed Images

Only compressed images (DXT) are supported for background loading. Their mipmap-levels have been pre-generated and stored in the Graphic Hub, which is required for background loading to work properly. The generation of mipmap levels on-demand (as required for un-compressed images) can interrupt the renderer. This primarily depends on the size of the un-compressed images. The larger the image, the more impact on the renderer during background loading.

Objects

Only native geometry and imported third-party geometry objects containing vertex-data, materials and textures are supported targets for background loading. Containers hosting any sort of other plug-in may fail to load in the background. Containers hosting Function plug-ins will NOT be loaded in the background but rather fall back to a normal foreground loading. Containers hosting RTT shaders will interrupt the renderer if the particular RTT shader is used for the first time (shader is being compiled), but will no longer interrupt the renderer when loading the same shader in the background again.

Function Plug-ins

If a geometry, which is to be loaded in the background, contains one or more Function Plug-ins, the console will show a warning and the geometry will not be loaded in the background but in the foreground instead, this can interrupt the renderer.

Image Combining

If a geometry uses the second image (image2) in a texture, this might have an impact on the performance of the renderer. Mostly if one of the images are too large and/or un-compressed.

Black/Transparent Flashing Image Textures

When setting an image with background loading, for technical reasons the renderer will get the handle to a black (or transparent) texture until the first mipmap level of the image is loaded. This can result in a short flash of black (or transparent) texture while background loading textures on a visible object.

Textures Not Visible on SDI Output with Background Loading Enabled

When Viz Trio prepares and takes a scene, it momentarily stops the Viz Engine renderer. If background loading is enabled, this can result in unexpected behavior, such as textures becoming visible only in the preview and not the video output. To avoid this issue when background loading and controlling the engine with UPDATE commands, such as sent by Trio, Viz Engine by default ignores the command `RENDERER *UPDATE SET 0`. This can be changed by locating the following configuration option in the Viz Configuration file, in the section `RENDERER _OPTIONS`, and changing the value from `0` to `1`:

```
RendererUpdatesForBackgroundloading = 1
```

This fixes potential background loading texture issues on the second channel of dual channel setups as well.

4.14 Built Ins

In Built Ins there are a standard set of Viz Artist installation plug-ins.

Most plug-ins can be added to Containers, except, for example, Scene plug-ins. Because Viz Artist uses an open interface for the plug-ins, new geometries can either be obtained or created and then installed with Viz Artist. Also, most plug-ins have a properties editor. When a plug-in is applied to a Container in the Scene Tree, a plug-in icon will be added to the target Container.

The **Built Ins** button on the **Main Menu** gives access to the various plug-ins:

- Geometry plug-ins
- Container plug-ins
- Shader plug-ins
- Scene plug-ins
- Media Assets

This section contains information on the following procedures:

- Content Tabs
- Plug-in Context Menu
- View and Search for plug-ins
- Add Plug-ins to Containers

4.14.1 Content Tabs

The Built Ins are contained under various tabs and in folders.

Click on a Built In tab to open its panel and folder list. Double-click a Built In tab to open its panel and go to the default folder in the folder list. Default folders are:

- **Geometry plug-ins (GP)**: Default
- **Container plug-ins (CP)**: Global

- **Shader plug-ins (SH)**: Default
- **Scene plug-ins (SP)**: Default
- **Media Assets (MA)**: Channel

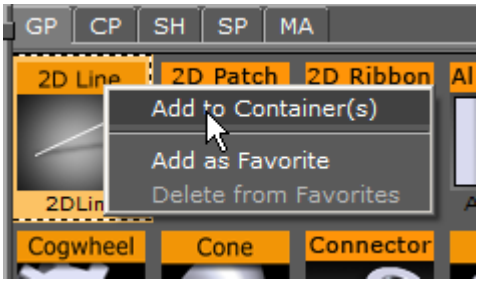
4.14.2 Plug-in Context Menu

A right-click on a plug-in icon to open the plug-in context menu.

This section contains information on the following procedures:

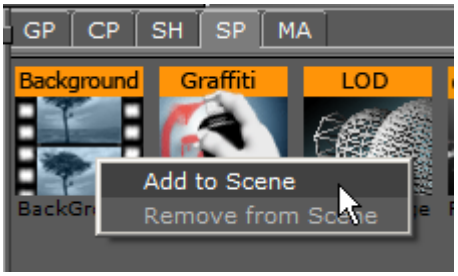
- Geometry, Container and Shader Menu
- Scene Menu
- Media Asset Menu

Geometry, Container and Shader Menu



- **Add to Container(s):** Adds the selected plug-in to a Container(s).
- **Add as Favorite:** Adds the selected Texture to the Favorites Bar.
- **Delete from Favorites:** Deletes the selected Texture from the Favorites Bar.

Scene Menu



- **Add to Scene:** Adds the selected plug-in to a Scene.
- **Remove from Scene:** Removes the selected plug-in from a Scene.

Media Asset Menu

- **Add <Media Asset> as Texture:** Adds the selected Media Asset to a Scene (automatically adds a new Container).
- **Add <Media Asset> as DVE:** Adds the selected source as DVE.
- **Show settings of <Media Asset>:** Shows the settings of the Media Asset, if it has been added to the scene and the Scene Tree is in Texture or DVE view.
- **Add as Favorite:** Adds the selected source to the Favorites Bar.
- **Delete from Favorites:** Deletes the selected source from the Favorites Bar.

For the **Background** and **Foreground** media assets, the context menu shows the Favorite options only.

4.14.3 View and Search for plug-ins

All plug-ins can be dragged straight into the Scene Tree (except Scene plug-ins). Double-click a folder tab to go to the **Default** folder for that plug-in category.

This section contains information on the following procedures:

- To View plug-ins
- To Search for a plug-in

To View plug-ins

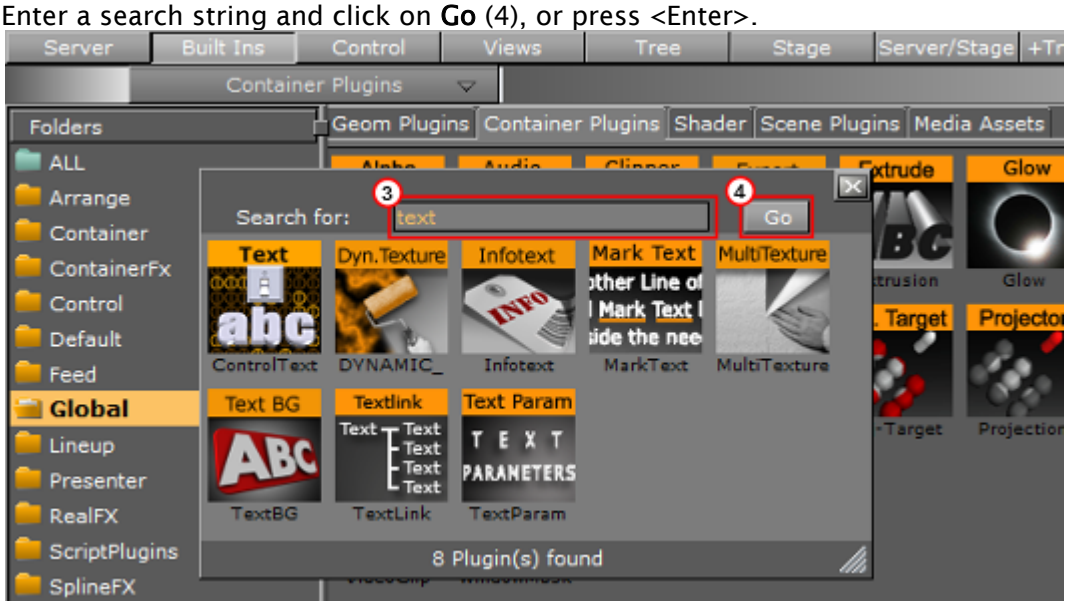
1. Select **Built Ins**.
2. Select the appropriate plug-in category, either:
   - From the drop-down menu that shows above the plug-ins view, or
   - Click on of the tabs.

The All folder in the left panel, offers an uncategorized view of all plug-ins. Pressing a letter on the keyboard to jump to a plug-in which starts with that letter.

To Search for a plug-in

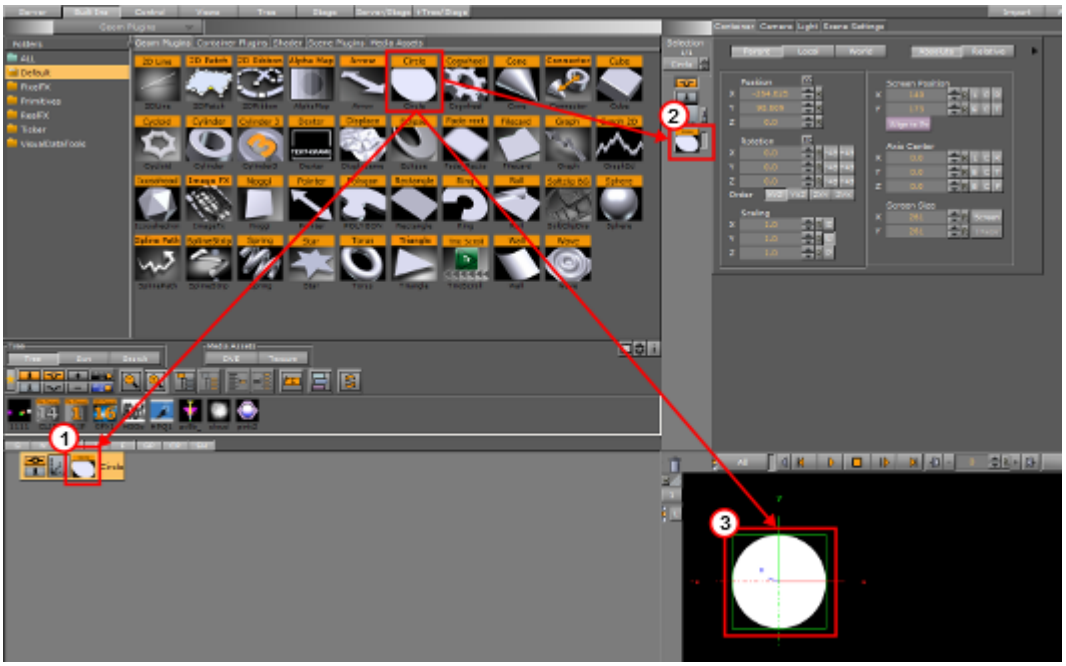
You can drag plug-ins from the search results to the desired destination.

1. Select **Built Ins**.
2. Open the plug-in search panel, by opening the **Built Ins** drop-down menu and selecting **Search...**, or pressing <Ctrl+F>.
3. Enter a search string and click on **Go** (4), or press <Enter>.



- To View plug-ins
- To Search for a plug-in

4.14.4 Add Plug-ins to Containers



Most plug-ins can be dragged to a target Container(s) in the scene tree (1), the Properties Panel (2) or the Scene Editor (3). Each plug-in also has a context menu where a plug-in can also be added to a Scene.

This section contains information on the following procedures:

- To Add a Geometry plug-in
- To Add a Container plug-in
- To Add a Shader
- To Add a Scene plug-in
- To Add a Plug-in to Multiple Containers
- To Add a Plug-in to Favorites
- To Open a Plug-in Properties Editor

To Add a Geometry plug-in

A Container can only contain one Geometry plug-in at a time. If a new Geometry plug-in is added the fist one will be removed from the Container.

1. Drag the Geometry plug-in icon to either:
   - A Container or multi selected Containers.
   - The scene tree area (creates its own container).
   - The Scene Editor area.
   - The Container Editor area.

   or

1. Highlight the target Container(s).
2. Right-click the plug-in icon and select **Add to Container(s)** from the plug-in context menu.

To Add a Container plug-in

1. Drag the Container plug-in icon to either:
   - A Container or multi selected Containers.
   - An object in the Scene Editor (only some container plug-ins will do this, when they will the object becomes active when the icon is moved over the object.
   - The Container Editor area.

   or

1. Highlight the target Container(s).
2. Right-click the plug-in icon and select **Add to Container(s)** from the plug-in context menu.

To Add a Shader

A Container can only contain one Shader at a time. If a new Shader is added the fist one will be removed from the Container.

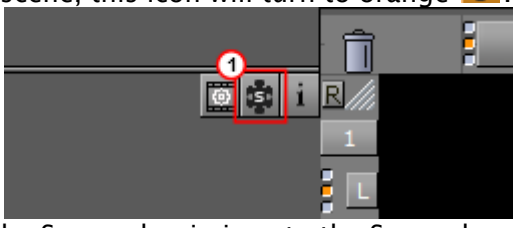
1. Drag the icon to either:
   - A Container or multi selected Containers.
   - An object in the Scene Editor.
   - The Container Editor area.

   or

1. Highlight the target Container(s).
2. Right-click the plug-in icon and select **Add to Container(s)** from the plug-in context menu.

To Add a Scene plug-in

1. Open the Scene plug-in tab:
   - Click on *Scene Settings -> Plug-in*.
   - Click on the Scene plug-in icon (1). When a scene plug-in has been applied to the scene, this icon will turn to orange  .



2. Drag the Scene plug-in icon to the Scene drop area.

   or

1. Right-click the Scene plug-in icon to open the plug-in context menu.
2. Click on **Add to Scene**...

To Add a Plug-in to Multiple Containers

1. Highlight the required Containers.
2. Drag a plug-in icon to a highlighted Container.

3. Release the plug-in icon.
   or

1. Drag a plug-in icon.
2. Hold the plug-in icon over a Container.
3. Continue to hold the left-click and do a right-click to drop the plug-in icon on the Container.
4. Continue to hold the left-click and repeat for each required Container.

To Add a Plug-in to Favorites

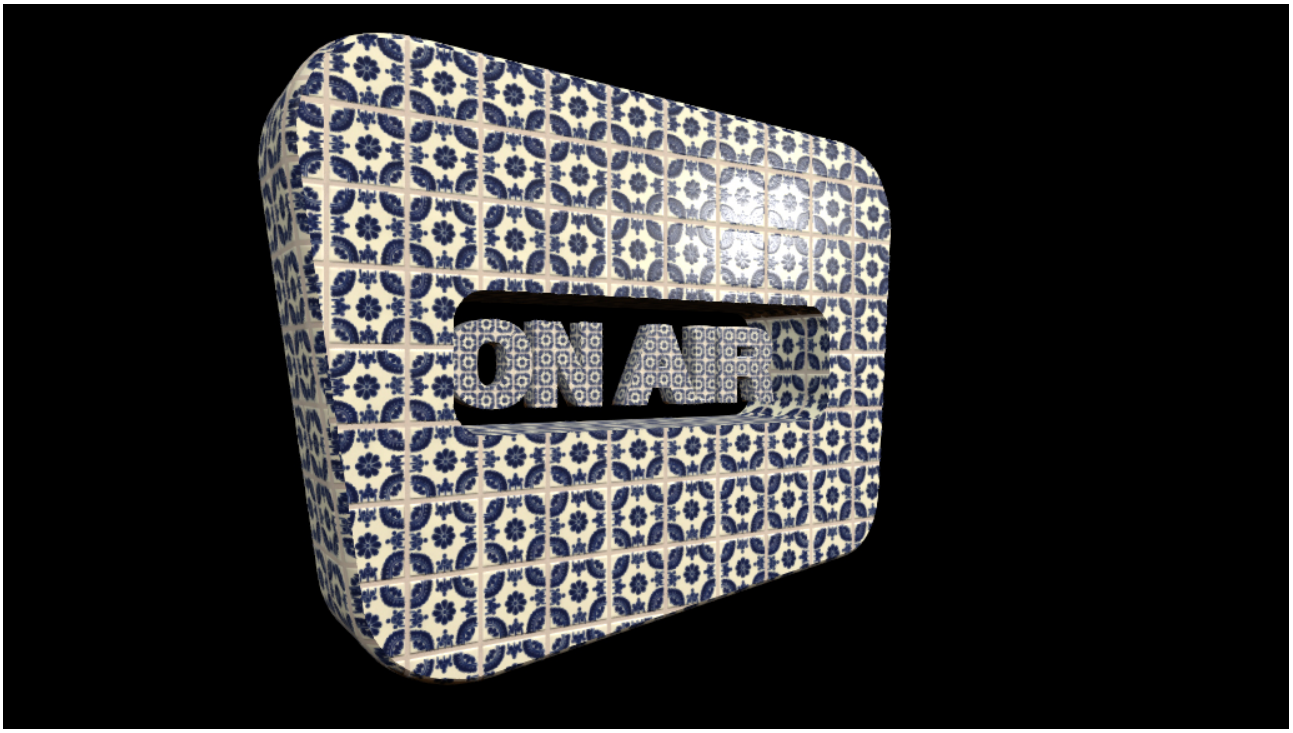
1. Right-click the icon to open the plug-in context menu.
2. Click **Add as favorite...**

You can also add plug-ins as favorites from the search panel .

To Open a Plug-in Properties Editor

Click on the Media Asset icon, in the Container or the in the Properties Editor.

4.15 Substance Editor



4.15.1 Working with Substances

This section details how to work with Substances.

The import and use of Substances in Viz requires a separate license. You obtain this license, as well as a license for the Allegorithmic Substance Designer and the Substance PBR database, from your local Vizrt representative.

Substances use two different file extensions, `.sbs` and `.sbsar`:

- `.sbs` files are editable substance source files. Use Substance Designer with these files.
- `.sbsar` files are performance optimized, published substances.

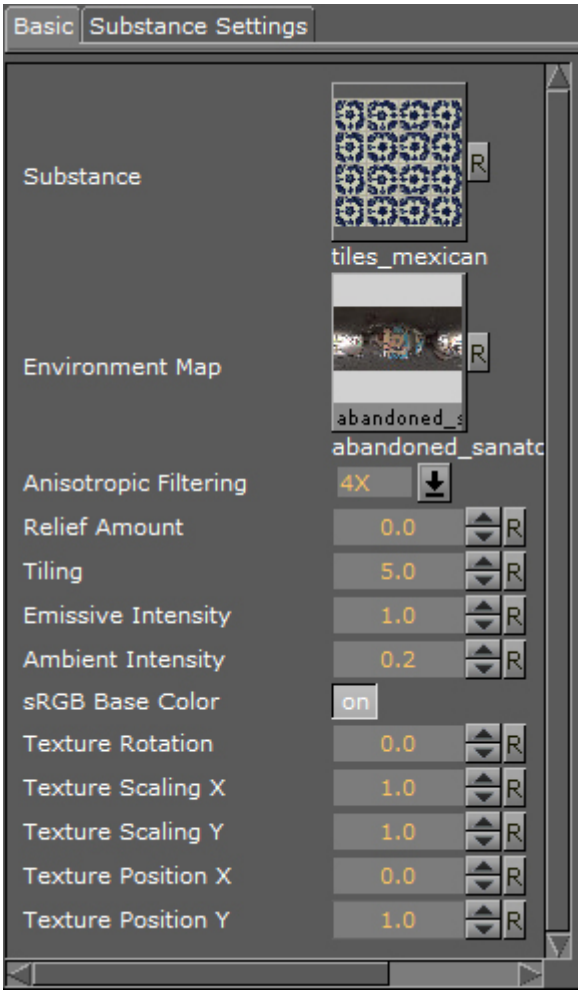
You can import `.sbsar` files into Viz Artist. They must be Physically Based Rendering (PBR) materials to work correctly in Viz. Make sure you create the correct material type when creating Substance shaders in Substance Designer.

To create a `.sbsar` file, first create a Shader with Substance Designer, and then publish this shader for import to Viz Artist. To view and adjust the parameters of a shader, switch to the **Substance Settings** panel. The parameters in the settings panel are different for each loaded substance, depending on the type of shader. The only supported type of shader in Viz Artist is PBR Metallic/Roughness.

Tip: If you get the error message "This is not a valid Substance" when you apply the shader in Viz Artist, you need to make sure you have a PBR Metallic/Roughness material created in Substance Designer.

4.15.2 Substance Basic

Here you can adjust the basic settings of your Substance shader.

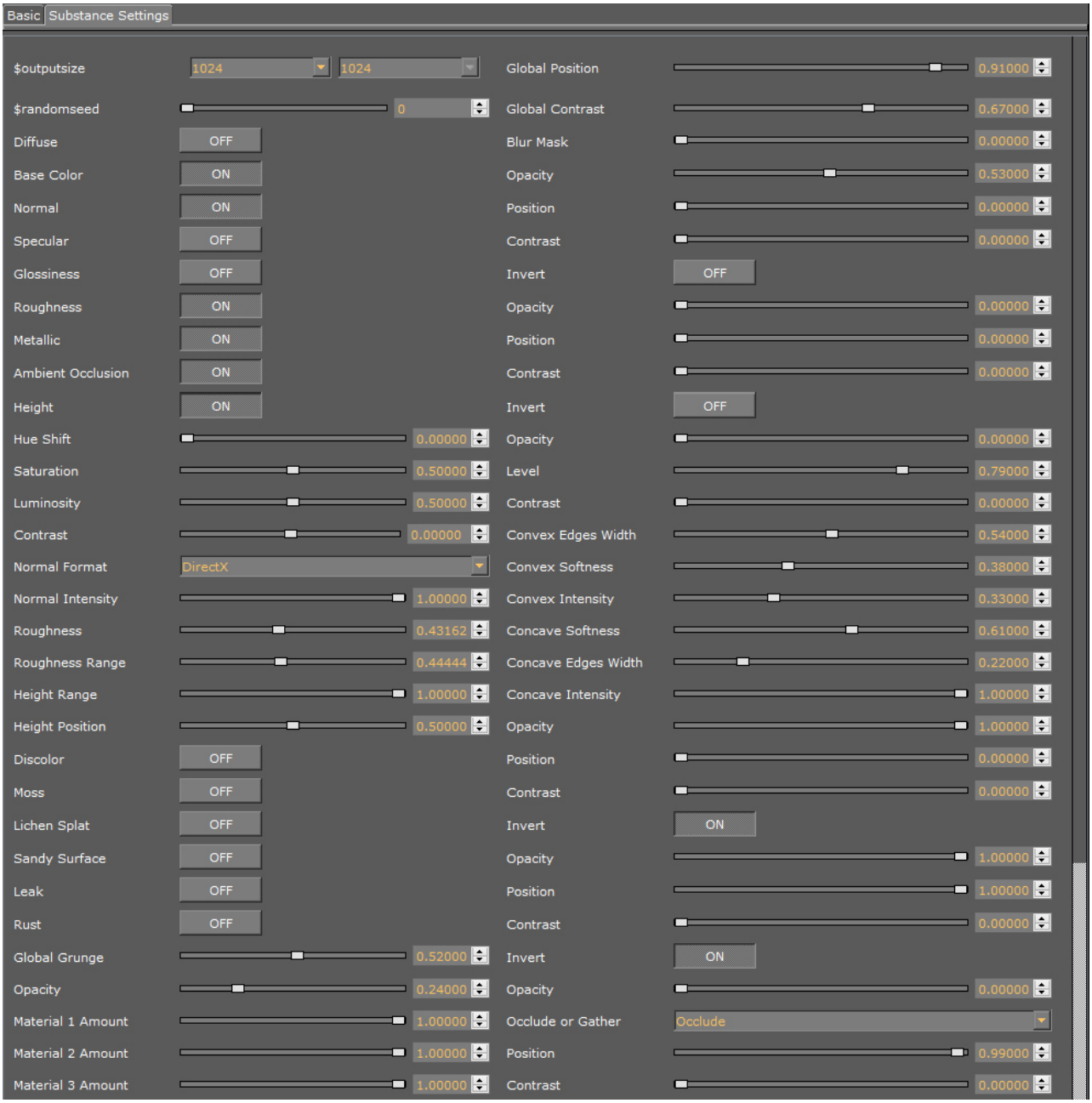


- **Substance:** Drag the desired Substance shader from the Viz database section to this drop-field.
- **Environment Map:** Drag any RGB or HDR environment map that should be reflected on the surface to this drop-field. The strength is controlled by the **Ambient Intensity** and by the metallic setting exposed individually in a Substance shader.
- **Anisotropic Filtering:** Higher settings increase the load on the GPU during shader calculation. This is usually set between `0-4`.
- **Relief Amount:** You only need to adjust this setting for relief shaders.
- **Tiling:** Lets you specify the number of tiles.
- **Emissive Intensity:** Change this setting for a higher emission rate of your shader.
- **Ambient Intensity:** Use this option to control the blending when you have an Environment texture applied to your Substance shader.
- **sRGB Base Color:** Enable to use the sRGB color space model.
- **Texture Rotation:** Use this setting to rotate the texture applied to your geometry. This rotation has a limitation in the rotation degree of freedom, If the limit is reached, you need to change the rotation in your Substance shader.
- **Texture Scaling X:** Use this setting to scale the applied texture/shader on the `X` axis.
- **Texture Scaling Y:** Use this setting to scale the applied texture/shader on the `Y` axis.

- **Texture Position X:** Use this setting to change the X position of the applied texture/shader on the x axis.
- **Texture Position Y:** Use this setting to change the Y position of the applied texture/shader on the y axis.

4.15.3 Substance Settings

The settings of each Substance shader are different, depending on the material itself. All Substance shaders have in common that they need to generate a texture from the procedural shader description created in Substance Designer. If you are creating texture based shaders, you need to make sure that your base texture has enough resolution. All other settings and sliders in the Substance shaders vary with each different shader, depending on which features are exposed in each individual shader. Some settings are exposed for almost any existing shader from the Substance PBR library. These include **Hue Shift**, **Saturation**, **Luminosity**, **Contrast**, **Normal Format**, **Output Size** and **$randomseed**. If you create your own shaders in Substance Designer, you must expose the modifiers manually from Substance Designer if they are to be editable in Viz Artist.



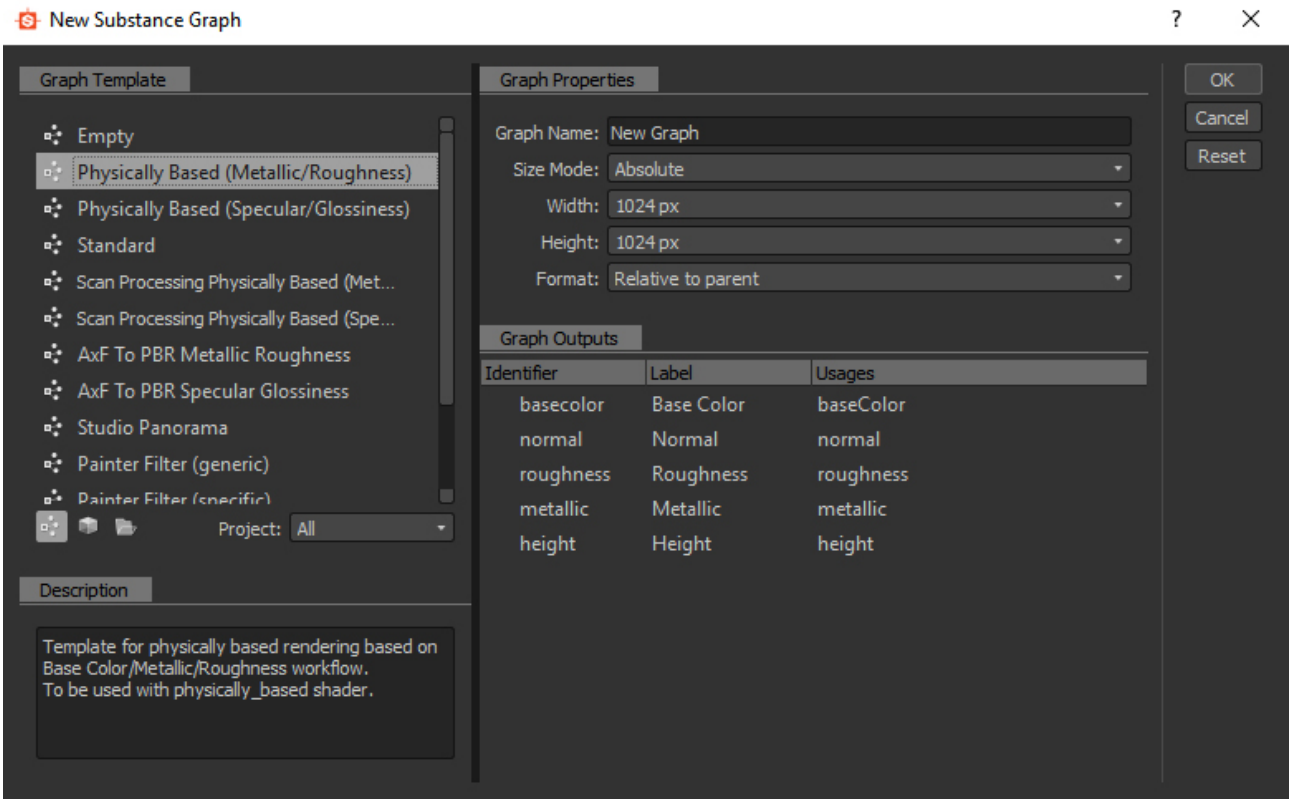
⚠ **Tip:** Check the Nvidia GPU meter to see performance used by the shaders.

4.15.4 How to create Substance shaders with Substance Designer

This section gives you a brief introduction on how to create a PBR Metallic/Roughness material in Substance Designer. Substance Designer allows you to create procedural materials as well as texture/scan based materials. Please see the Allegorithmic website and their Youtube channel for

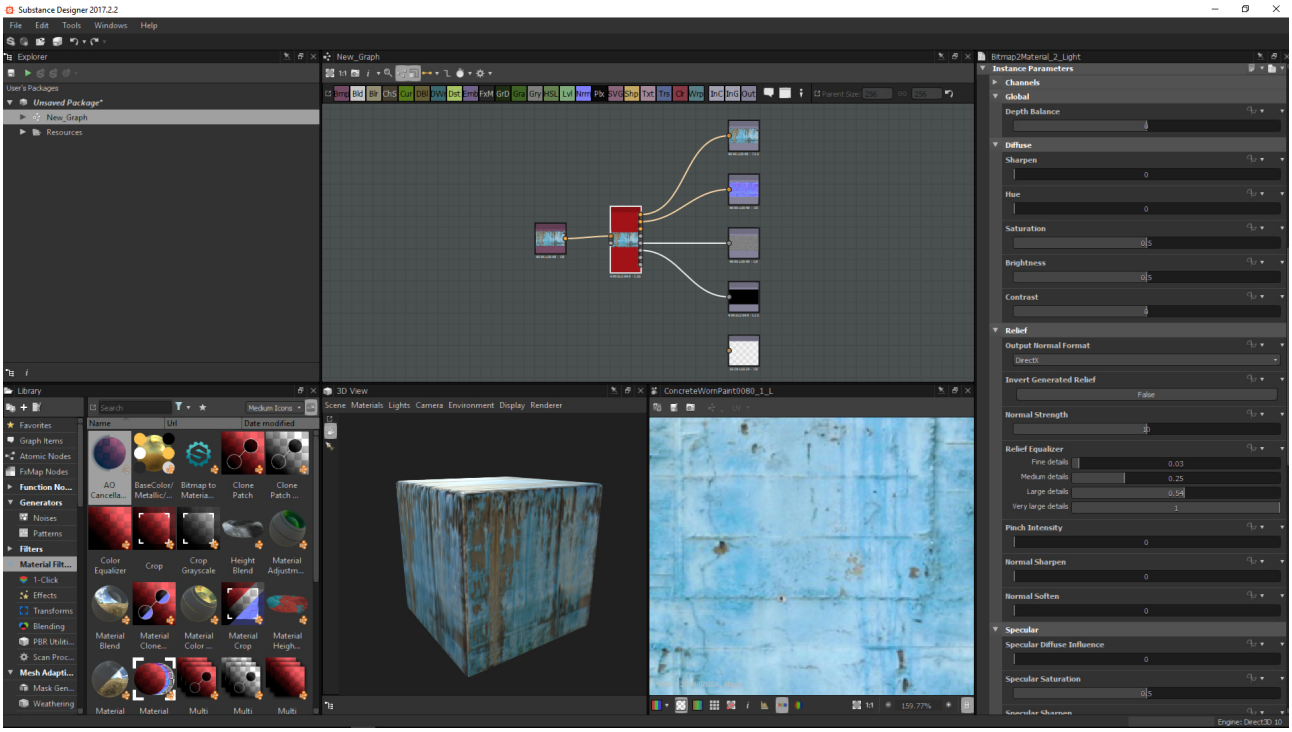
more detailed information on how to work with Substance Designer: https://www.allegorithmic.com/.

When you create a new Substance in Substance Designer, you need to select the **Physically Based (Metallic/Roughness) Graph Template**. Otherwise, this shader will not work properly in Viz Artist, causing the error message "This is not a valid Substance". When saving a Substance shader/material, it will be saved as an `.sbs` file. These files cannot be imported into Viz Artist. To import Substance shaders in Viz, you need to publish the material to an `.sbsar` file.



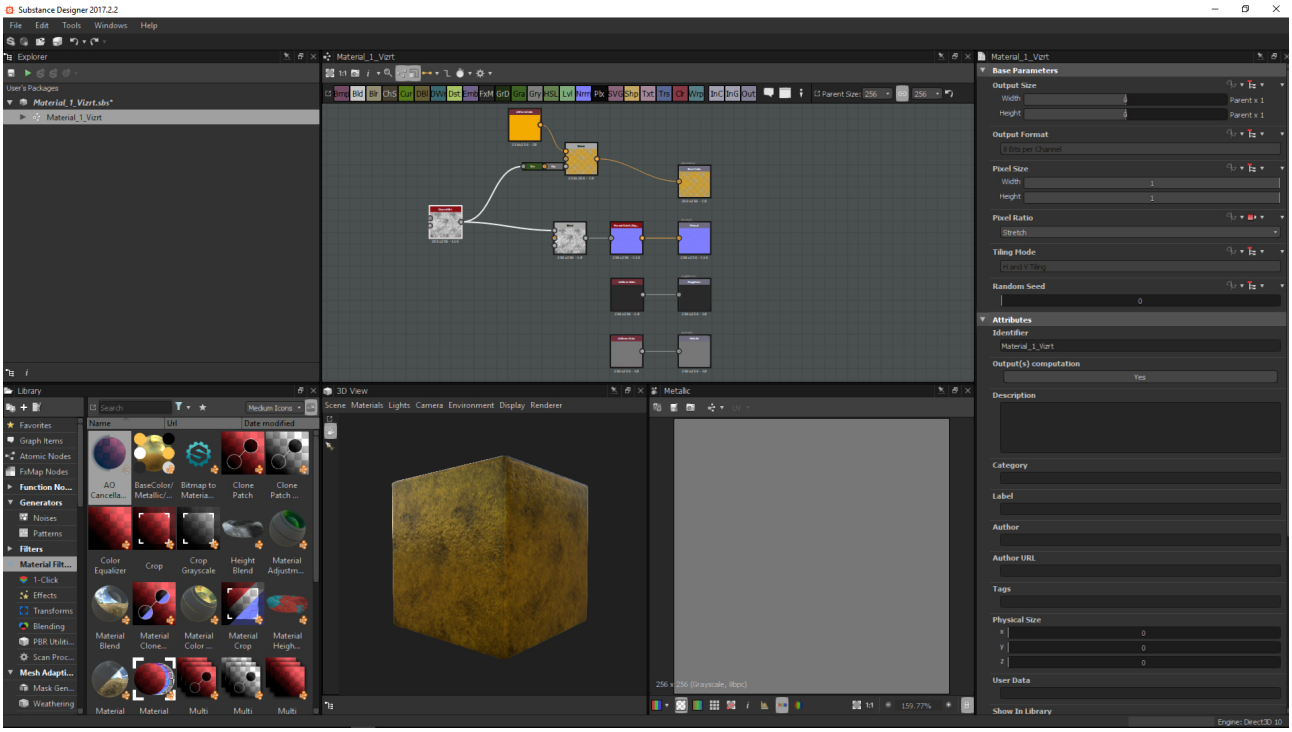
The shader graph of each material looks different. Here are a few examples for different shader graphs with PBR Metallic/Roughness materials, reaching from simple texture shader graphs to complex procedural shader graphs.

4.15.5 Simple texture shader



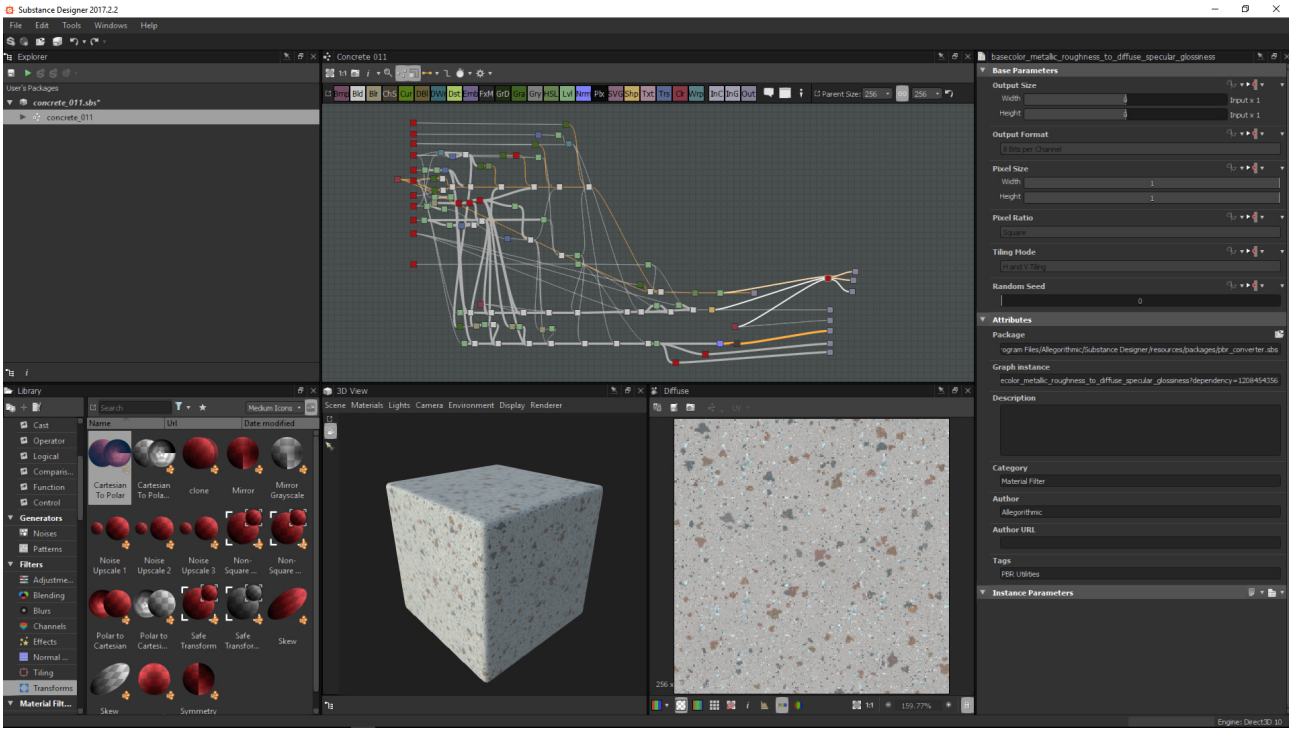
In the shader graph, you see the 5 Output channels for the PBR material: **Base color**, **Normal**, **Roughness**, **Metallic**, and **Height**. All outputs are generated automatically from a simple input texture.

4.15.6 Basic procedural shader



In the shader graph, you see the 4 Output channels for the PBR material: **Base color**, **Normal**, **Roughness**, and **Metallic**. All outputs are generated and mapped manually.

4.15.7 Complex procedural shader



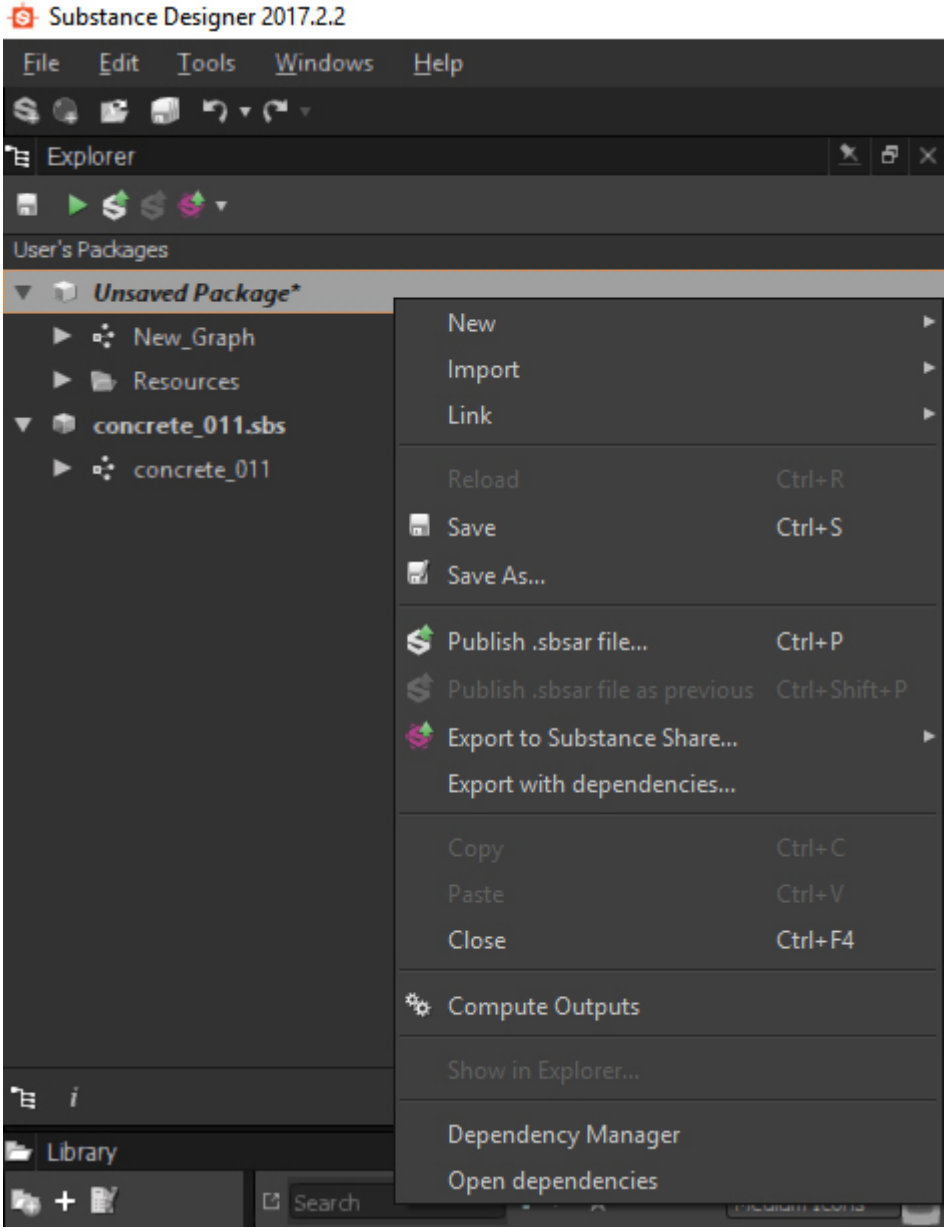
Here you see a very complex procedural shader graph, constructed with different modifiers and functions, generating the necessary output for **Base Color**, **Normal**, **Roughness**, **Metallic** and **Height**.

> ⚠ **Tip:** Only PBR Metallic/Roughness materials from Substance Designer are supported in Viz.

4.15.8 Publishing shaders

When you want to import the Substance shaders in Viz Artist, you need to publish them from Substance Designer instead of saving them. The diffference between publishing and saving is that the shader is performance optimized when published. Published shaders cannot be opened by Substance Designer, so it is important to keep your .sbs files and .sbsar files together, in case you need to make changes to your shader later on.

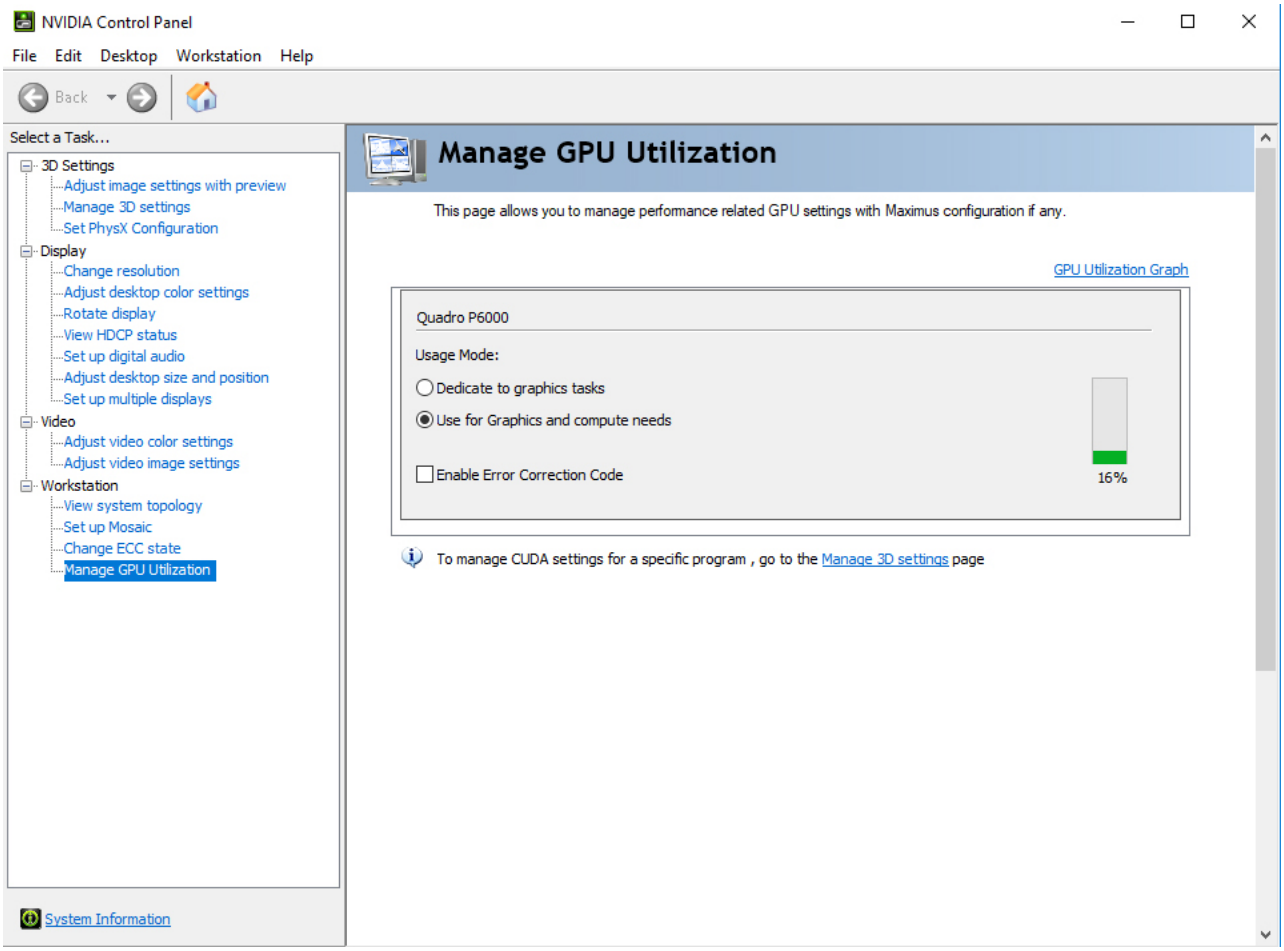
To publish the shader, right click the shader/package and select **Publish .sbsar file**.



Tip: You can find documentation and training with Substance Designer on https://www.allegorithmic.com/ as well as on the Allegorithmic Youtube channel.

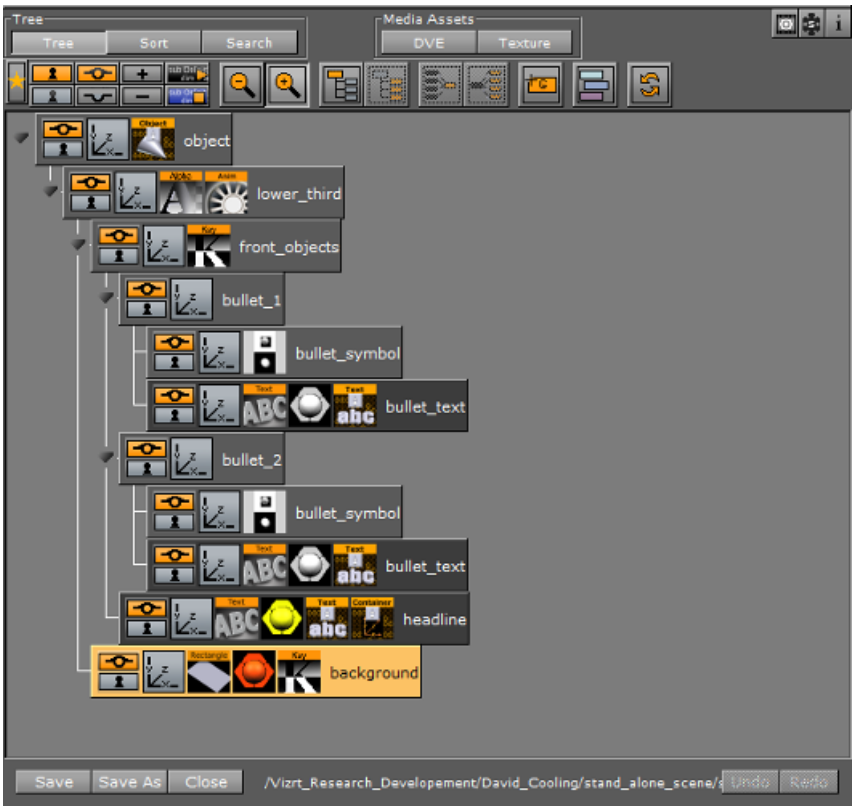
4.15.9 Substances Performance

Substance shaders are computed on your GPU. Vizrt recommends an Nvidia M-series graphics card or higher to work efficiently with Substances. Since Substances run exclusively on the GPU, you should check the Nvidia GPU meter when working with Substances inside Viz to have full overview of the rendering performance. There should be a section called **Manage GPU Utilization** in your **Nvidia Control Panel**. Check the performance of the Substance shaders here.


NVIDIA Control Panel
File Edit Desktop Workstation Help
Back
Select a Task...
3D Settings
Adjust image settings with preview
Manage 3D settings
Set PhysX Configuration
Display
Change resolution
Adjust desktop color settings
Rotate display
View HDCP status
Set up digital audio
Adjust desktop size and position
Set up multiple displays
Video
Adjust video color settings
Adjust video image settings
Workstation
View system topology
Set up Mosaic
Change ECC state
Manage GPU Utilization
System Information
Manage GPU Utilization
This page allows you to manage performance related GPU settings with Maximus configuration if any.
GPU Utilization Graph
Quadro P6000
Usage Mode:
Dedicate to graphics tasks
Use for Graphics and compute needs
Enable Error Correction Code
16%
To manage CUDA settings for a specific program , go to the Manage 3D settings page

5 Scene Tree

The Scene Tree is the logical visualization of all the properties in a Scene. The Scene Tree consists of Containers that hold properties. The properties can be Geometries, Images, Materials, Texts, Transparency functions, Key functions, and many more. Scenes are root items that hold other items. Created Scenes are stored in the Graphic Hub and are available through the Server Panel. For general information about Scenes and procedures on how to create, import, rename, and delete Scenes, see Working with Scene Items.



This section contains information on the following topics:

- Scene Tree Menu
- Favorites Bar
- Containers
- Modify Container Properties
- Text Editor
- Geometry Editor
- Transformation Editor
- External Control

To build a Scene, you add properties, such as image items or geometries (Built Ins), to the Scene Tree. The containers in the Scene Tree are organized in a hierarchy and grouped in logical divisions. Containers and Groups are explained in more detail in the following paragraphs. The Scene Tree is available when these buttons in the Main Menu are clicked on:

- Server Button
- Built Ins Button

- Control Button
- Tree Button
- Stage Button
- +Tree/Stage Button

5.1 Scene Tree Menu

At the top of the Scene Tree area is a control menu:

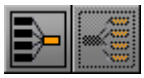
- **Tree**
  - **Tree**: Shows an overview of all the Containers in the Scene (see Tree Button).
  - **Sort**: Sorts the items in the Scene Tree based upon criteria that are set in the Display area (see Sort Button).
  - **Search**: Searches items in the Scene Tree (see Search Button).
- **Media Assets Manager**: In the Media Asset Manager, add Media Assets to a Scene as:
  - DVE
  - Texture
- **Scene Log**: Shows if an error occurs. Opens the Log window, which gives information about the loading process of the Scene, such as errors, for example if a plug-in is not found. Click on it to open the Scene Log.
  - If the log window is closed, but not cleared, the exclamation mark icon will show in gray. It will show red again if another fault occurs.
- **Media Asset Properties Editor:** Opens the Media Asset Panel Editor panel of the Scene Settings.
- **Scene plug-in:** Opens the **Plug-in** panel panel of the Scene Settings. In the plug-in panel, Scene plug-ins can be added to the current Scene. This icon turns orange when a Scene contains a Scene plug-in. Click on the icon to view the Scene plug-in properties panel.
- **Scene Info**: Opens the Scene Information window, which gives information about the Scene entered by the designer. Shows orange if information is available.

This section contains information on the following topics:

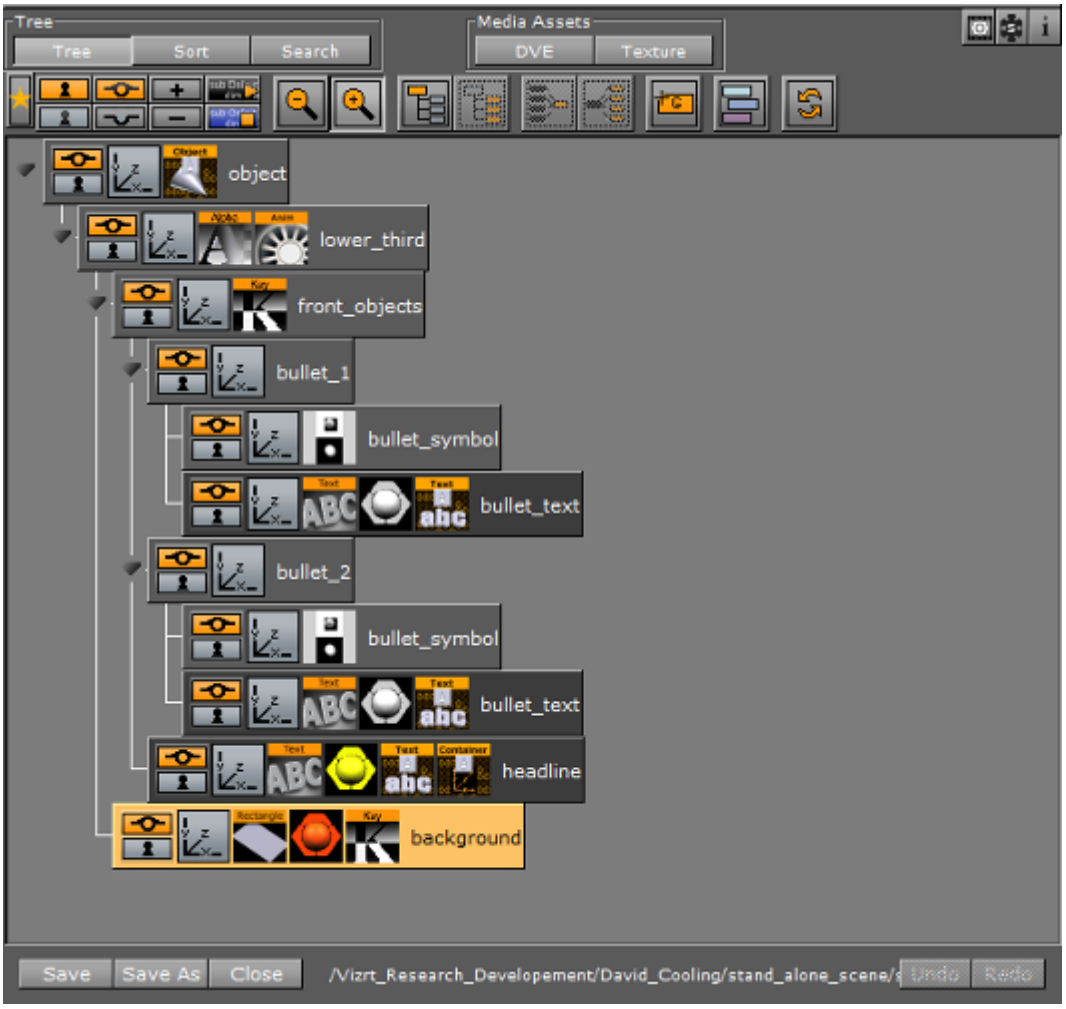
- Tree Button
  - Scene Tree View
- Sort Button
  - Scene Tree Sort View
- Search Button
  - Search Options
  - Show Search Results
  - To Do an Extended Search
  - To Do a Script Search

5.1.1 Tree Button

If the Tree button is clicked, all Containers are shown in the hierarchical Scene Tree View.

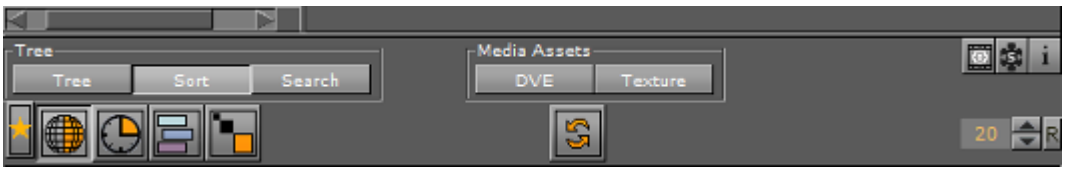
-  **Show/Hide Favorites**: Shows or hides the Favorites Bar.
-  **Lock/Unlock**: Locks or unlocks the ability to modify the transformation of selected Containers in the Scene Editor. Middle mouse click to lock or unlock all Containers.
-  **Show/Hide**: Shows or hides selected Containers in the Scene Editor, and eventually in the final rendered output. Middle mouse click to show or hide all Containers.
-  **Expand/Collapse**: Expands or collapses the Scene Tree.
-  **Start/Stop**: Starts (compiles) or stops all Container scripts.
-  **Zoom In/Zoom Out**: Zooms in (larger Containers) or Zooms out (smaller Containers) on the Container tree. Zoom out applies smaller icons to the Containers. This gives a better overview, especially in complex Scenes.
-  **Group/Ungroup**: Creates a new container group, and places the selected Containers as Sub-Containers. The Ungroup button moves the Sub-Containers up one level.
-  **Merge/Split**: Merges the selected Containers to one compound object, or splits an existing compound object.
-  **Create Group**: Creates a new empty group at the top of the Scene Tree. Drag the icon to place a new group at a specific place in the Scene Tree.
-  **Color**: Applies a color to selected Containers. You can define text labels for the colors in the Global Settings Panel to make them more descriptive.
-  **Refresh**: Refreshes the Scene Tree. Use this button if scripts have been deployed to change the Scene Tree.

Scene Tree View



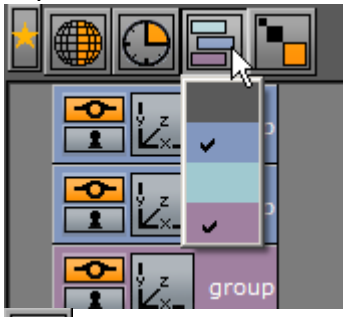
5.1.2 Sort Button

Click the Sort button to enable the Scene Tree Sort View, where the items in the Scene Tree can be sorted based on various criteria. A sorted Scene Tree makes it possible to analyze the Scene and optimize the performance.



-  **Show/Hide Favorites**: Shows or hides the Favorites Bar .
-  **Vector**: Sorts the Scene Tree by the number of vectors the Containers (which have Geometries) have. The Container with most vectors is shown on top.

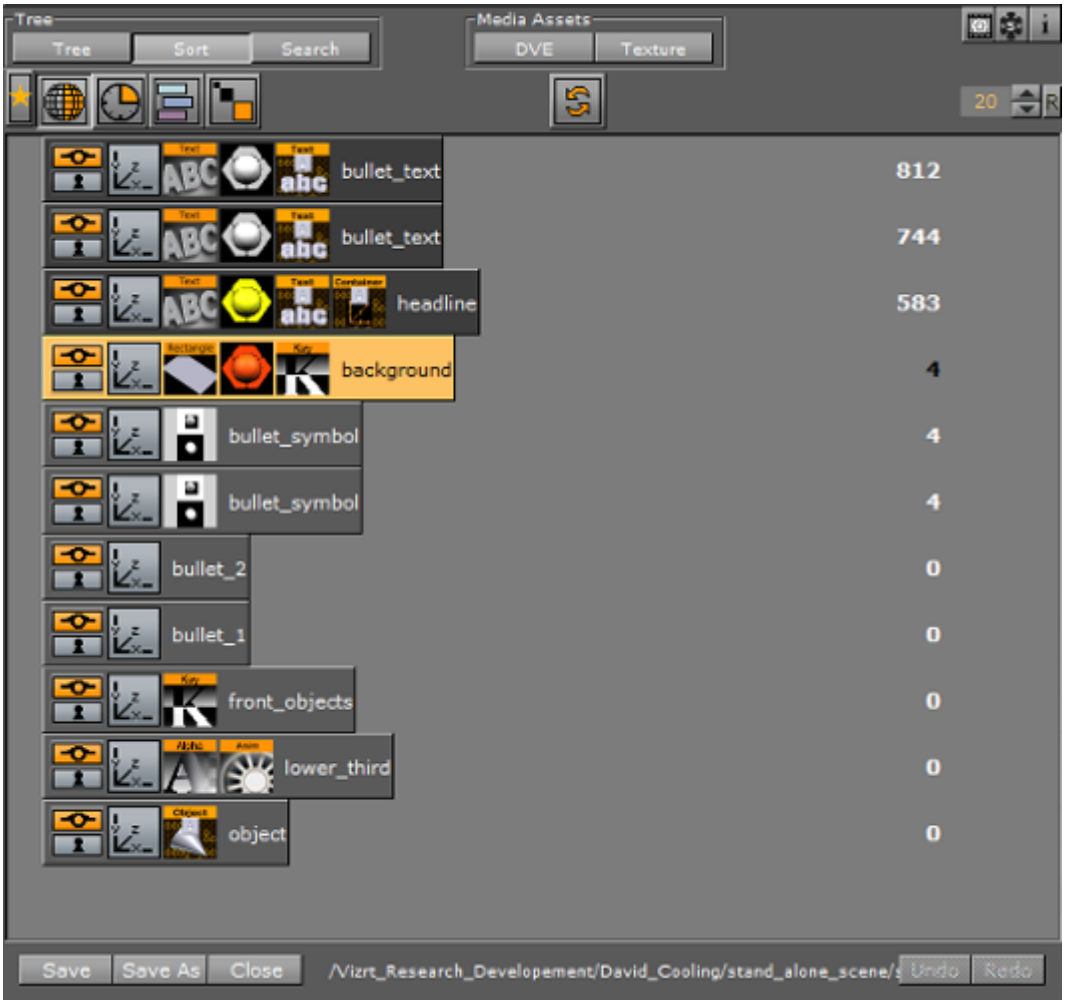
-  **Render Time**: Sorts the Scene Tree by the time it takes to render the Containers. Each Container is shown with the percentage of total time the Scene takes to be rendered. The Container with the highest percentage is shown on top.
-  **Sort by Colors:** Filters Containers by color. Click on the icon, and click on each required color to filter.



-  **Texture Size**: Sorts the Scene Tree by the size of the Container texture. The Container with the largest texture size is shown on top.
-  **Unique**: Shows unique group objects with the same Texture only once. This option is only available if the Containers are sorted by Texture size (click the Texture Size button).
-  **Refresh**: Refreshes the Scene Tree.
- 

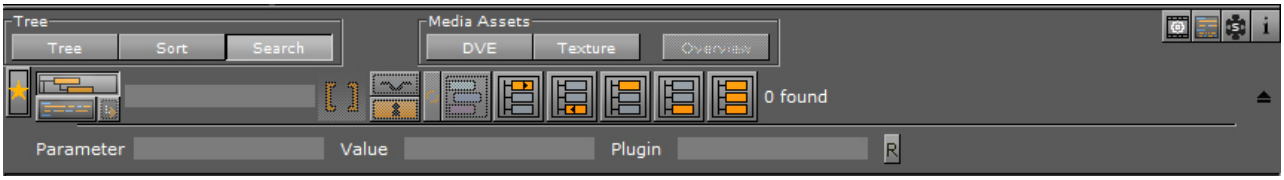
The input box defines the number of search results displayed. Click <R> to set to default number of search results (20).

Scene Tree Sort View



5.1.3 Search Button

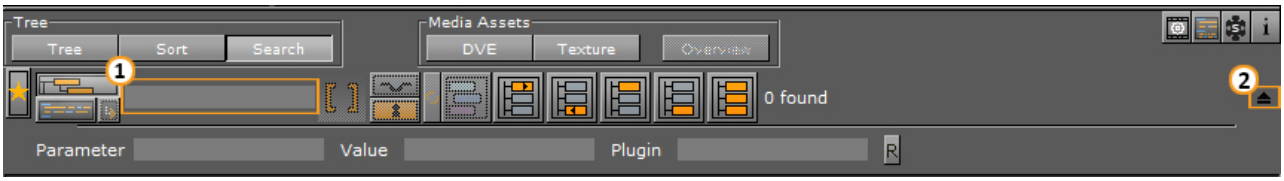
Click **Search** button to access the search controls. Here you can search for items in the scene tree, and inside containers scripts and scene scripts.



Use the different buttons to search for specific properties of the Scene (see Search Options) or highlight the search results. Search results are highlighted in the Scene Tree, and the number of search results are displayed.

⚠ **Example:** If a plug-in (for example *Extrude*) is dragged to Properties, clicking the **Show all** button will highlight all occurrences of that plug-in within the Scene Tree.

Search Options



- **Name**: Search for Containers with a specified name. Type the search string in the **Name** field (1). Wildcards, for example ***, ?** and **[ ]**, may be used in the search.

  ⚠ **Tip:** By default, the text wildcard character, *, is added to the *start* and at the *end* of the search string. This means that searching for "*player*" will find any node named "*player*" in the scene tree, and also nodes named for instance *player-name* or *secondplayername*.

  Click on the drop arrow (2) to show extended search parameters. Options are to search for Parameter, Values and plug-in names within the Scene. When the extended parameters option is active, all other search options than **Name** are inactive.

- **Search for Containers**: Searches in the tree for container properties.
- **Search in Script:** Searches all scripts in your scene, container scripts and scene scripts.
- **Show in Script:** Shows the search results inside the script editor.
- **Drop area for Properties:** Drag an item from the Server Panel, from a Container in the Scene Tree or from the Favorites Bar to search for its occurrence within the Scene.
- **Hidden**: Searches for all hidden Containers.
- **Locked**: Searches for all locked Containers.
- **Inactive Properties:** Searches for Containers that contain properties that are set to inactive.
- **Color**: Searches for all Containers with the assigned color tile.

Show Search Results

- **Show Next**: Shows the next result.
- **Show Previous**: Shows the previous result.
- **Show First**: Shows the first result.

-  **Show Last**: Shows the last result.
-  **Show All**: Shows all results.
- 

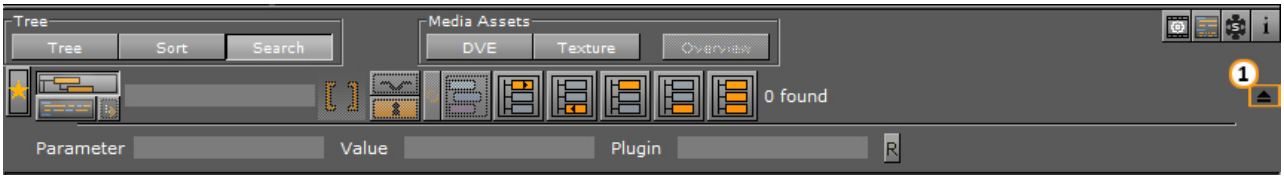
Search Results: Shows how many results were found.

> ⚠ **Tip:** If a specific plug-in is no longer to be used in a Scene, make a search for the plug-in and select **Show All** to highlight all the results. Drag one of the highlighted plug-ins to the trash can, and all the highlighted plug-ins will be deleted from the Scene.

> ⚠ **Tip:** If **Show all** search results is selected, the transformation properties can be changed for all highlighted Containers.

> ⚠ **Tip:** If **Show all** search results is selected, a plug-in or an item (except a Scene) can be added to all the highlighted Containers (drag a plug-in or an item to one of the highlighted Containers).

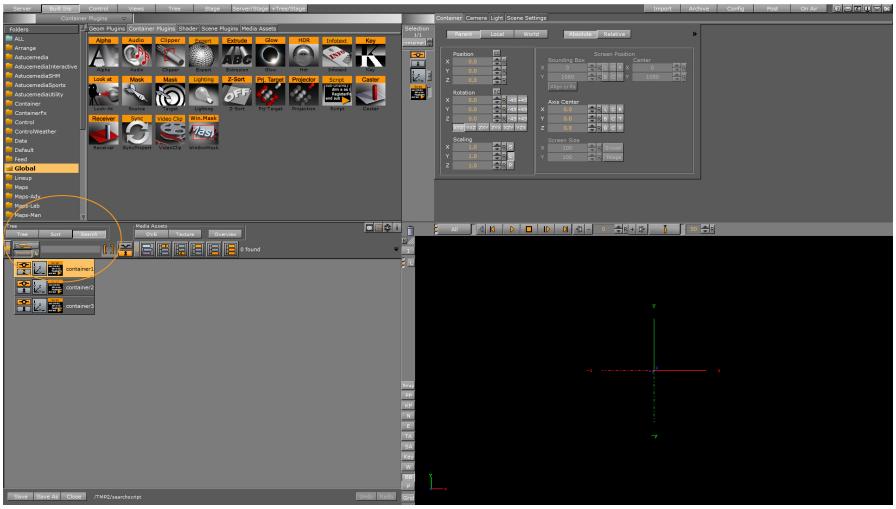
To Do an Extended Search



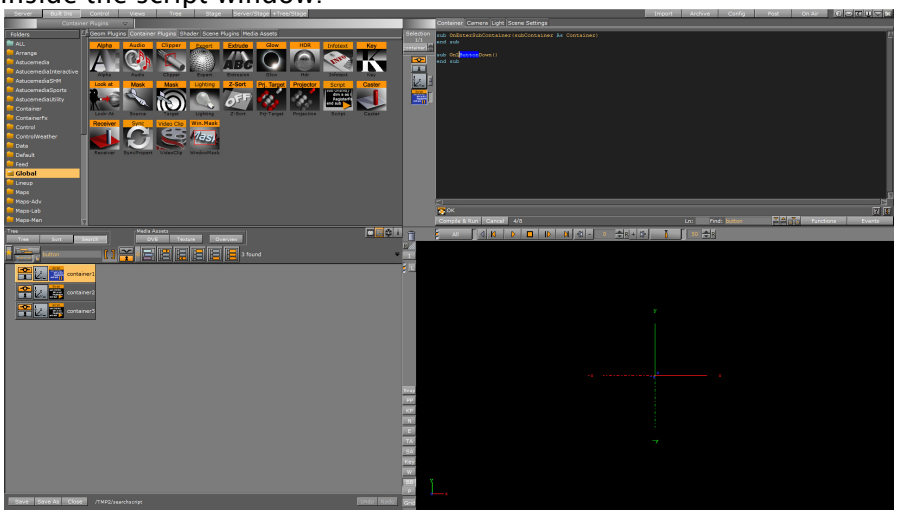
1. Go to the Scene Tree and click **Search**.
2. Click drop arrow to show the extended search options (1).
3. Enter a search criteria. The search criteria can be a combination of Parameter, Value and plug-in name. Below are some examples:
   - **Parameter:**`tesselation, width, *show*`
   - **Value:**`20, *10.0*`
   - **Plugin:**`*rect*`
4. Press <Enter>.
5. Use the Show Search Results buttons to navigate the Scene Tree.

> ⚠ **Tip:** Use the label names in the plug-in editors as a search criteria. If there is no hit, use the asterisk (*) symbol.

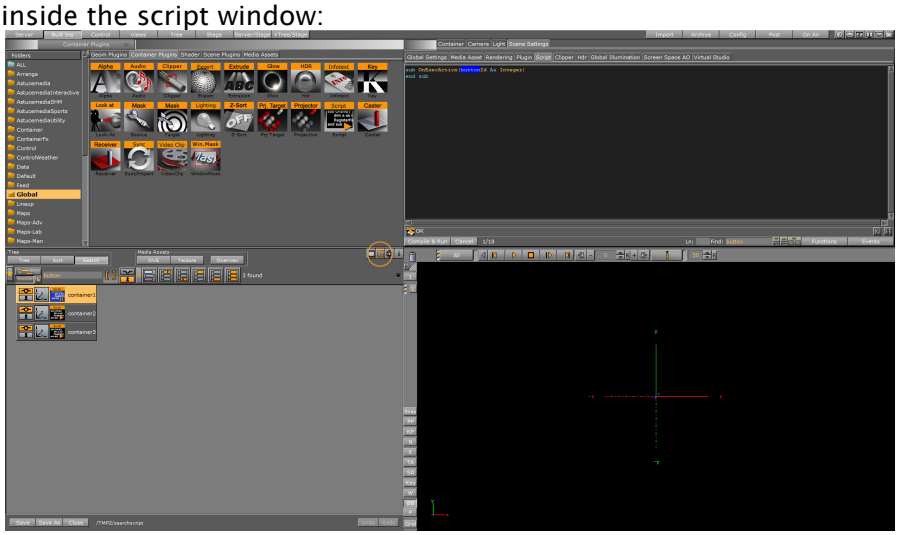
To Do a Script Search



1. Load a scene that contains scripts, either scene scripts, container scripts, or both script types.
2. Go to the Scene Tree and click **Search** to open the search fields.
3. Enter your search term, then click the **Search Script** icon.
4. The search result contains all container scripts and scene scripts that contain the search term within the script code.
5. Clicking the **Show in Script** button to open the selected script and highlight the search result inside the script window:



6. You can also click **Scene Script** to open the scene Script Editor, highlighting the search result inside the script window:



⚠ **Tip:** You can also perform an advanced search inside the script editor, to search not just within a single script but in *all* scripts used in the scene.

5.2 Favorites Bar

For quicker access to items that are used often, you can add items to the Favorites bar. To open or close the Favorites bar, click on the Star icon (1) at the top of the Scene Tree area.

All items, except Scenes and Audio files, can be added as favorites. Geometry, Container and Shader plug-ins can also be added to the favorites. Favorites are saved as part of the user settings to the database.

Click on the buttons below the Favorites bar (2) to show or not show favorite items from the categories detailed here:

- **G**: Geometries
- **M**: Materials
- **MA**: Materials Advanced
- **I**: Images
- **F**: Fonts
- **GP**: Geometry plug-ins

- **CP**: Container plug-ins
- **SH**: Shader plug-ins

Items can also be quick selected. Click on an item button with:

- **<Ctrl+Click>:** Selects all
- **<Shift+Click>:** Deselects all
- **<Alt+Click>:** Only shows the clicked type

5.2.1 To add an item as a Favorite

1. Right-click the item in the Server or Built Ins area.
2. Select **Add as Favorite...**,
   or

1. Click on the Star icon to open the Favorites bar.
2. Drag an item from the Server or Built Ins area to the Favorites bar.

Items can also be added as favorites from the search results window.

5.2.2 To Remove an item as a Favorite

1. Right-click the item in the Server or Built Ins area.
2. Select **Delete from Favorites...**,
   or

1. Click on the Star icon to open the Favorites bar.
2. Drag an item from the Favorites bar to the trash can.

5.2.3 To Add a Favorite item to a Container

1. Select the target Container(s).
2. Right-click the required item in the Favorites bar.
3. Select Add to Container(s).
   or

1. Select the target Container(s).
2. Drag the required item from the Favorites bar to the target Container.

If more than one Container is selected, the item will populate all selected Containers.

5.3 Containers

A Container combines a set of properties to one block used in the Scene (a Scene can contain many Containers). Containers can also be multi selected to modify or move many Containers at once.

This section contains information on these topics:

- Container Names
- Container Icons
- Container Context Menu

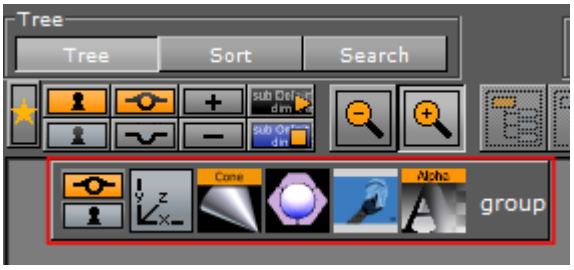
- Add Containers to a Scene
- Select Multiple Containers
- Copy Containers
- Move Containers
- Rename Containers
- Edit Container Properties
- Delete Containers
- Merge Containers
- Group Containers

5.3.1 Container Names

You can rename containers to ease identification of their role(s) in your scene design. Container names can be any combination of uppercase and lowercase letters, or numbers. However, they cannot contain any of the following special characters:

- $ (dollar sign)
- / (slash mark)
- \ (backslash)
- @ (at sign)
- * (asterisk)
- ? (question mark)
- % (percent)
- " (quotation mark)
- { } (curly brackets)

5.3.2 Container Icons



Every Container is shown with a name and three basic icons:

-  **Show/Hide**: Shows if the Container and all its Sub-Containers are visible/invisible in the Scene Editor. Click the icon to change the state.
-  **Locked/Unlocked**: Shows if the Container is locked so that properties cannot be manipulated in the Scene Editor, or unlocked so that properties can be manipulated. Click the icon to change the state.

> ⚠ **Example:** An unlocked item can be resized and repositioned in the Scene Editor.

-  **Transformation Editor**: Opens the Transformation Editor in the Container view.
  A Container that has Sub-Containers is marked with a black arrow to the left of the Container.
-  **Collapse/Expand**: If the arrow points to the right, the Container is collapsed and the Sub-Containers are not shown in the Scene Tree. If the arrow points downwards, the Container is expanded and the Sub-Containers are shown in the Scene Tree. Click the icon to change the state.
  The following icons are only available if the respective items are applied to the Container:

> ⚠ **Note:** Materials and geometries can be dragged from a Container to the Database area. This will create new independent instances of these items in the database. Dragging an image from a Container to the Database area will only create a linked item.

- **Geometry items**: Shows the geometry held by the Container (e.g. a merged or imported geometry). The thumbnail icon reflects the geometry.
  
- **Material items**: Shows that a material has been applied to the Container. The color of the icon reflects the color of the material and its properties. If there is no material icon, a standard white base color is used for the Container. Click the icon to open the Material Editor (see also Working with Material and Material Advanced Items).

> ⚠ **Note:** A Sub-Container can inherit color from a parent Container.

- **Image items**: Shows that an image has been applied to the Container. The thumbnail on the icon reflects the image. There are two ways to apply an image to a Container. If the Container holds geometry, the image is used for texturing the geometry. Click the icon to open the Image Editor.
  

- **Font items**: Shows that a font has been applied to the Container. Click the icon to open the Text Editor.
  The following icons are only available if the respective plug-ins are applied to the Container:

- **Geometry Built Ins**: Shows the geometry held by the Container. The icon reflects the built-in Geometry plug-ins, in this example a cone.

- **Container Built Ins**: Shows that the Container holds a Container function. The icon reflects the Container plug-ins, in this example an alpha.

- **Media Assets:** Shows that the Container holds a Media Asset. The icon reflects the Media Assets, in this example a Clip Channel.

- **Animation**: Shows that the Container contains animated items (see Create Animations).

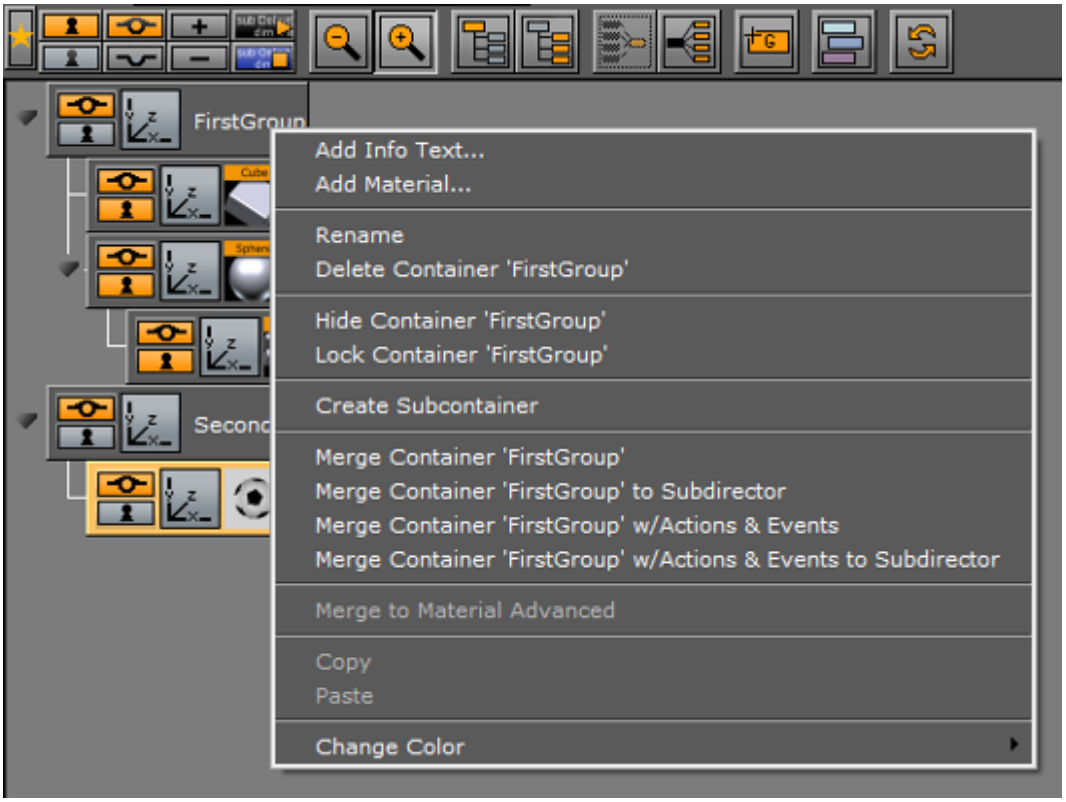
5.3.3 Container Context Menu

A right-click on a Container in the Scene Tree opens a Container context menu.

This section contains information on these topics:

- - Right-click on a Container Name
  - Right-click on a Container Property
  - Right-click on a Container Media Asset Property

Right-click on a Container Name



Right-click on the Container name to show these context menu options (dependent on the Container contents):

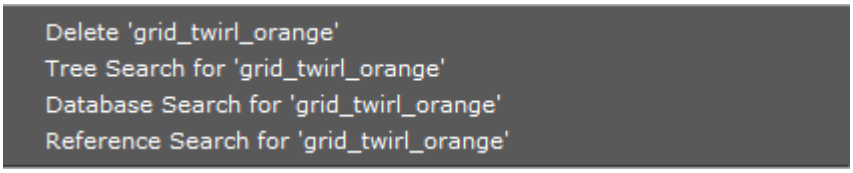
- **Add Info Text**: Makes it possible to add text-based information on a Container. Also available as a plug-in by clicking **Built Ins** and then double-clicking the **Container plug-ins** tab.
- **Add Material**: Adds a Material to the selected Container. See also Working with Material and Material Advanced Items.
- **Rename**: Makes it possible to change the name of the selected Container.
- **Delete Container**: Deletes the selected Container.
- **Show/Hide Container**: Indicates if the selected Container and all its Sub-Containers are visible/invisible in the Scene Editor.
- **Lock/Unlock Container**: Disables or enables the ability to modify the transformation of the selected Container in the Scene Editor.
- **Create Sub Container**: Creates a new empty Sub-Container for the selected Container.
- **Merge Container**: Merges the selected Container and all its Sub-Containers into one compound object, or extracts an already merged object.
- **Merge Container to Sub Container**: Merges the selected Container(s) and all the respective Sub-Containers into one compound object and creates a new sub director for the merged animation.

- **Merge Container w/ Actions & Events to Sub Director**: Merges the selected Container(s) and all the respective Sub-Containers into one compound object, including actions and events and creates a new sub director for the merged animation.
- **Merge Container w/ Actions & Events to Sub Container**: Merges the selected Container(s) and all the respective Sub-Containers into one compound object, including actions and events.
- **Merge to Material Advanced**: Merges multiple objects to one, creating advanced material(s). See Working with Material and Material Advanced Items.
- **Change Color:** Applies a color to the selected Container. Colorized Containers make the Scene Tree more organized. For example, all font Containers can be colored red, while all Containers that hold images can be colored blue. Four active colors are configured by default.

> ⚠ **Tip:** To modify the color of multiple containers, multi select the required Containers and click on  in the tree .

Right-click on a Container Property

Right-click on a Container property to show these additional menu options:



- **Delete property**: Deletes the property from the Container.
- **Tree Search for property**: Selects the **Search** Button, and searches the Scene Tree for instances of this property.
- **Database Search for property**: Searches for the property in the database, and shows the result in the Server Panel.

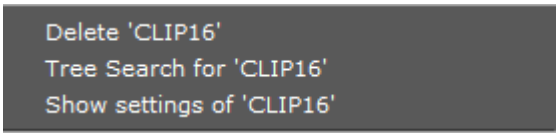
> ⚠ **Note:** This menu option is only available for images.

- **Reference Search for property**: Searches for the property in the Graphic Hub database, and shows the result in the Search window.

> ⚠ **Note:** This menu option is only available for images.

Right-click on a Container Media Asset Property

Do a right-click on a Container Media Asset property, to show these additional menu options:



- **Delete <Media Asset>**: Deletes the Media Asset from the Container.
- **Tree Search for <Media Asset>**: Selects the **Search** button, and searches the Scene Tree for instances of this property.
- **Show settings of <Media Asset>**: Opens the Texture View (Media Asset Manager) and the Media Assets tab of the Scene Settings in the Property panel (see Media Asset Workflow).

5.3.4 Add Containers to a Scene

Containers can be added to the Scene by dragging an item onto the Scene Tree, Scene Editor or Property Editor.

By dragging an item to the Scene Editor, the Container will be added at the top of the Scene Tree.

> ⚠ **Tip:** items can be added to a Scene by dragging them from the result window after a database search.

By selecting multiple items from the Server Panel and dragging them onto the Scene Tree, all of them will be added.

This section contains the following topics:

- To Add Root or Sub-Containers
- To Add Empty Containers
- To Add an Empty Container
- To Add a Container at a Specific Location in the Scene Tree

To Add Root or Sub-Containers

items that are dragged onto the Scene Tree can be positioned in various ways, and create either root or Sub-Containers.

- **Sub-Container**: Dropping an item at the right side of a Container in the Scene Tree will create a Sub-Container.
- **Root Container Below/Above**: Dropping the item at the left side of a Container in the Scene Tree will create a root Container at the same hierarchy level, either below or above the selected Container.

To Add Empty Containers

Clicking the Create Group button (<Ctrl+Insert>) will add an empty Container above the selected

Container. If a Container is not selected it will add the new Container to the root level. The Create Group button is only available if Basic is selected from the Scene Tree menu.

To Add an Empty Container

- Click the Create Group button from the Display area.

To Add a Container at a Specific Location in the Scene Tree

- Drag the Create Group icon and place it as a above or below of an existing Container, or as a Sub-Container.

5.3.5 Select Multiple Containers

More than one Container can be selected in the Scene Tree.

When multiple Containers are selected in the Scene tree, properties can be applied to all the selected items. This multiple selection feature is global.

This section contains the following topics and procedures:

- To Select Multiple Containers
- To Select all Containers
- To Select a Range of Containers

To Select Multiple Containers

- Press <Ctrl> and click on each Container.

To Select all Containers

- With the cursor over the Scene Tree, press <Ctrl+A>.

To Select a Range of Containers

1. Click on a Container.
2. Press <Shift> and click the last required Container.
   All Containers in between are selected.

5.3.6 Copy Containers

To Copy a Container Within a Scene

- Hold <Ctrl> and drag the Container to its new position, or
- Use <Ctrl+C> and <Ctrl+V> to copy and paste the Container.
  - Copied Containers will be placed as Sub-Container of the selected Container.

> ⚠ **Note:** When copying a Container with Sub-Containers, all Sub-Containers are copied as part of the same operation.

To Copy Multiple Containers at Once

- Hold <Ctrl> and the left mouse button and drag the Containers to their new position, and then click the right mouse button to release.

5.3.7 Move Containers

To Move a Container Within a Scene

- Drag it to the new position, or
- Use <Ctrl+X> and <Ctrl+V> to cut and paste the Container(s) to the new position.

To Retain Transformation Properties While Moving a Container Within a Scene

Pressing the <Alt> button while moving a Container (or group of Containers) within the Scene will retain the transformation properties. This lets a Container, which is part of a Group, to be moved to another group while the absolute transformation of the Container will not change, although the parent Containers (Groups) have different transformations.

For example:

1. Group 1: The Group position is X 100.0 and Y 100.0. The Container position is X 0.0 and Y 0.0.
2. Group 2: The Group position is X 0.0 and Y 0.0.
   When a Container is moved from Group 1 to Group 2, while pressing the <Alt> button, the moved Container will inherit its former Group's position values (i.e. X 100.0 and Y 100.0), so its absolute transformation stays the same.

Moving it back, while holding the <Alt> button, its transformation properties will be readjusted. Meaning, if you adjust the moved Container's position X to 200.0, before moving it back to its original Group, its position X will be 100.0 and position Y 0.0 after the move.

1. Add two group Containers to the Scene Tree
2. Rename the groups to Group 1 and Group 2

3. Open Group 1's transformation editor and set Position X and Y to 100.0
4. Add a geometry (e.g. Cylinder) as a Sub-Container of Group 1
5. Add material to the Cylinder
6. Press and hold the <Alt> button and place the Cylinder as a Sub-Container of Group 2
    - When the Container is moved a letter R will show in the drag and drop window.
7. Release the <Alt> button
8. Open the Cylinder Container's transformation editor and set Position X to 200.0
9. Press and hold the <Alt> button and place the Cylinder as a Sub-Container of Group 1 and check that the transformation properties are:
    - Group 1: Position X 100.0 and Y 100.0
    - Cylinder: Position X 100.0 and Y 0.0

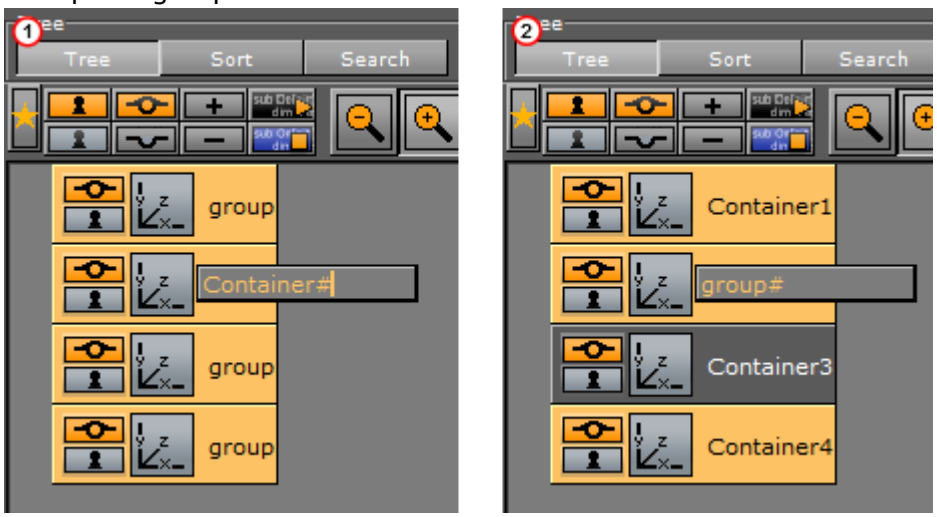
5.3.8 Rename Containers

To Rename a Container

- Double-click the Container name, or
- Right-click a Container. Click on Rename in the Container context menu.

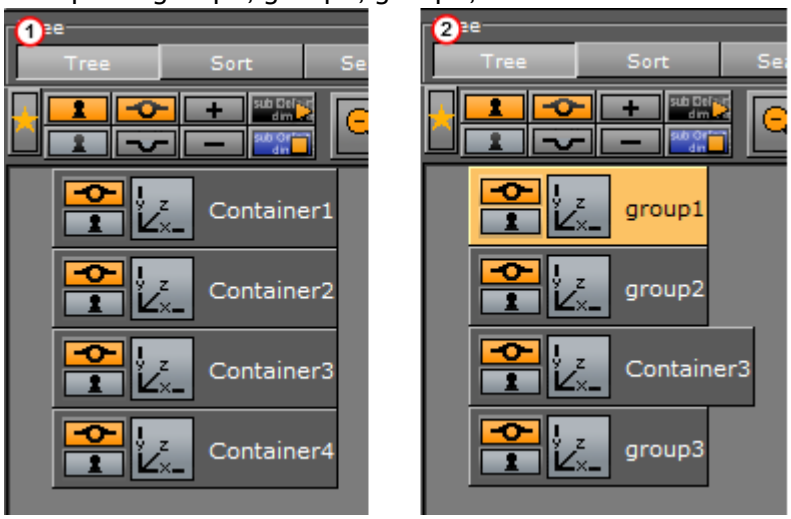
To Rename Multiple Containers

1. Multi select the required containers.
2. Rename one of the selected containers and add # to the end of the text:
    - Example 1: Container#
    - Example 2: group#



3. Press <Enter>.
4. All selected Containers will be renamed.
    - Example 1: Container1, Container2, Container3, etc.

- Example 2: group1, group2, group3, etc.



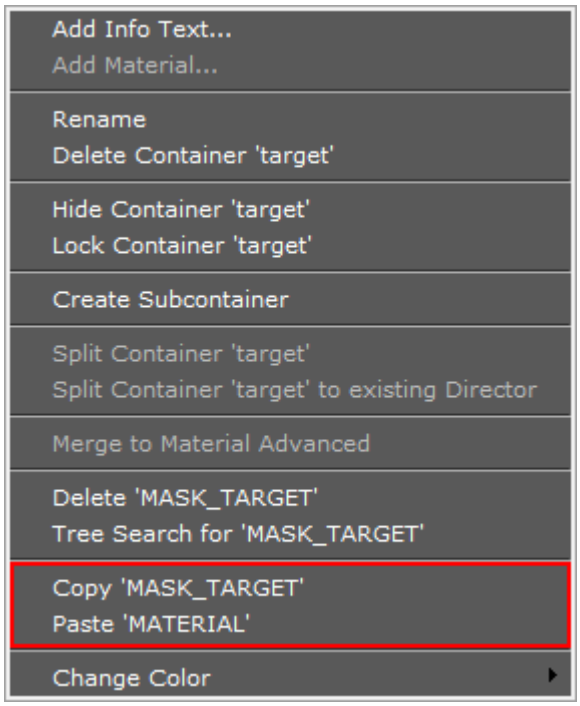
5.3.9 Edit Container Properties

To Edit the Properties of a Container

- Select the Container in the Scene Tree, or in the Scene Editor. Its editor will then show in the Properties Panel (see Modify Container Properties).

To Copy a Property from one Container to Another

- Drag the property from the source Container to the target Container, or use the context menu.



5.3.10 Delete Containers

To Delete a Container from a Scene

1. Drag a Container to the trash can, or
2. Right-click the Container, and in the context menu select Delete Container [Name], or
3. Use the shortcut keys <Ctrl+Delete>.

> ⚠ **Note:** If deleting a parent Container, all Sub-Containers will be deleted too.

5.3.11 Merge Containers

Containers can be merged into compound objects and saved as reusable components to the Graphic Hub database. Any animations will also be merged and added to the new compound object's Container.

A saved compound object can be used in other Scenes and will per default have an active reference between the stored object and all Scenes where it has been used. Through use of stored objects with active references, changes can be applied to many Scenes at once, which will typically be a meaningful way to work when designing a set of Scenes that share many design items.

⚠ **Note:** The geometry of a compound object cannot be edited. A compound object must be split for the geometry to be edited again. Splitting a referenced compound object disables the reference.

Sometimes, when opening a Scene with merged objects, the message "*Old merge style, convert to new merge style*" shows. This is a known issue when combining the Autofollow plug-in with merged objects.

In merged objects the ordering of Containers is reversed which leads to broken autofollow functionality (for first and next reference modes); hence, this merge-style conversion corrects the ordering. Designers that worked around this problem in their Scenes do not need to convert; hence, the conversion is not automated.

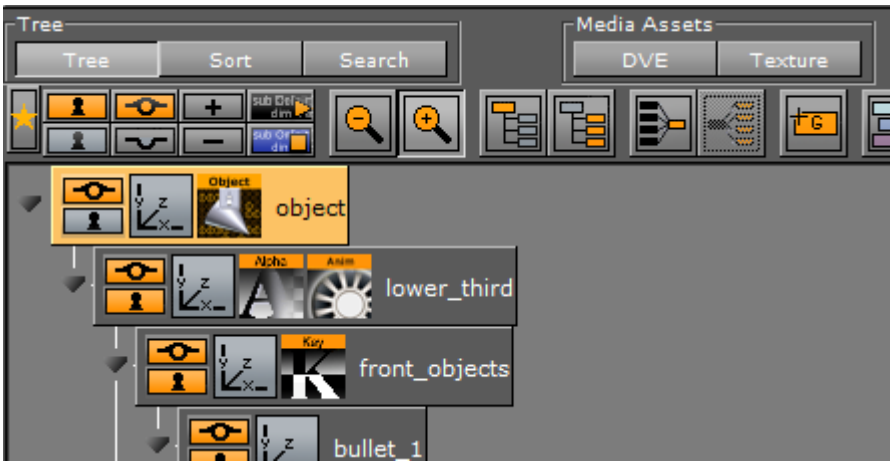
This behavior is not backward compatible because older versions assume reversed sort order. Ungrouping and grouping with older Viz versions on new Scenes will not solve this issue.

To overcome this issue, it is recommended to update all Viz Artist/Engine versions to the official 3.2.2 release and later by opening the Scene and converting it to the new merge style.

This section contains the following procedures:

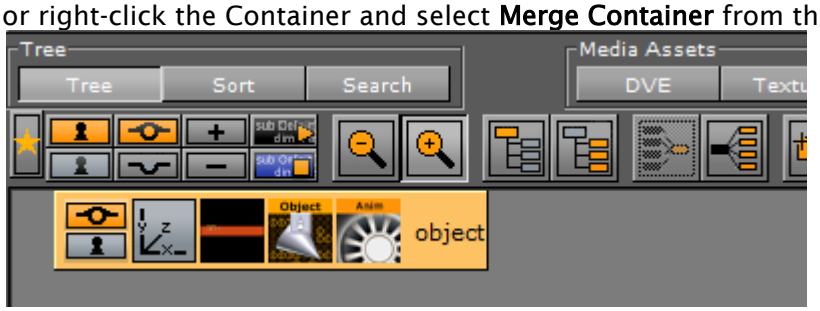
- To Merge Containers into a Compound Object
- To Split a Merged Compound Object
- To Save a Compound Object to a Graphic Hub

To Merge Containers into a Compound Object



1. Select a Container which contains Sub-Containers.
2. Merge the Container with all its Sub-Containers:
   - Click the **Merge** button, or
   - Press <Ctrl+M>,

- or right-click the Container and select **Merge Container** from the context menu.



> **Note:** The Merge button is only available if **Tree** is selected from the Scene Tree menu.

To Split a Merged Compound Object

1. Select a merged Container.
2. Split the Container:
   - Click the Split button, or
   - Press <Ctrl+Shift+M>, or
   - Right-click the merged Container and select **Split Container** from the context menu,

> **Note:** The Split button is only available if Basic is selected from the Scene Tree menu.

To Save a Compound Object to a Graphic Hub

1. Right-click a Container (with or without Sub-Containers),
2. In the context menu select **Merge Container**.
3. Drag the compound object's icon from the property editor or Scene Tree and drop it onto a folder/project in the **Server Panel**.
4. Type a descriptive name for the object in the dialog box
5. Click **Ok**.
   The object will now be visible in under the **Server Objects** tab.

⚠ **Tip:** Any animations or scripts used within the object will be kept when saving.

⚠ **Note:** Make sure to drag only the object icon, not the whole Container.

5.3.12 Group Containers

Containers in the Scene Tree can be organized in a hierarchy and grouped in logical divisions. The hierarchy depth is practically unlimited. Grouping Containers can be compared to creating a parent/child hierarchy. Such a hierarchy provides a better overview, especially in complex Scenes, and is very practical since changing the properties of the group (for example adding a new material) also applies the changes to all the Sub-Containers.

⚠ **Note:** Changing the properties of one of the Sub-Containers (for example adding a material) will override the settings for this particular Sub-Container.

Moving one group is also much easier than moving a lot of single Containers.

A parent Container, that has Sub-Containers, is marked with a black arrow to the left of the Container. If the arrow points to the right (▶), the Container is collapsed and the Sub-Containers are not shown in the Scene Tree. If the arrow points downwards (▼), the Container is expanded and the next level of Sub-Containers are shown.

Any Container can basically be used as parent, although group Containers (that hold transformation only) are best suited.

This section also contains information on the following procedures:

- To Group Existing Containers
- To Ungroup Containers
- To Detach a Single Container From a Group

To Group Existing Containers

1. Select the Containers to be grouped.
2. Click the **Group** (<Ctrl+G>) button from the Display area.
   This adds an empty group Container one level higher than the original Containers.

   ⚠ **Note:** The Group button is only available if Basic is selected from the Scene Tree menu.

To Ungroup Containers

1. Select the parent Container
2. Click the **Ungroup** (<Ctrl+Shift+G>) button from the Display area.

> **Note:** The Ungroup button is only available if Basic is selected from the Scene Tree menu. Also, tree structure further down in the hierarchy will remain.

To Detach a Single Container From a Group

- Drag it from the group and place it somewhere in the Scene Tree.

See Also

- For more information about setting default merge and split behaviors, see the **User Interface** page in the **Configuring Viz** section of the Viz Engine Administrator Guide.
- For more information about how to configure colors, see the **Colors** page in the **Configuring Viz** section of the Viz Engine Administrator Guide.

5.3.13 Container Editor

When a Container is highlighted in the Scene Tree a list of all properties and plug-ins added to that Container, shows in the Container Editor (Container Menu (1)). Click on each icon to view its editor menu. If more than one Container is selected, select which Container to view from the **Selection** panel.

Container Editor Properties

- **Selection**: When multiple Containers are selected, chooses which Container properties to view:
  Either use click the Container name (1) to show a context menu or use the up and down arrows (2), to select a Container.
  
- **Container Menu (1)**: Shows a list of all properties and plug-ins added to the Container selected in the Scene Tree, or the Container selected in **Selection**. Click on an icon to open an editor (see Modify Container Properties)
  - **Show/Hide**: Shows or hides a Container in the Scene Editor, and eventually in the final rendered output. Click to show or hide the Container.
  - **Lock/Unlock**: Enables or disables the ability to modify the transformation of a Container in the Scene Editor. Click to lock or unlock the Container.
    Some editors like the Transformation Editor enable properties for one or multiple

selected Containers to be set. Other editors have the option to enable or disable certain properties.

5.4 Modify Container Properties

When a Container is selected in the Scene Tree, the selected Container property, for example the transformation properties, can be modified in the **Properties** panel.

Items stored in the Graphic Hub database can be added to Scenes:

- Geometry or Fontstyle items will create a new Container using the item as geometry.
- Materials and Audio clips can only be applied to existing Containers.
- Images can be added as a geometry of a Container, or applied on an existing Container as a Texture.
- Geometry plug-ins can also be added to the Scene.
  See Container and Scene Properties for details on how to add properties to Containers.

This section contains information on the following topics:

- Multiple Selection Container Properties
  - To Add Properties to Multiple Containers
  - To Change Specific Properties on Multiple Containers
  - To Select a Container in a Selection
- Edit Properties in the Scene Editor
- Plug-in Editor
- Material Editor
  - To Modify a Scene Specific Material
  - To Save the Container Material
- Quick Edit a Material in the Scene Editor

5.4.1 Multiple Selection Container Properties

When multiple Containers are selected in the Scene tree, properties can be applied to all the selected items. This multiple selection feature is global. Select the required Containers then apply the appropriate property.

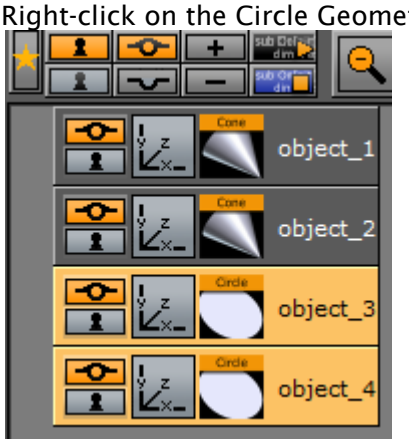
To Add Properties to Multiple Containers

1. Multi select the required Containers.
2. Add an item or plug-in:
   a. Right-click on an item or plug-in the **Server** panel.
   b. Select **Add to Container(s)** (if available).
   or
   - Drag an item or plug-in to one of the selected Containers

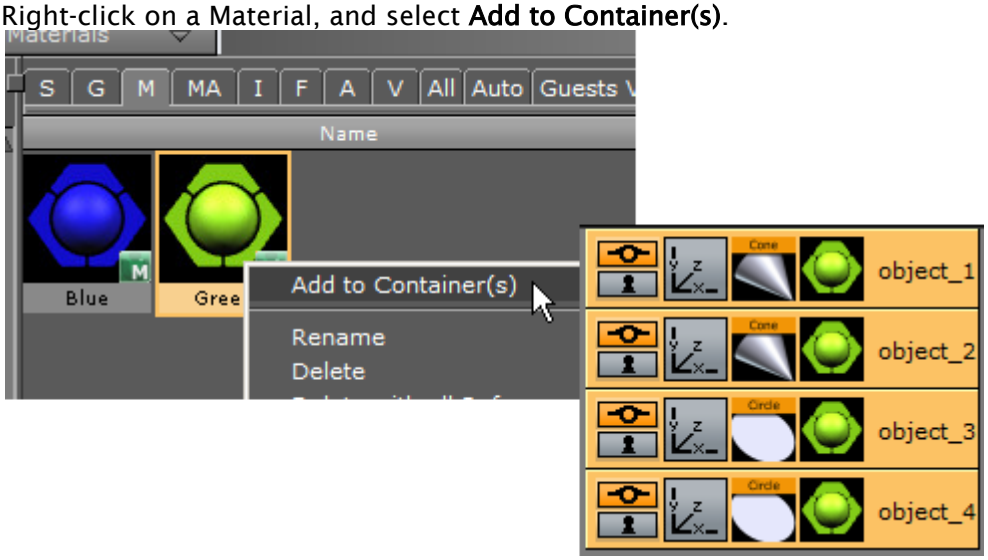
   Example:

1. Create four Containers, named object_1 to object_4.
2. Multi select two Containers.
3. Right-click on the Cone Geometry, and select **Add to Container(s)**.

4. Multi select two Containers.
5. Right-click on the Circle Geometry, and select **Add to Container(s)**.



6. Multi select all four containers.
7. Right-click on a Material, and select **Add to Container(s)**.



To Change Specific Properties on Multiple Containers

1. Multi select the required Containers.

> ⚠ **Note:** If several or all, of the Containers in a Scene are selected and contain the property to be modified, then all selected Container properties of this type will be changed.

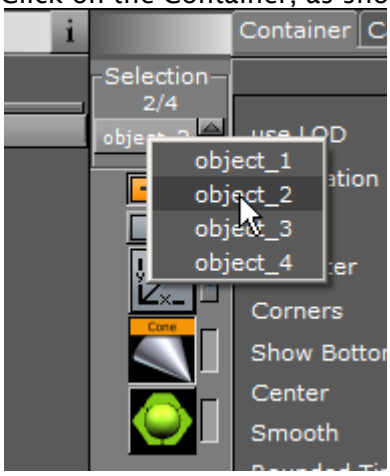
2. Change a parameter of the Circle Geometry.
   The modified parameter will apply to both selected Containers, which contain the Circle Geometry (object_3 and object_4)

If, in the example above, the Material property is modified, the change will show on all selected Containers in the Scene.

To Select a Container in a Selection

Select a different Container in the selection without loosing the multi selection.

- In the **Selection** panel select another Container. Either:
  - Use the up and down arrows, or
  - Click on the Container, as shown in the image below



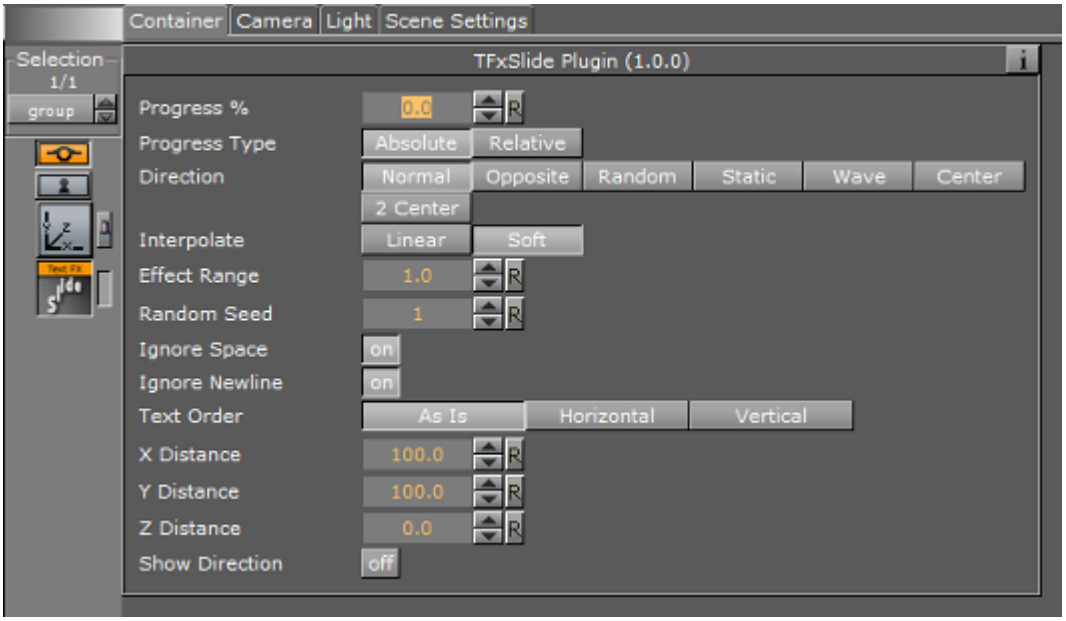
> **Tip:** Visit www.vizrt.com to see a short video illustrating the Viz Artist Multi-selection Features feature.

5.4.2 Edit Properties in the Scene Editor

Some transformation parameters can be changed directly in the Scene Editor through the manipulation of handles.

5.4.3 Plug-in Editor

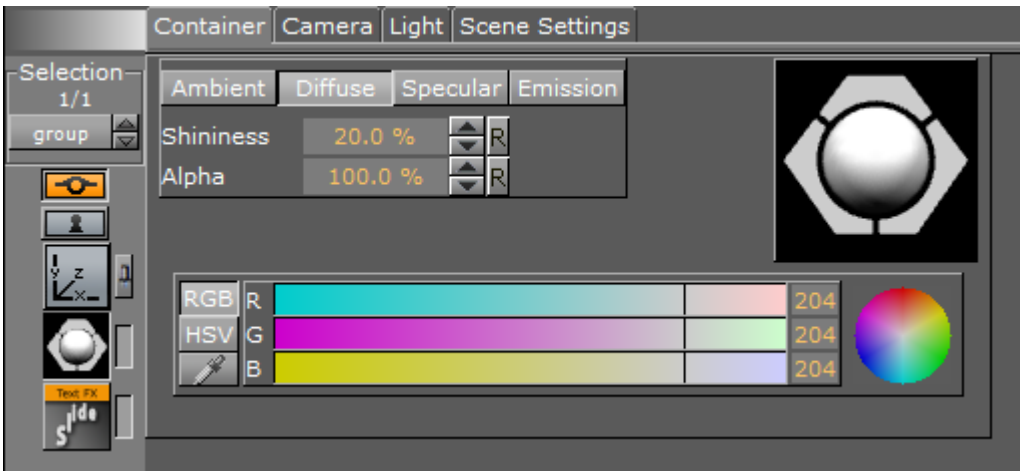
When a Geometry, Container or Shader plug-in is added to a Container its properties will be available in the **Properties** panel. Each plug-in may or may not have properties that can be modified.



Click on the information icon () to show details about the selected plug-in, if available. Additional information may also be available in the Plug-in section of this User Guide, see these sections for more information:

- Geometry plug-ins
- Container plug-ins
- Shader plug-ins

5.4.4 Material Editor

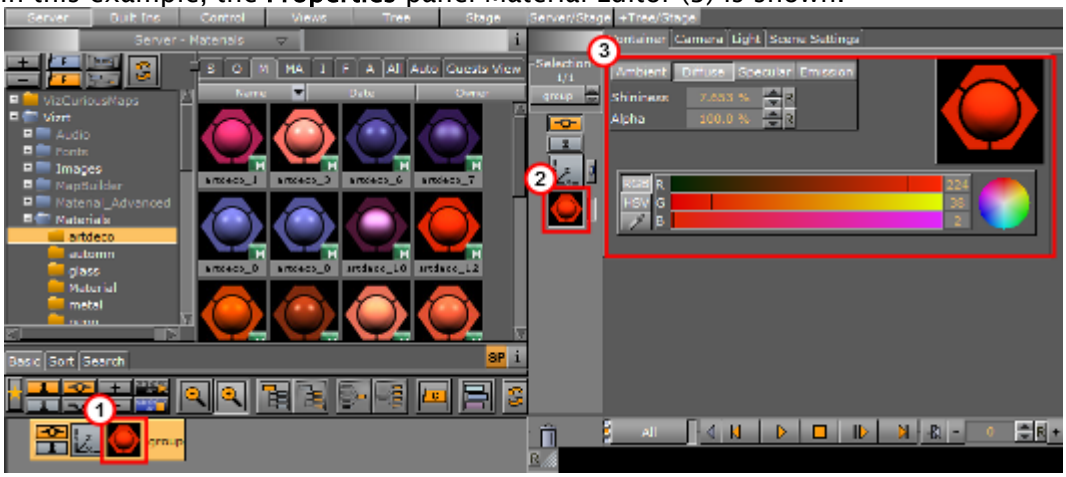


When a Material has been added to a Container, the Container-specific Material can be modified in

the Material Editor that opens in the **Properties** panel. Only the Container-specific Material will be modified, the original Material in the Graphic Hub database will not be affected. See Working with Material and Material Advanced Items.

To Modify a Scene Specific Material

1. Click the Material icon in the **Container** (1) or **Properties** panel (2).
   In this example, the **Properties** panel Material Editor (3) is shown.

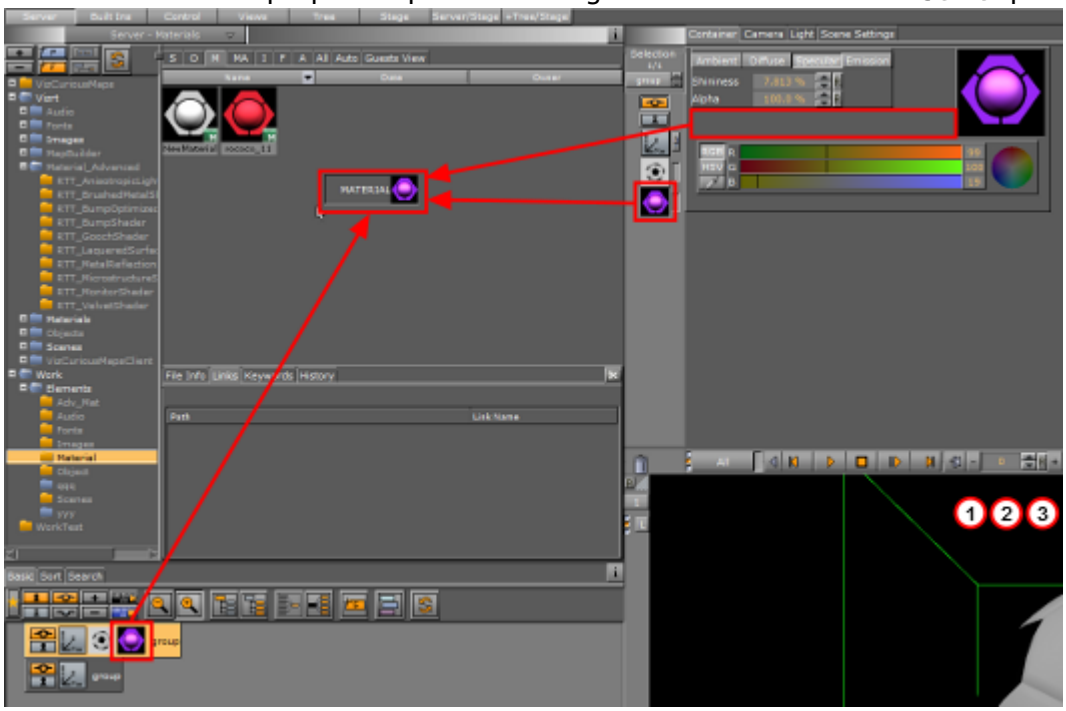


2. Adjust the color properties and schemes as necessary.

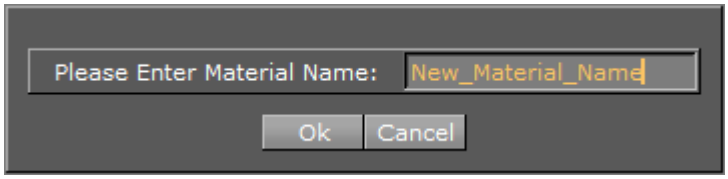
To Save the Container Material

1. Move the modified Material to the **Server** panel:
   - Drag the material icon from the Container to the **Server** panel, or
   - Drag the material icon from the Properties area to the **Server** panel, or

- Click in the Material properties panel and drag the material icon to the **Server** panel.



2. Enter a name for the material.



3. Click **OK**.

5.4.5 Quick Edit a Material in the Scene Editor

To quick edit a material in the scene tree, click with the middle mouse button over the container material. This opens a quick editor for colors directly inside the scene tree, enabling immediate color changes.

- Pressing the **TAB** button switches between the R, G and B values.
- Pressing **CTRL+TAB** switches between the Ambient, Diffuse, Specular and Emission colors.

5.4.6 Manipulate Container Properties

Container properties are items and plug-ins which have been applied to a Container. There are a few operations that apply to all Container properties.

This section contains information on the following procedures:

- To Replace a Property Applied to a Container
- To Remove a Property from a Container
- To Save Properties in the Database
- To Enable/Disable Properties on a Container

To Replace a Property Applied to a Container

- Drag the new item or plug-in to the Container.

To Remove a Property from a Container

- Drag the property from the Container to the trash can, or
- Drag the property from the Container Editor to the trash can, or
- Right-click the property on the Container in the Scene Tree, and select **Delete <Name of the Property>**.

> ⚠ **Note:** Every Container must contain transformation properties, so the transformation property cannot be removed.

To Save Properties in the Database

If a scene specific property has been created, and it is likely that the property should be reused in another scene, it is possible to save the property in the database.

- Drag the property from the Container to the required folder/project in the Server Panel.

To Enable/Disable Properties on a Container

> ⚠ **Note:** item properties are automatically enabled when added to a Container.

This is a practical tool when multiple items are applied to a Container.

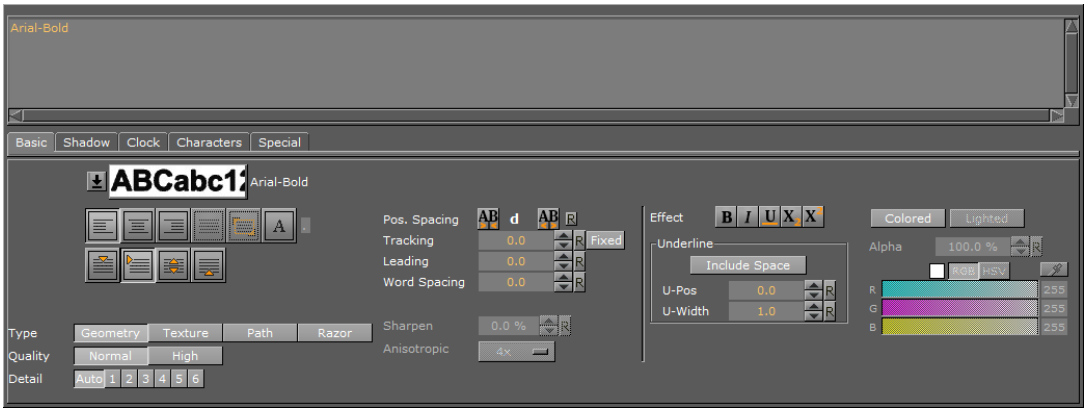
- Click the switch next to the icon which represents the item in the Container Editor.

See Also

- Containers

5.5 Text Editor

Use the Text Editor to add or modify text in a scene.



At the top of the Text Editor is an input box. Characters in this input box appear in the Scene Editor. By default, the text equals the font name.

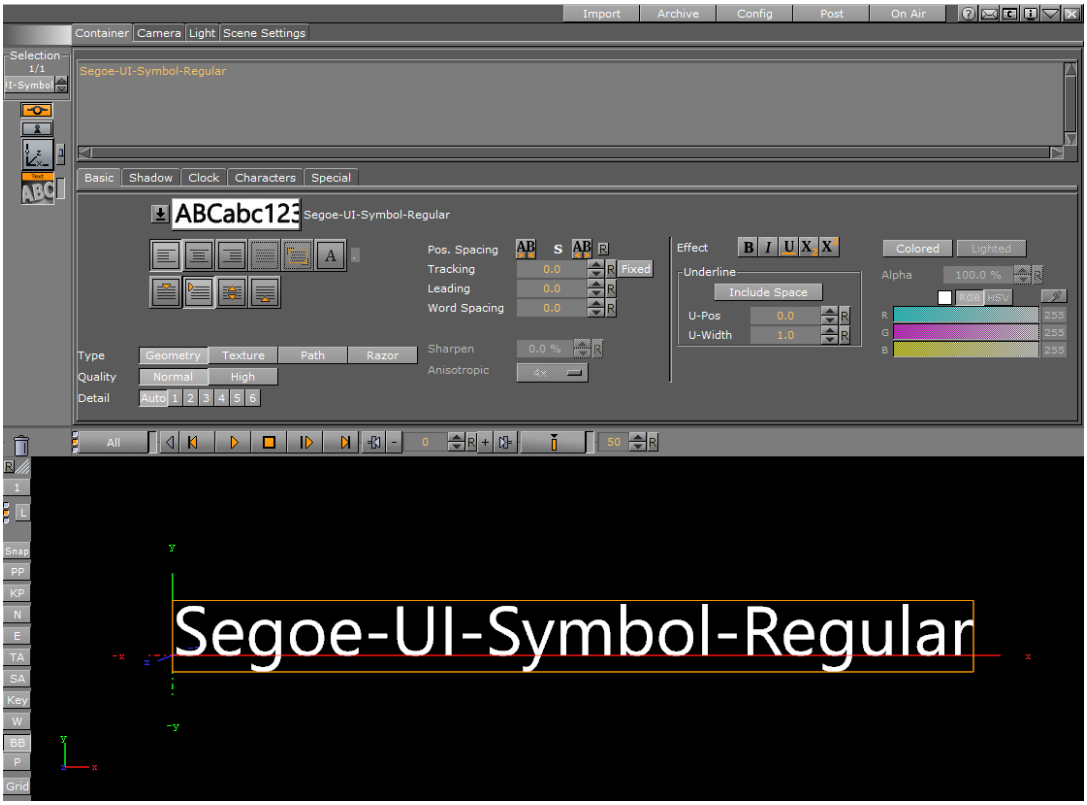
The Text Editor contains five tabs. More advanced modifications can be done in the Fontstyle Editor. Modifications made with the Fontstyle Editor apply to *all* instances of the font, not just the one used in this Scene.

This section contains information on the following procedures and topics:

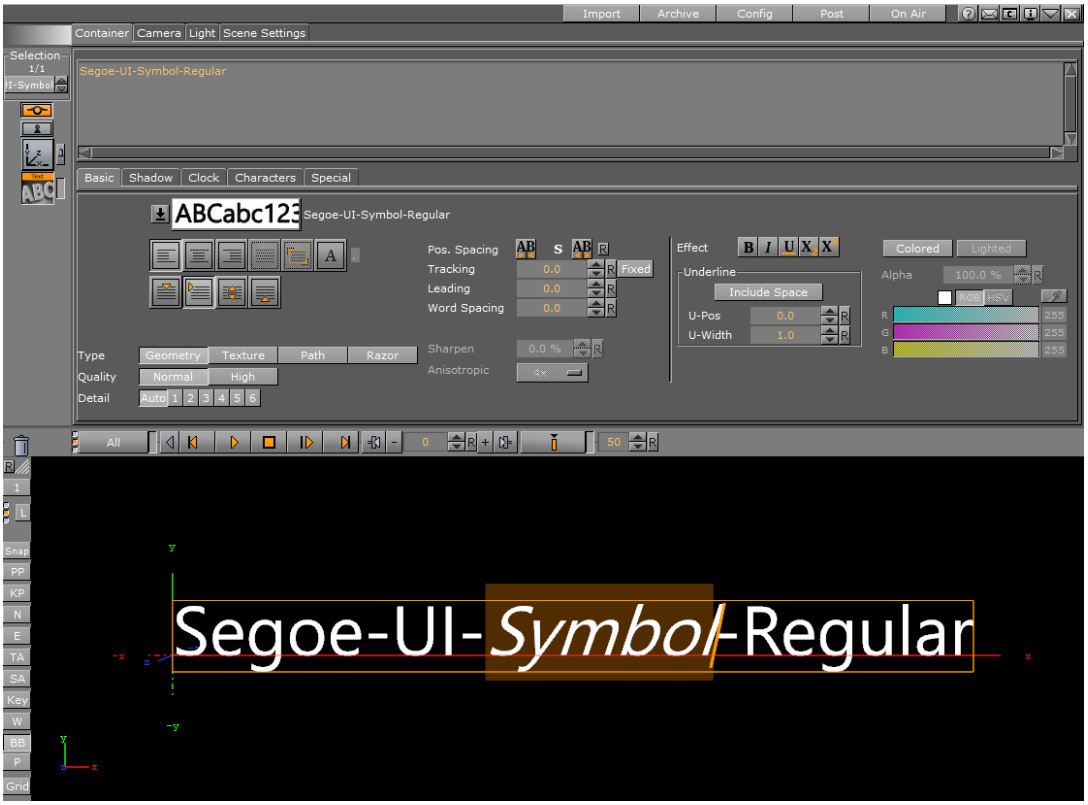
- Text Editor/Scene Editor
  - Colorizing individual characters
- Basic Tab
- Shadow Tab
- Clock Tab
  - Clock Format Examples
- Characters Tab
- Special Tab
  - Special Fonts
- To Create a Linked Fontstyle
- To Modify Text in the Text Editor

5.5.1 Text Editor/Scene Editor

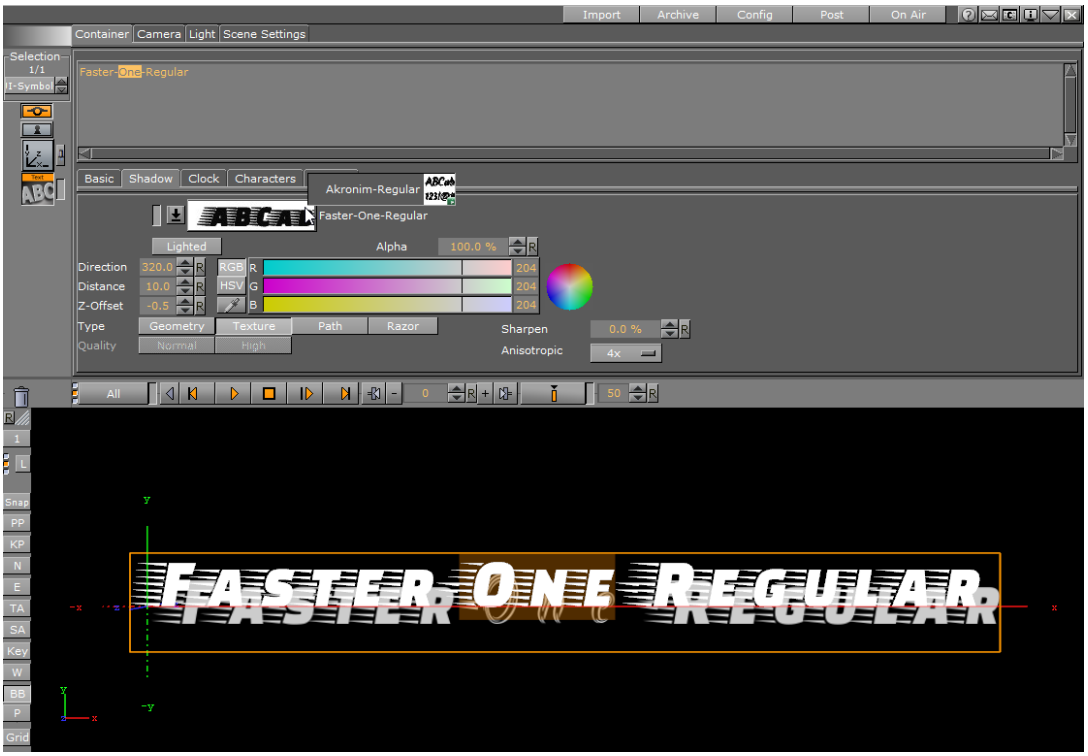
Use the Scene Editor to type text changes directly in your output window. When you double click on the text, you enter text edit mode, and the bounding box color of the text turns brown.



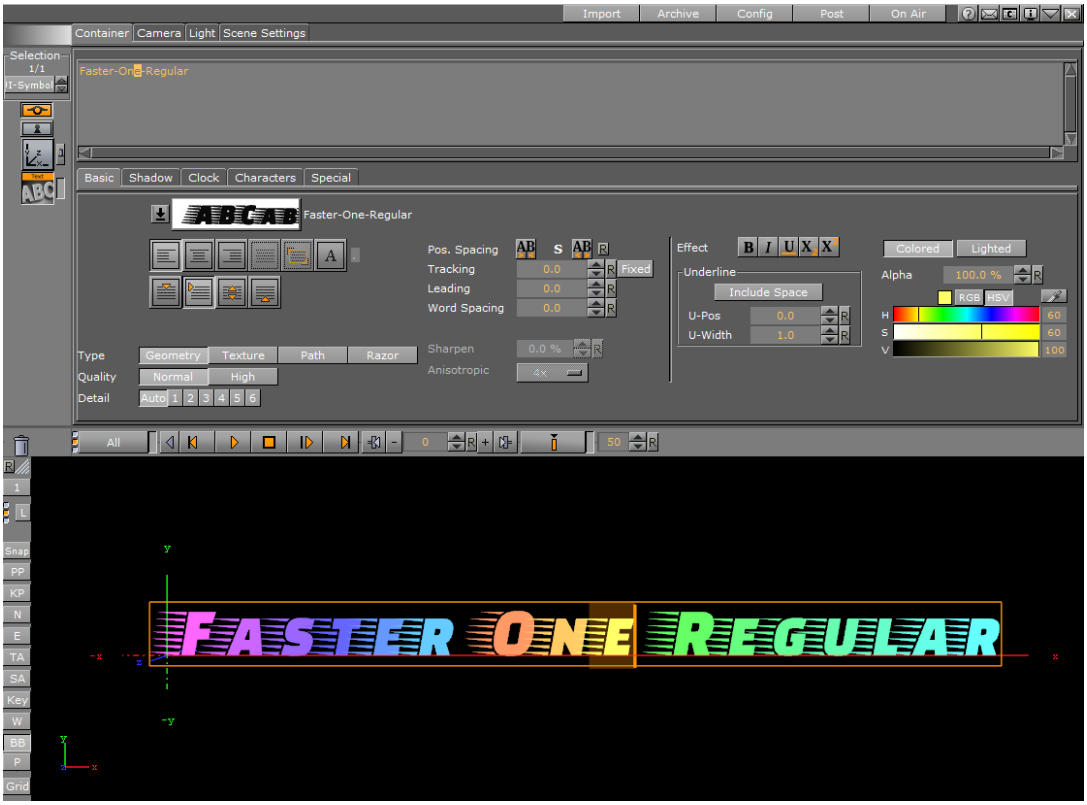
By double clicking on the text in the Scene Editor, you can edit the text directly in the output window. In this mode, you can select individual characters or sections of the text, and drag different fonts to the selection to change the font. This also works with color. By single clicking outside of the text element, the text editing function is disabled, and the bounding box turns back to green.



You can also set the shadow of the font for individual characters. To do this, drag the font into the text slot where the shadow is defined. You can do this with individual characters, or with the whole text object.

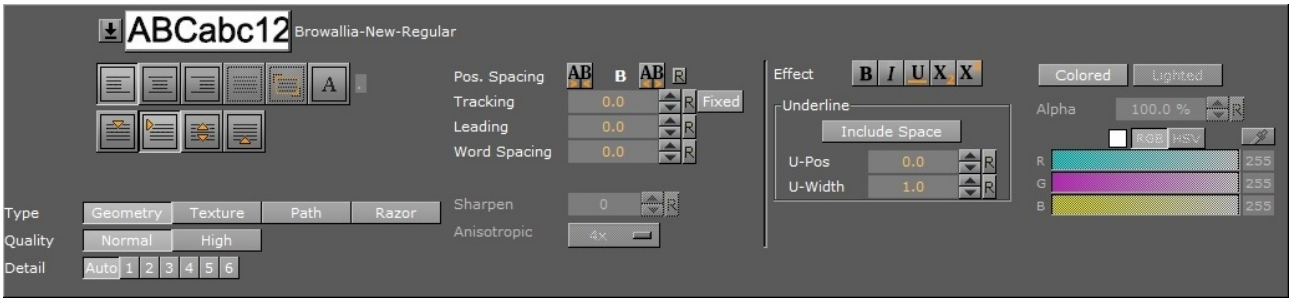


Colorizing individual characters



By double clicking on the text, you enter text edit mode, and the bounding box color of the text turns brown. Now you can select individual characters and colorize them by switching the setting Colored/Lighted in the basic section of the font. When using the Lighted/Colored setting you can colorize each character individual and choose if it should be filled by a solid color (Colored) or by a color reacting to the activated lights (Lighted) like a material.

5.5.2 Basic Tab



- Click the arrow to open a list of all available fonts created from the same base font. To use another font, select it from the list. To use a totally different font in the scene, drag the new font from the **Server** panel onto the visualization of the text.
- Sets the text justification to **left**, **center** or **right**.
- Sets the text, when in a **Text Box** (see Special Tab), to **justified** or **scaled justified**.

> ⚠ **Note:** Justified and Scaled Justified are only available when using a Text Box. If either of these settings are in use when a Text Box is turned off, the text will be set to Left orientation (the default).

- ☐ Defines a character. Typically a decimal separator like comma, and justify the text aligned after this character. Enter the justification character in the small input box to the right of the button.
- ☐ Sets the vertical justification on the top of the whole text block, below the first line, center of the middle line in the text block and at the bottom of the text block.
- **Type**: Sets the rendering method for the text.
  - **Geometry**: Renders geometry type fonts as polygons.
  - **Texture**: Renders the font as texture instead of polygons. This can, to some degree, create a poorer look, but it reduces the number of polygons in the Scene. The texture resources will get an increased demand, so to render a font as texture is a trade-off between polygon and texture performance. For half height mode rendering, texture fonts are normally the better choice. When selected, the anisotropic value is set to 4X by default.
  - **Path**: Renders the font as vector graphics.

    > ⚠ **Note:** To use the **Path** type, the machine must be equipped with NVIDIA GPU and have a driver version 275.33 or higher installed.

  - **Razor**: Renders the font as texture using graphics shader. This method of rendering produce higher quality of anti-aliasing. To use the **Razor** type, the font must have been imported with Viz Artist version 3.11.0 or higher.

    > ⚠ **Note:** A **Razor** type text may produce distorted result when rendering glyphs with high-complexity. Artifacts may appear when two glyphs overlap each other. Lighting is currently not support when using this type.

    > ⚠ **Note:** Some effects, like Extrusion, are not supported by **Texture**, **Path,** and **Razor** rendering.

- **Quality**: Sets the quality of the font to either *Normal* or *High*. The quality of a texture font can be changed in the Fontstyle Editor.

  > ⚠ **Note:** The Quality option is only applicable if the font type is set to **Geometry**.

- **Detail**: Sets the level of detail, or tessellation of the font, to either automatic level or one out of six fixed settings. The fonts are rendered as polygons, so by switching to wireframe mode, the difference becomes obvious as the detail level changes. One is the highest level of detail, six is the lowest. If setting a fixed level of detail, the automatic level will be overwritten and rendered with the same level of detail independent of the font size.

> ⚠ **Note:** The Detail option is only applicable if the font type is set to **Geometry**.

- **Pos. Spacing**: Local kerning of the font style. Clicking the left button will decrease the position spacing between characters. Clicking the right button will increase the position spacing between characters. Fonts can be permanently modified using the Fontstyle Editor.
- **Tracking**: Changes the tracking between all characters by the same amount. Clicking the Fixed button changes the text from true type to one where each character uses the same space.

> ⚠ **Note:** To change the kerning between two individual characters only, place the cursor between those characters, and then use ARROW UP/DOWN while holding <Alt> to increase/decrease the kerning. Press <Alt> and <Shift> to increase the size of the steps.

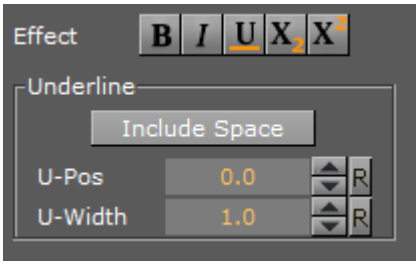
- **Leading**: Modifies the distance between the text lines.
- **Word Spacing**: Changes the size of the blanks between the words in the text.
- **Sharpen**: Makes it possible to sharpen the font.

> ⚠ **Note:** Sharpen is only available if font type is set to **Texture**.

- **Anisotropic**: Improves the texture sampling for anisotropic (in this case non-square) pixel filter areas in mipmapping mode, for example a rotated text texture. Choose either to use the standard mipmap level selection, or select the anisotropic filter rate from a number of available settings. Higher values give better image quality at the expense of some performance. The default value for anisotropic filtering is 4X

> ⚠ **Note:** Anisotropic is only available if font type is set to **Texture**.

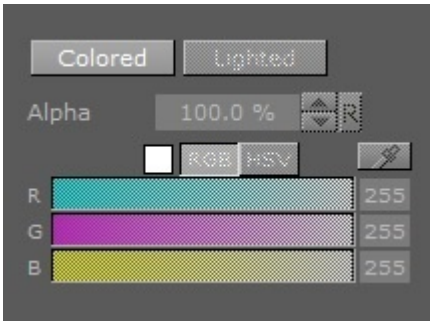
Effects:



- **Effect Bold**: Sets individual characters or the whole text object to bold.
- **Effect Italic:** Sets individual characters or the whole text object to italic.
- **Underline:** Underlines individual characters or the whole text object. In addition, you can choose the position (**U-Pos**) and the width (**U-Width**) of the line.

  X^2: Uses individual characters or the whole text object as superscript text.
- **X_2**: Uses individual characters or the whole text object as subscript text.

Colored/Lighted:



- **Colored:** Activates the selected color for the text as a solid color, overriding the color of any material on the text container in the scene tree.
- **Lighted:** Applies the selected color to the text, adjusted for the scene lighting. Transforming the object will thus create the visual impression of changes to the color. When inactive, the selected color is applied to the text as a solid color.

5.5.3 Shadow Tab

From the Shadow panel, shadow can be added to the text. Click the small switch left of the font visualization to enable text shadowing.

> ⚠ **Note:** All the modifiers that are applied to the text in the Fontstyle Editor are applied to the shadow too.

- ☐ Click the arrow to open a list of all available fonts created from the same base font as selected. To use another font, select it from the list. To use a totally different font in the scene, drag a new font from the Server Panel onto the visualization of the text.
- **Lighted**: Lights the font with the current light settings (see Light Editor).
- **Alpha**: Changes the alpha value of the shadow.
- **Direction**: Sets the direction of the shadow from 0 to 360°.
- **Distance**: Sets the distance between text and shadow.
- **Z-Offset**: Sets the displacement of the shadow on the Z axis.
- **Color**: Sets the color of the shadow.

> ⚠ **Note:** The Color option is only applicable if the color modifier has not been applied to the font.

- **Type**: Sets the rendering method for the shadow (see Basic Tab).
- **Quality:** Sets the quality of the font to either Normal or High (see Basic Tab).
- **Sharpen**: Makes it possible to sharpen the shadow font (see Basic Tab).
- **Anisotropic**: Sets the quality of shadow textures (see Basic Tab).

5.5.4 Clock Tab

Click the **Clock** tab in the Text Editor to open the **Clock** panel.

In the **Clock** panel, the display of a clock can be inserted and edited. Click the small switch in the upper left corner of the panel to enable the clock. The clock will be added to the text string, if any, at the cursor position.

- **Clock**: Sets the ID of the clock. Up to ten different clock IDs (0-9) can be used in a Scene. Clocks with the same ID are linked together, which means they are synchronized and all start when one clock is started. This can be used to show different time zones, for example
- **Update Text**: When enabled, any running clock in the scene will be updated in real time in the **Scene Editor** and **On Air** outputs. If disabled, the output to the Scene Editor and On Air outputs will not be updated, even if the clock is running. An external control application, such as for instance *Viz Trio*, could be used to enable or disable Update Text when running in On Air mode, as required.
- **Time:** Defines the start time in seconds. For example, if the clock should start to count at two minutes, set Time to `120`.
- **Limit:** Defines the count limit in seconds. For example, if the clock should stop counting after two minutes, set Limit to `120`.

  > ⚠ **Note:** If counting down to 0, Limit must be set to 0 seconds, or the clock will continue to count below 0. If counting upwards starting at 0, make sure that Limit is not 0, as the clock will not count at all.

- **Up and Down**: Specifies if the clock should count up or count down.
- **Start, Stop and Continue**: Starts, stops and continues the main clock.
- Select a Format: Set the format of the clock display.
  - **H**: Hours
  - **M**: Minutes
  - **S**: Seconds
  - **D**: Digits of a second. The first digit shows a tenth of a second, the second digit shows a hundredth of a second, the third digit shows a millisecond, and so on.
  - **User Defined:** Click on User Defined and type a non-standardized clock format in the box below. Make sure to add correct separation characters.
    Viz Engine distinguishes between upper- and lowercase clock formats, allowing the designer even more customization options. When defining the clock format by lowercase letters, for instance `hh:mm`, the digit will be shown even if the value is `0`. However, if the clock format is defined by uppercase letters, Viz Engine will not display the value if it is zero. This allows the designer to omit unused digits while still planning for their use when designing the scene.

Clock Format Examples

Clock format:	Displayed result
HH:mm:ss	03:05
Hh:mm:ss	0:03:05
HH:Mm:ss	3:05
HH:Mm:s	3:5
hh:mm:ss	00:03:05
hh:M:ss	00:3:05

⚠ **Tip:** Up to 20 clock formats can be defined in **Clock Formats** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

- **Fixed**: Sets a fixed width for each letter. The kerning of non-proportional fonts vary for the various letters. For example, 1 has less kerning than 6. To avoid the letters from "jumping" around while counting, the Fixed option can be enabled. The Fixed value refers to the size of the text box around each letter.

  ⚠ **Note:** The problem with changeable kerning can be avoided by using a proportional font

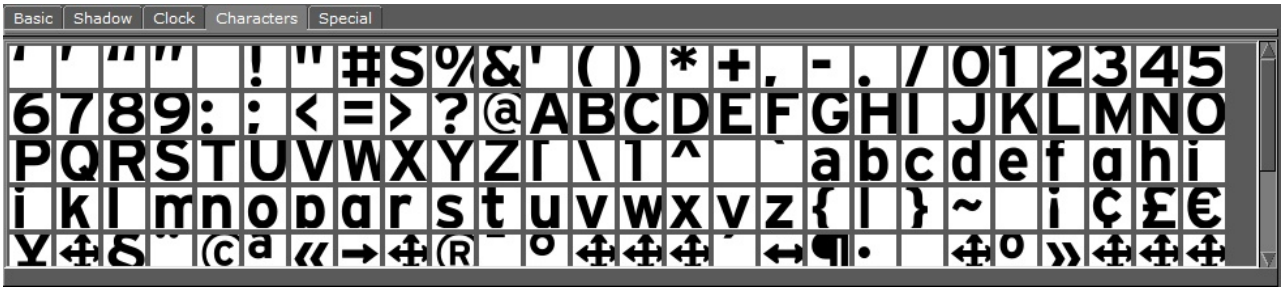
- **Normal**: Shows the normal main clock view
- **Count: Create a counter, which counts a defined time lap in relation to one of the running clocks. For example, if a clock is running, another Container with the same clock can be created. The state of the second clock should be set to Count, and a time range defined. When clicking the Start** button under the **Count** section, the count clock will start counting for the defined time. It is related to the main clock, so if stopping the main clock, the count clock will also stop. A typical use for this function is for counting penalty times in sport graphics. In a sport game running on efficient time, a penalty count down should stop when the main game time stops
  - **No Sync**: When enabled, the count clock will not synchronize seconds to the main clock. It is almost impossible to start the count clock exactly at a whole second, so the count clock will most likely run unevenly compared to the main clock. This will in

many cases not create the desired output, but may be useful if an offset output is desired

- **Sync Sec**: Synchronizes the count clock with the main clock
- **Loop**: Loops the count clock within the defined count range
- **Min/Max**: Sets the count clock minimum and maximum parameters
- **Up and Down**: Specifies if the clock should count upwards or perform a countdown
- **Start, Stop and Continue**: Makes the clock start, stop and continue counting
- **Init**: Resets the clock's counter

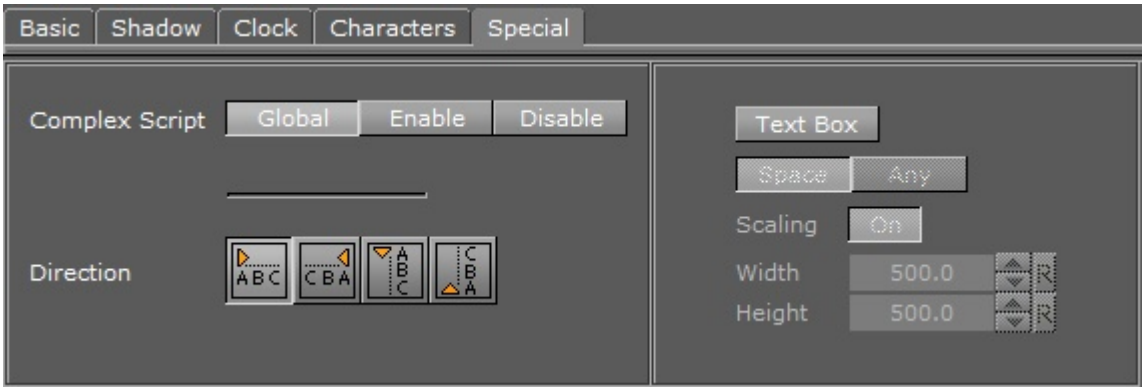
- **Offset: Click the Offset** button to access an input box where an offset value in seconds can be entered. For example, if the value is set to 20, the clock will start running after 20 seconds
- **Sys. Time**: Sets the clock to the system time
- **Local Time**: Sets the clock to the local time. This is useful if running on a system which automatically adjusts to daylight savings time

5.5.5 Characters Tab



The Characters panel shows the character table in the fonts library. As an alternative to the keyboard, the symbols from the table can be dragged onto the input box at the top of the Text editor. This is a helpful tool when using special characters that do not exist on the keyboard, or is hard to find/use.

5.5.6 Special Tab



- **Complex Script:** Chooses whether to enable or disable complex script processing in this text object, or use the global configuration.

Complex Script is required in many conditions

- **Languages:** Renders complex language scripts such as Arabic and Thai more correctly.

- **Handwriting:** Generates a different and better result in some fonts.

- **Symbols:** Renders symbol characters in Unicode Private Use Area (PUA) only in complex script such as [🐰🐑🐉🦁🐼🐯] from Segoe UI Symbol

- **Direction:** Switches the text direction from left to right, right to left, downwards, and upwards.

> ⚠ **Note:** When complex script is disabled, all characters are rendered directly to glyphs according to the **Direction** option. For example, the text "ABC" will be rendered as "CBA" when the direction is right to left.
> On the other hand, when complex script is enabled, ligature and bi-directional text algorithms are also performed, and the **Direction** option dictates the primary writing direction. Thus, the rendered glyphs are ordered in the direction of its native writing system.

- **Text Box**: When enabled, the text will be contained within a Text Box. The text box shows as an orange outline in the Scene Editor (if Bounding Box is enabled in the Scene Editor).
With the Text Box, text can be divided into multiple lines, and the text on each line automatically gets the selected justification (see Basic Tab). Without the text box, the text is shown on one line.

> ⚠ **Note:** Only the text within the Text Box boundaries is shown.

  - **Space**: Each line of text contains whole words.
  - **Any**: Each line of text contains all possible letters. Words are split at the Text Box right border.
  - **Scaling**: Sets the scale the size of the Text Box as scale of the text is changed when enabled.
  - **Width/Height:** Sets the width and height of the Text Box.

Special Fonts

Special options apply for multibyte and geometry fonts:

- **Geometry:** If adding a geometry font, the text will be treated like a compound object created from characters. The text can be split to become a group of single character objects (not text!). Splitting of text makes it possible to edit the properties of single character objects.
  

  ⚠ **Note:** Be aware that it is a one way operation. It is not possible to merge characters into text again, so by performing the splitting operation, the text cannot be edited anymore.

5.5.7 To Create a Linked Fontstyle

1. Click on the Basic tab.
2. Click on and drag the Fontstyle (1) to the target folder in the Server Panel.
   

5.5.8 To Modify Text in the Text Editor

1. Click the text icon,  , either on the Container, or in the Container Editor. You can also drag a font to the scene. This will automatically create a new container containing the font as a geometry, and open the Text Editor opens automatically.
2. Enter text in the text input box. The text is shown in the Scene Editor panel. Right click in the text editor to change the font size.
3. Use the five panels to modify the text as you require.

See Also

- Fontstyle Editor

5.6 Geometry Editor

When adding an Geometry to a Scene, a new Container is created. Geometries that have been created with external 3D programs and imported into Viz Artist can be modified in the Geometry editor.

⚠ **Note:** Face Orientation and Shading are only applicable if the reference is deactivated.

This section contains information on the following topic and procedure:

- Geometry Editor Properties
- To Open the Geometry Editor

5.6.1 Geometry Editor Properties

- **Geometry**: Shows a thumbnail of the Geometry, and also its name. Click the thumbnail to select the Geometry in the database.

  > ⚠ **Note:** This can only be done when the reference to the database is active.

- **Face Orientation:** After importing Geometries that have been created by external 3D programs, it might happen that the polygons face orientation is wrong. By clicking the Flip button, the polygons are set correctly.
- **Shading**: Imported polygon Geometries normally use shading for creating a smooth surface on the object, which is set together by a planar polygon mesh. The shading value in the Geometry editor sets a threshold between which edges are to be smoothed by shading and which are to be kept as sharp edges. All edges with an angle smaller than the defined value are smoothed by shading. Shading in Viz Artist is performed in the following way: A shade value is produced at each vertex of the polygonal mesh representing the surface of the object. This is done by creating a surface normal on each polygonal facet. The surface normal at each vertex is created by averaging the surface normals for the surrounding facets. The shade at each vertex can then be calculated. Once the shade at the vertices of the polygonal mesh are known, the shade at points interior to the polygonal facets are interpolated from the values at the vertices. This technique makes curved surfaces look “smooth shaded” even though based on a representation of planar facets. The interpolation of shade values is incorporated into the polygon scan conversion routine. Hence an increase in realism is obtained at far less performance cost than carrying out a pixel-by-pixel shading calculation over the whole original surface. Shading may be set to Flat, which means that all shading is disabled, or Smooth, which enables the Edge Angle option. After adjusting the Edge Angle, make sure to click the Rebuild button for the changes to take effect.
- **Reference:** If using an imported or compound object in a Scene, there is a reference between the object in the Scene and the object in the database. If changes are applied to the object in the Scene, the object in the database including any Scenes where this object is used with an established reference, will reflect those changes, as it is the same object.
  - **Deactivate:** To create a unique copy of the object for the Scene, the reference link must be deactivated. Changes that are applied to a deactivated object do not lead to changes in the original object in the database.
  - **Re-establish**: Reactivates the reference and overwrites the object in the Scene with the object in the database. An active reference is practical when creating a set of Scenes with the same object. If for instance all the Scenes should have the same background and the background item is created as an object with active references, changes can easily be applied to all the Scenes that use that object.
  - **Update:** Reactivates the reference and overwrites the object in the database with the object in the Scene.
  - **Status**: Shows the current status of the reference.

> ⚠ **Note:** The reference is automatically deactivated if splitting the object. If the object is split, it must first be merged before the Re-establish and Update operations can be performed.

5.6.2 To Open the Geometry Editor

- Click the icon representing the geometry either on the Container or in the Properties editor.

5.7 Transformation Editor

Every Container in a Scene contains transformation properties. The transformation editor holds information about the most basic properties of a Container, such as position and rotation.

The transformation editor is also able to stick transformation properties to the container view. This feature allows Position, Rotation, Scaling and Axis Center parameters to stick to the Container view while opening other editors. This enables designers to adjust transformation properties while working on other objects (e.g. text, geometry).

This section contains information on the following topics and procedures:

- Transformation Editor Properties
  - To Open the Transformation Editor
  - To Stick Transformation Properties to the Container View
- Transformation Principles
- Multiple Select Transformation
- Copy Transformation Properties to Another Container
  - To Copy Transformation Properties to Another Container
- Quick Editor
- How to use the Quick Editor

5.7.1 Transformation Editor Properties

- **Parent / Local / World**: See Transformation Principles.
- **Absolute / Relative**: See Transformation Principles.
- **Position**: Sets the position of the object along the X, Y, and Z axis
  - **CC** (Center Camera): Clicking this button centers the Container to the camera
- **Rotation**: Rotates the object according to the X, Y, and Z axes. With the bounding box option enabled, the rotation axes will be visible as three lines; X is red, Y is green, and Z is blue. To rotate in steps of 45 degrees, use the buttons labeled **45** and **+45**
  - **FC (Face Camera)**: Clicking this button changes the rotation in such a way that the object faces the camera
  - **Order**: Clicking the order buttons sets the order of rotation the three angles **X**, **Y**, and **Z**, are applied to the Container. If multiple Containers are selected, the order buttons are disabled

- **Scaling**: Before changing the scale values of the Container, make sure to set the preferred kind of scaling:
  - **S (Single)**: Scales each axis by itself
  - **L (Locked)**: Scales all axes at once, so that X, Y, and Z have the same value
  - **P (Proportional)**: Scales all axes at once, so that the values remain proportional
- **Screen Position**: Sets the screen position along the X and Y axis for images and some objects by specifying pixels instead of the regular positioning which do not use pixels as input values. The Screen Position option is only available if the image or object faces the camera. To achieve this, click the Face Camera button. When changing the Screen Position values, the values for the regular positioning will also change. Set the screen position to left, center, or right by clicking **L**, **C**, or **R**; or bottom, center, or top by clicking **B**, **C**, or **T**.
  In Viz Artist, position calculations are always from the upper left corner and based on the current Z value for the Container and camera views. For design purposes, it may sometimes be desirable to position an objects *center*. Therefore, screen positioning of objects can be defined by either:
  - **Bounding Box**: Defines the position of the object upper left corner, or
  - **Center**: Defines the position of the object center
    Both are calculated from the upper left corner of the current view.

    > ⚠ **Example:** Setting an objects screen position Bounding Box values to X=0 and Y=0 will place the object in the upper left corner. Note however, that if the Z position changes the object will no longer align to the viewpoint corner. The pixel values must be set to zero again to reposition the image or object.

  - **Align to Px (Pixel)**: When using the normal positioning editor to position an image/object that is facing the camera head on, it can occur that the image/object is off by fractions of a pixel. This can happen because of rounding errors in the screen size and face camera functions. This can produce visual artifacts in some cases on some graphics cards. When the image/object is not aligned to the screen pixels, Align to Px button turns red. Clicking the button adjusts the image/object to be moved to the nearest matching whole pixel.
- **Axis Center**: Sets the axis origin of the object. The rotation axis of the object will be visible in the Scene Editor if the Bounding Box option is enabled.
  - The origin can be set to left, center, or right on the object's X axis by clicking L, C, or R.
  - The origin can be set to bottom, center, or top on the object's Y axis by clicking B, C, or T.
  - The origin can be set to back, center, or front on the object's Z axis by clicking B, C, or F.
  - When a group is selected but certain Containers in it are not visible, clicking C with the middle mouse button will define the axis center only for those Containers in the group which are visible.
- **Screen Size**: Makes it possible to scale an object/image based on pixels. The Screen Size option is only available if the object/image faces the camera. To achieve this, click the Face Camera button.

    - **Screen**: Scales the object/image to the size of the output format.
    - **Image**: Scales the image to the original pixel value.
- **Additional Information on the Selected Container**
  Click the Expand button to show this information to the right of the transformation editor. This information includes parent Containers plus all visible child Containers.
    - **Width, Height, and Depth**: Shows the real size of the object in units (not pixels).
    - **Containers**: Shows the number of Containers that are selected. If the selected Container holds Sub-Containers, all Sub-Containers are counted in addition to the main Container. A Container that holds geometry is counted as two.
    - **Vertices**: Shows the number of vertices that are rendered within the selected Container, including the vertices in all Sub-Containers.
    - **Primitives**: Shows the number of primitives in the Container including the primitives in all Sub-Containers.

To Open the Transformation Editor

1. Click the Transformation icon on the Container, or
2. Select the Container from the Scene Tree, and then click the icon located on the left side of the Property panel.

To Stick Transformation Properties to the Container View

- Click the button (1) next to the transformation editor button in the Container view.
  

5.7.2 Transformation Principles

Containers can be manipulated in several contexts:

- **Parent/Group:** Refers to changes that can be made to the Container or Containers in its Parent or Group coordinate system.

  > ⚠ **Note:** Multi selected objects: When more than one object is selected Group will show and a bounding box will cover all selected objects.

- **Local:** Refers to changes that can be made to the Container in its local coordinate system.
- **World:** Refers to changes that can be made to Containers in their world coordinate system.
- **Absolute:** Refers to changes that can be made to Containers in absolute values.
- **Relative**: Refers to changes that can be made to Containers relative to current values.

5.7.3 Multiple Select Transformation

With multiple select transformation, a group of Containers can be moved to a new position, which is determined by parameters of one selected Container in the selection. All Containers keep their relative position to the selected Container.

All changes made in the Transformation Editor will be on each selected Container:

- **Group** is selected and **M** (multi) will show with the mouse pointer when inside the Transformation editor.

  The selected Container and its position in the group is shown in the Selection panel (i.e. 1/6, 3/6).

In the Container Editor use the up and down arrows (2) to scroll through the selected Containers, or click on the Container name (1) to show a context menu. Change the parameters, as required, for the selected Container.

5.7.4 Copy Transformation Properties to Another Container

If a Container has certain settings, for example position or rotation, and another Container in the Scene should have identical settings, it is possible to drag transformation properties from one Container to the other.

All transformation properties can be copied to another Container, either the whole property (for example, **Screen Size**), or just one axis of a property (for example, **X** of **Screen Position**).

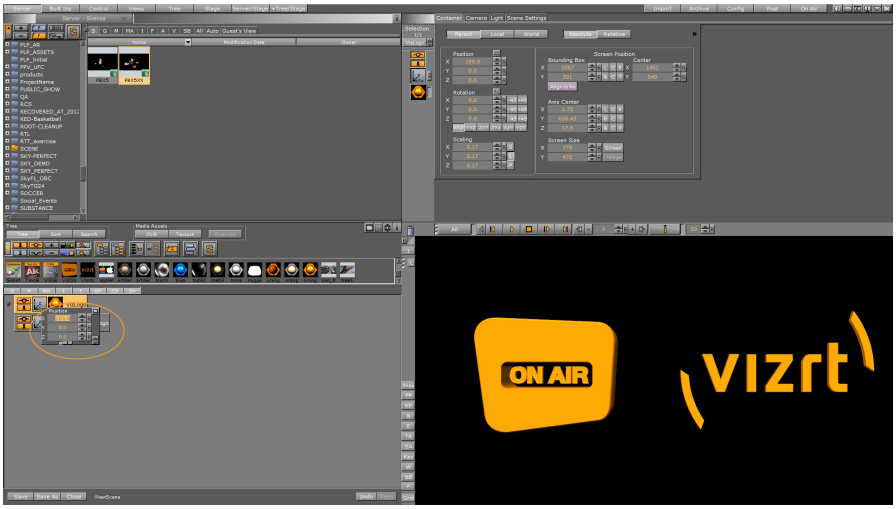
To Copy Transformation Properties to Another Container

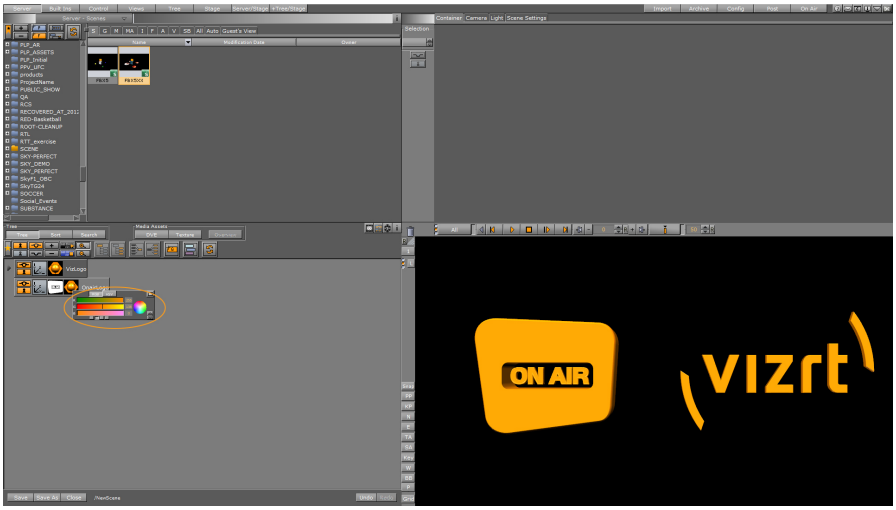
1. Open the Transformation Editor of the **source** Container (4).
2. Drag the required property to the **target Container** (3):
   - Drag the whole property (1), or
   - Drag a sub-property (2 (X, Y, or Z))

> ⚠ **Tip:** Transformation properties can also be dropped on multiple Containers. Drag a selected property to the Scene Tree (aligned with a target Container). Hold the left-click and right-click to add the property to the target Container. Keep the left-click and repeat as required.

5.7.5 Quick Editor

The Quick Edit menu lets you edit the transformation and material settings of a container within the scene tree, instead of in the Transformation Editor. Click the middle mouse button on the container's transformation or material icon to open the Quick Editor. When using the Quick Editor, you can still have the Transformation Editor open and an active object selected. The Quick Editor activates for the object you click on in the scene tree. When you select an object's transformation icon with the middle mouse button, the Quick Editor opens and overlays half your object, to indicate which object is selected.



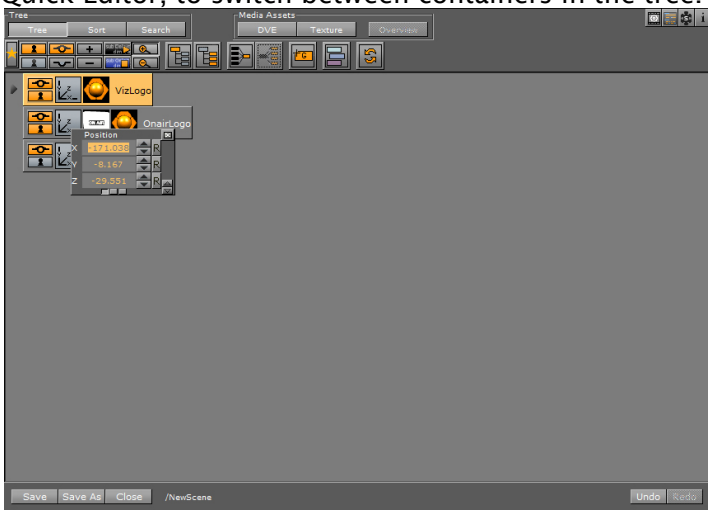


5.7.6 How to use the Quick Editor

1. Create or open a Viz scene with some objects where you can transform the material color and it's transformation values.
2. Click with your middle mouse button onto the geometry icon or the material icon. The Quick Editor pops up, overlaying half of the clicked object. In this example, the Quick Editor is for

the **VizLogo** object:

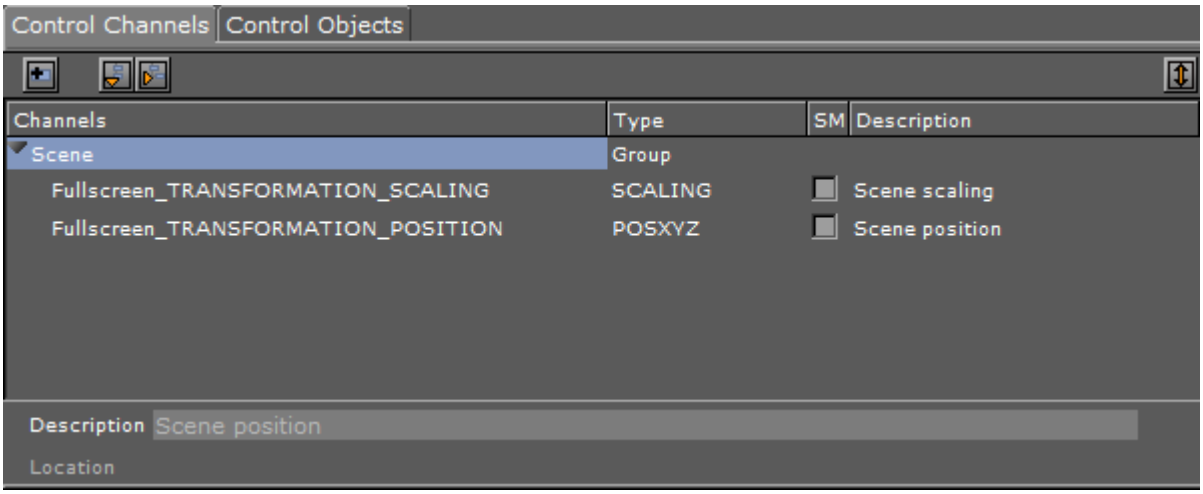
3. By clicking the middle mouse button on the transformation icon of another container, the selected container is still highlighted as active. The Quick Editor updates the transformation values of the container you clicked on, as seen with **OnAirLogo** in this example. You can also use Page Up and Page Down on your keyboard, or the small arrows on the left bottom of the Quick Editor, to switch between containers in the tree.



4. You can switch between the different transformation properties by clicking the small squares on the bottom of the Quick Editor, or by pressing CTRL+TAB.

5.8 External Control

Parameters in a Scene can be exposed for use with external controls, meaning these exposed parameters can be modified by an external control application, such as Viz Libero, Viz Trio or Viz Pilot.



The Viz Artist GUI provides two options to expose parameters and controls:

- **Control Channels**: Provides a convenient way to expose parameters from the Properties panel in a Scene
- **Control Objects**: Shows exposed parameters when a Control plug-in is used (for use with Viz Trio only)

This section contains information on the following topics:

- Control Channels
- Control Objects

5.8.1 Control Channels

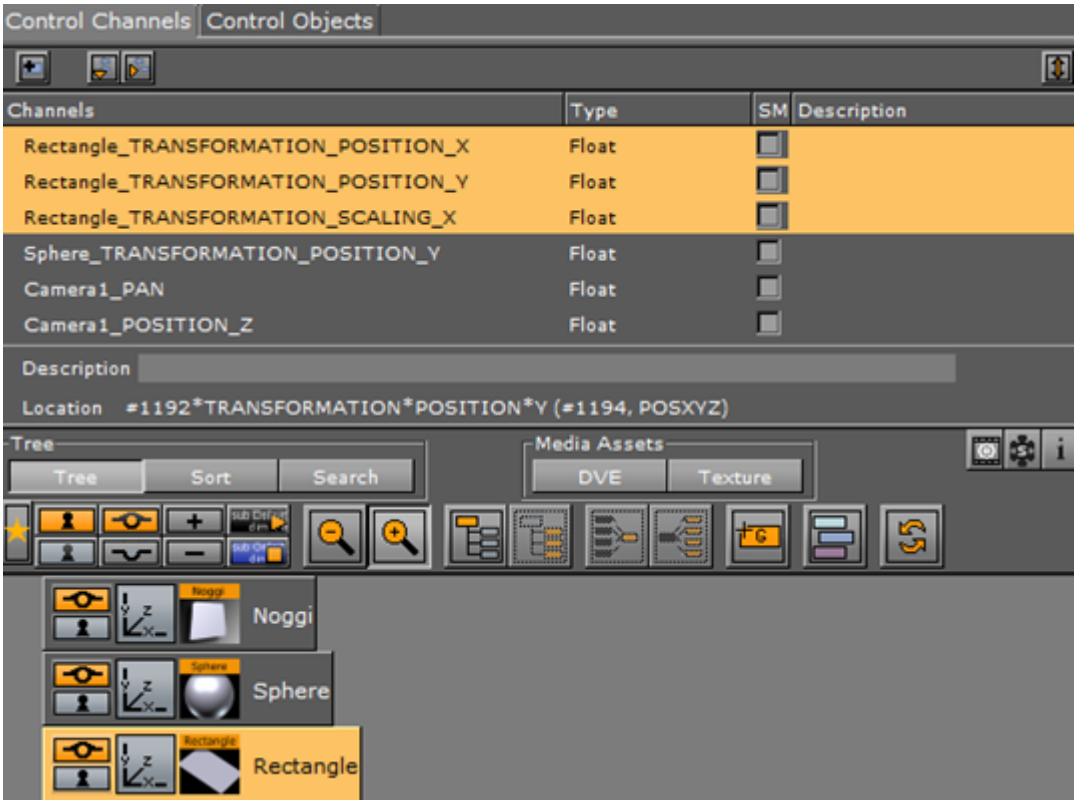
Parameters in a Scene can be exposed for use with external controls, meaning these exposed parameters can be modified by an external control application, such as Viz Libero, Viz Trio or Viz Pilot. To expose parameters, the parameters can be dragged to the Control Channel panel and configured.

Although parameters can be set through a Command Interface, the Control Channels offers three advantages:

- Only parameters, which should or can be modified, are exposed in certain external control applications (which use control channels)
- Instead of a long command string (as used in a Command Interface), a control channel name points to the parameter location, where "@" in the command location is the indicator for a control channel name
- Multiple Control Channels can have the same name, allowing them all to be updated in a single operation
  Control Channels can also be added from the **Stage**. To make this possible, the Control Panel can be floated. This enables the Control Panel to be viewed in all views of Viz Artist.

Most parameters can be added to the Control Channels. Properties that can be added to the Controls Channels are:

- **Properties Panel**: Container, Camera, Light and Scene Settings tabs
- **Server Tree**: Containers, Geometries, Materials
- **Server Panel**: Geometries, Materials
- **Stage**: Editors and parts of Editors, i.e., Director parameters, Channel Editor, Key Frame Editor, Stops and Tags, Action Key Frames.
  Navigating scenes with extensive use of Control Channels is made easy, as Viz Artist automatically selects all Control Channels belonging to a container when the container is selected. Correspondingly, selecting a single Control Channel will also select the container it belongs to.



Control Channels can also be grouped. This provides a convenient way to group a series of parameters which can be controlled from an external source.

⚠ **Example:** Click on the *Container tab -> Transformation editor.* Drag the parameter Position X of an Object to the Control Channels panel. This will expose Position X of that object.

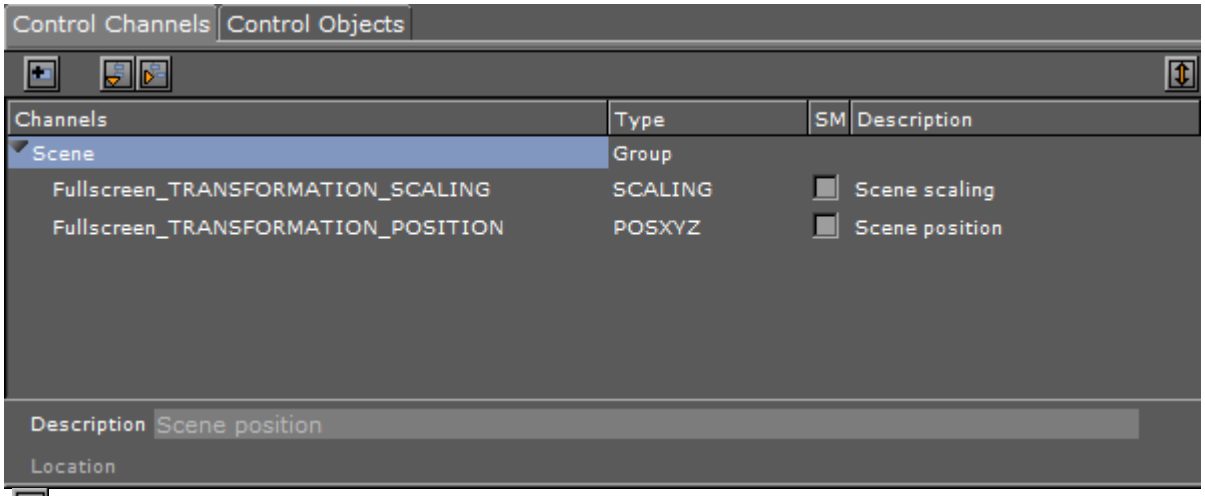
⚠ **Example:** In the **Properties** panel, click on the **Camera** tab and drag a camera's **Position Y** parameter to the Control Channels panel. This will expose Position Y of that camera.

⚠ **Example:** In the **Stage**, click on the header of a Key Frame editor (in-line with the ) and drag it to the Control Channels panel. This gives access to all parameters contained in the Key Frame editor. Note, to add control channels from the Stage, Control Channels need to be floated.

This section contains the following procedures:

- Control Channels Properties
- Control Channels Procedures

Control Channels Properties



- : Click to add a new group
- : Opens or collapses the **Control Channels tree**
- : Floats the **Control Channels panel**
- : Docks the **Control Channels panel**
- **Channels, Type, SM, Description**: Exposed parameter details.
  If required, click in the **SM** box to synchronize the Channel values with the corresponding key-value pair in the Scene Shared Memory Map.

  > ⚠ **Note:** The Shared Memory key syntax for grouped control channels is <groupname>.<channelname>, for example, MyGroup.Rectangle_TRANSFORMATION_POSITION_X

- **Description**: Adds a description for the exposed parameter
- **Location**: Shows command location of the command

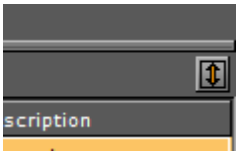
Control Channels Procedures

This section contains the following procedures:

- To Dock and Float the Control Panel
- To Create a Control Channel
- To Remove Exposed Parameters
- To Create a Group

To Dock and Float the Control Panel

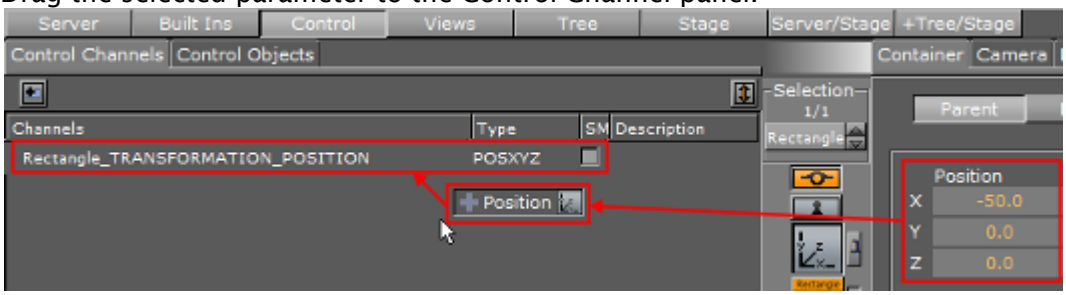
To access parameters in the Stage, the Control Panel can be floated and used as a floating panel, available in all Viz Artist panels:



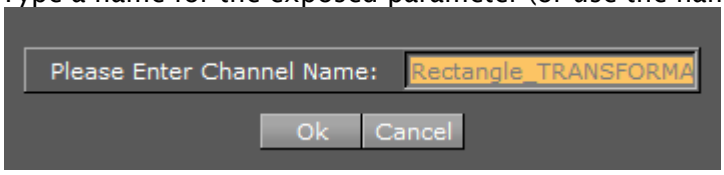
1. Click  to float the Control Panel.
2. Click the  to dock the Control Panel.

To Create a Control Channel

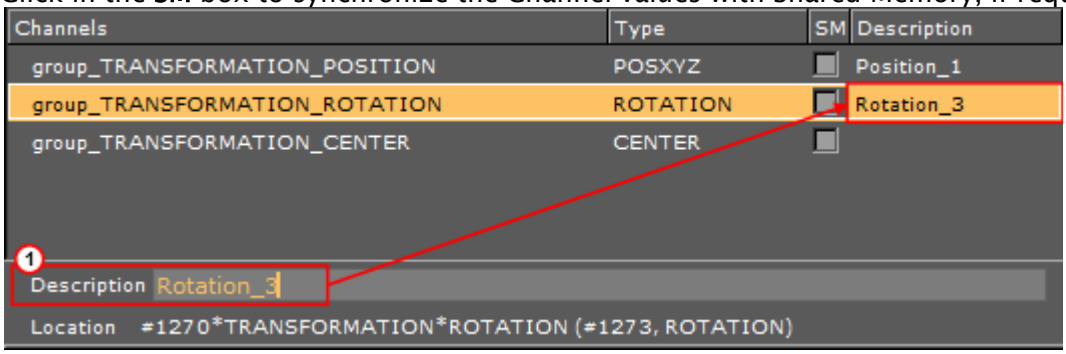
1. Click **Control**.
2. Select a parameter to expose, from:
   - Properties Panel: Container, Camera, Light and Scene Settings tabs
   - **Server Tree**: Containers, Geometries, Materials
   - **Server Panel**: Geometries, Materials
   - **Stage**: Editors and parts of Editors, i.e., Director parameters, Channel Editor, Key Frame Editor, Stops and Tags, Action Key Frames.
3. Drag the selected parameter to the Control Channel panel.



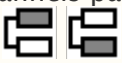
4. Type a name for the exposed parameter (or use the name provided).



5. Click on the exposed parameter.
6. Type a **Description** (1) for the exposed parameter, if required.
7. Click in the **SM** box to synchronize the Channel values with Shared Memory, if required.



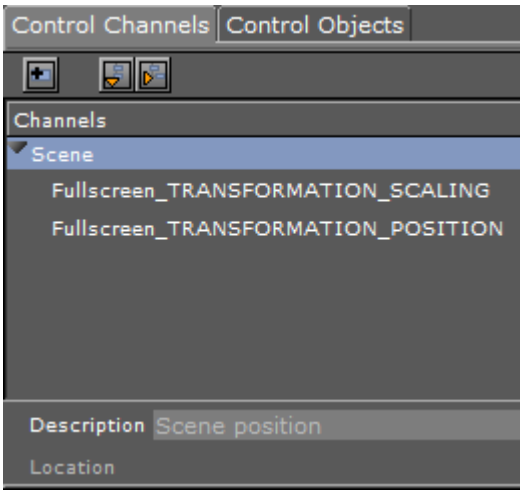
8. To add a second, third, fourth, or more exposed parameters, repeat from step 1.

> ⚠ **Note:** When a new parameter is dragged to the Control Channels panel an icon will show where the new parameter will be dropped in the list  .

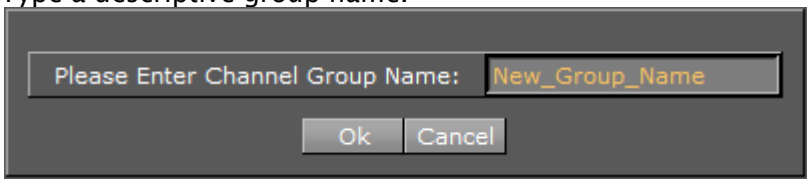
To Remove Exposed Parameters

- Drag an exposed parameter, or multi selected exposed parameters, to the Trash Can.

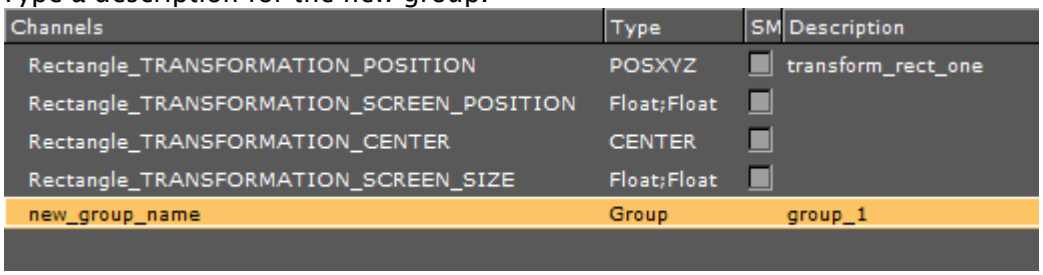
To Create a Group



1. Create a new group:
   - Click on  , or
   - Drag the  icon to the main Control Channels panel
2. Type a descriptive group name.

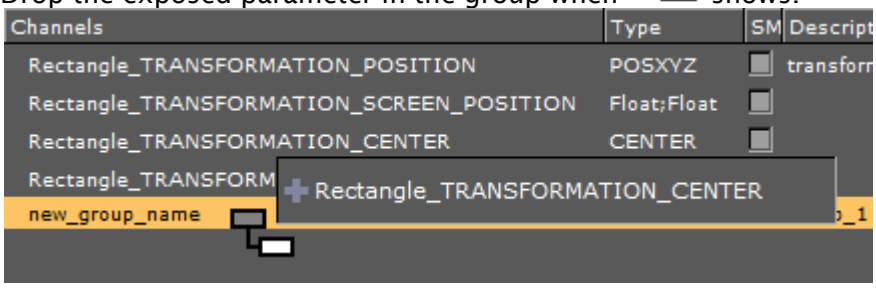


3. Type a description for the new group.

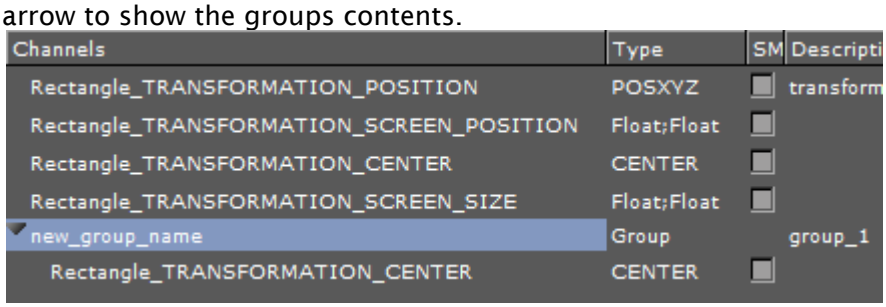


4. Drag an exposed parameter, or multi selected exposed parameters, to the group:
   - Drag from the Controls Channels panel, or
   - Drag directly from the Properties panel or Stage

5. Drop the exposed parameter in the group when  shows.



6. When a group contains an exposed parameter an arrow will show to its left. Click on the arrow to show the groups contents.



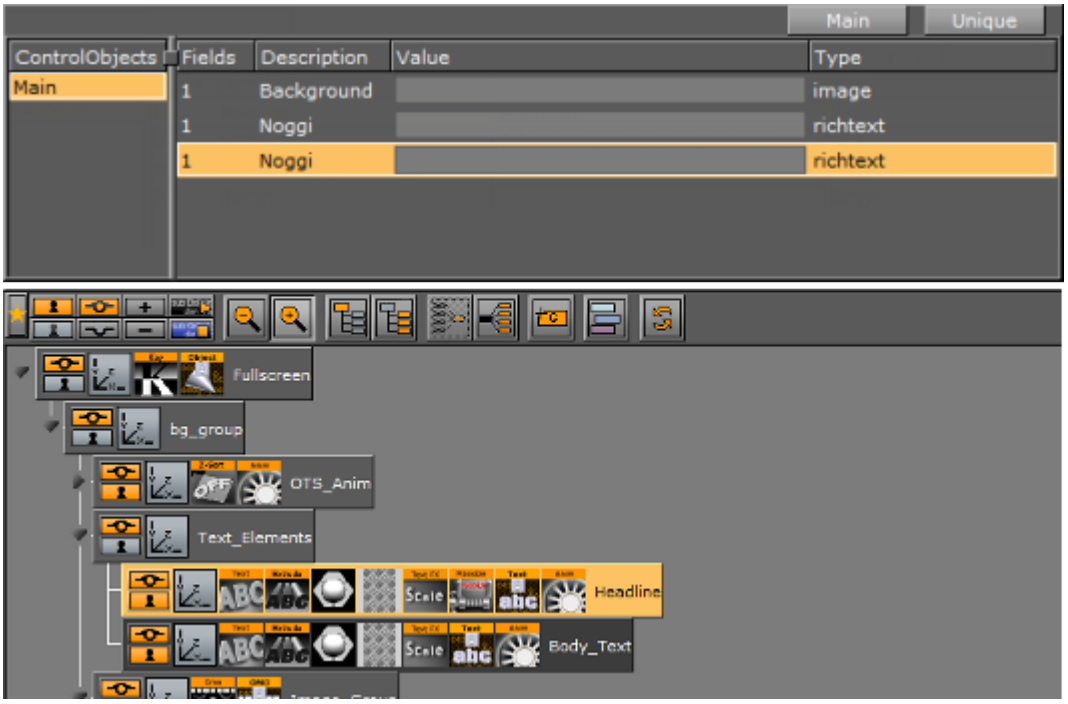
- To Dock and Float the Control Panel
- To Create a Control Channel
- To Remove Exposed Parameters
- To Create a Group

See Also

- Control Objects
- External Control

5.8.2 Control Objects

The Control Objects panel is used to show the control parameters for Viz Trio Scenes only.



Control Object Parameters

- **Main:** Set to **ON** to view only the main control fields of the Control Object. Set **OFF** to view all fields.
- **Unique:** Click to enable view of only the original field if there is more than one field with the same identifier. Click again to disable view all fields, including field duplicates with the same name. This allows the designer to easily select the respective containers in the scene. By default, Viz Artist displays all fields in the scene.**Control Objects:** The Container which contains the control Object.
- **Description:** Enter a description in the plug-in Properties panel.
- **Value:** The editable plug-in value.
- **Type:** The type of Control Object.

See Also

- Control Channels
- External Control

6 Scene Management

This section details the management of Scenes, which includes the parameters of Scene Settings and the Scene Editor.

The Scene Editor has a set of useful functions that let the designer see how graphics will look when keyed, seen from a different angle in Viz Virtual Studio (virtual sets), positioned on-screen, performance of the scene and more.

This section contains information on the following topics:

- Open a Scene
- Scene Settings
- Scene Editor
- Working with the Scene Editor
- Scene Editor Views

6.1 Open A Scene

6.1.1 To Open a Scene in the Database

1. Select **Server** from the Main Menu Left.
2. In the Server Tree, select the folder or project where the Scene is.
3. Open the Scene:
   - Double-click the Scene to open it, or
   - Drag the Scene to the Scene Tree area, or
   - Drag the Scene to the renderer window.

   The content of the Scene will be shown as Containers in the Scene Tree, and visualized in the renderer window.

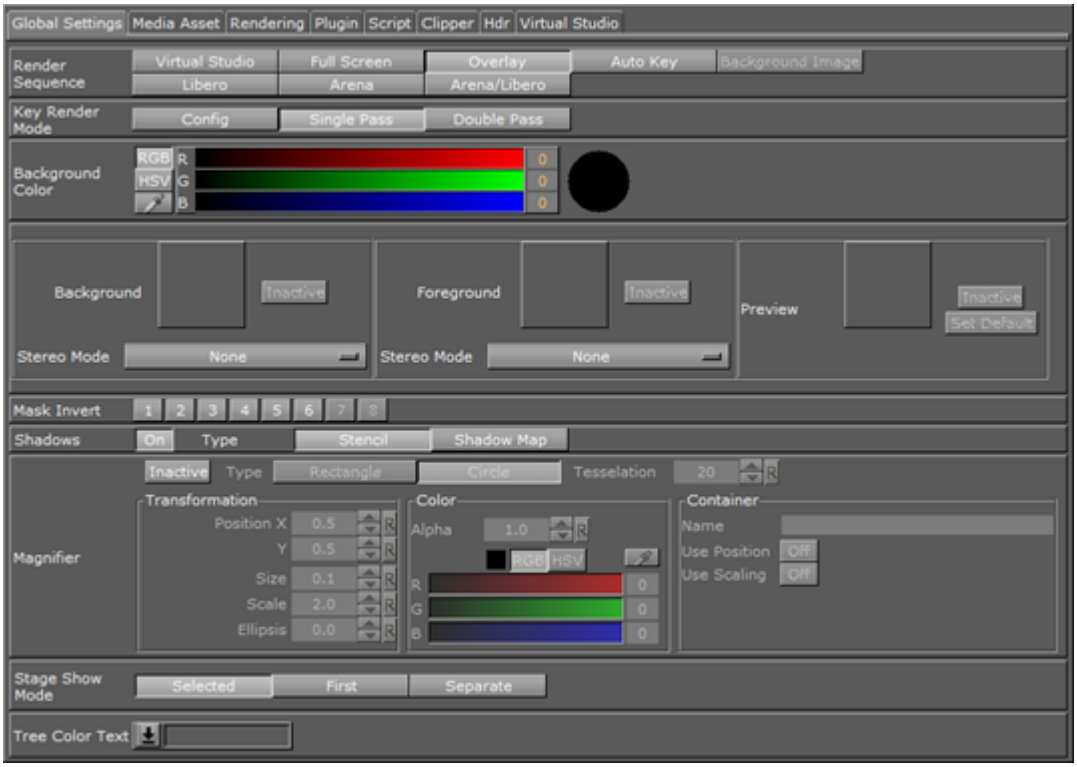
⚠ **Note:** Image Editor or Scenes are marked with a Keyhole icon  .

⚠ **Tip:** If the folder or project's path is unknown, search for the Scene in the Free Text Search function under the **<Username>s View** tab in the Server Panel.

⚠ **Tip:** The Status bar in the lower left part of the main window (between the Close and Undo buttons) shows the name and path of the current Scene. Click on the path to select the Scene in the Server Panel.

6.2 Scene Settings

To open the Scene Settings editor, select the Scene Settings tab in the Property panel. The Scene Settings editor makes it possible to modify the global parameters for the scene.

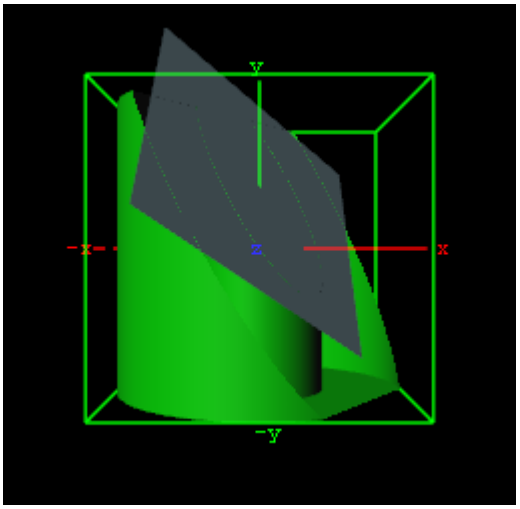


There are various types of scene settings. Click the tabs at the top of the Scene Settings editor to open the respective panels:

- Global Settings Panel
- Media Asset Panel
- Rendering Panel
- Plug-in Panel
- Script Panel
- Clipper Panel
- HDR (High Dynamic Range) Panel
- Virtual Studio Panel
- Viz Libero and Viz Arena Render Sequences

6.2.1 Clipper Panel

In the Clipper panel, settings for scenes that have the Clipper plug-in attached can be modified.

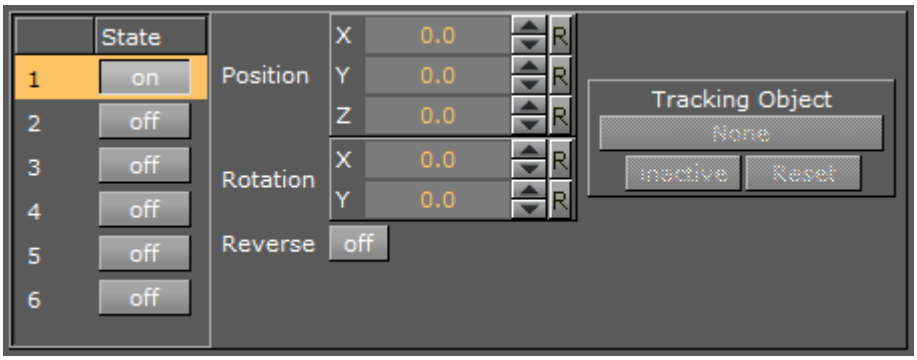


The Clipper plug-in is an alternative to the mask function; the clipper works in 3D and the mask in 2D. Objects using the clipper plug-in, when placed behind the clipper plane, will be masked/ clipped. If an object is only partially behind a clipper plane, only the part of the object that is behind the plane will be clipped out since the clipping is done in true 3D space. The plug-in uses the OpenGL clipping planes that supports up to six planes.

This section contains the following topics:

- Clipper Properties

Clipper Properties



- **Number**: Shows the clipper plane number, from 1 to 6
- **State:** Sets the selected clipper plane **On** or **Off**
- **Position**: Modifies the position of the clipper plane along the X, Y, and Z axis
- **Rotation**: Modifies the rotation along the X and Y axis
- **Reverse**: Shows everything in front of the clipper plane when set to **Off**. Shows everything behind the clipper plane when set to **On**. Tracking Objects with Clipper Plane
- **Tracking Object:** A clipper plane can be set up to track containers regarding position and rotation. Tracking position means that the position of the clipper plane will be in the center of the container, and tracking rotation means that the clipper plane will rotate if the object rotates. This is especially useful when animating a scene.

See Also

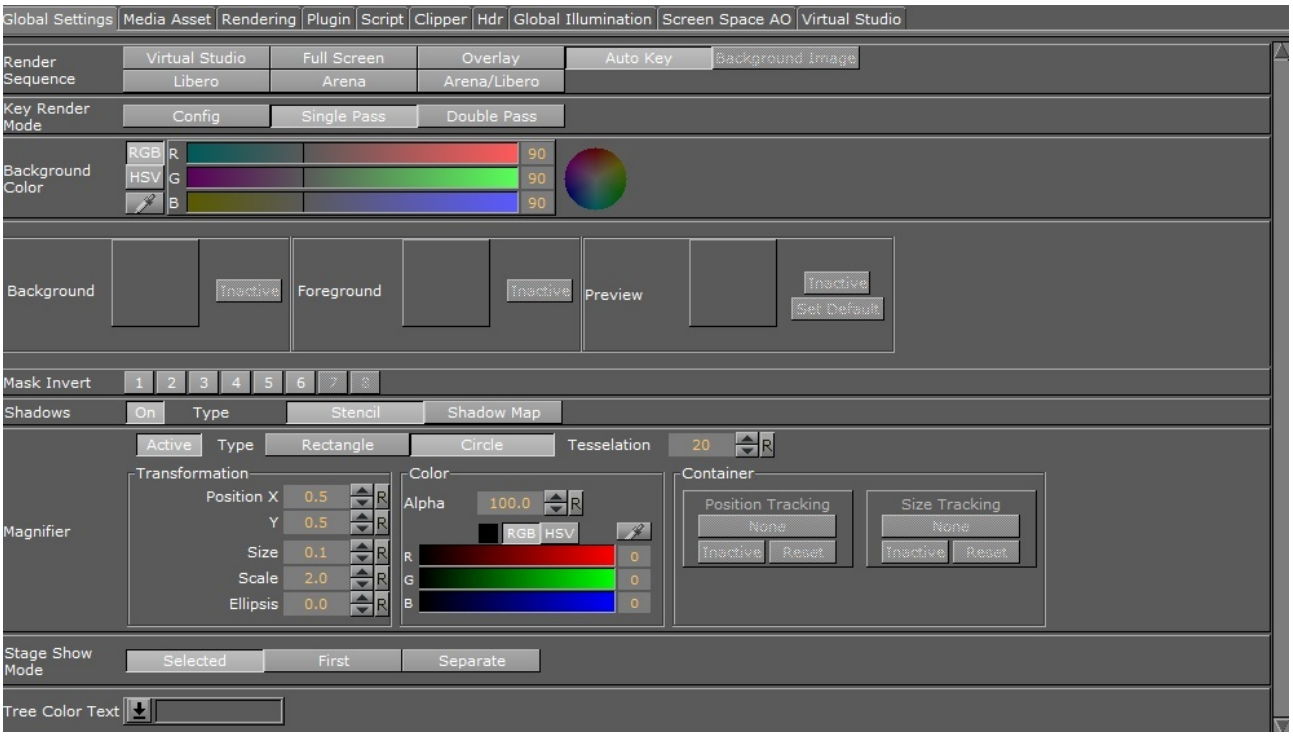
- Create Animations
- Clipper

6.2.2 Global Settings Panel

This section contains information on the following topics and procedures:

- Global Settings Properties
- Images as Background or Foreground

Global Settings Properties



In the **Global Settings** panel, the basic parameters for a Scene can be modified:

- **Render Sequence:** Defines how the alpha key signal should be generated (default is **Overlay**):

  - **Virtual Studio**: Switches from **Overlay** to **Full Screen** when the control mode of the current camera switches from **Remote** to **Editor** (see Camera Editor). In a Virtual Studio set, to do a camera flight with the virtual camera, when the real camera is stationary, the graphics must be set to **Full Screen** to avoid a situation where the virtual and the real camera have different position values. When the camera is switched from **Remote** to **Editor** (to do the flight) the keyed graphics automatically switch from **Overlay** to **Full Screen**.

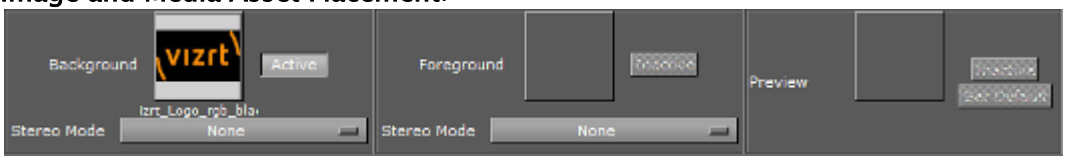
    ⚠ This option is relevant only in combination with Viz Virtual Studio.

- **Full Screen**: Enables downstream compositing and generates a full screen key that outputs a non-transparent key signal when Render Sequence is set to Full Screen. This means everything without a Key plug-in will be treated as background graphics and appear behind DVEs on the output. Everything with a Key plug-in will be rendered in front of DVEs. For setups using a Matrox X.mio video board, the key preview is all black, instead of all white, while show key is enabled in the Scene Editor. This allows the designer to work with containers with Key plug-ins during scene design.
- **Overlay**: Sets as key signal only those Containers that have a Key function applied. Only Containers in the Scene Tree with a Key function applied are rendered into the key signal. Use this mode to correctly blend graphics and DVEs.
- **Auto Key**: Generates the key signal automatically from the Containers in the Scene Tree, it does not require the application of Key functions to the Container. This function provides a worry-free overlay. This is the preferred mode for overlay graphics.
- **Background Image**: Uses the alpha values of the background image as a key signal and Containers with a Key function applied and rendered on top. This option is only applicable if a background image with an alpha layer is added.
- **Libero**: The render sequence for a Viz Engine used in with Viz Libero (see Viz Libero and Viz Arena Render Sequences).
- **Arena**: The render sequence for a Viz Arena/multi-zone keying setup (see Viz Libero and Viz Arena Render Sequences).
- **Arena/Libero**: The render sequence if the tracked camera is a Viz Arena image tracker (see Viz Libero and Viz Arena Render Sequences).

- **Key Render Mode:**

  - **Config:** Renders the key according to the settings in the Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide).
  - **Single Pass:** Uses one rendering step. Use this for all new scenes and for scenes imported from version 2.8, as it is much faster.
  - **Double Pass:** Uses two rendering steps. Use for backward compatibility in scenes created in Viz Artist versions 3.0 to 3.3.
- **Background Color**: Sets the color of the background.

- **Image and Media Asset Placement**:



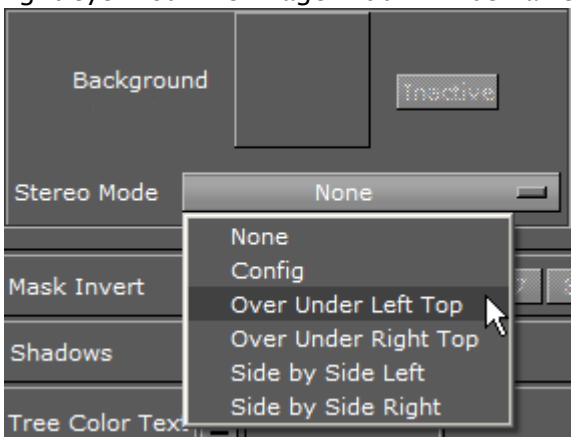
  - **Background**: Adds an image or Media Asset as the Scene background.
  - **Foreground**: Adds an image or Media Asset as the Scene foreground.

> ⚠ **Note:** If the foreground image or Media Asset has alpha, the Scene can be shown through the image or Media Asset.

- **Preview:** Adds a fixed layout to the Scene. This visualizes how the total preview (camera picture and/or graphics/Media Asset) will look like. To set the image as global default, click the **Set Default** button.

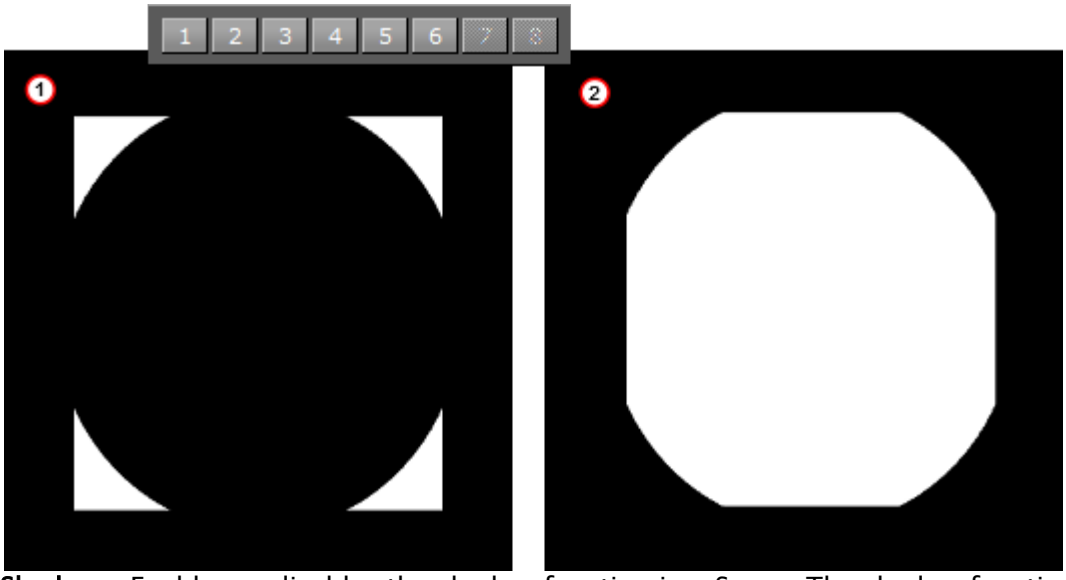
> ⚠ **Tip:** It is possible to drag results from a database search onto the Background, Foreground, and Preview drop zones.

- **Stereo Mode:** (license required) Selects the correct stereo mode for the inserted image (not available if **Stereo Mode** is set to **Off** in the Viz Configuration (see the **Camera** page in the **Configuring Viz** section of the Viz Engine Administrator Guide):
  - **None:** No Stereo mode.
  - **Config:** Images are drawn as set in the Viz Configuration (see the **Camera** page in the **Configuring Viz** section of the Viz Engine Administrator Guide).
  - **Over Under Left Top** or **Over Under Right Top**: Two images are stored, one over (top) and the other under (bottom), **left** or **right** eye (top) first. The image height will be halved, providing half resolution.
  - **Side by Side Left** or **Side by Side Right**: Two images are stored, side by side, left or right eye first. The image width will be halved, providing half resolution.



- **Mask Invert**: Makes it possible to set one or more of the mask layers in a Scene to be inverted. With no mask inverted (1), the mask cuts out a hole where it covers the mask target object. With mask invert enabled (2), the mask target is only visible where the mask

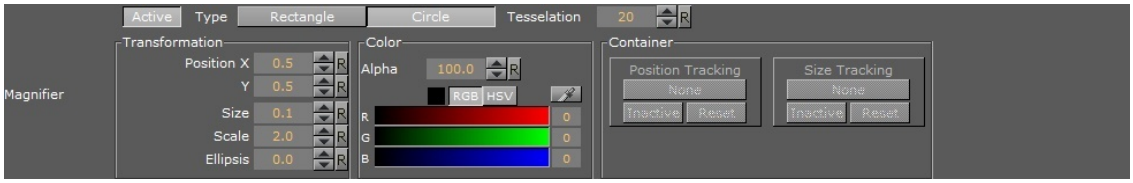
covers it:

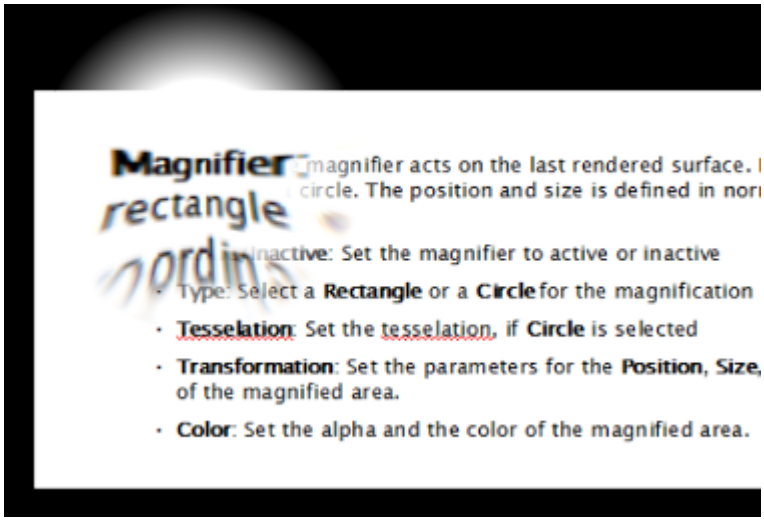


- **Shadows**: Enables or disables the shadow function in a Scene. The shadow function uses two layers from the mask function. With shadows enabled, there are only six layers available for the mask function (see Mask Source and Mask Target). In a scene where more than six mask layers are needed, shadows can be disabled (only very complex mask configurations need more than six layers).
  The **Stencil** shadow type is the least resource-demanding way to render shadows in scenes. For more advanced and better-looking shadows, cascaded shadow maps can be enabled. For more in-depth information on shadows in Viz Artist, please refer to Shadow Maps and Shadow Caster and Shadow Receiver.

> ⚠ **Note:** The Shadows On/Off button is only available when shadow type is set to Stencil, as Shadow Mapping is not affected by this setting.

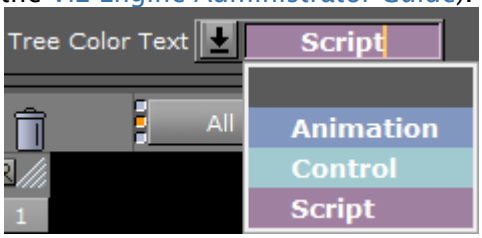
- **Magnifier**: Acts on the last rendered surface. It can be either a rectangle or a circle. The position and size is defined in normalized view-port coordinates.
  - **Active/Inactive**: Sets the magnifier to active or inactive.
  - **Type**: Selects a **Rectangle** or a **Circle** for the magnification area.
  - **Tesselation**: Sets the tessellation, if **Circle** is selected.
  - **Transformation**: Sets the parameters for the **Position**, **Size**, **Scale** and **Ellipsis** of the magnified area.
  - **Color**: Sets the alpha and the color of the magnified area.
  - **Container**: Selects a Container to control the position and, or, the size of the magnifier. Drag the Container to be tracked in the desired box.





- **Stage Show Mode:** Defines how the `STAGE*SHOWMODE` Viz Engine command should be handled for the scene. The command takes a named stop point or tag as a parameter, and the show-mode defines how the scene will be executed.

  - **Selected:** Sets the time-line of all selected directors to the first found tag location. This is the default setting.
  - **First:** Sets the time-line of the first found director, containing the searched tag, to the tag's position.
  - **Separate:** Positions the time-line for each director, containing the searched tag, to the tag's location.
- **Tree Color Text**: Defines custom text labels for the scene tree colors in the current scene. To define global labels or custom colors, go to **Viz Configuration > User Interface > Colors** (see the Viz Engine Administrator Guide).

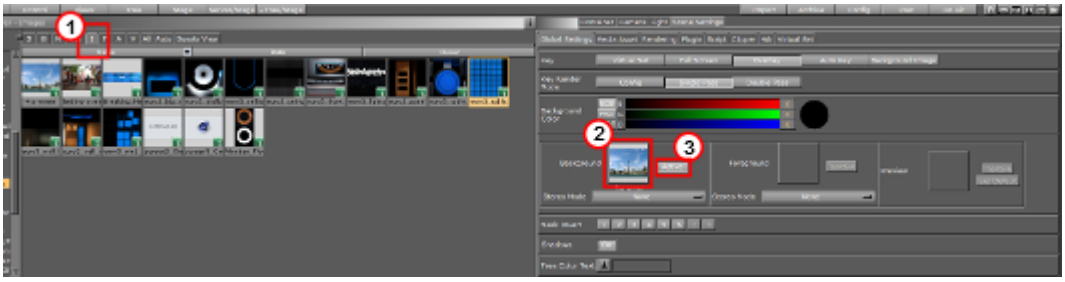


Images as Background or Foreground

This section details procedures with images use as background, foreground and preview:

-
  - To Add an Image
  - To Remove an Image

- To View an Image Source Project or Folder
- To Show/Hide an Image in the Scene Editor



To Add an Image

1. Click **Server**.
2. Click the **I** tab for Images (1).
3. Drag the required image onto the drop zone (2).
   A thumbnail of the image will then be shown in the drop zone button. Dragging a new image onto the drop zone will automatically replace the old image.

To Remove an Image

- Drag the image from the drop zone to the trash can.

To View an Image Source Project or Folder

- Click the image on the drop zone.
  The project or folder it is placed in is visible. If the image is linked, the first project or folder will be selected.

To Show/Hide an Image in the Scene Editor

- Click the Active/Inactive button next to the drop zone (3).
  If the image has a different aspect ratio than the screen resolution, the image will be stretched to fit.

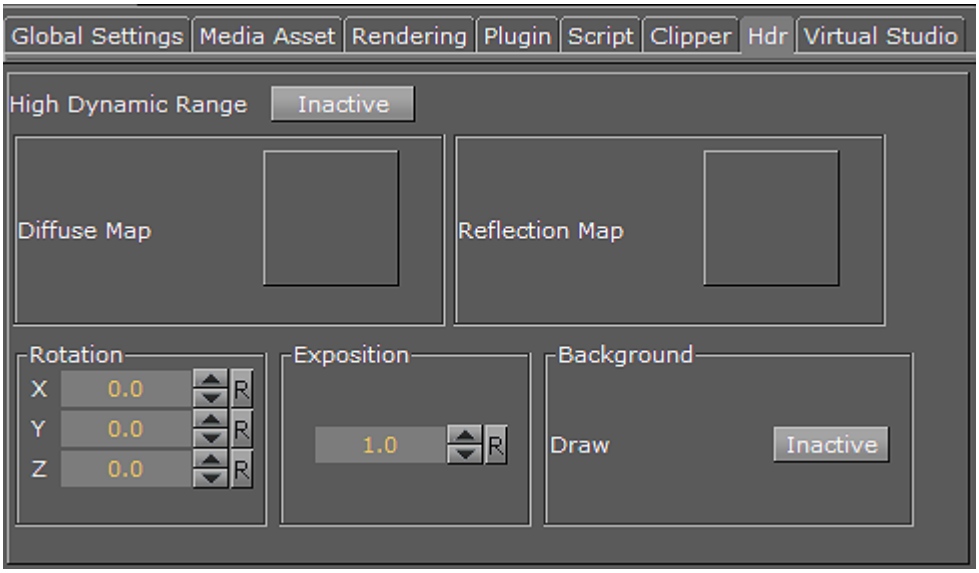
6.2.3 HDR (High Dynamic Range) Panel

Enables High Dynamic Range Rendering/Imaging for the selected container. This improves the contrast of the generated scene in a much more realistic way.

This section contains the following topics:

- HDR Properties

HDR Properties



- **High Dynamic Range (Active/Inactive)**: Enables HDR lighting instead of the default OpenGL lighting on Containers tagged for HDR lighting. To tag Containers for HDR lighting, the HDR plug-in must be placed in a Container to enable HDR lighting on the respective container and its sub-tree.
- **Diffuse Map**: Defines the diffuse light map. This is a HDR Cube map (must be a vertical cross and should have been imported from an EXR file).
- **Reflection Map**: Defines the diffuse light map for the reflection part of the lighting.
- **Rotation**: Rotates the environment map with the set parameters.

> ⚠ **Note:** Diffuse map and reflection map together form the 'environment map'.

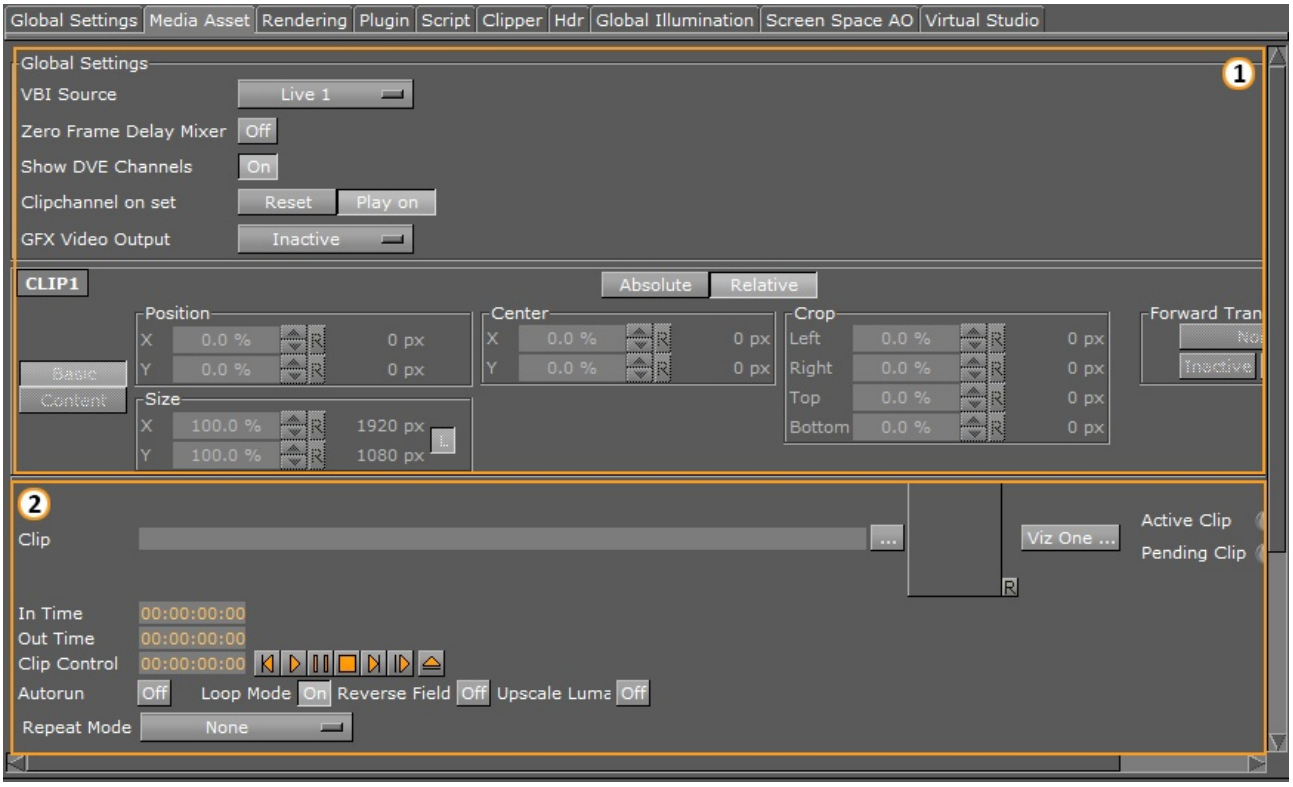
- **Exposition**: Controls the brightness of the maps (similar to the exposure time of an ordinary camera).
- **Background (Active/Inactive)**: Draws the Reflection map as background.

See Also

- HDR

6.2.4 Media Asset Panel

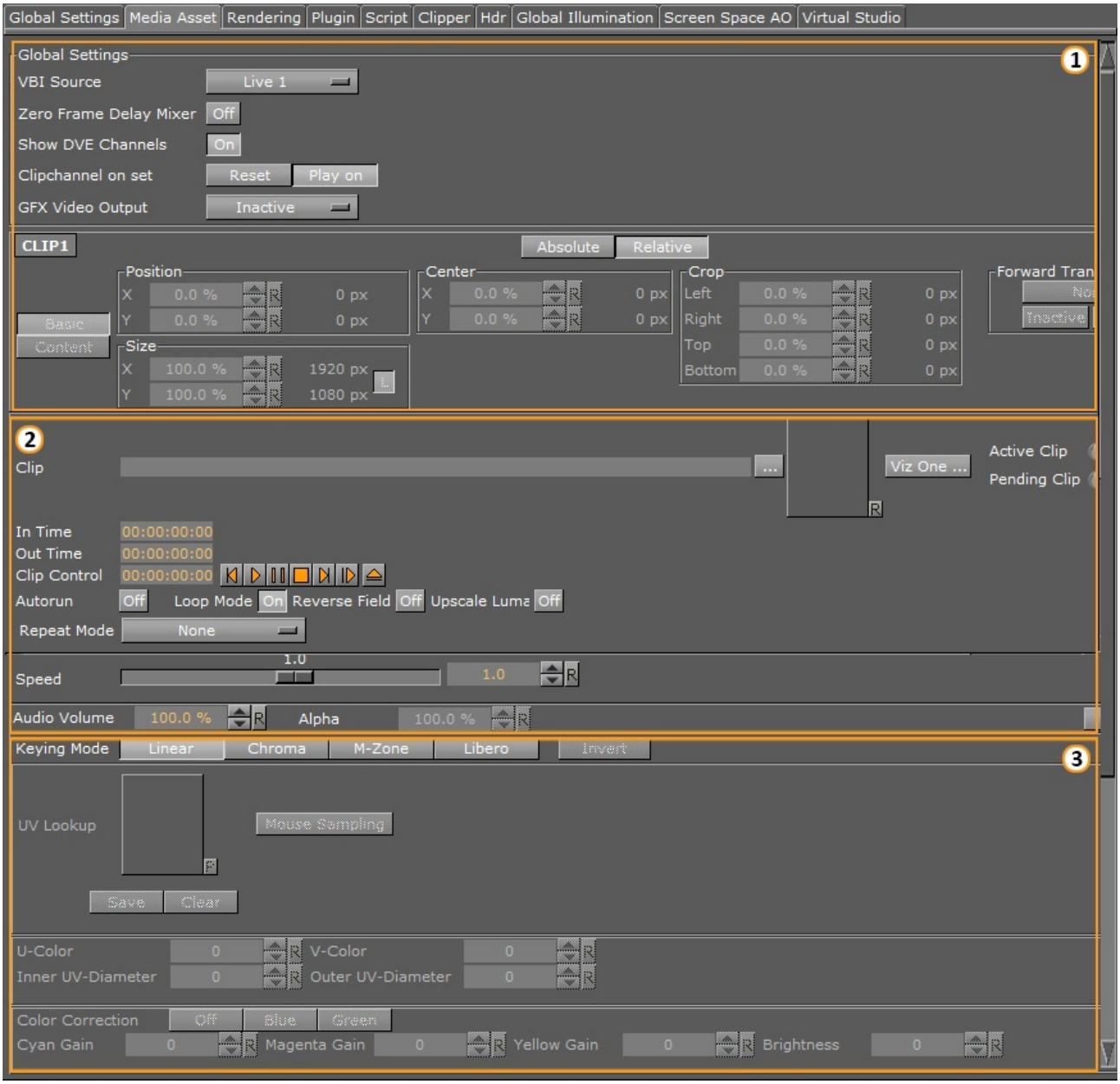
This panel contains parameters and settings for Media Assets. There are different types of Media assets available like Clip Channels, GFX Channels, Image Channels, Live Input Channels, IP Stream Channels and Super Channels. In the image below, section **1** contains details shared across *all* Media Assets. Section **2** contains details specific to the selected media asset (See section Media Asset Clip Settings).



This section contains information on the following topics:

- Common Media Asset Properties
- Global Settings (section 1)
- H-/V- Phase Values for Zero Frame Delay Mixer
- Media Asset Clip Settings
    - Clip Timecode Types
- Media Asset Keying Settings
- Switching between different Media Assets
- Live Properties
- Stream Properties
- GFX Properties
- To View Media Asset (DVE or Texture) Properties
- To View Media Asset (Texture) Properties

Common Media Asset Properties

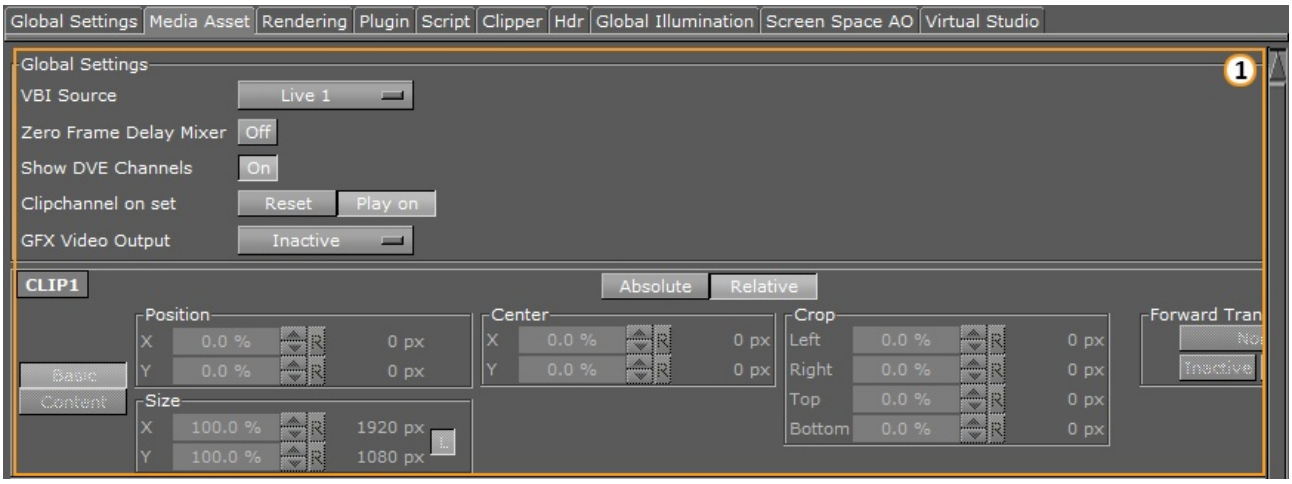


Item	Description
1	Global Settings
2	Clip Channel Properties / Live Channel Properties / IP Stream Channel Properties / GFX Channels Properties, Image Channels Properties

Item	Description
3	Keying Mode

Global Settings (section 1)

These are the common Media Asset properties.



- **VBI Source:** Selects which channel to use for output. Types of data sent include closed captions, teletext, time codes and other digital data. The setting **Clip Timecode Type** can enable Viz Engine to provide additional time code data in the VBI-signal, if required. The setting can be changed by editing the Viz Configuration file and setting the variable `clip_timecode_type = <VALUE>`. Please refer to the table Clip Time Code Type values below for valid values. It can also be set on the fly using the `Send` application to issue the following command: `CONFIGURATION*CLIP_TIMECODE_TYPE SET <VALUE>`

  > ⚠ **Note:** This setting only applies for installations with Matrox cards.

  > ⚠ **Note:** If you are using more than one specific Media channel in your scene, just click on the text of "CLIP1" (in section 1) and you can switch to the other clip channels used in the same scene.

- **Zero Frame Delay Mixer**: Set to **On** to give the shortest input to output delay, with graphics on top. For this to work correctly, specific values for H-/V- phase must be set. For the input/output configuration, input A is bound to output A, and Input B is bound to Output B. Please observe that the resolution of the input is required to be the same as for the output. If running a Dual Channel setup, make sure that input A and input B are 100 % in sync, and using the same resolution.
  Refer to H-/V- Phase Values for Zero Frame Delay Mixer for further information
- **Show DVE Channels:** Enables or disables DVE placeholder box drawing
- **Clip Channel on set:**

- **Play on**: Continues to play the clip even if the Scene does not contain that specific clip channel. This is the default behavior. When setting a Scene with this channel active, the clip will be shown from its current position onwards, when no clip or the same clip is set in the Scene.
- **Reset**: Starts playout from the beginning of the clip when it has the same clip name, or show black when no clip is set in the channel at all.
  Please observe that the commands and values are case sensitive. Any on-the-fly changes will be saved to the Viz Configuration file upon proper shutdown of Viz Engine.

⚠ **Caution:** Manual changes to the Viz Configuration file should never be made while Viz Engine is running. Close the engine, commit any changes, and start the engine. Make sure to always create a backup copy of the Viz Configuration File before making any manual changes.

- **Basic:** Transforms the whole used Media Channel.
- **Content:** Transforms the content inside the used Media Channel.
- **Position**: The position controls are used to set the X and Y coordinates for the Media Asset. The coordinates are relative to the configured output format (see the **Configuring Viz** section of the Viz Engine Administrator Guide).
- **Size**: Sets the X and Y values to define the size of the Media Asset. The size is based on the channel resolution as defined in the **Video Input** section of **Configuring Viz** (see the Viz Engine Administrator Guide), which may be different than the configured output resolution.
- **Center**: Uses the Center X and Y values to define the center position of the Media Asset. As for the **Size** control, the **Center** position is relative to the configured channel resolution.
- **Absolute / Relative Position Controls:** Controls to adjust the **Position**, **Size**, **Center** position, and **Crop**. The values can be provided either as **Absolute** pixel values, or as **Relative** percentage values. The corresponding value will be displayed next to the input value field. If provided with values that exceed the logical limit, the last value will be adjusted automatically to the maximum value. For example, if setting the **Left** crop value to 51 %, and then the **Right** crop value to 75 %, the **Right** crop value will be set to 49 %. Applies to DVE only.

  ⚠ **Note:** By default, the center position for DVE transformations is defined as the top left corner of the DVE placeholder. Setting the DVE Center `X` and `Y` positions to `50 %` will define the center for the transformations as in the middle of the DVE placeholder. This behavior deviates from that of transforming Containers, as they have their transformation center in the middle.

- **Crop**: Allows the designer to crop the Media Asset as required. The crop values are relative to the size as defined using this editor.
- **Forward Transformation**: Forwards transformations to the container which was dragged and dropped to this parameter. The transformation is mapped to the XY plane of the container. This allows a video wall control application like Viz MultiPlay to control virtual (3D) Video Walls as well. This applies only to Media Assets added as Texture, and the Forward

Transformation controls will only be visible for selected Texture Media Assets.

- **Identifier**: The name of the target container.
- **Active**: Forwarding is active.
- **Reset**: Resets forwarding to the specified container.

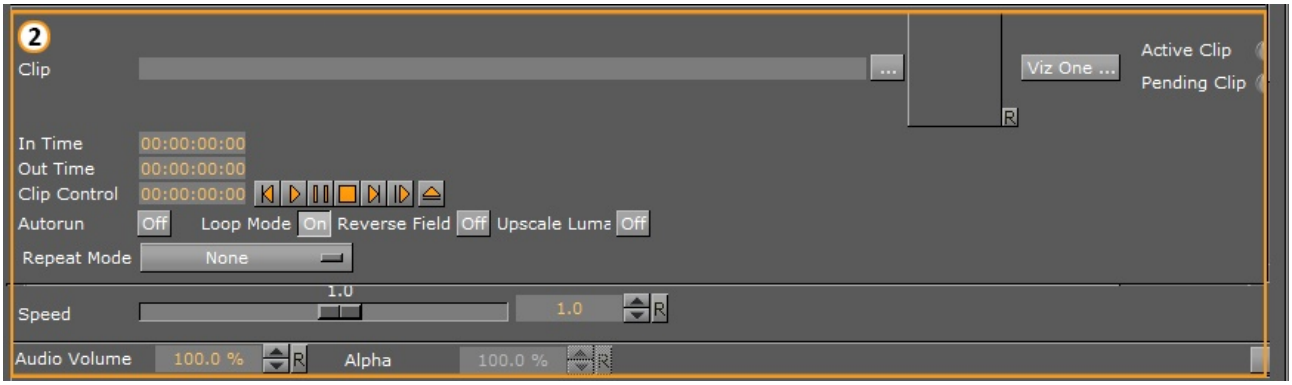
H-/V- Phase Values for Zero Frame Delay Mixer

For Matrox X.mio2, DSX.LE2 and DSX.LE3 video boards:

- **PAL**: Leaves the values at the default
- **NTSC**: Leaves the values at the default
- **720p50**: Valid H-Phase range from -900 to -1300
- **720p59.94**: Valid H-Phase range from -600 to -1000
- **1080i25**: Valid H-Phase range from -1600 to -2000
- **1080i29.97**: Valid H-Phase range from -1100 to -1600

Media Asset Clip Settings

In addition to the common Media Asset properties, Video Clip also has these properties:



- **Clip:** Selects the video clip to be shown:
  - In the **Clip** line, click ... , or
  - Drag a Video Clip item from the Server Panel, or
  - Click Viz One ... , and select a video clip to import.

> ⚠ **Note:** See Transfer Clips From Viz One. To transfer and download videos from Viz One, Viz One must first be integrated and configured with Viz Artist/ Engine.

> ⚠ **Note:** With the Clip Channel Media Asset it is possible to play images as DVE or Texture with the Matrox clip player. Supported image containers are jpg and png. The images have to be available on a physical disk.

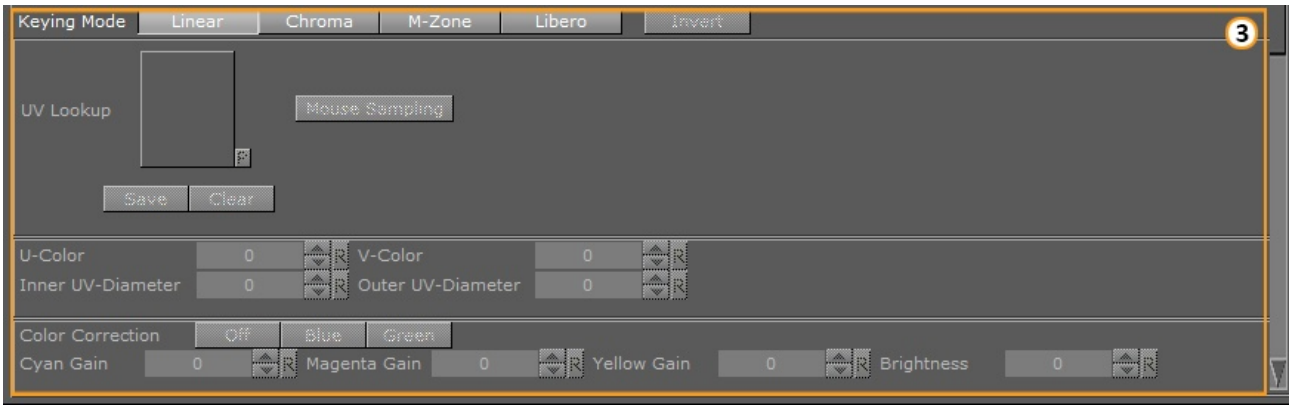
- **Active Clip/Pending Clip:** Displays the status of active and pending clips:
  - **Green** light shows that the clip has loaded successfully.
  - **Grey** light shows that no clip has been loaded.
  - **Red** light shows that there was an error during the clip loading process.
- **In Time:** Sets the video clip 'in time'.
- **Out Time:** Sets the video 'out time'.
- **Clip control:** Shows player controls for the selected clip.
- **Autorun:**
  - **Off:** As soon as the video clip is loaded, the first frame is presented.
  - **On**: As soon as the video clip is loaded, it begins to play.
- **Loop Mode:**
  - **Off** The video clip will play only once.
  - **On** The video clip will play in a loop.
- **Reverse Field:** As interlaced video contains two fields of a video frame, reverse playback of interlaced video can result in flickering unless the fields are also played back in reverse. On the Video version of Viz Artist and Engine with Matrox video hardware, Reverse Field is automatically enabled when playback speed is given a negative value.
  - **Off**: During reverse playback, the fields in interlaced mode are played out 'as is'.
  - **On**: During reverse playback, the fields in interlaced signals are swapped to reduce flicker.
- **Upscale Luma:**
  - **Off:** The key will be used 'as is'.
  - **On**: The key information will be upscaled from 16-235 to 0-255.
- **Repeat Mode**:
  - **None:** The playback of black frames when the clip is in an erroneous state.
  - **Field**: The first field of the last frame will be played out when the clip is paused.
  - **Frame:** The last frame will be played out when the clip is paused.
- **Speed**: Controls playback speed. 1.0 is the normal speed. Values between 0 and 1.0 are forward playback in slow motion. Values above 1.0 are forward playback in fast motion. Likewise, values between 0 and -1.0 are reverse playback in slow motion. Values less than -1.0 are reverse playback in fast motion.
- **Audio:** Sets the audio volume
- **Alpha**: Adjusts the level of opacity for each layer. A value of `100 %` is fully opaque and `0 %` is fully transparent.

Clip Timecode Types

Value	Description
DEFAULT	Mimics the behavior of earlier versions, and will provide a timecode if available. If not, no timecode information is provided in the VBI-signal.
TIMECODE	Will provide a timecode if available. If an explicit time code is not embedded in the clip, no timecode information will be provided in the VBI-signal.
ZEROBASED	The embedded timecode will be '0' at the clip start and count up to the last frame, based on the number of frames in the clip.
COUNTDOWN	The embedded timecode will be the duration of the clip at the beginning and countdown to '0', based on the number of frames in the clip.

Media Asset Keying Settings

In addition to the common Media Asset properties, Keying has these properties.

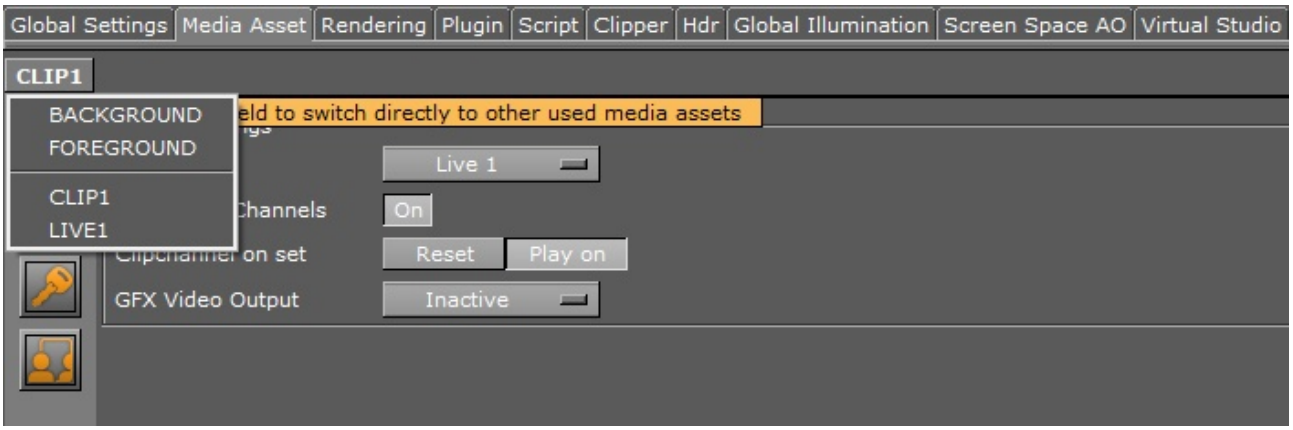


- **Keying Mode** (Texture only: see Keying Mode):
  - **Linear:** No chroma keying
  - **Chroma:** Uses the internal chroma keyer
  - **M-Zone**: Uses the multi-zone keyer
  - **Libero:** Uses the Viz Libero keyer
  - **Invert**: Only works with Chroma and Libero
- **Talent Reflection**: Sets Talent Reflection On or Off (use with Viz Virtual Studio):
  - **Offset**: Sets the offset of the talents lowest body part from where the reflection starts (pixels (Default: 0, Minimum: 0, Maximum: 1000))
  - **Extend**: Sets the extend of the total reflection (pixels). The reflection eases out with distances (Default: 300, Minimum: 0, Maximum: 1000)
  - **Person Scale**: Sets a global scale which applies to the talent (Default: 1, Minimum: 0.2, Maximum: 10)

- **Damping**: Sets how many previous fields are to be used to calculate the current **Y** position. The larger the number the higher the damping, but the **Y** coordinate will be less responsive to quick movements. The setting is a compromise between a stable and a non-responsive **Y** coordinate for the reflection (Default: 2, Minimum: 2 Maximum: 100)
- **Key Scale**: Sets the scale applied to the alpha of the reflection, on top of the parameters set in **Offset** (Default: 0.5, Minimum: 0.1, Maximum: 10)
- **Distortion X / Y**: Parameter **X** defines by how much x axis is affected by the sinus distortion which goes vertically down the screen (Default: 1, Minimum: 1, Maximum: 10). Parameter **Y** defines how much the sinus waveform is stretched vertically down the reflection (Default: 0, Minimum: 0.001, Maximum: 1)
- **Brightness**: Sets the brightness for video going through the chroma keyer (Default: 0, Minimum: 0, Maximum: 10)

Switching between different Media Assets

You can quickly switch between the different media assets used in your scene by clicking the name of the Media Asset selection box instead of individually selecting them in the Scene Tree Media Asset View.



By clicking on the upper-left corner in the Media Asset settings, you can see the list of all used Media Assets in your scene. Here you can easily switch between the different Media Assets.

Live Properties

The Live video properties are listed in the common Media Asset properties.

Stream Properties

In addition to the common Media Asset properties, Video Stream also has these properties:

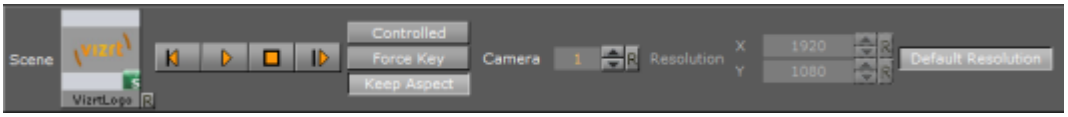
- **URL:** Displays the URL of the Stream to be shown (Video Stream only):

- **Load**: Loads the URL of the video Stream
- **Clear**: Removes the URL of the video Stream.

- **Repeat Mode:**
  - **None:** The playback of black frames when the stream is in an erroneous state
  - **Field**: The first field of the last frame will be played out when the stream is paused
  - **Frame:** The last frame will be played out when the stream is paused

GFX Properties

By default, a GFX Channel automatically inherits the camera settings of the Scene it is applied to. In addition to the common Media Asset properties, GFX Channels also have the following properties:



- **Scene:** Drag a scene, which should be rendered in the GFX channel
- **Play Controls**: Controls the animation of the set scene
- **Controlled**: If activated, the GFX Channel animation can be controlled by the main or parent Scene animation controls, for example, if the animation in the main Scene starts, it triggers the start of the GFX Channel Scene
- **Force Key:** If a GFX channel is rendered as Texture, it will be in RGBA. Therefore, the Scene needs to have keying information for an alpha channel (the key scene setting is set to auto key or a key plug-in is used in the Scene Tree). If the Scene does not have any keying information, enable **Force Key** to render the whole Scene keyed
- **Keep Aspect**: Renders the GFX Scene with its own aspect ratio (undistorted) when set to **On**. Scales to fit the GFX channel size if set to **Off**.
- **Camera**: Selects the camera view of the GFX Channel Scene.
- **Resolution:** The GFX Channels will default to the resolution configured for the current *Output Format* in the Viz Engine configuration. To change the current resolution, click the **Default Resolution** button to make the **Resolution X** and **Y** fields available for input, and enter the desired resolution setting. Clicking the Default resolution again will reset the resolution for the GFX Channel, as well as disable the **Resolution X** and **Y** input fields. If the Default Resolution button is in a pressed state, and the Resolution X and Y fields are dimmed, the default resolution is in use.

  > ⚠ **Note:** The resolution setting only applies to GFX Channels in Texture mode.

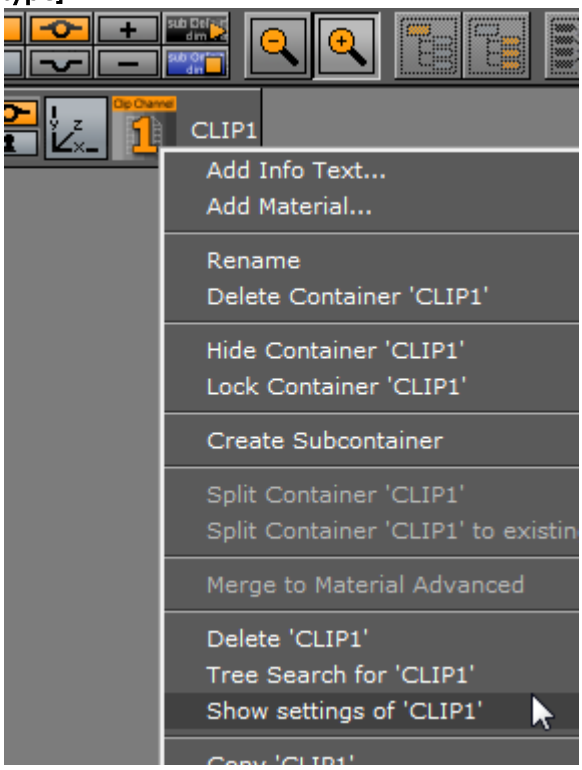
To View Media Asset (DVE or Texture) Properties

1. Add a Media Asset as DVE or Texture to a Scene.
2. In the Media Asset Manager select **DVE** or **Texture** (to view the added Media Asset)
3. Click on the required Media Asset.

⚠ **Note:** DVE or Texture view: If a Media Asset has been added to a Scene, its properties can also be selected by a <right-click> on the Media Asset icon in the Media Asset list (Built Ins), and select **Show settings of <clip channel>**.

To View Media Asset (Texture) Properties

- Right-click on the Media Asset icon in a container and select **Show settings of [media asset type]**.



or:

1. Click on the Media Asset icon in the Container.
2. In the Texture Editor click on the Media Asset image icon.
   In the properties panel the Media Assets tab is automatically selected, and the Media Asset Manager (Scene Tree panel) opens.

Texture Editor

To apply a texture to a Container drag a texture (e.g. an image) from the Server Panel to a Container. In the Container, an icon (a thumbnail of the texture) will show. If a texture is dragged to the Scene Tree area it will create its own Container, Geometries can be added later on, or the texture can be left as it is in the Scene. Open the Texture Editor by clicking the icon representing the texture, either on the Container or in the Properties editor.

⚠ **Tip:** Modifications to the texture in the Texture editor only apply to the texture in the selected Container.

Image Properties

Shows the added texture:

Two image textures can be used in one Container, however, Media Asset textures are limited to one. Click on a Media Asset image to open the editor for the Media Asset.

- **Default Texture** (1): Shows a thumbnail of the primary texture (the same thumbnail that is shown on the Container). To change the texture, drag another texture to the drop zone. If only one texture is used the other functions in the Image panel are inactive.
- **Alpha Texture** (2): Shows the thumbnail of a secondary (alpha) texture, if any. This texture is used as an alpha channel for the Default texture. When two textures are used the Image panel Alpha and Combine functions are active.
- **Alpha Buttons:** When a second image is added the Alpha buttons are active. The top button is the default button; the two images are shown as an alpha image. Click the lower button to view the two images blended, use the slider to blend as required.

  
-  **Combine:** When a second image is added, click this button to create a new single image from the two images. The new image will be created in the currently open folder in the Server Panel.

  ⚠ **Tip:** To save the new image in another folder, move to a different folder in the Server Panel before the Combine button is clicked.

- **Information**: Shows the original width, height, and file size of the primary texture.

  ⚠ **Tip:** To merge the two textures, click the little button between the textures. In the dialog box that opens, select a name for the new texture, and then click the Ok button. This will save the new texture to the database.

Unit Properties

Texture units take a texture and map them to a piece of geometry:

- **1-4:** If the graphics card supports it, it is possible to have up to four available texture units which can be used in parallel to do multi-texturing.
- **Inheritable:** Units can be inherited onto the textures of child Containers. Multiple inherit is possible, but be sure to use different texture units for each texture that should be inherited to not overwrite a unit.

Mapping Properties

The mapping option adjusts the way texture mapping is done:

- **Vertex**: Means that the texture mapping is done using the object texture, or UV, coordinates. These coordinates index a texture image, and are interpolated across the object to determine, at each of the object pixels, a texture image value. In most cases on 3D objects, this will give the best looking result. The built-in geometry plug-ins in Viz Artist have defined their texture coordinates. Imported objects can have texture coordinates if they have been generated by an external 3D modeling program. If for example adding a picture to a cube, all sides will look the same, and the whole image will be shown on each side.
- **Linear**: Is a 'straight forward' mapping where the image so to say is laid over the object directly on the frontal plane. The object is then moved through the image and every pixel on the object inherits the pixel it "touches" on the image. On 3D objects, this will cause a repetition of the pixels on the surfaces in the Z axis. If for example adding a picture to a cube, the front and back side will look good, but the remaining sides will be a repetition of the pixels laying on the edge of the front and the back side.
- **Reflection**: Maps the texture as if it were a reflection of a surrounding object. The reflection simulates that the texture image is an all-embracing object that casts its reflection on the object. The object acts in a way like a mirror.
- **Spherical**: Spherically texture projection onto a geometry. The mapping position and rotation can be adjusted with the corresponding parameters.
- **Cylindrical:** Cylindrical texture projection onto a geometry.
- **Cube**: Cube texture projection onto a geometry. This option needs a vertical cross cubemap which specifies how each of the six side projections should look like. A second image cannot be used with **Cube** Mapping (see Image Properties). When using Cube mapping method for textures, it is not possible to use the same texture on another object with a different mapping method.
- **Mapping Position**: Sets the texture projection origin for Spherical and Cylindrical mapping.
- **Mapping Rotation:** Sets the texture projection rotation for Spherical and Cylindrical mapping.

> ⚠ **Note:** Mapping position and rotation are only available for Spherical and Cylindrical mapping types.

Quality Properties

The quality parameters specify a trade between rendering speed and texture quality:

- **Pixel**: Shows the texture 'as it is' with no smoothing and anti aliasing. It takes little resources to render, but it can often create a poor visualization. This can be visible as 'toothed' lines and transitions, especially when close to the texture. A motion of either object or camera, especially in the Z axis, will create a 'blurring motion' on the texture, as the texture scales to fit the object.

- **Linear**: Performs a linear interpolation to smooth the texture when being magnified or shrunk. The texture looks good, but some distortions can be visible when the textured object is animated further away on the Z axis. As the object then gets smaller and smaller, the shrinking and interpolation of the texture creates a lot of “noise” on the texture. As a consequence, the linear quality is appropriate when the objects that have the texture do not change their size much.
- **Mipmap**: Performs a linear interpolation to smooth the texture. In addition, it offers a solution to the problem that shows on the two other qualities when the object is being moved away along the Z axis or shrunk. To avoid the “noise” that we see when a texture constantly scales to try to fit onto a shrinking/magnifying, the mipmap quality creates a set of prefiltered images in a decreasing scale down to almost on pixel. These are used as the object scales up and down and a more static look of the texture is thereby achieved. Mipmap quality absorbs more memory performance than the two other qualities.

> ⚠ **Note:** The following quality options are only available for the Mipmap quality.

- **Blur:** Makes the image blurry. The blur filter can be set to either Smooth, which blurs the image smoothly, or Mosaic, which makes the blur look like tiles.
- **Sharpen:** This option is useful when the rendering window is scaled to half height mode.
- **Anisotropic Filter:** The Anisotropic Filter can be set to either Off or 2-16x. Anisotropic filtering is used, when textured objects need to be rendered in z-direction, where textures usually get blurred.

Environment Properties

The environment parameters specify the way the texture RGBA information is combined with the object RGBA information:

- **Environment**: Select from:
  - **Blend**: Blends the RGBA parameters of texture and object.
  - **Decal**: Uses only the RGBA information of the texture and disregards the RGBA of the object.
  - **Modulate**: Uses the RGBA information of the texture and enables in addition use of lighting on the textured object.
  - **Add**: Adds source unit 1 and source unit 2 together.
  - **Add Signed:** Adds source unit 1 and source unit 2 then subtracts 0.5.
  - **Interpol.**: Linear interpolation between source unit 1 and source unit 2.
- **Blend Factor:** Defines in interpolation mode how much of source unit 1 and unit 2 is used (0.0%: shows current unit, 100%: shows source unit).
- **Source Unit**: Sets the second source unit for interpolation mode (can be all other units together or just a specific unit).

> ⚠ **Note:** Blend Factor and Source Unit environment options are only available for the Interpolated environment.

Wrap Properties

Wraps the object by repeating or clamping (no repeat – the texture's edge color is used to fill the rest of the object) the texture onto the object's surface. This setting comes into operation if the texture mapping cannot fill the whole object (e.g. the texture was scaled below 0.0):

- **Repeat**: Repeats the whole texture over again to fill out the object
- **Clamp**: Repeats the pixels on the end of the image to fill out the object

Compress. Properties

Texture compression can be used to save texture memory on the graphics card:

- **None**: No compression
- **DXT1**: Compresses to an RGB image. This format has a 6:1 compression ratio (DXT1 does not store alpha)
- **DXT3**: Compresses to an RGBA image with a 4:1 compression ratio
- **DXT5**: Compresses to an alternative RGBA image that treats the alpha channel different than DXT3 (alpha looks smoother). DXT5 has the same compression ratio as DXT3

Component and Constant Color Properties

- **Component**: Select from:
  - **Color**: Only the RGB color of the texture will be used. Set the alpha to a constant value in item 1
  - **Alpha**: Only the alpha of the texture will be used. Set the RGB color of the texture to a constant color specified in item 1
  - **Color Alpha**: Both the alpha and the RGB color of the texture will be used (default setting)
- **Constant Color**: Set the constant RGB color of the texture when **Alpha** is selected

Position Properties

- **Position**: Defines the position of the texture relative to the one stored in the picture.
- **Rotation**: Defines the rotation of the texture relative to the one stored in the picture.
- **Center**: Shifts the center of the image.
- **Scaling**: Scales the image that is used as texture. Before changing the scale values of the image, make sure to set the preferred kind of scaling:
  - **Single**: Scales each axis by itself.
  - **Locked**: Scales all axes at once, so that X, Y, and Z have the same value.
  - **Proportional:** Scales all axes at once, so that the values remain proportional.

Stereo Mode Properties

Select the correct stereo mode for the inserted image:

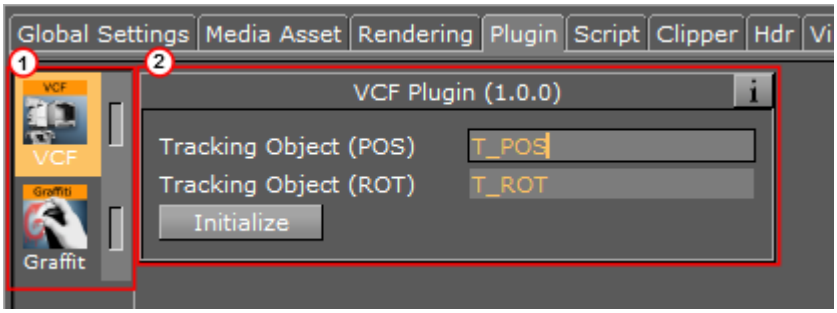
- **None:** No Stereo mode.
- **Config:** Images are drawn as set in the Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide).
- **Over Under Left Top** or **Over Under Right Top**: Two images are stored, one over (top) and the other under (bottom), left or right eye (top) first. The image height will be halved, providing half resolution.
- **Side by Side Left** or **Side by Side Right**: Two images are stored, side by side, left or right eye first. The image width will be halved, providing half resolution.

6.2.5 Plug-in Panel

Scene plug-ins can be applied to a Scene in the plug-in panel. Scene plug-ins are 'global' functions that have influence on the whole Scene. They are mainly used when external programs are interfaced with Viz Artist. Viz Artist uses an open interface for plug-ins. New plug-ins can either be obtained or written, and then installed into Viz Artist.

Installed Scene plug-ins can be found under Built Ins in the Main Menu. To view the Scene plug-ins, click on the **Built Ins** button and select **Scene plug-ins**, or click  in the Item Panel.

For a detailed list and description of Scene plug-ins, see Scene plug-ins.



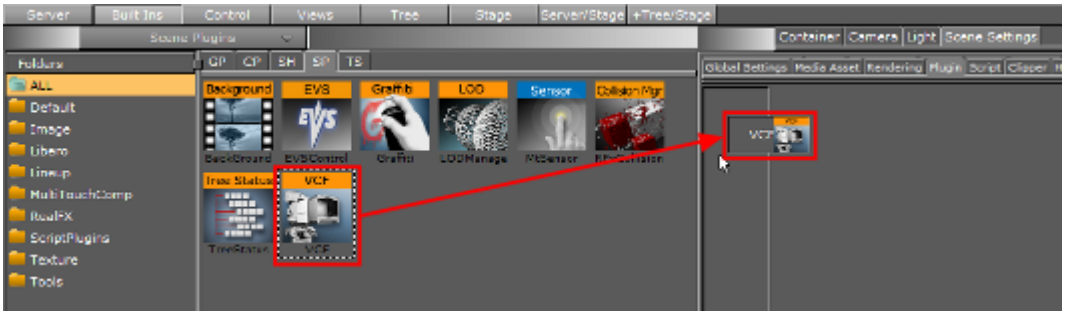
The Scene plug-in panel is divided in two sections:

- Scene plug-ins applied to a Scene, as icons (1)
- Scene plug-in editor (2 (opens when the Scene plug-in is highlighted)).

This section contains information on the following topics and procedures:

- To Add a Scene plug-in
- To Edit a Scene plug-in Properties
- To Show Scene plug-in Information
- To Enable/Disable a Scene plug-in
- To Remove a Scene plug-in

To Add a Scene plug-in



1. Click **Built Ins**.
2. Select **Scene plug-ins** (SP). Browse for the Scene plug-in to apply to a Scene.
3. Drag the Scene plug-in to the Scene plug-in panel:
   - Drag the selected Scene plug-in to the drop zone, or
   - Right-click on the Scene plug-in icon and select **Add to Scene...**.

Tip: If a Scene plug-in has been applied to a Scene, the Scene plug-in icon is orange:

To Edit a Scene plug-in Properties

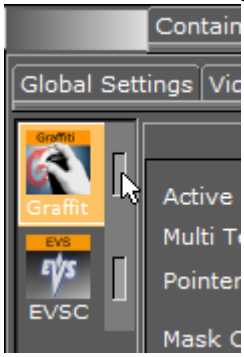
1. In the Scene Settings plug-in area, select the plug-in.
2. Edit the plug-ins properties in the editor to the right.

To Show Scene plug-in Information

1. In the Scene Settings plug-in area, select the plug-in.
2. In the plug-in editor, click the info icon.

To Enable/Disable a Scene plug-in

- In the Scene Settings plug-in area, click on the enable/disable switch in the icon.



For greater convenience, Viz Artist allows you to enable or disable the plug-ins already applied to a Scene. This gives overview when adjusting the plug-in properties to the values you need.

⚠ **Note:** After you add a plug-in to the scene, it will automatically be enabled.

To Remove a Scene plug-in

- Drag the plug-in to the trash can, or
- In the Scene Plug-in tab, right-click the plug-in and select **Remove from Scene**.

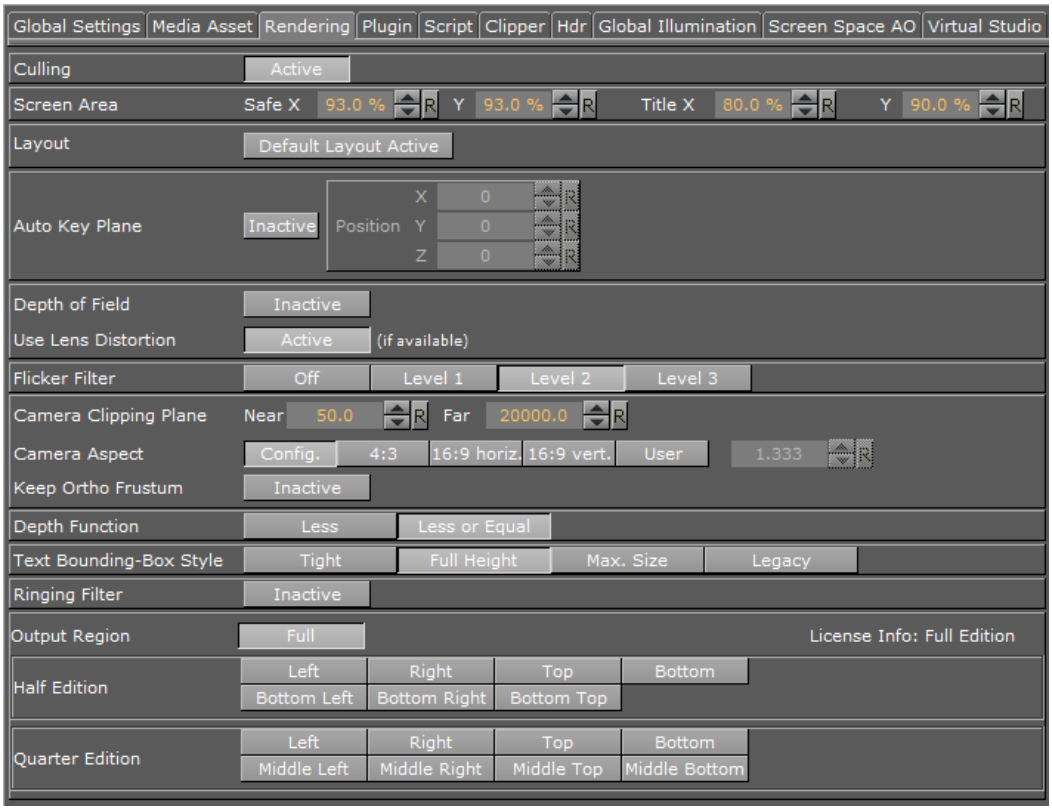
6.2.6 Rendering Panel

In the Rendering panel, basic rendering parameters can be modified.

This section contains information on the following topics and procedures:

- Rendering Properties
- To Design Graphics with a User Defined Aspect Ratio

Rendering Properties



- **Culling:** Saves performance for a scene when set to **Active**. Objects whose bounding box is outside the camera view are not rendered. This is of great importance to performance in large scenes and virtual studio. Normally Culling should be **Active**.
- **Screen Area:**
  - **Safe X/Y:** Sets the size of the Safe area X and Y values in percent of the screen resolution. Single click the **R** button on the X or Y to reset to its default value. Double-click the **R** button to reset both values to default values.
  - **Title X/Y:** Sets the size of the Title area X and Y values in percent of the screen resolution. Single click the **R** button on the X or Y to reset to its default value. Double-click the **R** button to reset both values to default values.
- **Layout**: The Video Wall Distributor defines a layout of the Viz Engine on air to be part of a cluster of Viz Engines:
  - **Default Layout Active**: Uses the video wall layout, as set in the Video Wall Distributor.
  - **Use Layout**: Makes the Video Wall layout inactive

  > ⚠ **Note:** Both these parameters are set in the Video Wall Distributor.

- **Auto Key Plane**: Makes it possible to define the X, Y and Z Position, that serves as a threshold between objects with and without key. The position can be set according to a tracking device carried by a person on the set, or it can be controlled by an operator.
  The auto key position decides which objects or parts of objects in the Scene that are to be part of the key signal. Only objects positioned between the auto key position and the camera will be rendered with the key signal.
  A typical setup for a Virtual Studio would be that the scene background is put into a separate channel of the external keyer.
  The objects of the background do not have a key function attached. The objects that are meant to be in front of the person on the set are placed in another channel/layer of the external keyer and have a key function attached.
  If the person on the set moves against the camera, the auto key function disables the key function of those objects that now are placed behind the person, thereby moving them into the background layer of the external keyer.
  To learn more about this, see the separate Viz Virtual Studio documentation.
- **Depth of Field**: Emulates the behavior of a real camera where objects further away from the focal plane show blurred and objects close to the focal plane show sharp. The respective parameters can be set in the camera settings under the Focus settings (see Camera Editor).
- **Use Lens Distortion**: Enables the lens distortion for this scene, see Advanced Lens Distortion.
- **Flicker filter**: Makes it possible to change the level of the flicker filter from Off and up to Level 3. The flicker filter is used to remove/reduce flickering from a scene. If a flicker filter is applied and the scene still flickers, try using the next level.
- **Camera Clipping Plane:** Sets the range of the virtual camera. **Near** sets the close range while **Far** defines the far range. Only containers within this range will be rendered.

  > ⚠ **Note:** The camera range is where the Zbuffer is within. So if Zbuffer problems arise, they may be solved by editing the camera clipping plane settings.

- **Camera Aspect**: Sets the camera aspect.
  - **Config:** Makes the camera aspect equal to the settings in the **Output Format** panel of the Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide).
  - **4:3, 16:9 Horizontal** and **16:9 Vertical:** Makes the camera aspect independent of the aspect for the rest of the scene.
  - **User:** Enables the user to set a user defined camera aspect ratio. When building a Scene with a customized aspect ratio for a **Video Wall**, set the user-defined camera aspect to the same value as the **Aspect** shown in the Viz Configuration (**Output Format** -> **User Defined** -> **Aspect** (which is set dynamically based on the other values). When building a Scene with a customized aspect ratio for other, e.g. hand-held/mobile devices or online publishing, set the **Keep Editing Aspect** (**Render Options** in Viz Configuration) parameter to **Editor** as this will not have an affect on the On-Air or Viz Engine output (see also To Design Graphics with a User Defined Aspect Ratio).
- **Keep Orthog. Frustum**: Takes the original frustum stored and does not allow its re-computation after changes have been made to the camera parameters in the orthogonal mode when active.
- **Depth Function:** Specifies the function used to compare each incoming pixel depth value with the depth value present in the depth buffer. Equal: passes if the incoming depth value is equal to the stored depth value. Less or Equal: passes if the incoming depth value is less than or equal to the stored depth value. This is OpenGL-specific.
- **Text Bounding Box Style**:
  - **Tight**: The bounding box is drawn tight around all text used in this Scene. This means just the height of the actual text used.
  - **Full Height**: The bounding box is higher. The bounding box is calculated for characters that need some space underneath, for example the letter ‘g’, and for characters that need more space above, such as a parenthesis, even if these characters are not used within the text in the scene. The Full Height setting does not work for special characters like such as 'Ä', which is even higher in some fonts (please refer to **Font/Text Options** in Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide)).
  - **Max. Size**: For Scenes imported from version 2.8 so that the various plug-ins which depend on font bounding boxes are compatible in version 3.5 and on, such as Max Size and Autofollow; it also includes a small trailing space between the letter and the bounding box.
  - **Legacy**: Produces the exact same bounding box as version 2.8.
- **Ringing Filter:** Reduces the ringing artifacts on the SDI output and applies only to SD formats. The ringing filter can be set to Off (see **Render Options** (set **Force Ringing Filter Off** to **Active**) under Viz Configuration in the Viz Engine Administrator Guide). The default Ringing Filter is located in **Scene Default Values** in Viz Configuration).
- **Output Region:** Becomes inactive if you design a Half or Quarter Edition scene. To revert to a full screen scene, click Full.
- **Half Edition**: If you have a full Viz Artist license, but want to design a scene for another license that has only a Half Edition, select which portion of the scene you want to design your scene in. Options are:

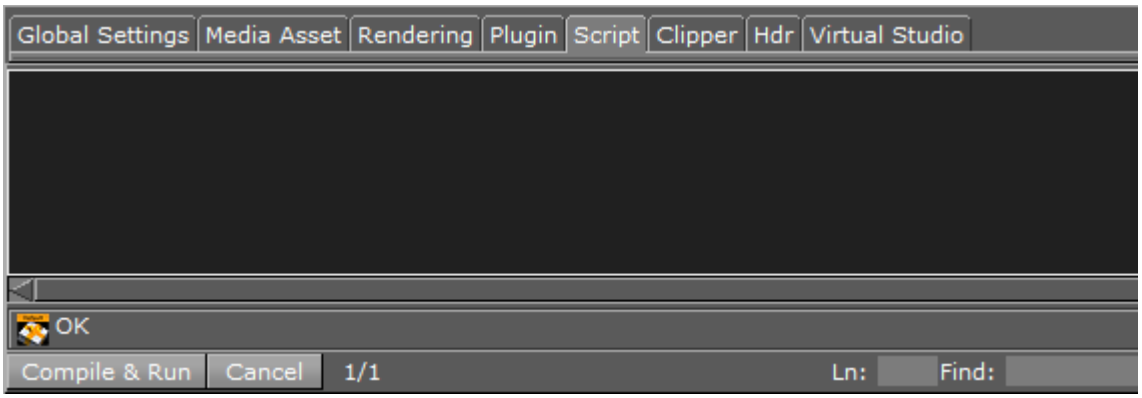
    - Left
    - Right
    - Top
    - Bottom
    - Bottom Left
    - Bottom Right
    - Bottom Top
- **Quarter Edition:** If you have a full Viz Artist license, but want to design a scene for another license that has only a Quarter Edition, select which portion of the scene you want to design your scene in. Options are:
    - Left
    - Right
    - Top
    - Bottom
    - Middle Left
    - Middle Right
    - Middle Top
    - Middle Bottom

To Design Graphics with a User Defined Aspect Ratio

1. Start Viz Artist.
2. Click **Config** (or press <F11>).
3. If you plan to render graphics in many different aspects (e.g. still images for online publishing) you should select the **Render Options** section and set **Keep Editing Aspect** to **Editor.**
    - This will make sure your graphics are rendered according to the scene's configured *camera aspect* (and not be affected by the Output Format aspect)
4. In the **Output Format** section set the format that best fits the screen resolution and aspect (e.g. 16:9).
    - Viz Artist is resolution independent and its workspace is optimized for designing in 16:9 and 4:3 aspects.
    - When designing graphics with a user defined aspect, only set the *Output Format* to size the aspect of your Scene Editor and adjust the *camera aspect* further to define it according to your scene's required and final aspect.
5. Close Viz Artist.
6. Start Viz Artist.
7. Click the **Server** button.
8. From the Properties Panel click the Scene Settings tab.
9. Under the Rendering Panel tab set **Camera Aspect** to **User.**
10. Enter your aspect ratio in the field beside the User field.
    - To calculate your aspect ratio simply divide width with height (e.g. 1600:900=1.778)

6.2.7 Script Panel

In the Script panel, you can add and edit scripts which are part of a Scene (see Script Editor).



6.2.8 Real Time Global Illumination

Global Illumination (GI), or indirect illumination, is a general name for a group of algorithms used in 3D computer graphics, that are meant to add more realistic lighting to 3D scenes. Such algorithms take into account not only the light that comes directly from a light source, *direct illumination*, but also subsequent cases in which light rays from the same source are reflected by other surfaces in the scene (*indirect illumination)*, whether reflective or not.

If you do not have a license for the Real Time Global Illumination, contact your local Vizrt Support or send an e-mail to license@vizrt.com to obtain a time-limited demo license or a full license.

This section contains the following information:

- Introduction to the Global Illumination function of Viz Artist
- Working with Global Illumination
- Global Illumination settings
  - Global Illumination
  - Render Mode
    - Render Mode Examples for Global Illumination
    - Debug modes for Global Illumination
  - Show Probes
  - Mesh Classification
  - System Generation
  - Precompute
  - Ambient Occlusion (AO)
- Creating a Scene with Global Illumination
- Limitations

Introduction to the Global Illumination function of Viz Artist

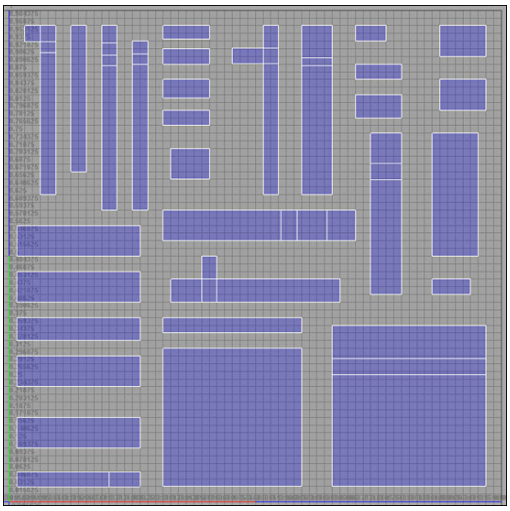
Introduced in Viz Artist 3.9, Real Time Global Illumination allows you to create Lightmaps and Ambient Occlusion maps that are modifiable in real time. This technique is based on the lightmap functionality widely used in leading game engines. When activating GI, you need to specify the computation settings, and precompute the Global Illumination and Ambient Occlusion maps. In Viz Artist, you find this feature section Scene Settings -> *Global Illumination*. The details of the settings are described below.

Working with Global Illumination

In general, Global Illumination requires two UV sets on any geometry that should be computed with GI: one UV set used for texture mapping, and one UV set used to create Lightmaps. There are two ways to create the Ligtmap UVs to get GI working properly. You can either import an `.fbx`-file with two UV sets that you have created by yourself. The first UV set must be used for texture mapping, and the second UV set is only used for Lightmaps. Or, you can use the first UV set (UV for texturing), and let the GI system calculate the according UVs for Lightmaps.

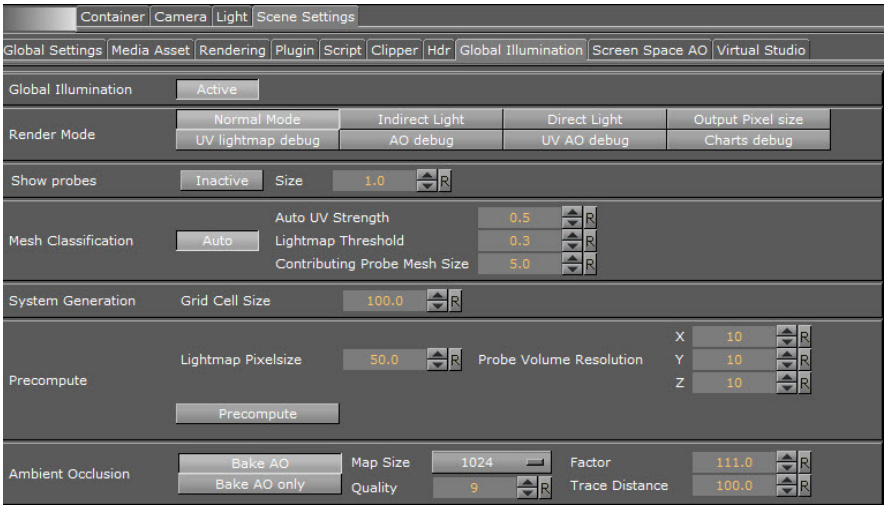
Global Illumination has an advanced algorithm that can extract the UV coordinates of your first UV set, and automatically create UVs for the Lightmaps. The better the layout of your unwrapped UVs for texture mapping are, the better the GI system can generate the Lightmaps based on this information. The most efficient way is to create your own UVs for Lightmaps. Here, it is important to have your UV layout in the range from `0` to 1. Also, all UVs for Lightmaps must be unique and non-overlapping, otherwise GI cannot compute all Lightmaps. In general, the layout of the UVs for Lightmaps look different than the layout of UVs for texture mapping. There are several tutorials available online concerning how to create UVs for Lightmaps.

The following is an example of correct UV Layout for Lightmaps. In the image below, you see the UV layout of the second UV channel, which is used for lightmaps only.



Precomputed Lightmaps do not use any memory on your GPU, so the texture memory of Viz is not reduced when having Lightmaps computed. The Lightmaps are not visible on your objects/ geometries in the scene tree, as they are applied to the rendering pipeline in the render window.

Global Illumination settings



Global Illumination

Available settings: `Active` / `Inactive`.

This function enables or disables Global Illumination for your scene. For the final result, Lightmaps must be precomputed. If no Lightmaps are precomputed, it will enable a general pixel shading for your scene.

Lightmaps are stored inside the Viz Graphics Hub. When saving as a new scene (**Save as**), the Lightmaps is precomputed again as theyare stored and linked to the UUID of your scene. If you have already calculated the Lightmaps, switching between `Active` and `Inactive` loads or unloads them. The time required to load and unload lightmaps depends on the size of your scene and Lightmaps. While loading the scene, the lightmaps are copied from the Graphic Hub to a local folder, for faster loading times. When the scene is open, the lightmaps files can be found in the Windows temporary folder:

> **Example**
>
> **Example**: `C:\Users\[user name]\AppData\Local\Temp\vizrt\Viz3\`

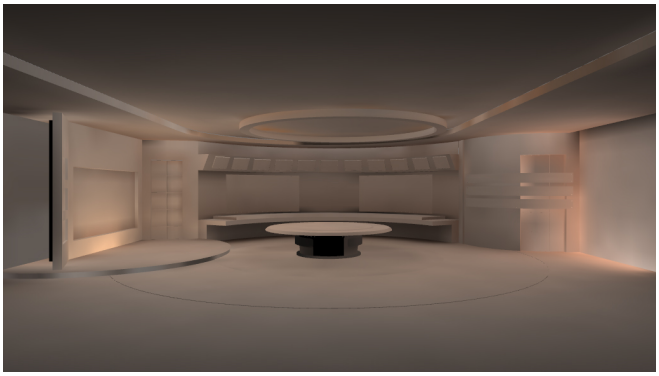
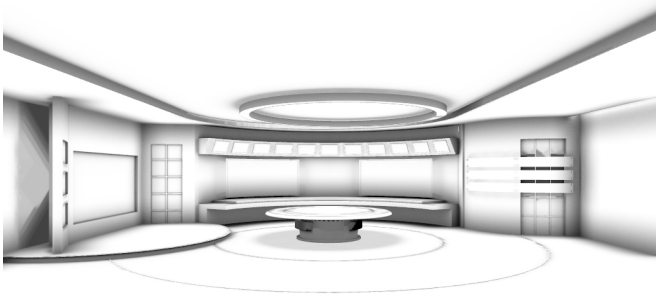
The folder name the Lightmaps are stored in corresponds to the UUID of the scene they belong to.

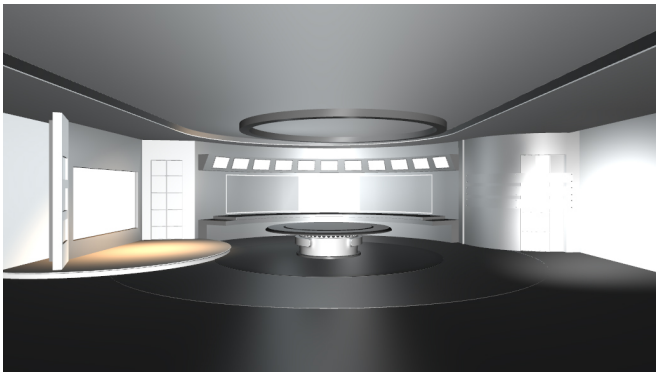
Render Mode

Available settings: `Normal Mode` / `Indirect light` / `Direct Light` / `Output Pixel size` / `UV lightmap debug` / `AO debug` / `UV AO debug` / `Charts debug`

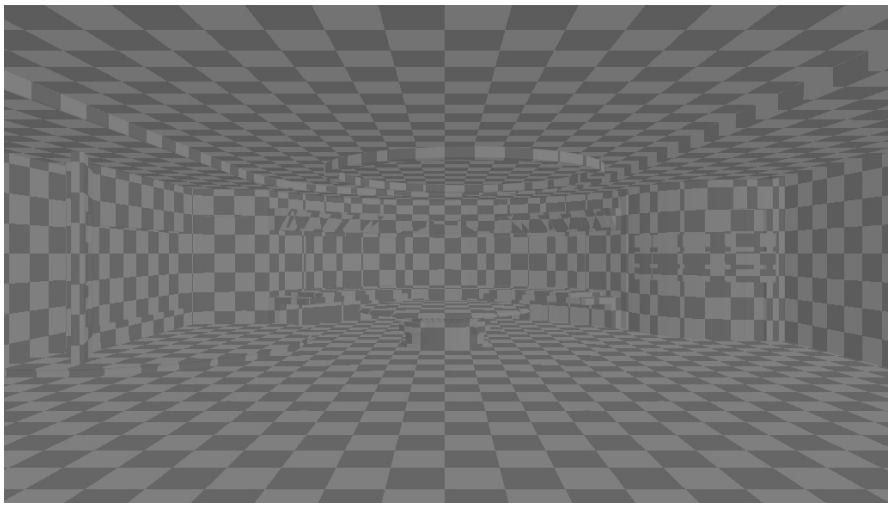
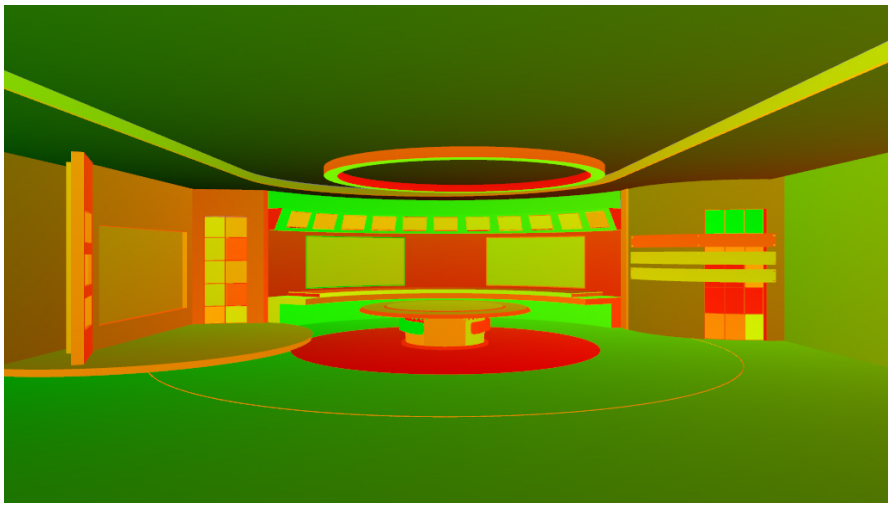
These are different visual modes for debugging the different render passes: the **Ambient Occlusion** render pass, the **Indirect Light** render pass, and the **Direct Light** render pass. You can also debug the **Pixel Size**, which controls the quality of the Lightmaps, or view the UVs for the Lightmaps, Ambient Occlusion, or the UV Charts. `Normal Mode` presents all maps together as a final result.

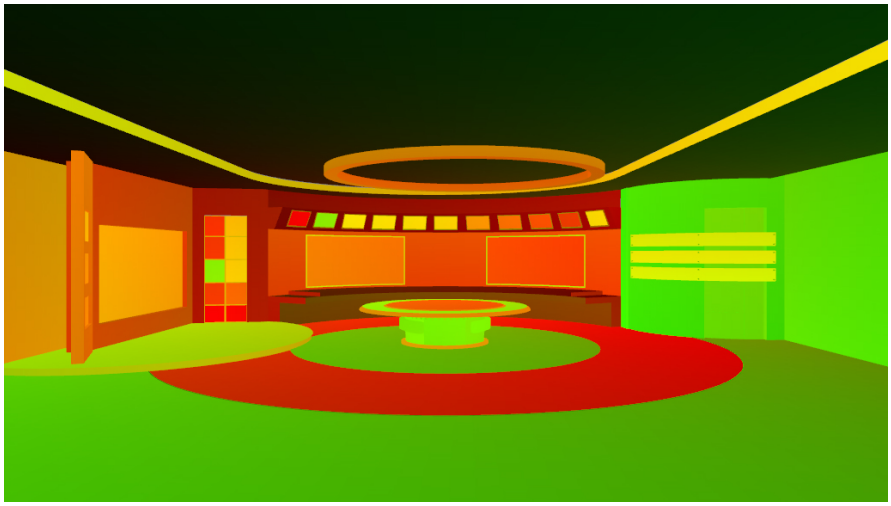
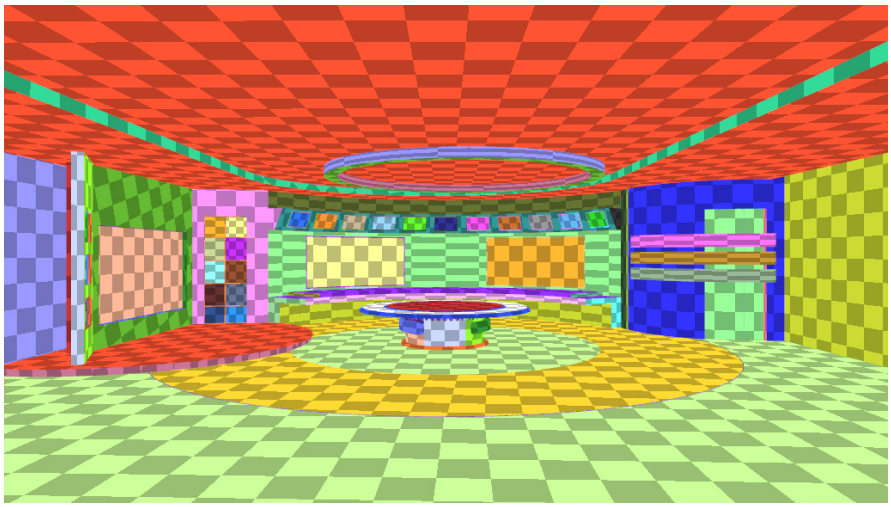
Render Mode Examples For Global Illumination

Render Mode	Visual Example
Global Illumination Normal mode This view represents the final output, with direct light, indirect light and ambient occlusion.	
Global Illumination Indirect Light This view is represents the direct light (Global Illumination) only.	
Global Illumination Ambient Occlusion This view is represents baked Ambient Occlusion map only.	

Render Mode	Visual Example
Global Illumination Direct Light This view is represents direct light only.	

Debug Modes For Global Illumination

Debug Mode	Visual Example
Global Illumination Pixel Size This view shows the output of the Lightmaps in pixel size.	
Global Illumination UV Lightmap debug mode This view shows the lightmap UV's only.	

Debug Mode	Visual Example
Global Illumination UV AO debug This view shows the ambient occlusion UV's only.	
Global Illumination Charts debug This view shows the individual UV charts.	

Show Probes

Enable or disable debug view for the light probes inside your set. The light probes provides you with a visual representation of how light is distributed within your scene. The **Size** parameter adjusts the size of the light probes in the viewport, but does not influence computation of the Lightmaps or light probes themselves.

Be aware that not every object can be lit by lightmaps, due to the complexity of the UVs. This applies especially to organic shapes, for example, a statue. Objects that cannot be lit by lightmaps will be probe-lit.

Mesh Classification

This is the most important setting of Global Illumination. Mesh classification automatically classifies meshes for Lightmaps and light probes, and creates the necessary UVs for them. If Mesh Classification is turned `Off`, only the UVs from the second UV channel is used to generate the Lightmaps. If you set it to `Auto`, Mesh Classification generates the UVs for the lightmaps from your

first UV set of your geometries/meshes. The computation of Global Illumination is faster when a second UV set is available, as analysis of the geometries and generation of additional UVs is not required.

Mesh Classification can be adjusted with the following parameters:

- **Auto UV Strength:** Controls how aggressively the Auto UV process flattens the meshes onto planes. Lower values lead to better UVs with less overlap. Thus, more meshes are classified as lightmap-lit, at the expense of larger UV space requirement. Higher values result in worse UVs, with more meshes classified as probe lit, and lower UV space usage.
- **Lightmap Threshold:** Controls the threshold where Auto UVs are considered bad enough that a mesh should be probe-lit instead of lightmap-lit. Lower values increase the number of meshes probe-lit, and higher values increase the number of meshes that are lightmap-lit.
- **Contributing Probe Mesh Size:** Controls how big a probe-lit mesh and it's surroundings have to be, before it comes a Contributing Probe.

System Generation

Grid Cell Size: Splits up the scene into systems of cubical cells of this dimension in World units.

Precompute

Lightmap Pixel Size: Defines the Output pixel size for lightmaps. If the pixel size is set too low, depending on your scene complexity, you will not be able to generate all Lightmaps. This is because there is only a certain amount of memory available for all Lightmaps.

Using a smaller pixel size value will increase quality, and will cause an error message when the maximum amount of pixels is exceeded. When adjusting the pixel size, make sure you are working with real world coordinates and transformations/scaling on your meshes/scene.

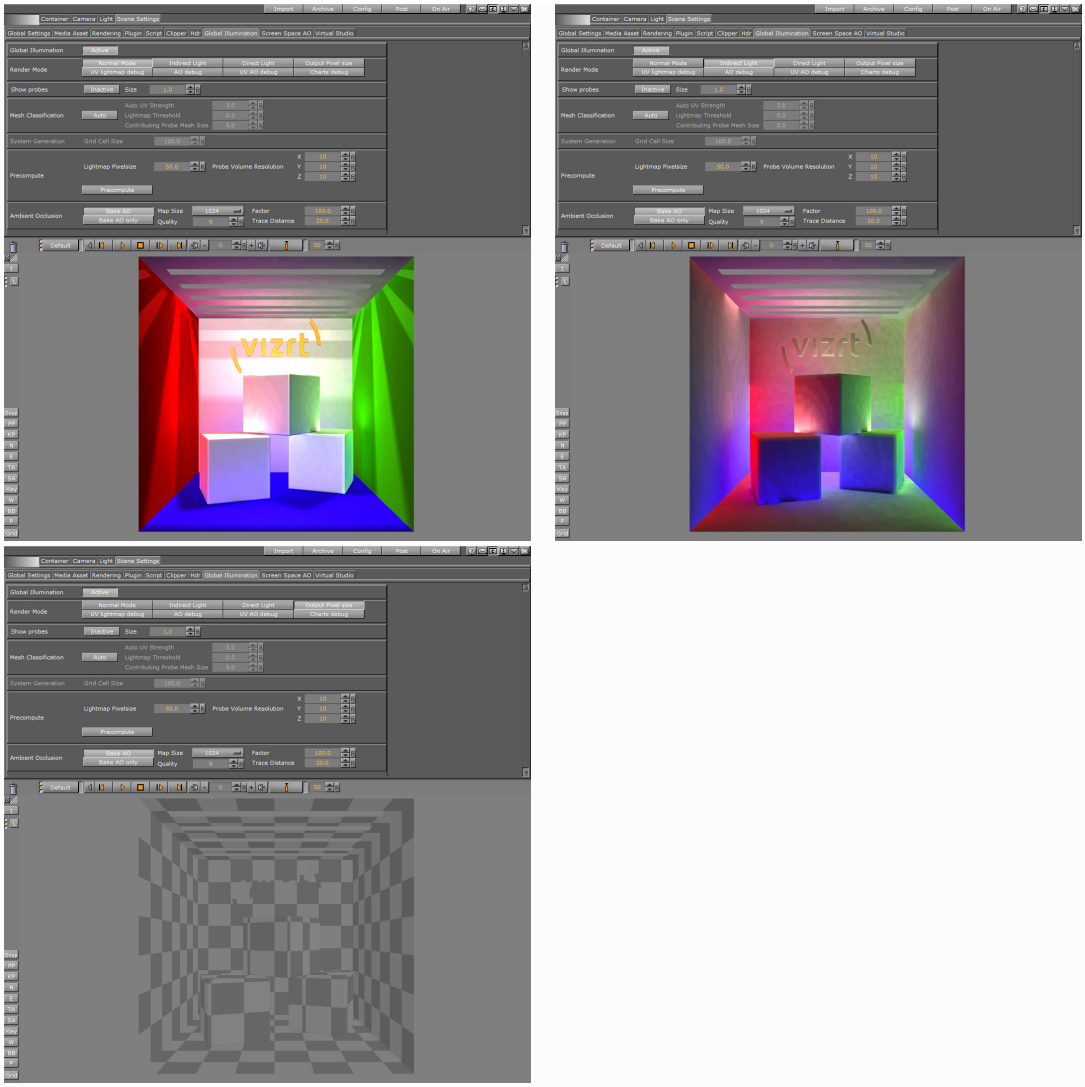
For a good start with low computation time, set the pixel size to `50` on a `1` x `1` meter cube (two pixels in one cube).

The following two examples shows the difference when using a lightmap pixel size of 50, and a pixel size of 10. Lower pixel sizes increase quality, but also increases computation time.

Example

Example 1:

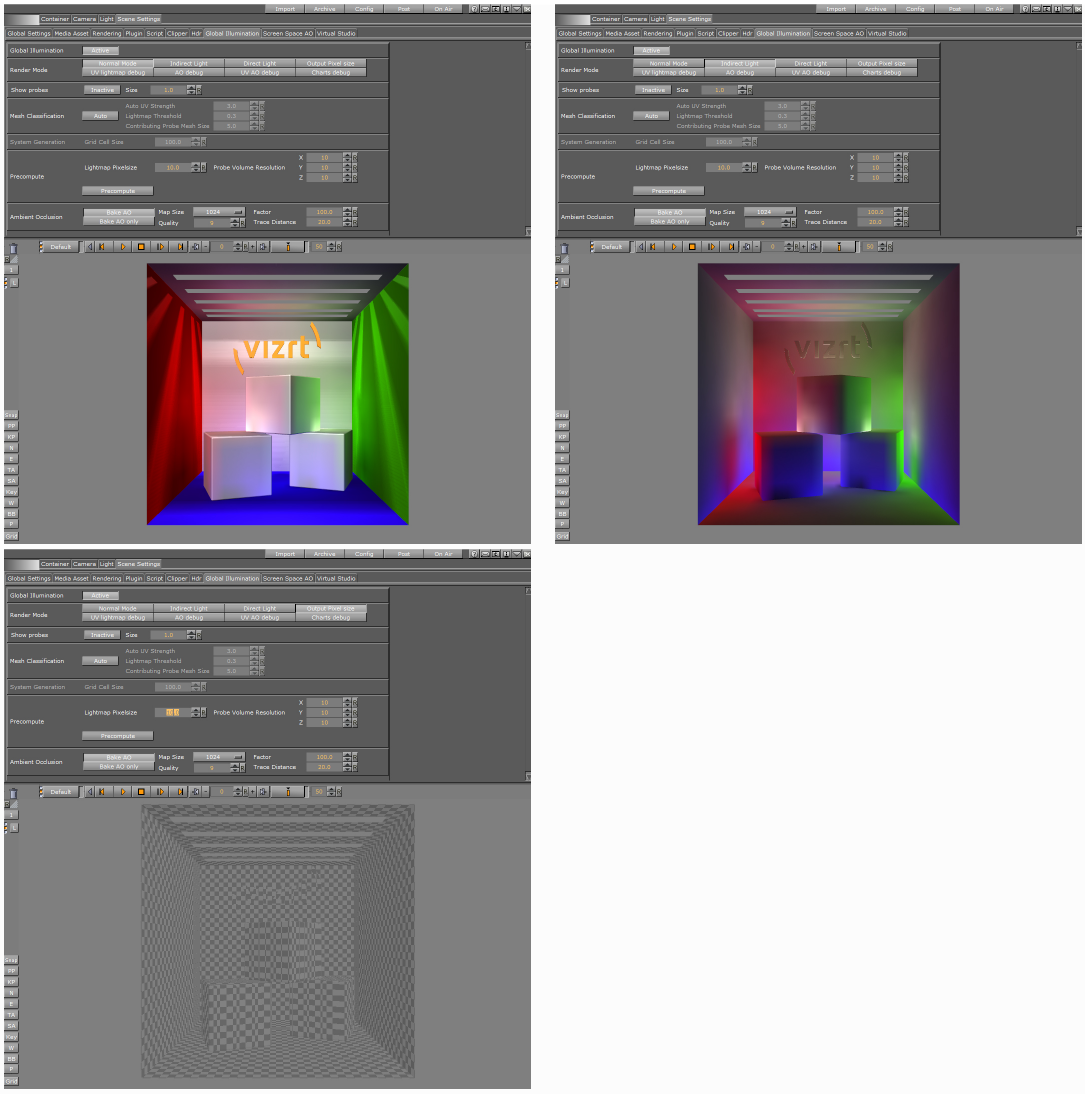
In this example there is visible noise in **Normal Mode**, and also in the **Indirect Light** render pass. The pixel size used in this scene is `50`.



Example

Example 2:

In this example there is no visible noise. The pixel size used in this scene is `10`.



Probe Volume Resolution: Defines the number of probe samples in the scene in x-, y- and z axis. When activating the ***Show Probes*** feature you can see the amount of probes in your scene. When changing the amount of probes you need to recompute the Lightmaps again to see the changes.

Precompute: Pressing the **Precompute** button to generate all lightmaps. An overlay window appears, showing you detailed debug info. The first overlay window displays the scene analysis procedure. When this task is done, a second overlay window appears, displaying the baking procedures.

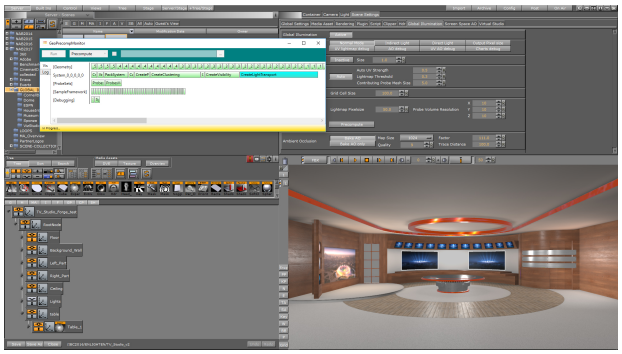
⚠ **Note:** You must precompute all maps again each time you add additional objects to your scene.

As individual objects cannot be computed on their own, precompute always recalculates the whole scene. If you have wrong UVs, or any other issues, for example with the pixel size, a window with error messages will appear. You can read the detailed error messages and computation

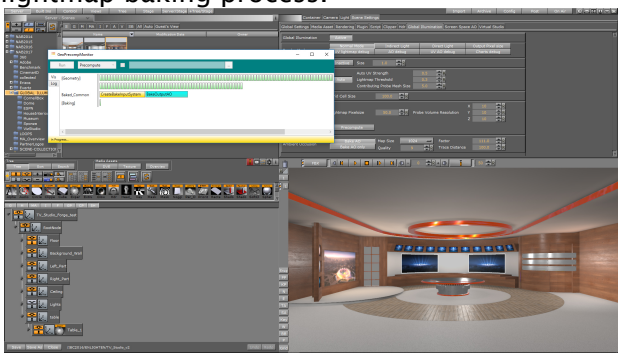
information in the log-file. The error messages are available in the scene info menu, which turns red in the case of any computation issues.

Precompute Computation Examples

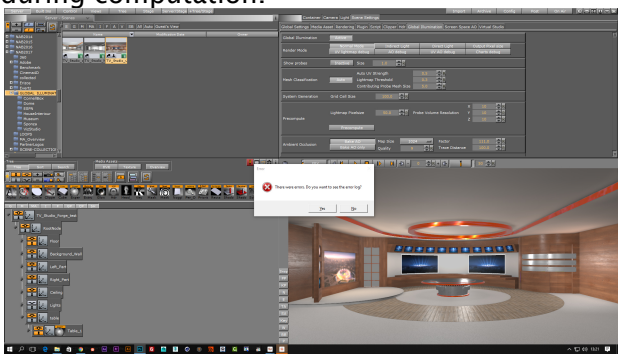
1: The Geo Precompute Monitor during scene analysis and UV preparation.



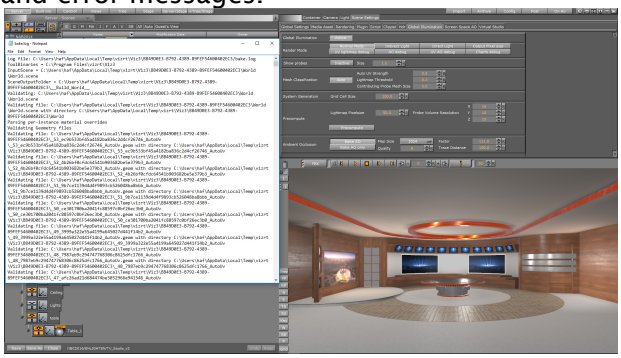
2: The Geo Precompute Monitor during the lightmap baking process.



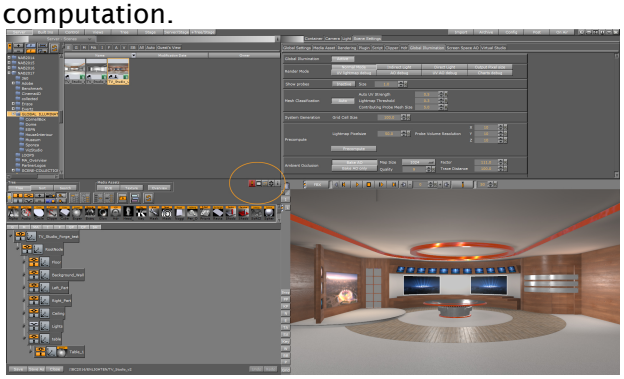
3: A pop-up window when errors occurred during computation.



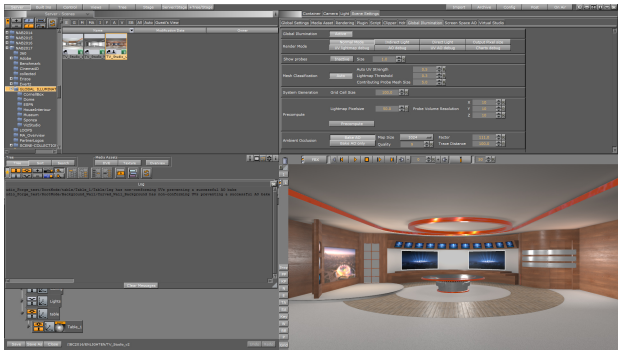
4: The log-file with computation information and error messages.



5. The scene info error log. A red icon indicates that errors occurred during computation.



6. Error messages in the scene info error log.



This example shows some geometries with non-conforming UVs preventing a successful AO bake.

Ambient Occlusion (AO)

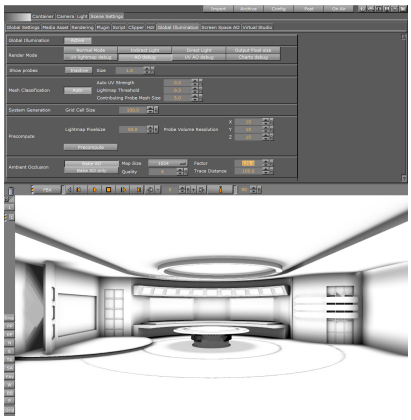
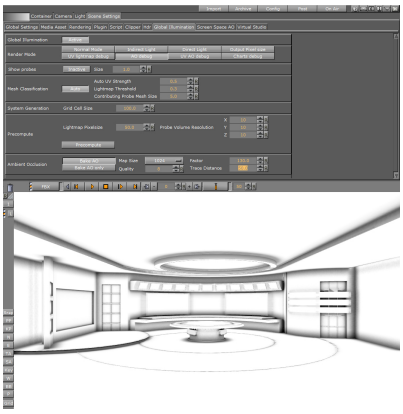
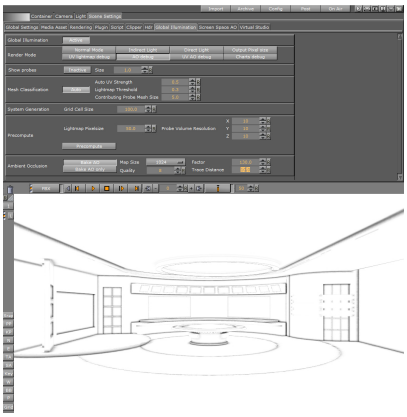
Bake AO: Recalculates the AO maps for every bake process (precomputation) when set to **active**. When Bake AO is set to **off**, only the indirect light maps are calculated when you press the **Precompute** button. Enable **Bake AO only** to recalculate the AO maps only. This is useful when you only to play with the AO without adjusting your Global Illumination parameters, as it reduces the computation time by only calculating the AO maps. For best results, always run a full computation process, as the GI and AO maps will not work properly together if computed with differences. This can happen if you change or delete objects in the scene.

Map Size: Specifies the size of your AO maps for the scene, ranging from 512 pixels up to 16K. The default setting is 512 pixels.

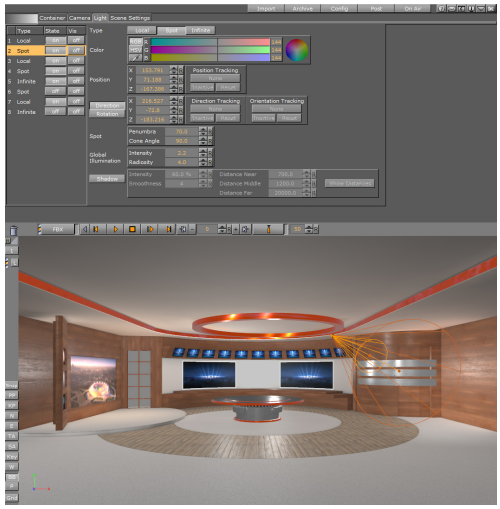
Quality: Increases or decreases the quality of your AO maps. Increasing quality will also increase calculation time. The quality setting has a range from 4 to 17. The default setting is 8.

Factor: Changes the factor to influence the strength/blending of the ambient occlusion. You can change the factor at any time without recomputing the meshes/scene.

Trace Distance: Sets the distance where a point shall be considered occluded. When you change this value, you cannot see the results in real time, as you must recompute the AO maps for the changes to take effect. Usually, the double value of the Lightmaps pixel size is used here; if your lightmap pixel size is set to 50, you should have a value of 100 in the Ambient Occlusion Trace Distance field. Since this comes down to the individual designer's style, use any value you like here. The following images show some examples:

Trace Distance: 100	Trace distance: 50	Trace Distance: 10
		

Finally, you can now play with the **Radiosity** setting in the Light Editor:



The input controls for Global Illumination are only visible in scenes where Global Illumination is set to `Active`.

Creating a Scene with Global Illumination

First, you need to import a geometry or scene with proper UVs and normals. This can be either just one UV set with unwrapped UV coordinates to be able to do a proper texture mapping and extract the lightmaps from this UV set. You can also import an `.fbx`-file containing the second UV set, with a layout for lightmaps created in your preferred 3D application. On your imported geometries make sure the normal orientation is facing outwards, flipping the normals with the Expert plugin will lead to wrong computation results. Do not forget to turn **Auto Mesh Classification** on or off, depending on your existing UV sets.

Next, if your scene is not already modeled or imported with the proper size, you need to set up the real world size of your scene before you do the baking. Then place some lights in your scene. After this, adjust all the parameters described above and run a test bake, to see if you get any error messages concerning UVs or baking. If your render passes look good from the debug views, no error messages appeared, or at least no visual errors are visible, you can start working on the detailed light setup.

All the light setups, color, type, transformation, can be done at any time now, without a need to recompute the whole scene. This is the benefit of working with lightmaps. It is very important to increase the radiosity setting of the lights that light up the whole scenery, using the Light Editor. Be aware that Lightmaps are static maps that react in real time to the GI parameters. They are not dynamically calculated with animations. When you have an object, like the Cornell box example above, and animate the rotation of one of the cubes inside the Cornell box, you will see wrong colors reflected during the rotation. This is because the lightmaps are statically projected onto geometries. When rotating the middle cube of the Cornell Box example, which is receiving a green color from the green wall next to it, it will not turn red when rotating to the other side, because lightmaps are static. The same applies to Ambient Occlusion maps. Whenever you want to animate an object, you should not compute the lightmaps or Ambient Occlusion maps for it, as otherwise the result will not look correct. If an object is hiden in the Scene Tree it will not be calculated with GI.

After setting up all the parameters and running a successful precomputation process, switch to the **Indirect Light** debug view from the **Global Illumination** pane, to see how the radiosity intensity of the light affect your scene. Experiment with different lights and radiosity settings to get your desired result. During this process, you should switch between the debug view for **Indirect Light**

and **Normal Mode**, to see how much GI affects in your scene. Now, you can start refining your scene with adding Substance Shaders to your scene to increase realism. Compared to using regular plain textures, substance's light reflection is aiming a physical correct behaviour depending on the shader properties. This is something a regular texture will not do.

If you want to work with plain textures only, this is possible. Be aware that no other material and effect shaders than Substance Shaders are supported in combination with Global Illumination.

To get your light color correctly spread into your scene, you need to apply a material to every geometry and use the eye dropper tool of the material editor to get the correct color from the texture or shader used on the selected geometry. Otherwise, radiosity cannot blend the surface color into the set, since it is not possible to extract the color directly from textures or shaders. Applying a material with color will not influence your Substance Shader, as this material color will only be used for the color bleeding of the radiosity.

Limitations

Global Illumination is built to work in conjunction with Physically Based Rendering (PBR) materials, to get the most realistic results. In general, it is common to use Global Illumination in conjunction with Substance shaders. Substance shaders are the only available PBR shaders for Viz Artist. Therefore, any other material shader, such as RTT and others, will not work with Global Illumination. When using a feature which is not supported by Global Illumination, you will get an error message in the scene info. Error messages in the scene info are indicated by a red info icon.

The following list are examples of features and/or plug-ins that are not supported in combination with Global Illumination:

RTT Shaders, Softmask-, Framemask-, ImageMask-, Fluid Effect-, Water Shader Plugin, Drop Shadow-, Emboss Plugin, Anisotropic Light Plugin, Cartoon Shader, Gooch Shader, Lighting Shader, Normal Map Plugin

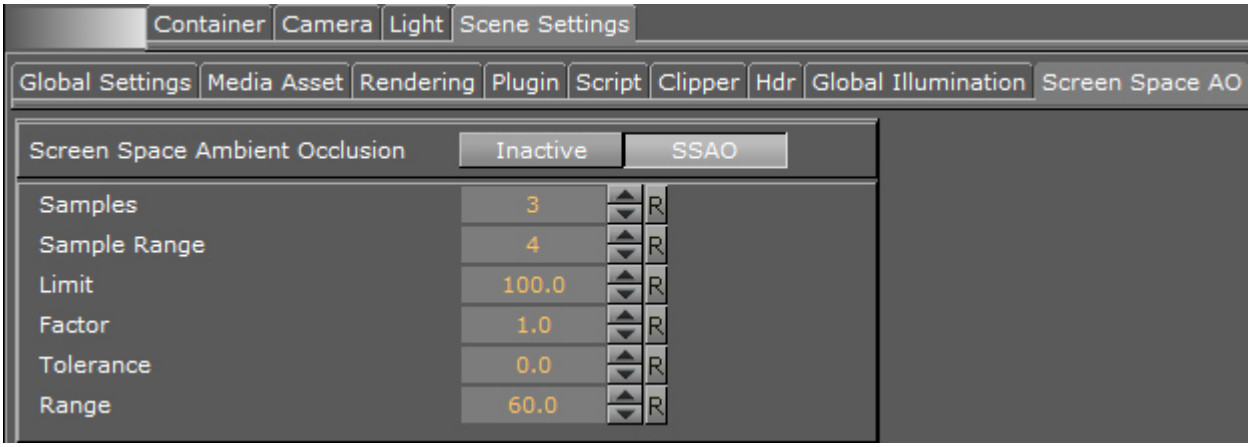
Filter Blend-, Filter Blur-, Filter Color Balnce-, Filter Radial Blur-, Filter Sepia-, Filter Sharpening Plugin, HDT Plugin, etc....

6.2.9 Screen Space Ambient Occlusion

Screen space ambient occlusion (SSAO) is a computer graphics technique for efficiently approximating the ambient occlusion effect in real time. SSAO is mainly used for virtual sets, and for augmented reality objects and scenes. In regular broadcast graphics, for example a lower third, SSAO is hard to notice. This is because the details in these types of graphics are too small for the effect to be visible.

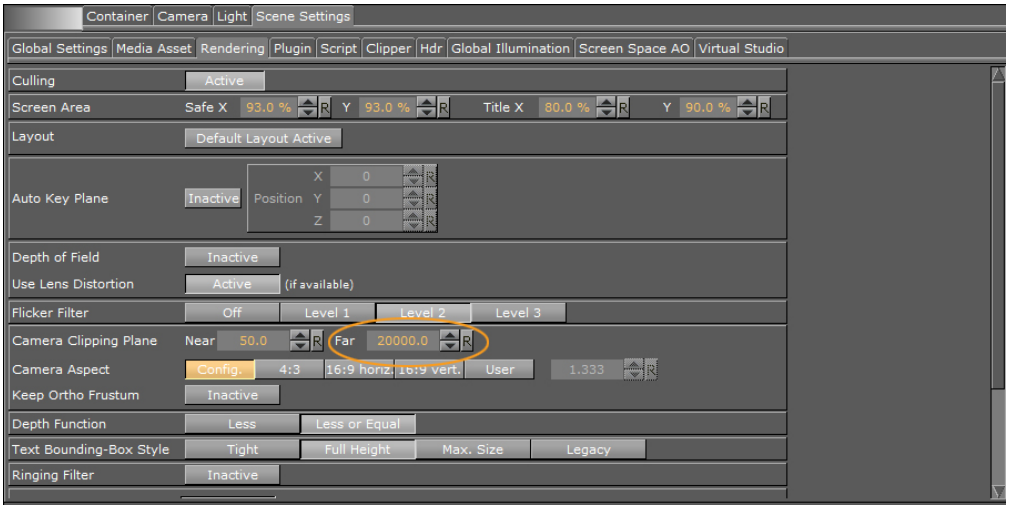
To activate SSAO for your scene, or adjust the settings, open Scene Settings and select the **Screen Space AO** tab. SSAO is applied to the final rendering image. This means that when SSAO is active, all objects in your scene are rendered with SSAO.

Adjusting the settings may impact performance, depending on the parameters changed: Increasing the **Samples** and **Sample Range** values will increase the load on the system GPU for SSAO computation, while improving the result. Changes to the **Limit**, **Factor**, **Tolerance** and **Range** settings affect the visual look of the SSAO, without impact on performance. **Limit**, **Factor**, **Tolerance** and **Range** influence the SSAO grayscale.

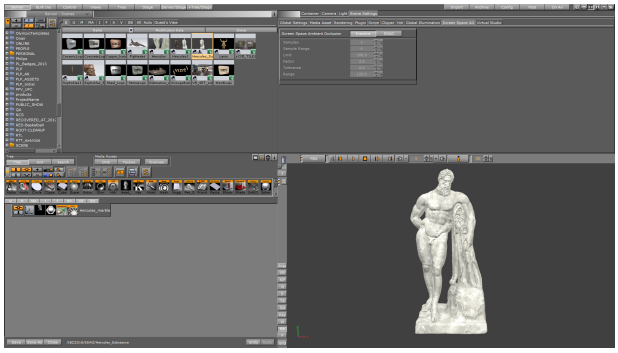
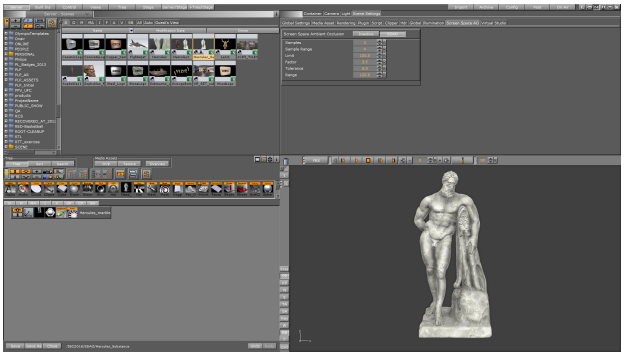
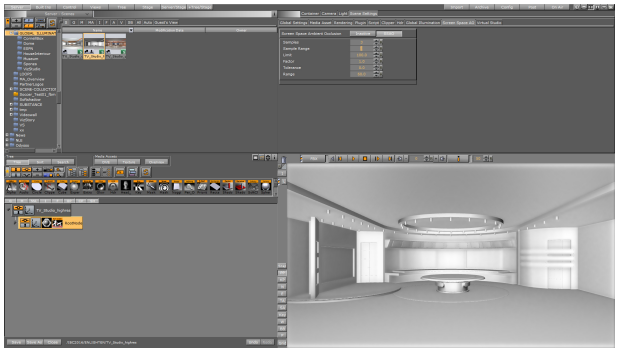
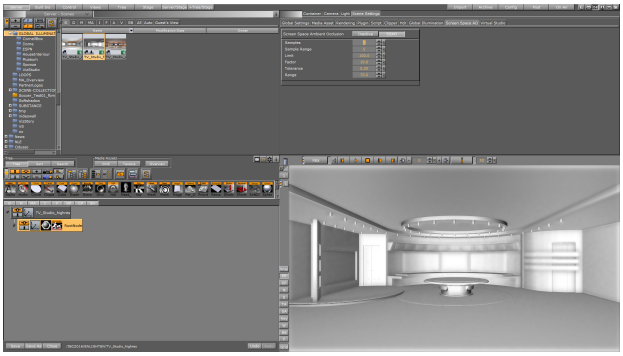


To see the result in your Render view, press the **PP** button to activate the Post Processing effects. If you are loading scenes into Viz Engine via an external control application, the SSAO will be visible both on the final output and the VGA preview.

> ⚠ **Tip:** To get the best result with SSAO, decrease the **Far** value for **Camera Clipping Plane** value to the maximum depth of your scene. By default, Camera Clipping Plane far is set to `20000.0`.



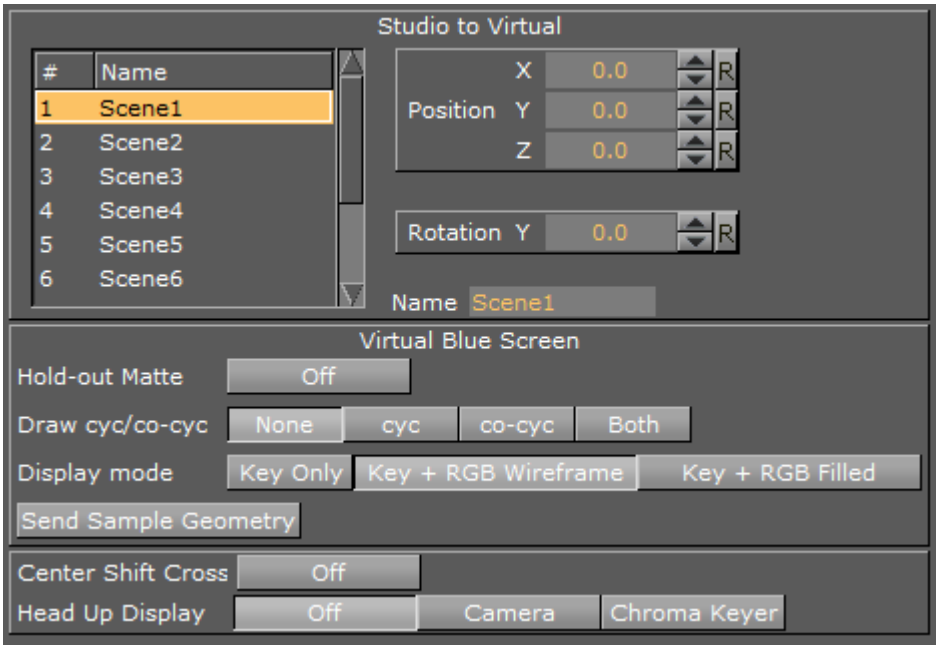
The following examples show the difference between scenes rendered with and without SSAO. SSAO gives the object a more realistic look. By changing the parameters of the settings, you can control the quality, detail and render performance of the SSAO. The objects in the examples provided below have the same light setting, and cast no shadows. The only difference is the Screen Space Ambient Occlusion.

Without SSAO	With SSAO Active
	
Untextured Virtual Set without SSAO	**Untextured Virtual Set with SSAO Active**
	

6.2.10 Virtual Studio Panel

The Virtual Set panel is only available if the Virtual Set option is installed. This panel makes it possible to set up and save configurations for virtual sets.

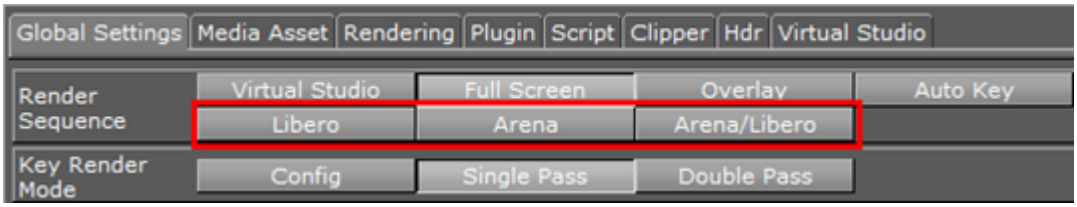
For information about the parameters, see the Viz Virtual Studio User Guide.



- **Studio to Virtual**: Sets the camera view offset in the selected Scene.
- **Virtual Blue Screen**: Check and view settings:
  - **Hold-out Matte**: If set to **On** chroma keying and color correction only happens in the cyc area, as set in Viz Virtual Studio. The trash matte is not applied.
  - **Draw cyc/co-cyc**: Selects the Trash Matte.
  - **Display Mode**: Sets the Trash Matte visibility.
  - **Send Sample Geometry**: Sends sample cyc/co-cyc geometry to the Viz Engine.
  - **Center Shift Cross**: Shows or hides the Center Shift Cross in the Scene Editor.
- **Head Up Display**: Set to **Off** (no head up display), or show either the **Camera** parameters or the **Chroma Keyer** head up display, in the Scene Editor.

6.2.11 Viz Libero and Viz Arena Render Sequences

This section details the render sequences for the Viz Libero and Viz Arena (see Global Settings Panel).



This section contains the following topics:

- Viz Libero Render Sequence (When integrated with Viz Engine)
- Viz Arena Render Sequence (Mechanical Based Tracking)
- Viz Libero/Viz Arena Render Sequence (Image Based Tracking)
- Rendering Support for Dirty Video Feeds with Viz Arena
  - To Add Image Masks for Dirty Feeds for Viz Arena

Viz Libero Render Sequence (When integrated with Viz Engine)

1. **Back Layer**: Not rendered, even if a Scene is present.
2. **Main Layer**:
    a. Background image (live video) is rendered.
    b. Containers with the Key plug-in, with **Combine with BG chroma key** set to **On** and **Render Mode** set to **Add**, are keyed with the background video. The final key of these items results from the key information of the background video, provided by Viz Libero, and the key information of the Container.
    c. Containers with standard keying are rendered with optional lens distortion and linearly keyed over.
3. **Front Layer**: Only containers with a linear key are rendered on top, without lens distortion.

Viz Arena Render Sequence (Mechanical Based Tracking)

1. **Back Layer:** The keying information for Containers in the Main Scene is created. Typically objects with the Key plug-in on them are used for this:
    a. Background image without alpha and without lens distortion.
    b. Containers which contain standard key information either as linear key or generated through chroma keying.
2. **Main Layer**:
    a. Containers with the Key plug-in, with **Depth information only** set to **On**, are rendered. Only the depth information is rendered and used to cutout "real" objects from chroma keyed virtual objects.
    b. Containers with the Key plug-in, with **Combine with BG chroma key** set to **On**, are rendered. Depth and chroma keying information are used to superimpose these Containers.
    c. Containers with the Key plug-in, with **Depth information only** set to **On**, are rendered. The **depth** information is used for Containers which should be keyed linearly.
    d. Containers with a linear key are rendered with lens distortion and linearly keyed over.
3. **Front Layer**:
    a. Background image is rendered. This only makes sense if the last visible layer for SDI output is set to Main Layer. If this is so, a different video source can be shown on the Viz Engine preview to prepare the next item(s), which will go on air.
    b. Containers with a linear key are rendered with lens distortion and keyed over.

Viz Libero/Viz Arena Render Sequence (Image Based Tracking)

1. **Back Layer**: Nothing is rendered from the Back Layer Scene.
2. **Main Layer**:
    a. Background images are rendered.
    b. If a foreground image (Stencil Mask) is present, but not active, only the alpha information will be used to mask out parts of the containers in the Main Scene. Alpha will be rendered without lens distortion.
    c. Containers with the Key plug-in, with **Combine with BG chroma key** set to **On** and **Render Mode** set to **Add**, are keyed with the background video. Chroma keying information is used to superimpose these Containers.
    d. Containers with a linear key are rendered with lens distortion and linearly keyed over.
3. **Front Layer**:
    a. Containers with a linear key are rendered.

Rendering Support for Dirty Video Feeds with Viz Arena

A "dirty" video input signal is a video containing graphical items such as logos, banners, clocks, etc., that are overlaid on top of the video signal before being sent to the Viz Engine. To avoid disturbances in the viewing experience, Viz Engine has to render its graphical items "below" the preexisting overlays. If supplied with a "dirty" video input signal, Viz Engine can suppress rendering of graphical items within certain image regions. To do this, alpha image masks need to be sent to Viz Engine, defining in which regions of the image graphical items must be rendered, and in which rendering shall be suppressed.

In other words, areas with existing graphics from the broadcast need to be excluded to avoid rendering new graphics on top of them. To achieve this, an alpha image needs to be loaded as a foreground image in the scene. The area with alpha information will then be excluded from rendering, working as an inverse mask. Viz Engine then incorporates the currently activated image mask(s) for rendering of graphical items. In image regions where the value of the image mask is below 255, the alpha of all rendered graphics is adjusted accordingly. Applying the image mask also works when the render scale is different than 1. Independent of the currently selected render scale, the size of the image mask always refers to the original video resolution.

⚠ **IMPORTANT!** The foreground image needs to be set to inactive to ensure only the key is applied.

To Add Image Masks for Dirty Feeds for Viz Arena

1. An alpha image must be created, with the exact width and height of the currently used resolution. For example, for 1080 resolutions, the size is 1920x1080 pixels. As images with a lower resolution will be stretched, possibly giving unexpected or unwanted results, make sure to always create the alpha image with the required resolution.
2. Load the image with the alpha information as a Foreground image in the scene. The area with alpha information will be excluded from rendering.
3. Select `Arena/Libero` as **Render Sequence** in the Global Settings Panel.
4. During scene design, the Foreground Image can be set to `Active`, however, when the scene is completed and ready to be taken on air, the Foreground image *must* be set to `Inactive` to allow only the key information to be applied.
   This procedure works with both Fast Texture Mode and Normal Texture Mode, and requires **Arena/Libero** to be set as **Render Sequence** in the Global Settings Panel. Viz Arena can be used to deactivate a currently activate image mask, transfer new image masks, or activate one of the previously transferred image mask using the corresponding mask ID.

6.3 Scene Editor

The Scene Editor has a set of useful functions that lets the designer see how the graphics will look when keyed, seen from a different angle in Viz Virtual Studio (virtual sets), positioned on-screen, performance of the scene and more.

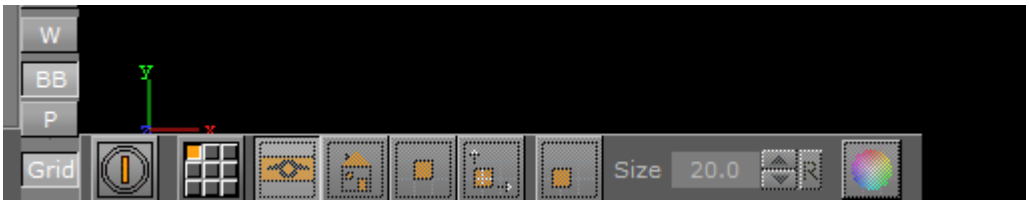
On the left side of the Scene Editor are the Scene Editor Buttons. These buttons make it possible to show additional information in the Scene Editor, as well as providing more options while working.

This section contains information on the following topics:

- Scene Editor Buttons
- Layer Manager
- Snapshot
- Performance Bar
- Grid Tool-bar

6.3.1 Grid Tool-bar

The Grid tool-bar can be shown in the lower part of the Scene Editor. Grids are used to align containers. If the **Grid Editor** is closed while a grid is still active, the Grid button will turn orange. This is especially useful when the grid is active, but hidden in the Scene Editor using Show/Hide Grid. All grid settings are saved with the scene; thus any such settings will be reloaded the next time the scene is loaded. The different grid settings available are:



- 

Plane Type: Shows which plane the grid should lie in:

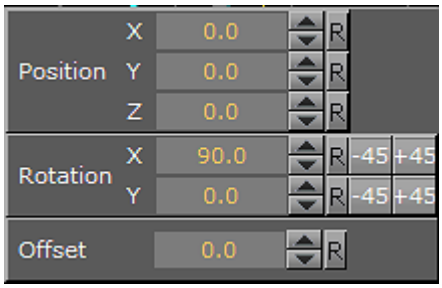
- **Align X/Y**: Aligns the grid to the X/Y-plane.
- **Align X/Z**: Aligns the grid to the X/Z-plane.
- **Align Y/Z**: Aligns the grid to the Y/Z-plane.
- **Free Grid**: Makes it possible to use a free grid, aligned for example to an object.
- **Off:** Shows that the grid is set to **OFF.**

-  **Select Grid**: Sets a predefined grid. 9 grids can be selected and manually configured as presets for the scene.
-  **Show/Hide Grid**: Shows or hides the grid in the Scene Editor.
-  **Home**: Sets the grid back to its default position.
-  **Snap to Grid**: If this button is enabled and the position of a container is modified by dragging it around in the Scene Editor, the position of the object will snap to the grid based on the **Axis Center** defined for the object's container.
-  **Align Object on Plane**: To align the axes of a container to the corresponding axes of the grid, first enable this option and next left-click the container you want to align and drag it around. As long as this option is switched on, the container will move only along the plane if you drag it around in the Scene Editor.
-  **Move Object to Plane**: If you select a container and next click this icon, the container will be moved so its center is on the plane.
- 

Size: Defines the raster size of the grid.
-  **Change Color of Grid**: Changes the color of the grid.

The three buttons, detailed below, are only available if the grid type is set to Free Grid:

-  **Picking On/Off**: Aligns the grid to the object. Enable picking and click the object to align the grid. The grid will be aligned to the vertex you click on.
-  **Move Grid to Object**: Moves the grid so the center of the selected container will be on it.
-  **Free Plane Value Editor**: Switches the free plane value editor on and off. In this editor you can adjust the grid's plane as illustrated in the image of the Free Plane Editor options below:



Adjusting the **Position** or **Rotation** in the Free Plane Editor will change the grid's own Position (x, y, z) or Rotation (x, y). Changing the Offset will adjust the grid's location along the normal vector of the grid XY plane (Z position).

To Enable the Grid Toolbar

- Click the Scene Editor's **Grid** button.

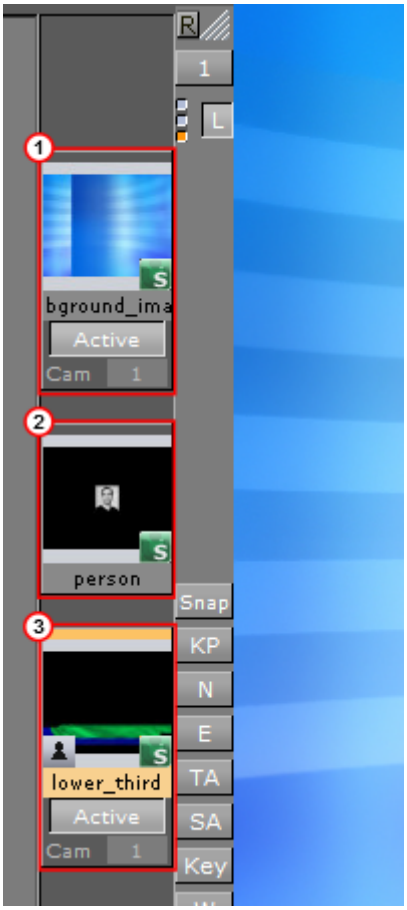
See Also

- Scene Editor Buttons
- Snapshot
- Performance Bar
- Layer Manager
- Working with the Scene Editor
- Time-line Editor

6.3.2 Layer Manager

All scenes are designed to run in the Main layer (2) by default, but can be played out in the Front (1) and Back (3) layer as well. The Layer Manager can be used to test combinations of Scenes in the three layers.

A Scene set in the Main Layer can be edited in the Scene Tree and the Scene Editor panel. Scenes set in the Front or Back Layer can be edited by a double-click on the relevant Scene.



Picking in the Scene Editor is done on the loaded layer. The Grid Tool-bar also works for the current layer.

This section contains the following topics and procedures:

- Self Layers
- Layer Manager Context Menu
- To Add Scenes to the Front or Back Layer
- To Edit a Scene Used as a Front or Back Layer

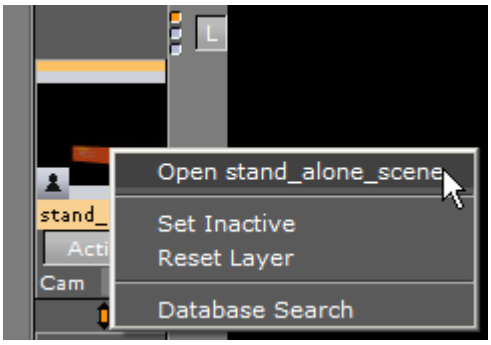
Self Layers

The same Scene can be set to the Front and Back layer, which creates what is known as self layer(s).

This allows the Scenes to be linked, and the adjustment of the camera settings for the Front layer. When saving a Scene, any self layer settings are preserved.

The three small indicators (4) show which layer is active in the Scene Tree, and are referred to as traffic lights.

Layer Manager Context Menu



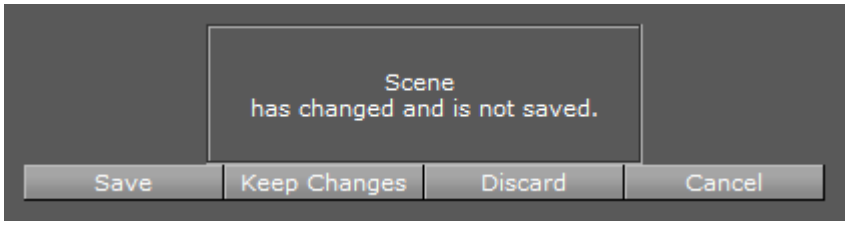
- **Open <scene>**: Opens the Scene for editing.
- **Set Inactive/Active**: Shows or hides the layer
- **Reset layer**: Removes the Scene from the layer (the same as drag and drop into the bin).
- **Database Search**: Opens the location of the Scene in the Server Panel.

To Add Scenes to the Front or Back Layer

- Drag and drop Scenes onto the layer placeholder.

To Edit a Scene Used as a Front or Back Layer

1. Double-click the Scene to be edited.
2. If another Scene in one of the three layers was modified, select an option from the message which opens:
   - **Save**: Saves any changes made in the last selected Scene.
   - **Keep Changes**: Keeps any changes made in the previously selected Scene, but does not save the changes.
   - **Discard**: Discards any changes made in the previously selected Scene and reloads the Scene when switching layers.
   - **Cancel**: Cancels the operation.

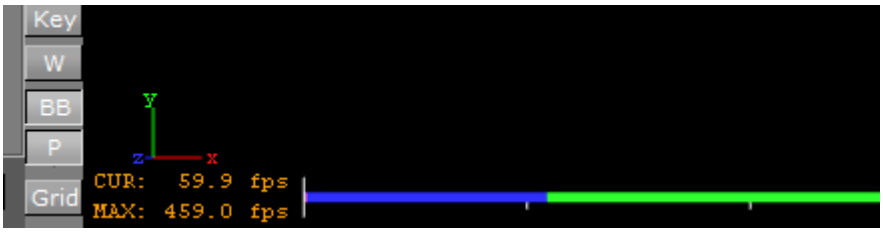


See Also

- Scene Editor Buttons
- Working with the Scene Editor
- Time-line Editor

6.3.3 Performance Bar

A Performance Bar can be shown at the bottom of the Scene Editor. The Performance Bar gives an indication of the performance (frames per second) the Scene is rendering in.



⚠ **Note:** A Performance Bar can also be shown in On Air Mode (see Viz Artist Performance).

This section also contains information on the following topics and procedures:

- Performance Bar Properties
- To View the Performance Bar
- To View all Performance Bar Indicators

Performance Bar Properties

- **VER (Vertices):** Shows the number of vectors in the scene.
- **TET (AllocTexSize):** Shows the total allocated size of Texture memory.
- **TEC (TexSize):** Shows the size of the current in use Texture memory.

> ⚠ **Note:** The actual drawing of Textures has an impact on the performance, more visible textures cause more render time. If texture memory is low it is possible that Textures are not visible in the Scene. The Texture memory usage is an indication for this.

- **ANI (Animation):** Shows how many microseconds all active directors and animation channels take. This indicator is linked to the yellow bar.
- **MAT (Matrix):** Transforms each container in the scene into world coordinate space. This indicator is linked to the cyan bar.
- **Z&C (Z-Sort):** Refers to Z-sort and Culling, and sorts all containers for correct transparency drawing and determines if containers are visible in the current camera view. This indicator is linked to the pink bar.
- **VID (Video):** Shows how many microseconds video input (live video Media Asset) and video output take. De-interlaced video inputs take longer time than progressive and interlaced. The only way to improve this value is to use a faster system. This indicator is linked to the red bar.
- **REN (Rendering):** Shows how many microseconds it takes to render all objects on the screen. A faster graphics card will improve this value. This indicator is linked to the blue bar.
- **PLU (plug-in):** Indicates how much time in microseconds all active plug-ins spend in each render cycle. This indicator is linked to the orange bar.
- **SCR (Script):** Shows the consumed time in microseconds from all active scripts. This indicator is linked to the dark green bar.
- **CUR (Current):** Shows how many frames per second the scene will render at in On Air mode. The number should be above 50 (PAL) or 60 (NTSC), according to the rate that has been specified in the Output Format section.
- **MAX (Maximum):** Shows how many frames per second the scene can render at without waiting for vertical retrace. The higher the maximum value, the more performance is left. If the maximum value is reduced to below 50 or 60, the scene is not rendering in real-time.
- **Blue bar:** Shows the number of frames used
- **Green bar:** Shows the number of frames available
  The Performance Bar refresh rate can be modified in **User Interface** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

To View the Performance Bar

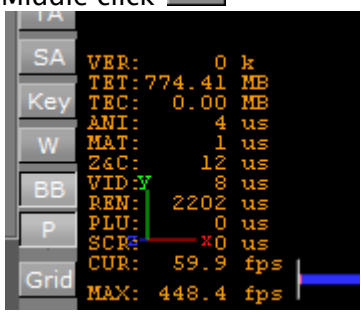
- Click 

in the Scene Editor.

> ⚠ **Note:** This shows the Current (CUR) and Maximum (MAX) parameters.

To View all Performance Bar Indicators

- Middle-click P



See Also

- Scene Editor Buttons
- Snapshot
- Grid Tool-bar
- Viz Artist Performance

6.3.4 Scene Editor Buttons

Click the Scene Editor buttons once to enable the features, click them again to disable.

-  **Reset**: Sets the Viz Artist layout to default size.
-  **Camera Selection**: See Camera Selection.
-  **Layer Manager**: Shows or hides the Layer Manager (see Layer Manager).
-  **Snap**: Takes a snapshot of the current content in the Scene Editor (see Snapshot).
-  **Post-Processing**: Enables rendering of advanced post-processing effects, such as Depth of Field or Lens Distortion, in the scene editor window. Under normal operation, these effects are only rendered in On Air Mode.
-  **Key Preview**: Activates the preview of the key signal.
-  **Normal Vertices**: Shows the normal vertices. For example, it is practical to use this feature when the lighting is not reflected in the required manner. Enabling this feature shows the vertices as purple handles. These handles makes it easier to predict how the light will be reflected.
-  **Events**: Enables the handling of interactive script/plug-in events. Interactive scripts and plug-ins are those related to mouse or keyboard actions.
-  **Title Area**: Shows the title area as the outline of a light blue rectangle in the Scene Editor.

  > ⚠ **Tip:** The size of the title area can be defined under the Rendering Panel scene settings.

-  **Safe Area**: Shows the safe area as the outline of a green rectangle in the Scene Editor

  > ⚠ **Tip:** The size of the safe area can be defined under the Rendering Panel scene settings.

-  **Key**: Shows what the key output will look like.
-  **Wire Frame Mode**: Renders the scene in wire frame mode.
-  **Bounding Box**: Enables the bounding box visualization for the selected Container.
-  **Performance Bar**: Shows the Performance Bar.
-  **Grid**: Opens the Grid Tool-bar.

See Also

- Layer Manager
- Snapshot
- Performance Bar
- Grid Tool-bar

6.3.5 Snapshot

A snapshot can be taken of the current content in the Scene Editor.

When clicking the Snap button, the Server Panel will automatically be selected from the main menu, and the database content will show. The image will be placed in the folder/project that was last visited.

This section also contains information on the following procedure:

- To Take a Snapshot

To Take a Snapshot

1. Snapshot RGB or RGBA:
   - **RGB**: Click the **Snap** button
   - **RGBA**: Press **Shift** (or **Alt**) and click the **Snap** button.
2. In the Snapshot dialog box that shows, enter the **name**, **size**, and **format** for the image.
3. Click **Ok** to save the image to the database.

> ⚠ **Note:** An RGBA image will use the key information as alpha.

> ⚠ **Note:** To save the image in another folder/project: first select the destination you want to use, then click the Snap button again.

Taking snapshot from a running video is allowed.

The procedure above is interactive. If required, you can alternatively instruct Viz to take a snapshot using the commands below. The examples below, calling *renderer* directly, assumes the Engine is in On-Air mode:

```
RENDERER*VIDEO*SNAPSHOT CREATE file:///c:/images/snapshot.png
RENDERER*VIDEO*SNAPSHOT CREATE vizgh:///my/images/snapshot
RENDERER*VIDEO*SNAPSHOT CREATE vizgh:///{159FC7E7-B724-4CC3-B474B7EE46CDA043}/snapshot
```

- Example 1 writes a PNG image to disk. The PNG file format is particularly useful when working with RGBA images or 4-channel data in general.
- Example 2 saves the snapshot to the graphics-hub (GH) using a path.
- Example 3 saves the snapshot to the GH using a GH Id (unique identifier).

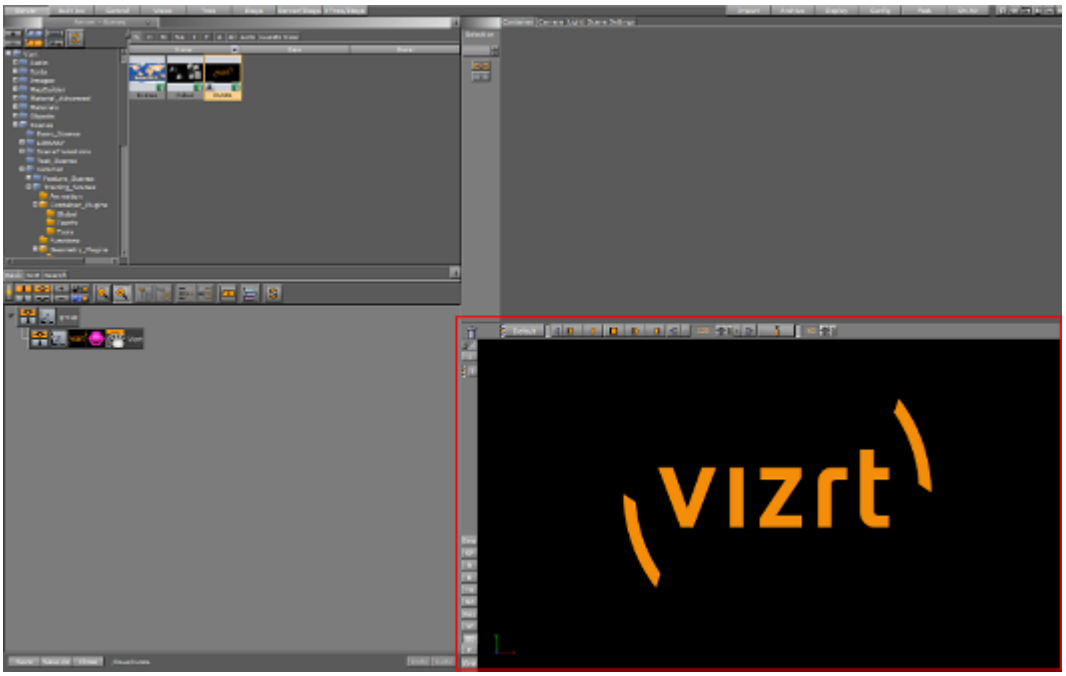
See Also

- Scene Editor Buttons
- Performance Bar

- Grid Tool-bar

6.4 Working With The Scene Editor

Some transformation parameters of a Container or Containers in a Scene can be edited in the Scene Editor. Furthermore, you can switch some properties of a container ON and OFF.



This section contains information of the following topics:

- Scene Editor Functions
- Scene Editor Context Menu

6.4.1 Scene Editor Functions

This section contains information of the following topics:

- To Select/Deselect an Object
- To Copy an Object
- To Move an Object
- To Move an Object on its Z Axis
- Multiple Objects Selected
- To Select Multiple Objects
- To Deselect Multiple Objects
- To Add Properties to Multi Selected Objects
- To Move Multi Selected Objects
- To Move Multi Selected Objects on Their Z Axis

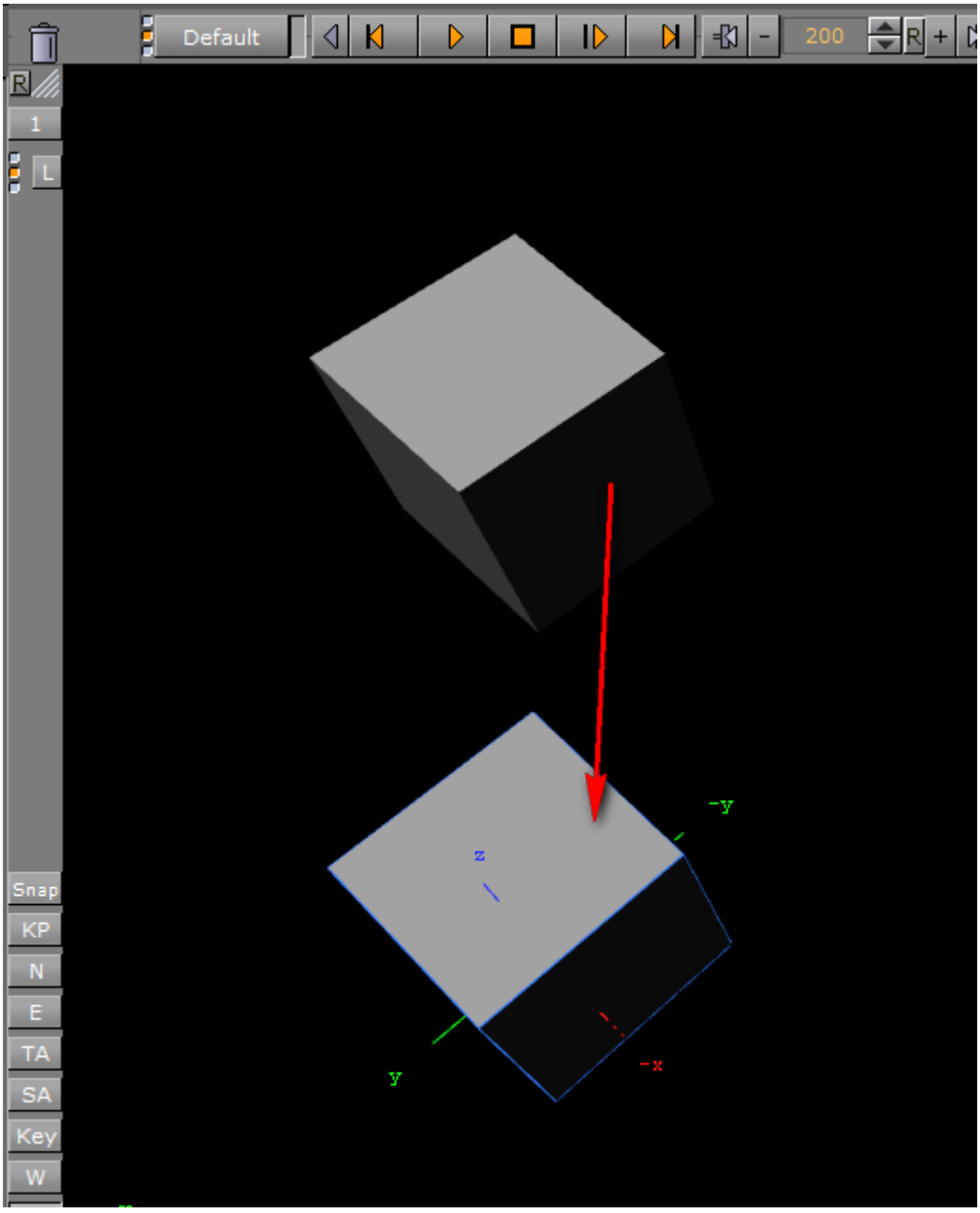
To Select/Deselect an Object

1. Left-click on an object, to select that object.

   > ⚠ **Note:** This is done on a per pixel basis.

   - A bounding box shows which object has been selected (if the Bounding Box option is enabled, see Scene Editor Buttons).
   - The selected object's Container is highlighted in the Scene Tree
   - Click on the Transformation editor in the Properties panel to show to the transformation properties.
2. Press **Space** or click in an empty space to deselect an object.

To Copy an Object



1. Hold <Ctrl> down and click and hold the **left mouse button**
2. Drag and drop the object to the new location.

To Move an Object

1. Left-click on an object, to select that object.
2. Keep the left button pressed and move the object within the Scene Editor.
3. Release the left button

To Move an Object on its Z Axis

1. Left-click on an object, to select that object.
2. Keep the left button pressed and press the right button. Hold both buttons down.
3. Move the mouse to the left or right to move selected object in its Z axis.

Multiple Objects Selected

When multiple objects are selected in the Scene Editor, the Transformation editor, in the Properties Panel, will show the editable properties for all selected objects.

All selected objects will be included in one bounding box (if the Bounding Box option is enabled, see Scene Editor Buttons). The selected object's Containers are also highlighted in the Scene Tree.

> ⚠ **Note:** When more than one object is selected the Group coordinate system will show and a bounding box will cover all selected objects. When Local and World is selected a bounding box for each selected object will show.

To Select Multiple Objects

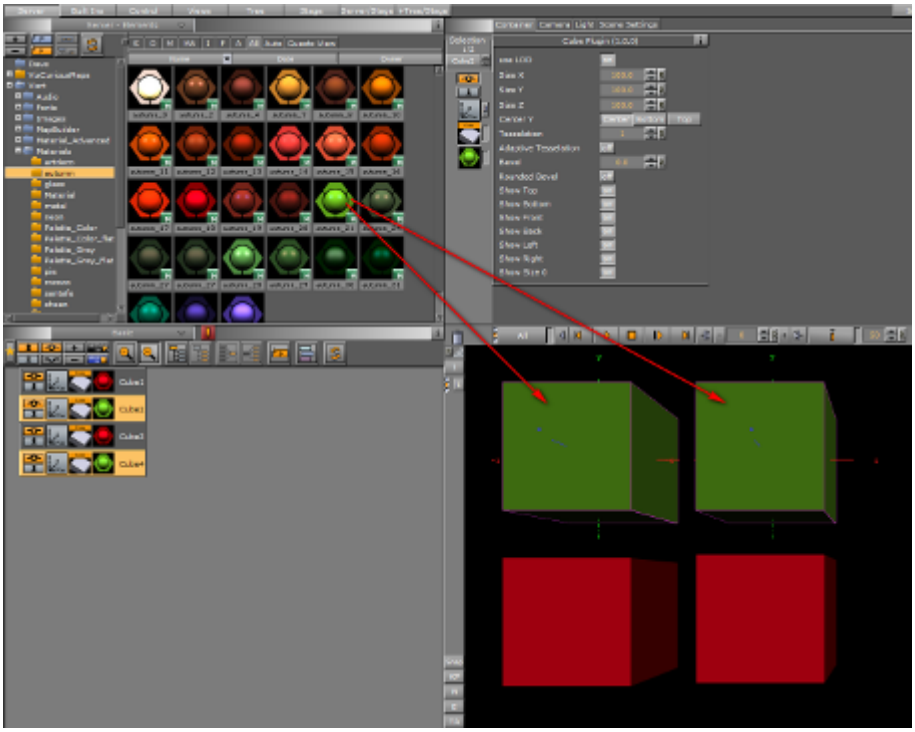
1. Hold the <Ctrl> key and click on each object to select, as required, or
2. Click in an empty space in the Scene Editor and drag around the required objects to select.

To Deselect Multiple Objects

1. Press the **SPACE** bar, or
2. Click in an empty space on the Scene Editor.

> ⚠ **Note:** Toggle the SPACE bar to select or deselect multiple objects.

To Add Properties to Multi Selected Objects



1. Multi select the required objects.
2. From the Server Panel select the required item.
3. Drag the item to one of the selected objects.
4. All selected objects will take on the property.

To Move Multi Selected Objects

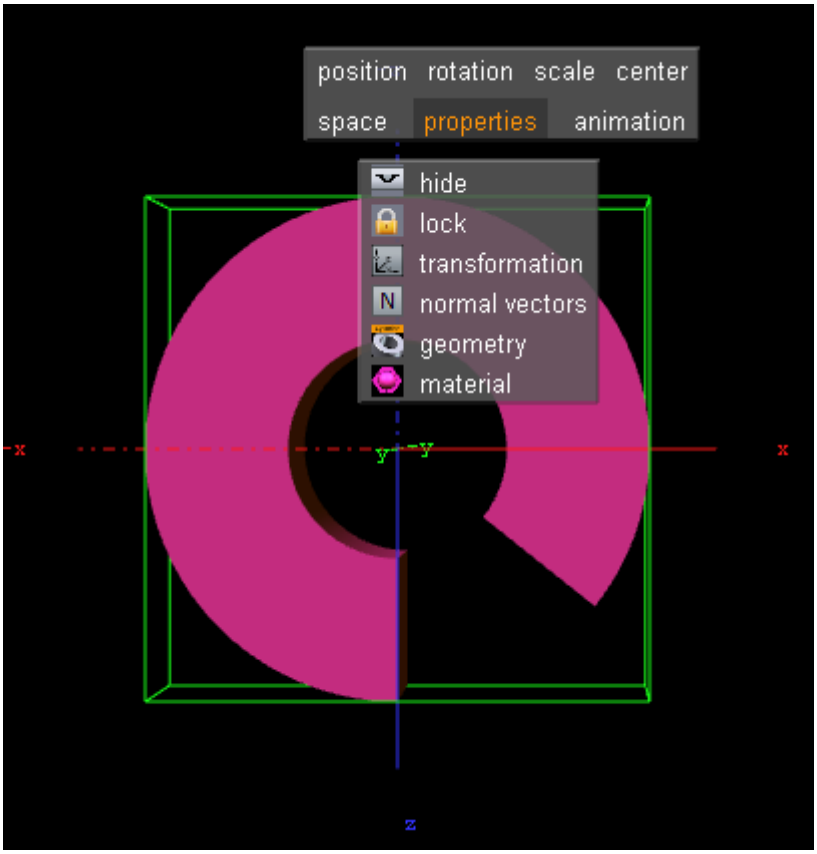
1. Select the required objects.
2. Left-click one of the objects to be moved and move the multi selected objects within the Scene Editor.
3. Release the left button.

To Move Multi Selected Objects on Their Z Axis

1. Select the required objects.
2. Left-click on one of the multi selected objects.
3. Keep the left button pressed and press the right button. Hold both buttons down.
4. Move the mouse to the left or right to move selected objects in its Z axis.

6.4.2 Scene Editor Context Menu

In the Scene Editor right-click and hold to open a context menu:



⚠ **Note:** The context menu selection is available on a single Container or multiple Containers.

- **Position**: Switches into the position mode.
- **Rotation:** Switches into the rotation mode.
- **Scale**: Switches into scale mode.
- **Center**: Switches into center editing mode.
- **Space**: Switches between coordinate systems:
  - **Parent/Group:** Refers to changes that can be made to the Container or Containers in its Parent or Group coordinate system.
  - **Local:** Refers to changes that can be made to the container in its local coordinate system.
  - **World:** Refers to changes that can be made to containers in their world coordinate system.

    ⚠ **Note:** Multi selected objects: When more than one object is selected Group will show and a bounding box will cover all selected objects. When Local and World is selected a bounding box for each selected object will show.

- **Properties** (only applicable if a Container is selected):
  - **Hide/Show:** Hides or shows the Container in the Scene Editor.

- **Lock/Unlock**: Locks or unlocks the container. When locked the Container cannot be modified in the Scene Editor.
- **Transformation:** Opens the transformation editor for the Container.
- **Normal vectors**: Toggles the display of the normal vectors at the bottom of the properties you will find a list of all properties applied to the container with the possibility to switch them **on** and **off**.
- **Material**: Toggles diffuse color on (enlighted) or off (colored).
- Any other properties of the container can set to on and off (except materials).

- **Animation**: Shows the animation path.

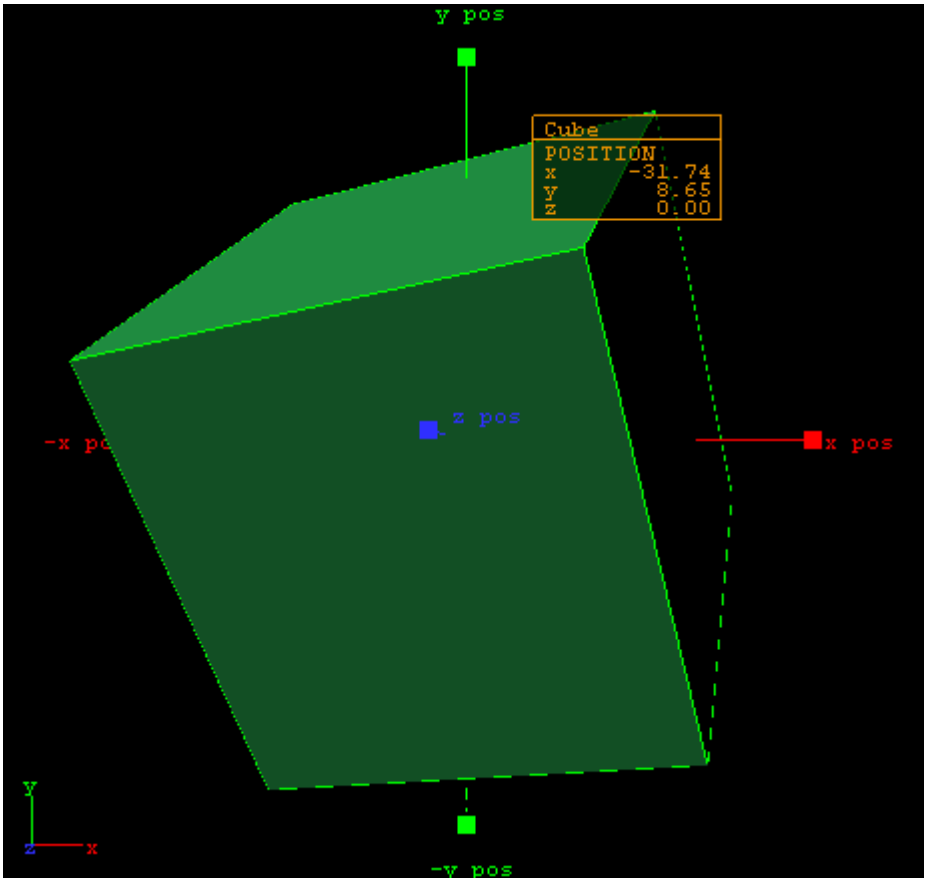
This section contains information of the following topics:

- To Select an Option from the Context Menu
- To Change an Object's Position
- To Rotate an Object
- To Scale an Object
- To Move the Axis Center of an Object

To Select an Option from the Context Menu

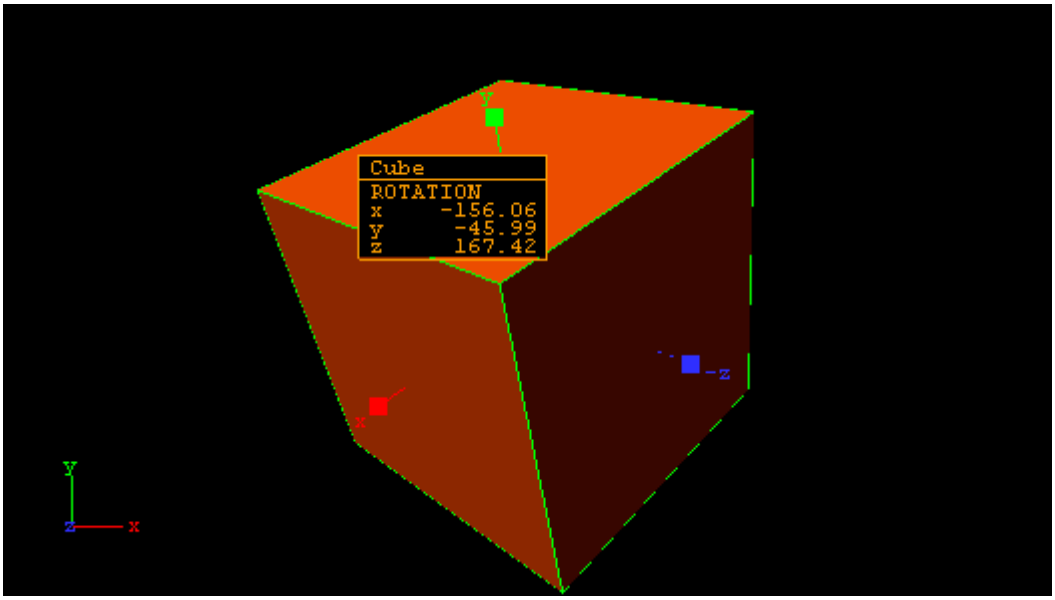
1. Right-click and hold.
2. Move the cursor to the required option.
3. Release to select.

To Change an Object's Position



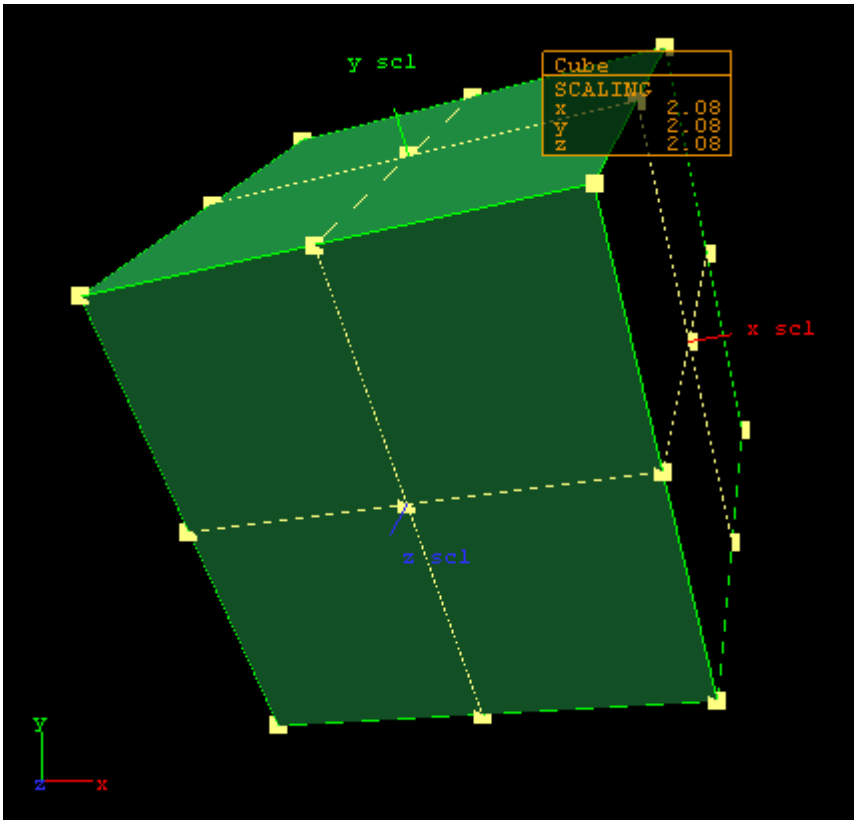
1. Select an object.
2. Right-click the object and select **position** from the context menu.
3. Grab and drag a handle.
   The container will follow the movement along the axes shown in the selected camera view. During this operation a status window is shown, giving you information about the position.

To Rotate an Object



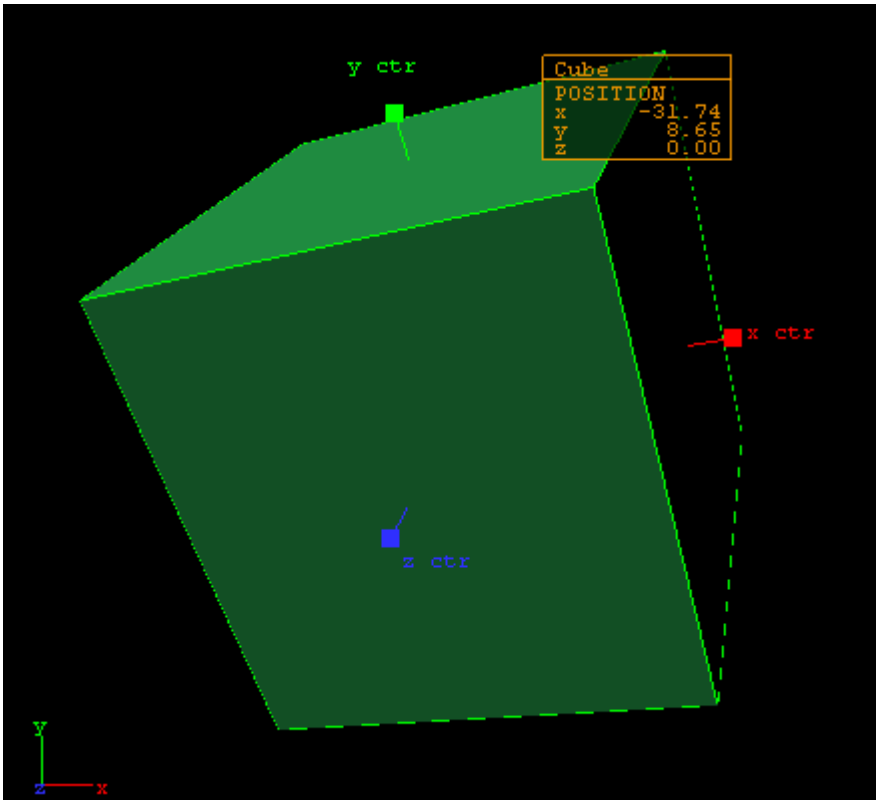
1. Right-click to open the context menu.
2. Select **rotation**.
3. Drag the handles side ways to rotate on all the axes.

To Scale an Object



1. Right-click to open the context menu
2. Select **scale**.
3. Drag the handles to scale the object.

To Move the Axis Center of an Object



1. Right-click to open the context menu
2. Select **center**.
   This will show the axis of the selected container with handles on their ends.
3. Grab one of the handles and change the center position by dragging them around the Scene Editor.

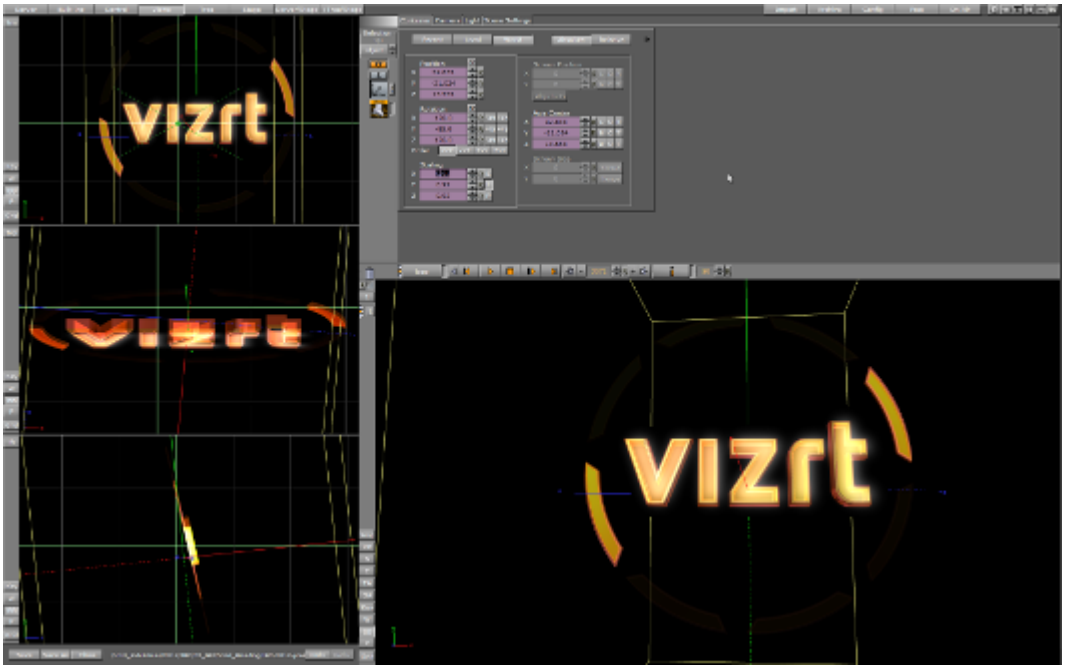
See Also

- Layer Manager
- Time-line Editor

6.5 Scene Editor Views

To get a better overview of an object, or objects, in a Scene, and to visualize their position in relation to lights and cameras, open the Scene Editor Views function.

To open the views panes, click on the **Views** button in the main menu.



In principle, each of the three views has the same functionality, and lets you do the same operations on objects as is possible in the Scene Editor.

As well as the Scene Editor, the three side views can be set to view through one of the 16 editable cameras or through one of the six predefined orthogonal views (see Camera Selection). When set to one of the predefined orthogonal views this allows the position and direction of lights and cameras, to be modified.

If the Scene Editor is set to show one of the predefined orthogonal side views, there are some special shortcuts that may be used to pan and zoom:

- **Pan:** Left-click, hold and drag in the background (not the objects).
- **Zoom:** Hold down **Z** (zoom in) or **X** (zoom out) and draw a rectangle with the left button pressed, or position the cursor over a panel and press the **Z** or **X** keys multiple times to zoom in or out in a stationary position.
  In the six predefined orthogonal views, the currently selected camera, and in addition all cameras that have been given the status **Vis** or **inf**, will show as symbols.

Click on a camera in the side views and that cameras focus is set in the Scene Editor, and handles to edit the position and direction will show (drag the handles to change position and direction).

The same functionality is available for lights (see Lights). The light currently selected and those lights which have their **Vis** status set to **On**, will be visible, as symbols. Their position and direction can be edited with the cursor.

See Also

- Light Editor
- Cameras

7 Media Assets

Use Media Assets to add input sources to scenes as video clips, live video, video stream, graphics channels, image channels or as super channels.

This section describes the workflow and each type of Media Asset, and details Media Asset requirements and considerations for video playback, which includes video clips, live video feeds and live feeds through video streaming.

Also detailed are the required and supported video codecs.

The Media Assets are located in the Built Ins panel (see Main Menu), and are organized in folders corresponding to the Media Asset Channel Types. You can view all available media assets by selecting the **ALL** folder.

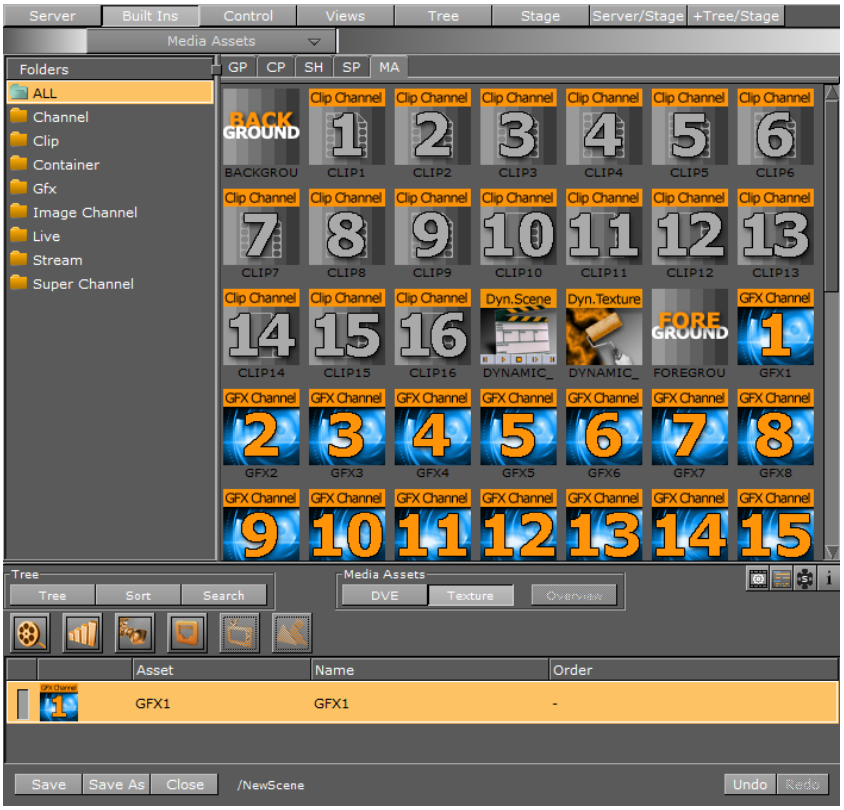
This section contains the following topics:

- Media Asset Manager
- Media Asset Workflow
- Media Asset Channel Types
- Playback of Media Assets
- Video Clips
- Keying Mode
- Seamless Input Channel Switcher
- Supported Codecs
- Advanced Issues with Video Codecs

7.1 Media Asset Manager

The Media Asset Manager is used to add and manipulate Media Assets. Media Assets can easily be added to a scene as either DVE or Texture (1) asset types, depending on the scene's requirements. The asset type can also be changed through the Manager interface, from DVE to Texture or vice versa. The Media Assets panel also provides an overview of the different media assets in the scene. The media assets can be renamed within the scene according to your preferred naming convention, ideal for consistent identification.

The Media Manager lets you set the rendering priority of Media Assets, by the way they are ordered. To change an asset's priority, drag it to a higher or lower position. The asset with the highest **Order** value is rendered on top, with declining **Order** values below it.



Tip: Hover the mouse pointer over any **Clip**, **Live** or **Stream** channel asset in the Media Asset Manager to display a tool-tip with the resolution for that asset. If the channel is set to `inactive` in the configuration, the tool-top displays **INACTIVE**.

This section contains the following topics:

- Media Asset Overview
- Media Assets Added as DVE
- Media Assets Added as Texture
- To Create a Name for a Media Asset

7.1.1 Media Asset Overview

Use the Media Asset Overview to manage and access all the different types of media assets in use within your current scene. The Media Asset Overview shows if a media asset is in use in other scenes. When multiple scenes use the same media asset, the Media Asset Overview provides access to these scenes from within your current scene. This way, you can change properties of a media asset or scene that you load within the scene you are working on, without closing the current scene. The Media Asset Overview is not for container or hierarchy manipulation, it presents

the media assets in your scene, and lets you to load these scenes. Switch back to the scene tree/ hierarchy view by clicking the **Tree** button.

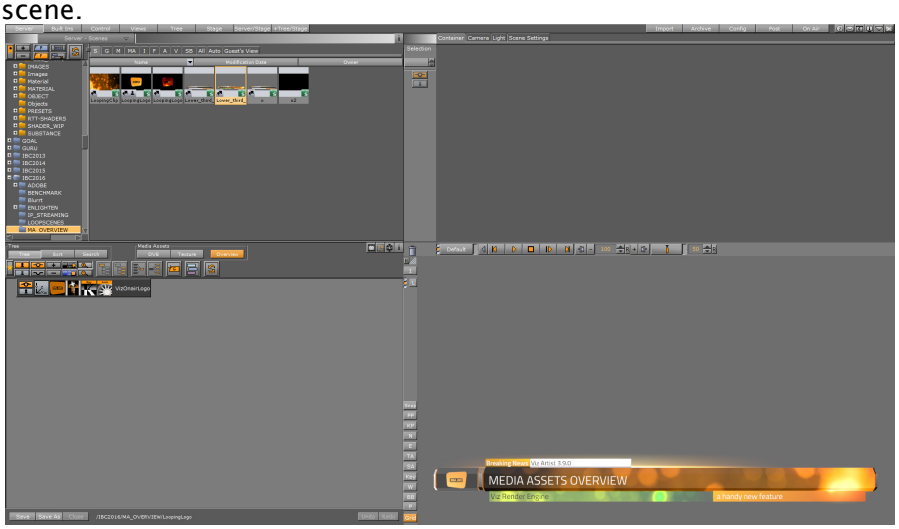
> ⚠ **Tip:** The Media Asset Overview window does not highlight containers when they are selected. Double click on a child node to load the respective scene.

In the following example, a main scene containing a lower third uses multiple media assets. Within this main scene, we use media assets to load two other scenes: One with a looping clip, the other with a looping logo.

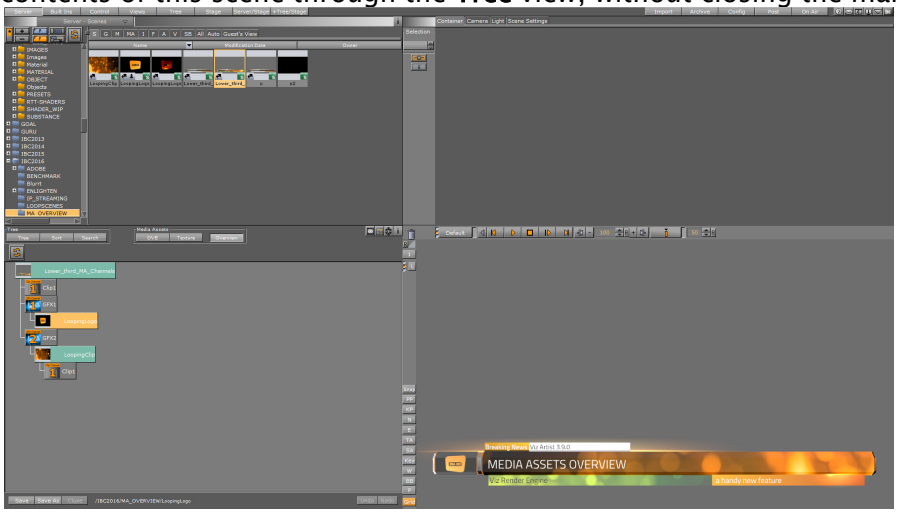
1. Click **Overview** to open the Media Asset Overview. Notice how the scene tree display changes, and provides an overview of which scenes are loaded. Here, the scene is using one clip channel and two GFX Channels. The child nodes of the GFX Channels shows the scene loaded in the respective GFX Channel:


2. Double click the child node of one of the GFX Channels to open the scene in the GFX Channel. A dialog lets you save the changes or close the scene without saving. The **Overview** button turns orange to indicate that you have loaded a different scene for manipulation.
3. This updates the **Tree** view of the scene, which now shows the scene tree of the scene inside the GFX Channel. However, the preview window still contains the main scene. This provides a convenient way to edit the content of different media assets, without closing the main

scene.



4. Going back to the Media Assets Overview, notice that the child node of the GFX Channel is orange. This indicates that this is the scene in your **Tree** view. Now, you can edit the contents of this scene through the **Tree** view, without closing the main scene.

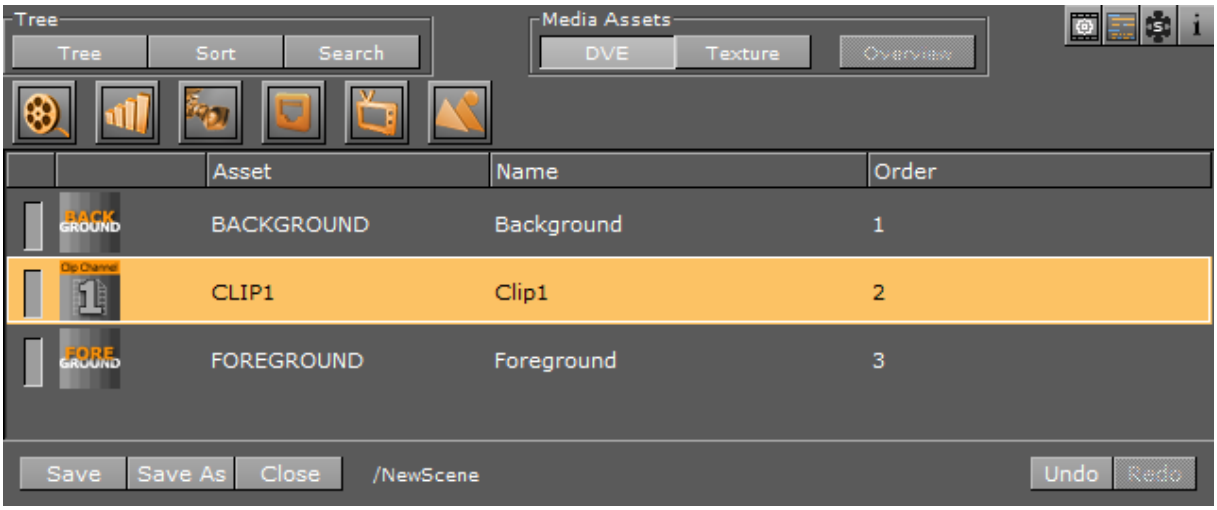


5. Return to the main scene again by double clicking the root container (here: **Lower_third_MA_Channels**) at the top of the Media Assets Overview. This opens the main

scene in the **Tree** view.

7.1.2 Media Assets Added as DVE

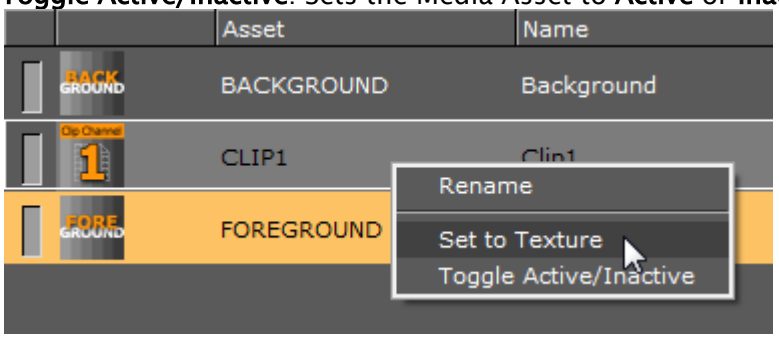
For detailed information see Media Asset Workflow.



Right-click the Media Asset to open its context menu:

- **Rename**: Creates a name for each Media Asset to make it easier to identify Media Assets when a Scene contains multiple Media Assets.
- **Set to Texture**: Sets the Media Asset to Texture.

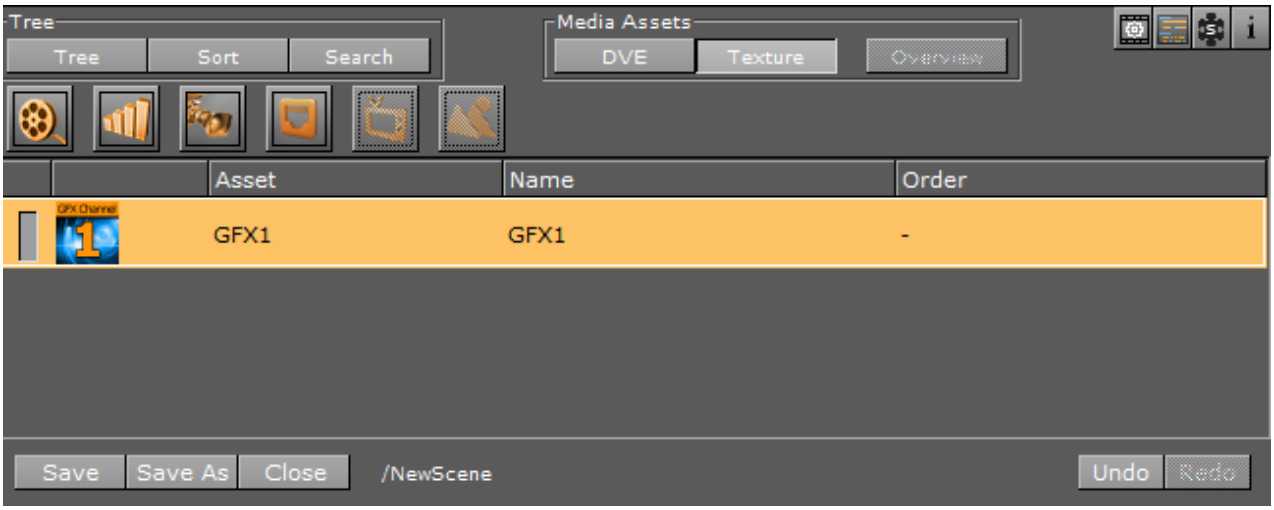
- **Toggle Active/Inactive**: Sets the Media Asset to **Active** or **Inactive**:



> **Tip:** Media Assets can also be multi selected and set to Texture or made Active / Inactive.

7.1.3 Media Assets Added as Texture

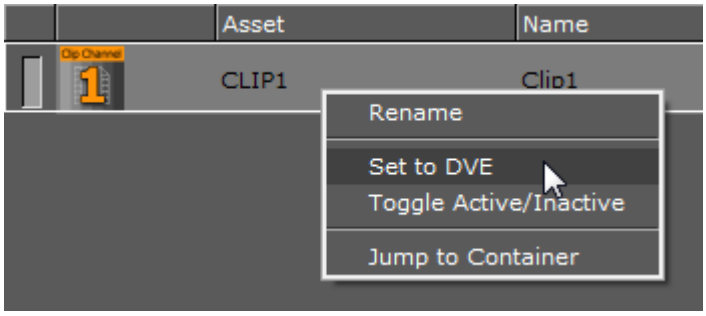
For detailed information see Media Asset Workflow.



Right-click the Media Asset to open its context menu:

- **Rename**: Creates a name for each Media Asset to make it easier to identify Media Assets when a Scene contains multiple Media Assets.
- **Set to DVE**: Sets the Media Asset to DVE.
- **Toggle Active/Inactive**: Sets the Media Asset to Active or Inactive.

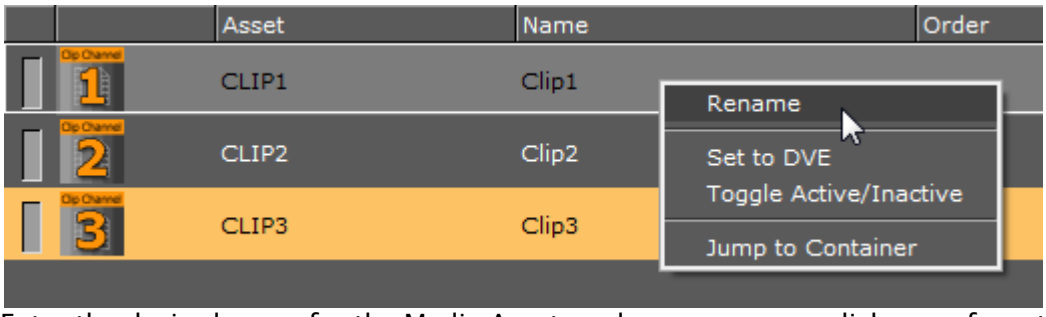
- **Jump to Container**: Switches to **Tree View** and selects the container holding the asset if the Media Asset has been added to a container.



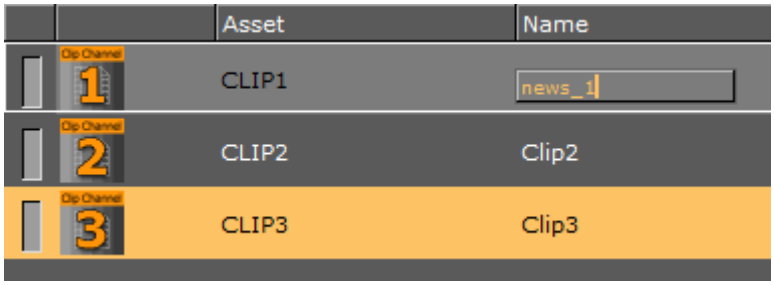
> ⚠ **Tip:** Media Assets can also be multi selected and set to DVE or made Active/Inactive.

7.1.4 To Create a Name for a Media Asset

1. In the Media Asset Manager, either select the Media Asset and press `F2`, or right-click the asset and select **Rename**.

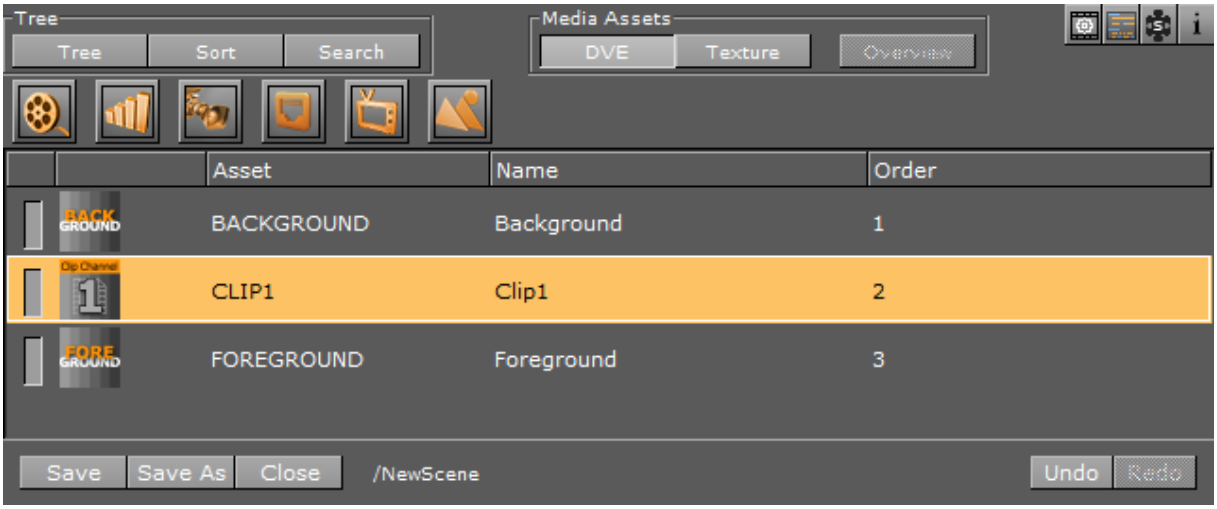


2. Enter the desired name for the Media Asset, and press `Enter` or click away from the asset to store the new name.



7.2 Media Asset Workflow

Media Assets can be added to a Scene as DVE or Texture through the Media Assets manager.



This section contains the following topics:

- Media Assets as DVE
- Media Assets as Texture
- Media Assets as a Background or Foreground

7.2.1 Media Assets as DVE

Important: Hardware surfaces in Viz Engine, such as a Live Video Media Asset or Clip Channel Media Asset, are rendered with the video hardware. Software based DVEs, such as Image Channels and GFX Channels, are rendered with the Graphics card, and then composited on the video hardware as one surface. The software renderer ensures the correct ordering within this surface, but as there is only one final surface, multiple hardware-based surfaces and software-based surfaces cannot be blended interchangeably.

As DVEs are composited on the video hardware, the graphics can be placed either in the front or in the back of the DVE layer. The main layer in Viz Artist defines this behavior. If the same scene is set into the background layer, this means the DVE effect is always drawn in the back of graphics.

To add a Media Asset as DVE it must be placed into the Media Asset Manager in the Scene Tree panel.

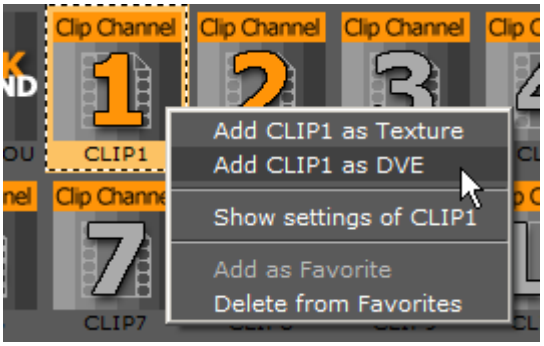
Tip: If a Scene is dragged to the Media Asset Manager, a GFX Channel Media Asset is automatically added as DVE.

This section contains the following topics:

- To Add a Media Asset as DVE
- To Set a DVE Media Asset to Texture
- Order DVE Media Assets

To Add a Media Asset as DVE

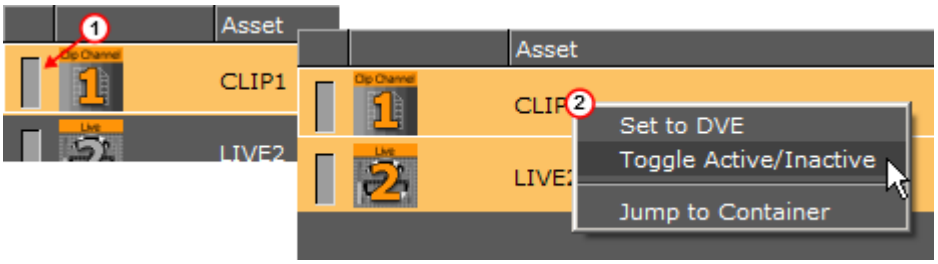
- Right click on a **Media Asset**, and select **Add <Media Asset> as DVE**.
  - The **DVE** button in the Media Asset manager is automatically selected
  - The selected Media Asset is added to the DVE view



Or:

- With DVE view selected, drag the Media Asset to the DVE panel
  - The selected Media Asset is added to the DVE view

> ⚠ **Tip:** A Media Asset can be set to active or inactive. Toggle the active/inactive button (1) or use the Media Asset context menu. If Media Assets are multi selected, use the context menu to set active or inactive (2).



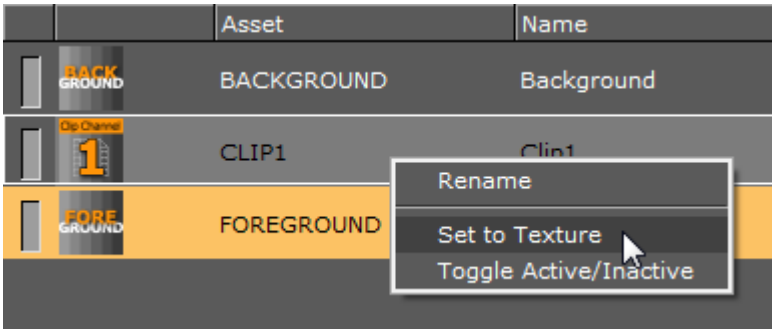
> ⚠ **Tip:** Click on the **Media Asset**, to open its properties panel.

> ⚠ **Tip:** Click the  icon (top right of the Scene Tree panel) to view Media Asset properties at any time.

To Set a DVE Media Asset to Texture

At any time, right click on the Media Asset to set it to Texture. The selected Media Asset will then be set as a texture and will show the in the Media Asset Manager Texture panel.

⚠ **Tip:** Media Assets can also be multi selected and set to Texture.



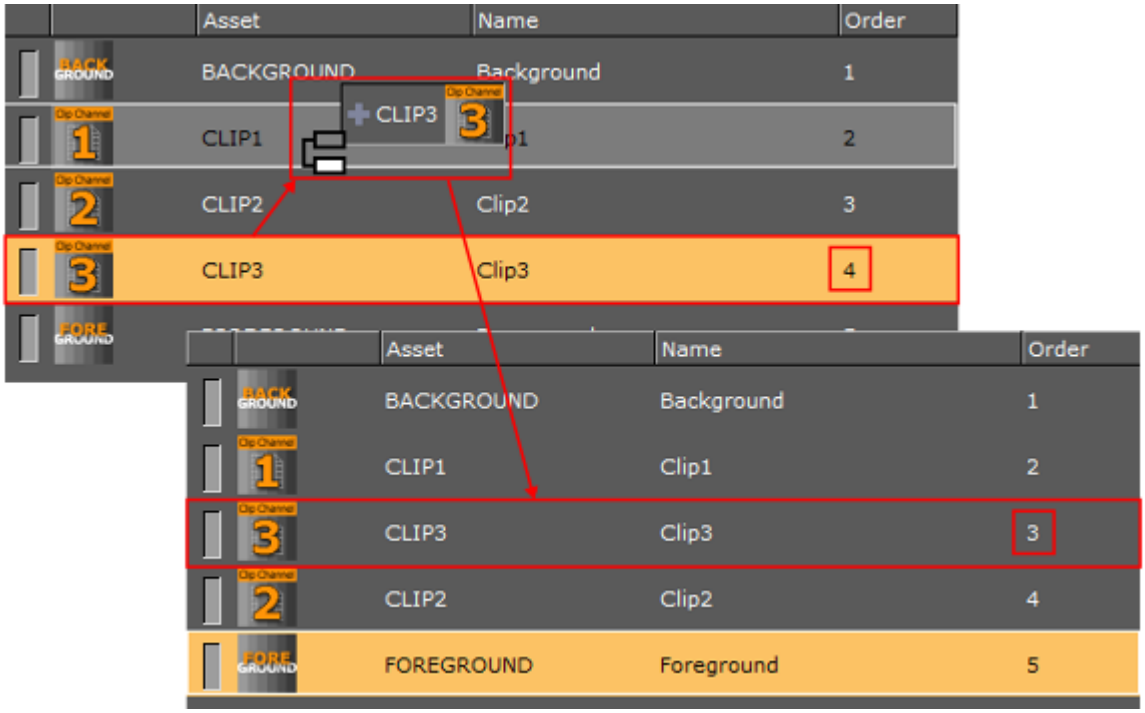
Order DVE Media Assets

Each Media Asset, added as DVE, can be sorted in a render order, as required.

⚠ **Note:** If the **Video Input Layer Targets and Priority** settings are made **Active** in **Scene Default Values**, each active Media Asset will be added to either the DVE or Texture section of the Media Asset Manager. They will be set in order of priority as set in **Scene Default Values** (see the **Configuring Viz** section of the Viz Engine Administrator Guide)

The Background and Foreground Media Assets are, by default, added as DVE layers. The Background Media Asset is always the first Media Asset to be rendered. The Foreground Media Asset can be ordered the same as other Media Assets.

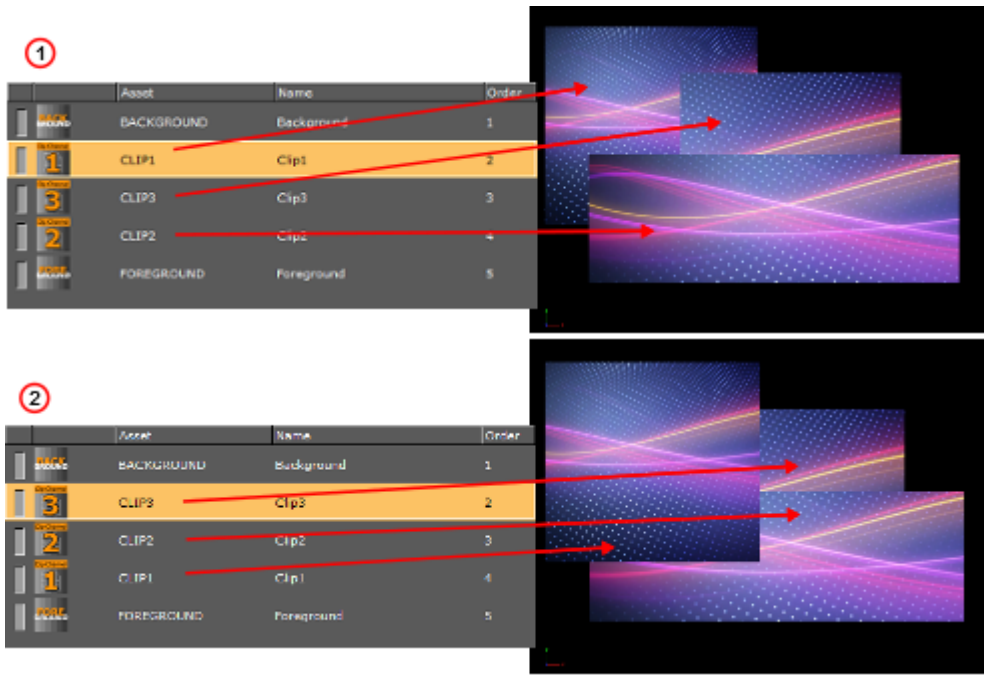
To sort a render order, drag a Media Asset to a new position.



The order of each DVE Media Asset determine the order each Media Asset is rendered. The first Media Asset in the list is rendered first, the second is rendered next, and so on. See the image below:

Example 1: The Clip order is 1, 2 and 3. So Clip 1 is rendered first and Clip 3 is rendered last

Example 2: The Clip order is 3, 2 and 1. So Clip 3 is rendered first and Clip 1 is rendered last.



7.2.2 Media Assets as Texture

To add a Media Asset as a Texture, it must be placed into the Media Asset Manager in the Scene Tree.

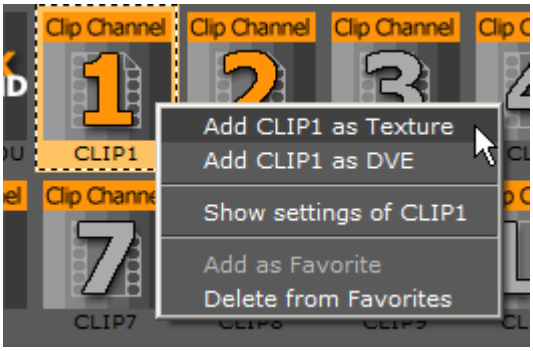
> ⚠ **Tip:** If a Scene is dragged to the Media Asset Manager a GFX Channel Media Asset is automatically added as a Texture.

This section contains the following topics:

- To Add a Media Asset as Texture
- To Set a Texture Media to DVE
- To View the Media Asset Container
- Order Texture Media Assets

To Add a Media Asset as Texture

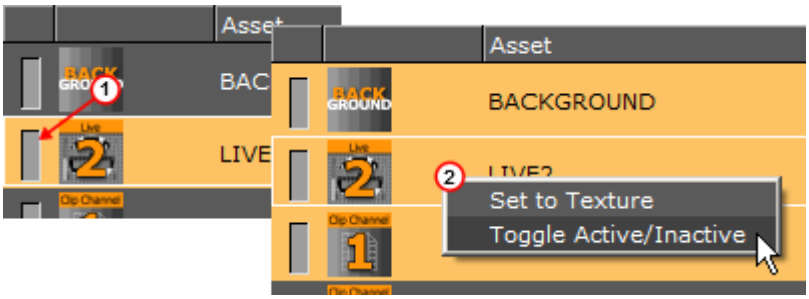
- Right click on a **Media Asset**, and select **Add <Media Asset> as Texture**:
  - The **Texture** button in the Media Asset manager is automatically selected
  - The selected Media Asset is added to the Scene Tree and a new Container is created (click **Tree** to view)



Or:

- Drag the Media Asset to a Container or the Scene Tree:
  - If a Media Asset is dragged to a Container in the Scene Tree, or it is dragged to the Scene Tree (which creates a Container), then the Media Asset is automatically created as a Media Asset Texture
    Or:
- Drag the Media Asset to the Media Asset Manager:
  - With **Texture** selected in the Media Asset Manager, drag the Media Asset to the Texture panel
  - If a Media Asset is dragged directly to the Media Asset Manager, a Container is not created automatically. To use the Media Asset in a Scene, make sure that the same Media Asset is added to a Container in the Scene Tree
    Or:
- Drag a Media Asset to the Background or Foreground drop zone in **Global Settings** (see the **Configuration Interface** section of the Viz Multichannel User Guide).
  - If a Media Asset is dragged to the Background or Foreground drop zones, it is applied in the Scene as a full screen texture

> ⚠ **Tip:** A Media Asset can be set to active or inactive. Toggle the active/inactive button (1) or use the Media Asset context menu. If Media Assets are multi selected, use the context menu to set active or inactive (2).



> ⚠ **Tip:** Click on a Media Asset in the Media Asset Manager to open its properties panel.

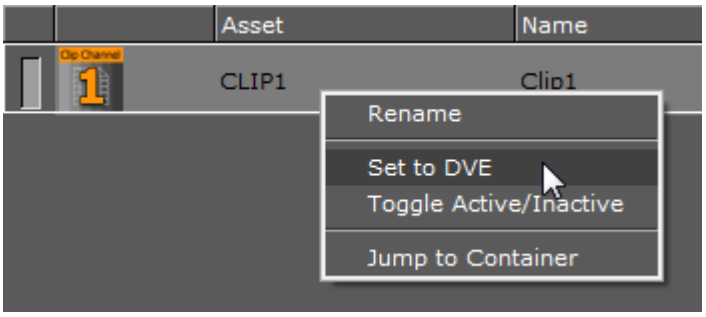
> ⚠ **Tip:** <Right-click> the Media Asset icon in a Container. Select **Show settings of <Media Asset>**.

> ⚠ **Tip:** Click the  icon (top right in the Scene Tree panel) to view Media Asset properties at any time.

To Set a Texture Media to DVE

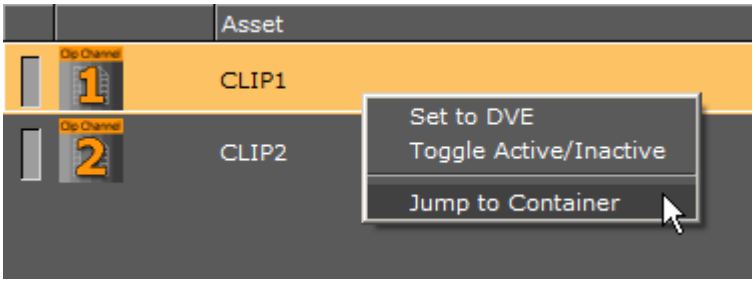
At any time, right click on the Media Asset to set it to DVE. The selected Media Asset will then be set to DVE and will show the Media Asset Manager DVE panel.

> ⚠ **Tip:** Media Assets can also be multi selected and set as DVE.



To View the Media Asset Container

At any time, right click on the Media Asset and select **Jump to Container** to view the **first** Container in the Scene Tree, which holds the selected Media Asset (if a Container was created for the Media Asset).



Order Texture Media Assets

Each Media Asset, added as Texture, will be ordered only by its addition to a Container in the Scene Tree.

If the **Video Input Layer Targets and Priority** settings are made active in the **Scene Default Values**, each active Media Asset will be added either the DVE or Texture view of the Scene Tree. Set in

order of priority as set in **Scene Default Values** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

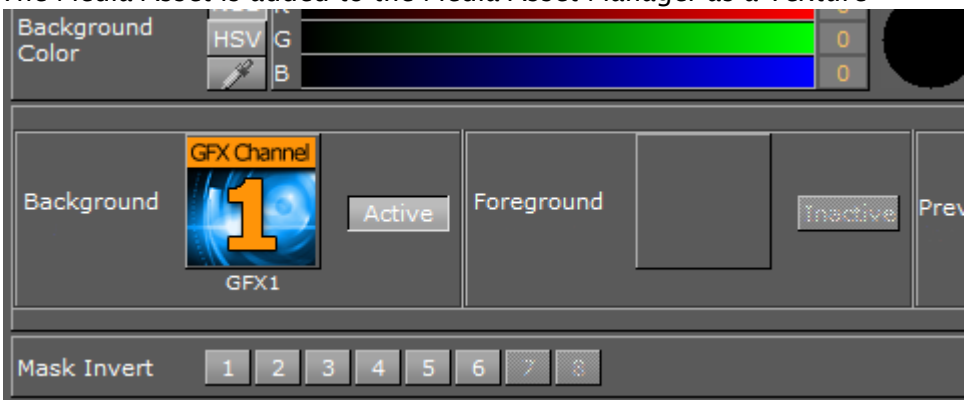
A Texture added to the Texture view is not added to a Container. To use a Media Asset as a Texture, it must be also added to a container as well.

7.2.3 Media Assets as a Background or Foreground

This section details how to add a Media Asset as a background or foreground in a Scene. A Scene can be shown through a Foreground Media Assets with alpha.

To Add a Media Asset as a Background or Foreground

1. Click *Scene Settings -> Global Settings*.
2. Drag a Media Asset to the **Background** or **Foreground** drop zone.
   - The Media Asset is added to the Media Asset Manager as a Texture



3. Click on the Media Asset in the Media Asset Manager.

4. In the Media Asset properties panel, configure the Media Asset as required.

See Also

- Video Clips
- Media Asset Manager

7.3 Media Asset Channel Types

The following media asset channel types are available:

- Channel Folder Media Assets
- Clip Channel Media Asset
- Container Folder Media Assets
- GFX Channels
- Image Channels

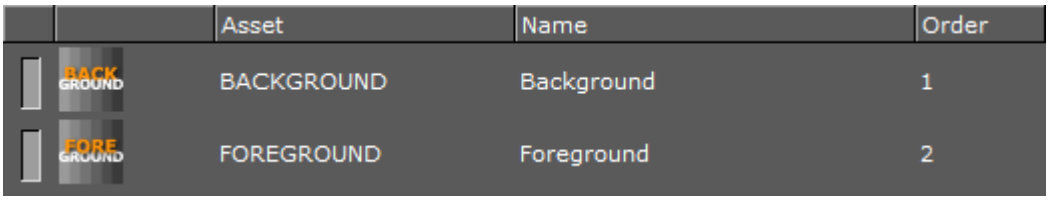
- Live Video Media Asset
- Stream Media Asset
- Super Channels

7.3.1 Channel Folder Media Assets

Background and Foreground Media Assets are located in the Channel folder of the Built Ins panel.

The Background and Foreground Media Assets represent the rendered background and foreground graphics on the DVE Compositor.

> ⚠ **Note:** By default a Background and Foreground Media Asset is applied to each new Scene.



The Foreground Media Asset can be ordered as required, but the Background Media Asset is always the first in the render order.

Both the Background and Foreground Media Assets can be deleted. If they have been deleted, drag the Background and, or the Foreground Media Asset icon, from the Server Panel, to the Media Asset manager, if required.

> ⚠ **Note:** Only one Background and one Foreground Media Asset can be used in the DVE Media Asset Manager at a time.

7.3.2 Clip Channel Media Asset

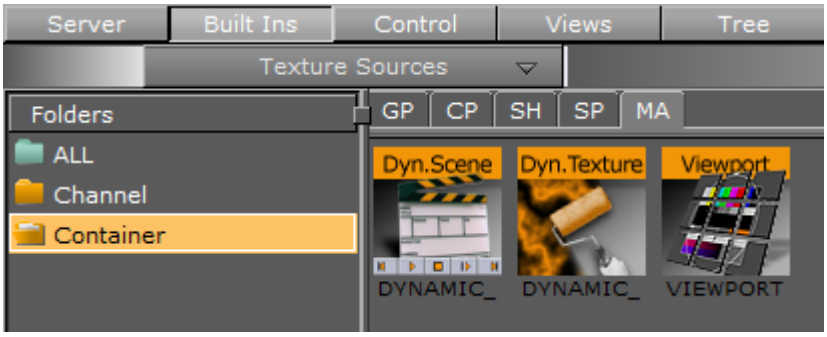
Clip Channels represent clip players. The clip players are either hardware bound native players (Matrox) or a DirectShow based software player. Clips can be used either as DVE or Texture effect.

With the Clip Channel Media Asset it is possible to play images as **DVE** or **Texture** with the Matrox clip player. Supported image containers are jpg and png. The images have to be available on a physical disk for this feature to work. One advantage of an image played as a clip, is that the same features available for a video clip can be used on the image. You can see a detailed description of the functionality in the Media Asset Panel.

⚠ **Note: DVE** is not available with DirectShow based software players.

7.3.3 Container Folder Media Assets

The following Media Assets are located in the Container folder of the Built Ins panel:



- Dynamic Scene Media Asset
- Dynamic Texture Media Asset
- Viewport Tile Media Asset

Dynamic Scene Media Asset

The Dynamic Scene is a Media Asset that is able to render a different scene, local scene or parts of a local scene into a texture and show the texture in the current scene being rendered.

Showing a different scene as ‘a window’ in the current scene can be used in a virtual studio scene where you would like to ‘open a window’ (as a texture) and render a different scene (e.g. a weather forecast).

You can also use it to show parts of a local scene if you want to animate the camera and at the same time show static items in the scene. The animating camera can be a different camera (two) and be used to show dynamic images in camera one’s view. This way you can show more than one camera at the same time.

For visual effects, the whole scene or parts of it can be grabbed into an image, and have shaders applied to the whole scene or parts of it.

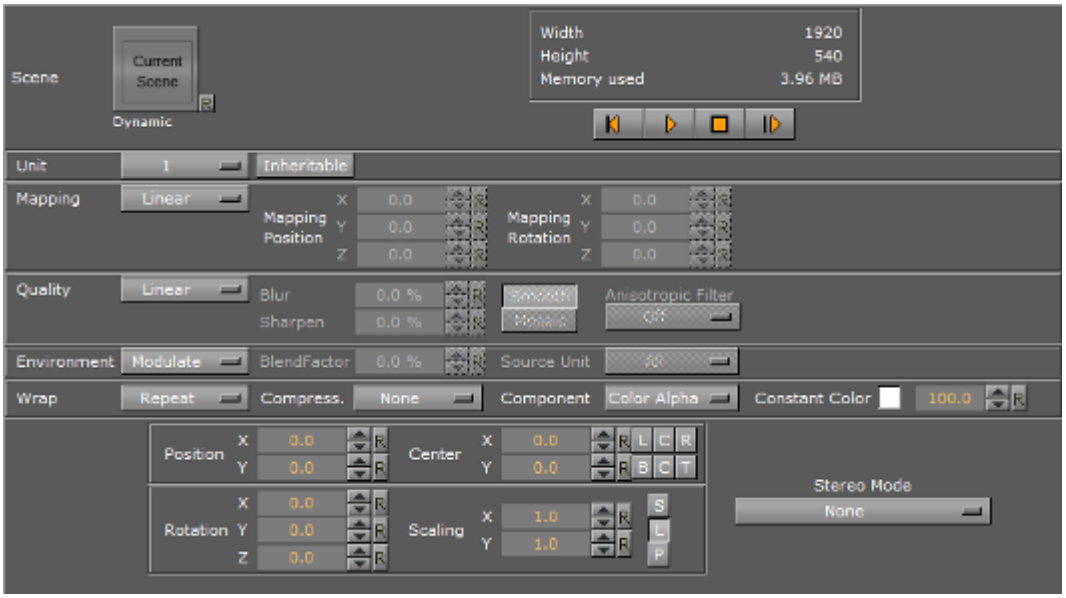
To transition from one scene (A) to another (B) both scenes are rendered into dynamic images and a third scene is used to hold both dynamic images to create a transition between them (see Create Transition Effects).

⚠ **Note:** This Media Asset is located in: Built Ins -> Media Assets -> Container

This section contains information on the following topics:

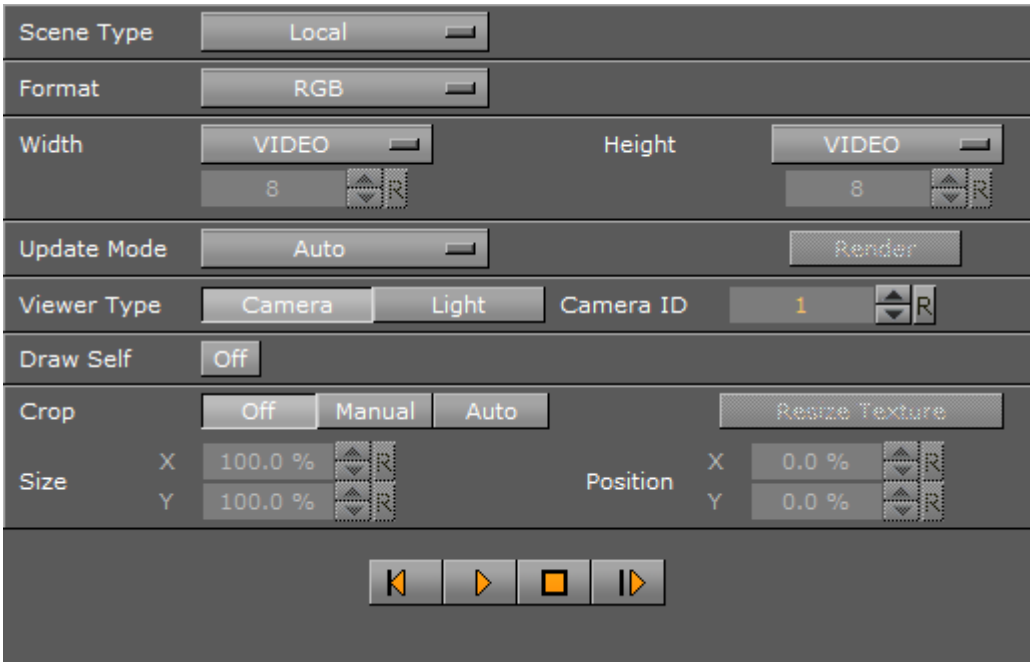
- Dynamic Scene Texture Properties
- Dynamic Scene Dynamic Properties

Dynamic Scene Texture Properties



- **Scene**: Placeholder for the rendered scene. If it is empty, the current scene will be rendered.

Dynamic Scene Dynamic Properties



- **Format**: Sets the format to be used for the dynamic image. If the format has alpha key it must be used on the render scenes or it will show as black.

- **Width / Height**: Sets the width and height of the image. The Video option will take the current output size. When selecting Custom you may set any size; however, it is always better (when possible) to use power of 2 images.
- **Update Mode**: Sets when the scene in the dynamic image will be rendered.
  - **Change**: When anything changes in the scenes it will be rendered.
  - **Always**: It will be render each frame regardless if there was any change.
  - **Command**: When the render command will be sent the scene will be rendered.
  - **Grab**: Renders the scene on command and can then be copied to another image.
- **View Type**: Sets the view port to Camera or Light. If Camera is selected and Camera ID is set to zero (0), it will take the current camera. If Light is selected it will use the light according to the Light ID.
- **Draw Self**: Creates an endless mirror effect.
- **Crop**: Enables different crop modes.
  - **Off**: The entire view port will be rendered.
  - **Manual**: Enables you to set the size and position of the rendered scenes.
  - **Auto**: The size and position of the rendered scene will be set by the bounding box holding the dynamic scene (acts as a mask).
- **Size / Position**: Sets the size and position of the dynamic image.
- **Resize Texture**: Should the image be resized automatically to reflect the size of the scene being rendered.
- **Back/Play/Stop/Forward**: Controls the playback of the scene.

Dynamic Texture Media Asset

Dynamic Texture is a feature that generates dynamic textures on-the-fly. It can be used as an alternative to importing textures to save texture memory usage. Another advantage of using a dynamic texture is that you will have a texture that is not pixelated as you zoom in or out of it.

⚠ **Note:** This Media Asset is located in: Built Ins -> Media Assets -> Container

Viewport Tile Media Asset

⚠ **IMPORTANT!** This Media Asset is considered legacy. Please use GFX Channel Media Assets when designing new scenes in Viz Artist.

The Viewport Tile renders a Scene into a part of the current viewport with the need for a texture, like a GFX Channel Media Asset.

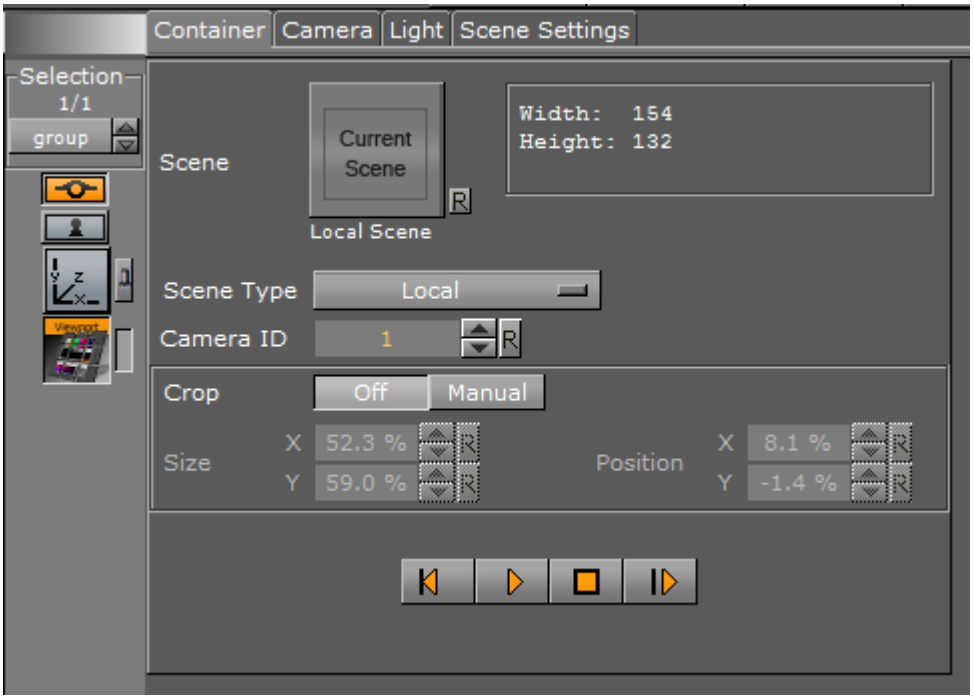
- **Advantage**: The rendering can be slightly faster in specific use-cases
- **Disadvantages**:
  - Viewport tile does not support post processing effects such as lens distortion and defocus.
  - The Viewport Tile is always orthogonal to the camera.
    Viewport tiles can also be interactive and make use of the regular mouse and multi-touch events. However, Viewport tiles can not be nested. A Scene with a Viewport tile, which contains a Scene, can not be used as a Scene in another Viewport tile.

⚠ **Note:** This Media Asset is located in: Built Ins -> Media Assets -> Container

This section contains information on the following topics:

- - Viewport Tile Properties
  - To Render a Scene in a Viewport Tile

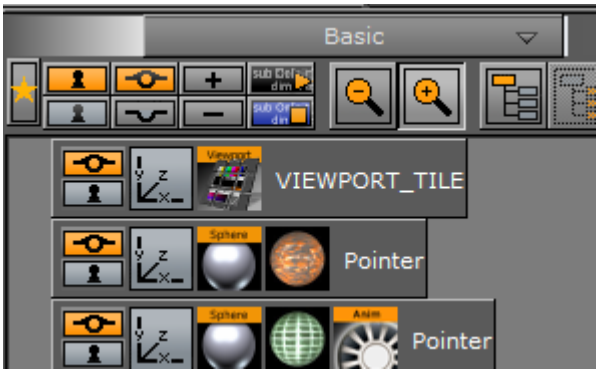
Viewport Tile Properties



- **Scene**: Placeholder for the rendered scene.
- **Scene Type**:
  - **Local**: Renders the current scene.
  - **Other**: Renders the Scene put in the Scene drop zone (drag and drop a scene).
  - **Main Layer**: Renders the scene of the main layer.
- **Camera ID**: Select a camera for the Viewport Tile scene.

- **Crop**:
  - **Off**: No cropping.
  - **Manual**: Define the new viewport **Size** and **Position** with the parameters X and Y.
- **Play controls**: Used to play, stop, etc. the animations of the Viewport Tile scene.

To Render a Scene in a Viewport Tile



1. Add the **Viewport Tile** to the Scene Tree.
2. Open the Viewport Tile editor and add a Scene to the **Scene** placeholder.
3. Open the **Transformation Editor**
4. Set the **Screen Size** (**width** and **height**) of the Viewport Tile.
5. Open the Viewport Tile editor
6. Click the **Play** button.

7.3.4 GFX Channels

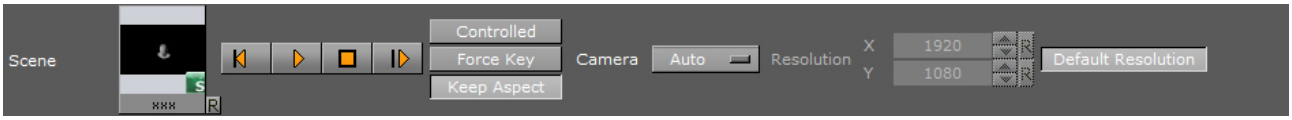
Graphics (GFX) Channels allow a designer to compose several independent scenes into a new scene. Utilizing a GFX Channel, a previously created scene can be added to the current scene as a DVE or Texture.

GFX Channel Media Assets can be nested. This means a scene with a GFX Channel Media Asset containing a scene, can be used as a scene placed within another GFX Channel Media Asset. However, nested scenes cannot use the same GFX Channels as the scene they are nested into. For example, if *Scene 1* uses GFX channel 1, it cannot be nested within GFX Channel 1 of *Scene 2*. It can however be nested within GFX channel 2.

GFX channels do not support the use of Transition Logic or Remote Cameras. Also note that cropping GFX channels have an impact on the render performance. You can see a detailed description of the functionality in the Media Asset Panel.

GFX Properties

By default, a GFX Channel automatically inherits the camera settings of the Scene it is applied to. In addition to the common Media Asset properties, GFX Channels also have the following properties:



- **Scene:** Drag a scene, which should be rendered in the GFX channel.
- **Play Controls**: Controls the animation of the set scene.
- **Controlled**: Controls the GFX Channel animation using the main or parent Scene animation controls if active. For example, if the animation in the main Scene starts, it triggers the start of the GFX Channel Scene.
- **Force Key:** Renders the entire Scene as keyed if the Scene does not have any keying information. If a GFX channel is rendered as Texture, it will be in RGBA. Therefore, the Scene needs to have keying information for an alpha channel (the key scene setting is set to auto key or a key plug-in is used in the Scene Tree).
- **Keep Aspect**: Renders the GFX Scene with its own aspect ratio (undistorted) if set to **On**. The Scene will be scaled to fit the GFX channel size if set to **Off**.
- **Camera**: Selects the camera view of the GFX Channel Scene. "Auto" means that the current camera setting of the loaded scene is used (may also change due to animations within the loaded scene).
- **Resolution:** Defaults the GFX Channels to the resolution configured for the current *Output Format* in the Viz Engine configuration. To change the current resolution, click the **Default Resolution** button to make the **Resolution X** and **Y** fields available for input, and enter the desired resolution setting. Clicking the Default resolution again will reset the resolution for the GFX Channel, as well as disable the **Resolution X** and **Y** input fields. If the Default Resolution button is in a pressed state, and the Resolution X and Y fields are dimmed, the default resolution is in use.

⚠ The resolution setting only applies to GFX Channels in Texture mode.

Performance considerations for GFX Channels

Rendering a GFX channel means typically that a second scene has to be rendered in multiple render passes. To get the best performance out of this, please consider a few hints:

- GFX channels can be rendered in three internal modes (from fastest to slowest):
  - The scene is directly rendered on top of the main scene, using the correct size as needed.
  - The scene is rendered in a temporary texture (using the correct size) and this texture is blended into the main scene (needed when render sequence of nested scene is "Overlay", or resulting texture is used with alpha < 1.0 in main scene).
  - The scene is rendered in a temporary texture (using **full render size**) and this texture is blended and scaled down into the main scene (needed when inner scene contains Defocus, Glow, or a Magnifier).

- This leads to two main performance tipps:
  - If your nested scene is a fullscreen scene, make sure that you are using render sequence **Fullscreen** instead of using **Overlay**.

- Consider if you really need postprocessing effects Defocus, Magnifier or Glow in your nested scene, especially if the GFX channel is scaled down in your main scene.

See Also

- Event Pool

7.3.5 Image Channels

This section details how to create, assign and work with Image Channel media assets.

Image Channels are mainly used in conjunction with Viz Multiplay for creating video wall layouts. However, you can also use Image Channels to add images to a 2D environment in DVE mode. You can still add additional 3D objects to your scene tree, but the Image Channel will always reside in the background of your scene unless you change the layer order via the DVE menu.

⚠ Image Channels are designed to work only in DVE mode, not Texture mode.

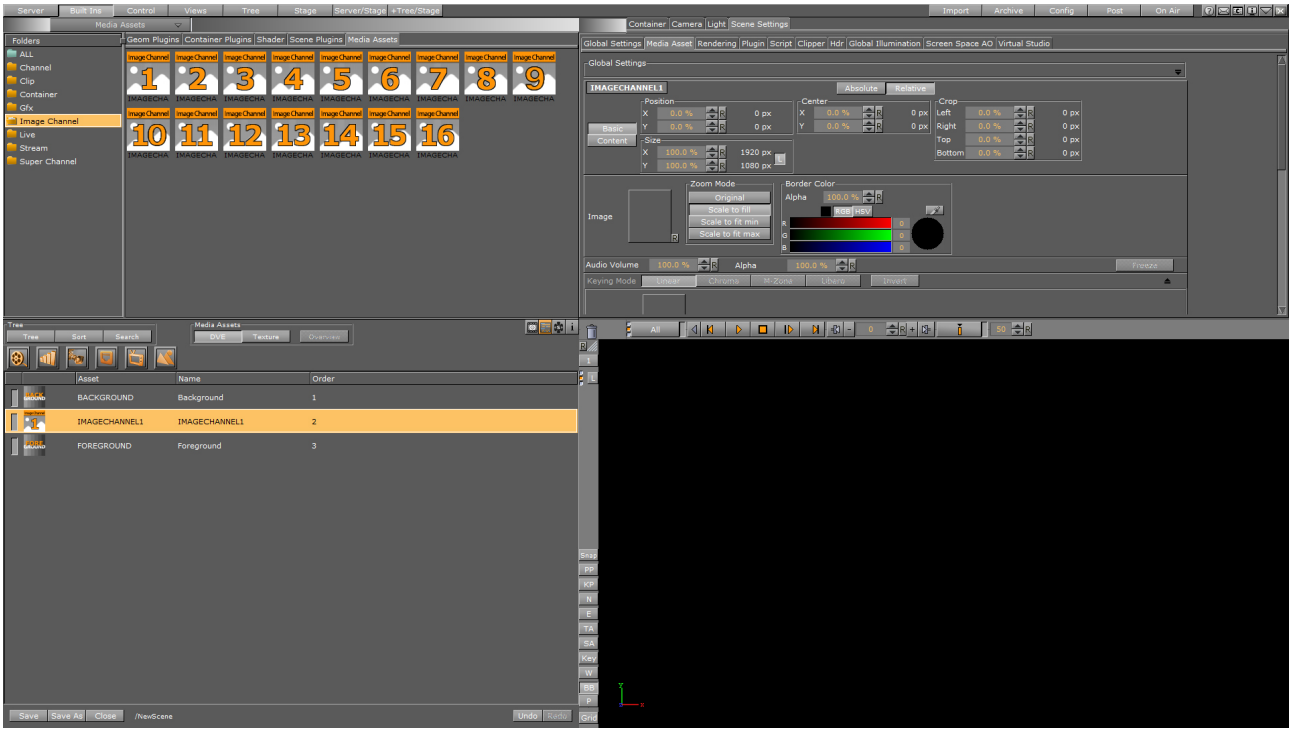
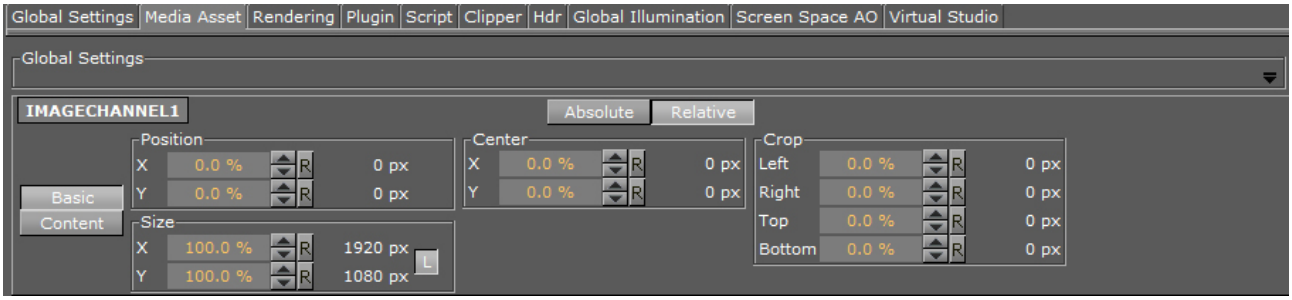
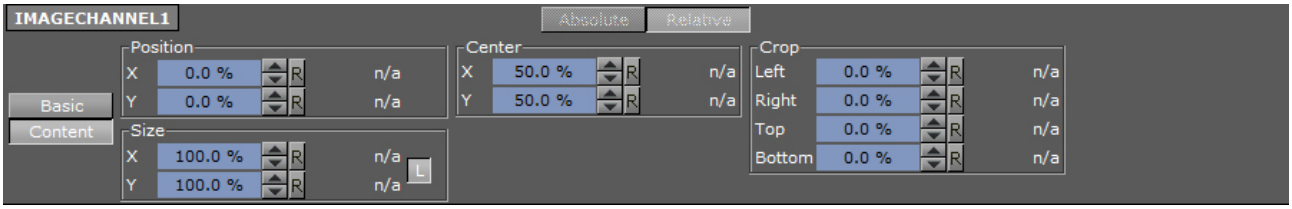


Image Channel Global Settings

The Image Channel media asset global settings are available in the Media Asset Panel. Here, you can manipulate transformation of the currently selected channel. By clicking on the label **IMAGECHANNEL1** in the **Global Settings** panel, you can switch between all used Image Channels in the scene. This applies to all Global Settings for the different types of Media Assets.



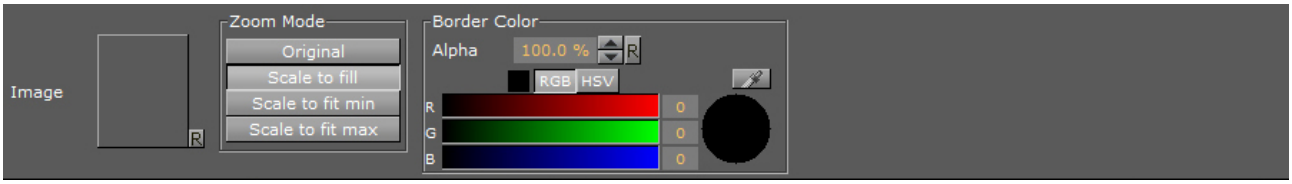
Switching between **Basic** and **Content** is indicated by blue input fields.



> ⚠ **Note:** If you use more than one image channel in your scene, you can switch between the image channels in the scene by clicking the **IMAGECHANNEL1** label.

Image Channel Specific Settings

The following settings lets you to manipulate the content inside the Image Channel:



- **Image:** Drag or load the image here.
- **Zoom Mode:** Selects the zoom mode for the image. The options are **Original**, **Scale to fill**, **Scale to fit minimum** and **Scale to fit maximum**.
- **Border Color:** Defines a border color, and defines the color and alpha value of the border.

Enforced Background Loading

Similar to generic images in the scene tree, images for image channels can be loaded using background loading even if background loading is disabled in the config. To enforce background loading, use the following command when loading an image:

VizEngine commands

```
$(scene)*IMAGECHANNEL*$(id)*IMAGE SETBGL $(image_path_or_uuid_or_url)
example:
MAIN_SCENE*IMAGECHANNEL*2*IMAGE SETBGL <31E74EAF-E4B1-254C-B50987E7DF19182F>
```

7.3.6 Live Video Media Asset

Live Media Assets integrate live video sources either as DVE or Texture Effect. You can see a detailed description of the functionality in the Media Asset Panel.

Live Video Feeds

To be able to use a Live input channel, a Clip channel or a Stream channel, they first have to be set to active in **Video Input** (see the **Configuring Viz** section of the Viz Engine Administrator Guide). A live video feed can be can be applied through either DVE or Texture. If alpha is required for the external video, open **Config** and enable alpha in **Video Input** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

- Specifically for Matrox cards: Enable **w/Alpha** when you require external video with alpha. The Matrox board combines two of the video inputs in pairs, to serve both fill and key. Inputs A and B, and input C and D are combined, respectively.
- Use the Stream media asset to capture input using the Matrox X.mio 3 IP card.

> ⚠ **Tip:** When hovering the mouse pointer over a **Live Video Feed** asset, the tool-tip will display the resolution configured for that Live Video Feed.

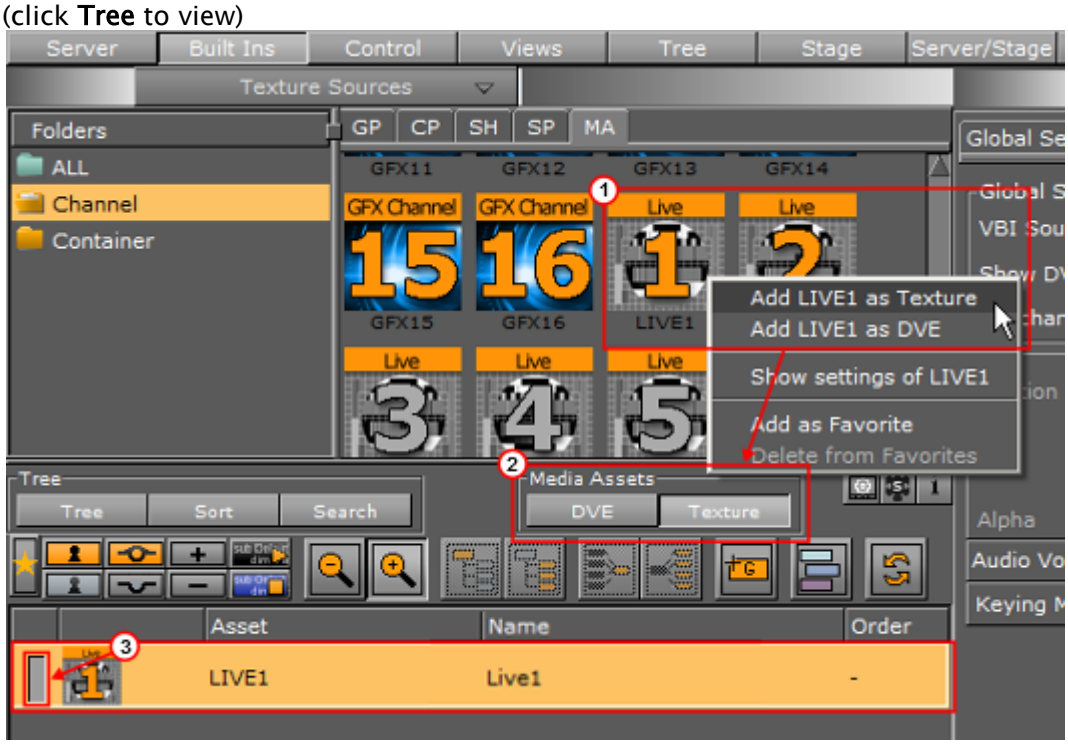
This section details the following procedures:

- To Feed Live Video as Texture
- To Feed Live Video as DVE

To Feed Live Video as Texture

1. Click on **Built Ins**.
2. Click on **Media Assets** (MA).
3. <Right-click> on the required Live Channel, and select **Add Live<1> as Texture** (1).
    - The **Texture** button in the Scene Tree panel (Media Asset Manager) is automatically selected (2)

- The selected Live Channel is added to the Scene Tree and a new Container is created (click **Tree** to view)



> ⚠ **Tip:** The **Live<1>** Channel can also be dragged to the Scene Tree, where it is added as a Texture in a new Container.

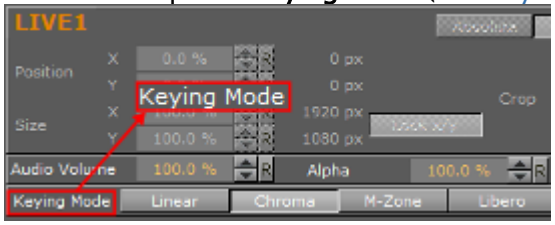
> ⚠ **Tip:** The **Live<1>** Channel can be made active or inactive. Toggle the active/ inactive button (3).

4. Click on **Live<1>** Channel, in the Scene Tree panel (Media Asset Manager) **Texture** view, to open its properties panel.

> ⚠ **Tip:** Also <right-click> the **Clip<1>** icon in its Container. Select **Show settings of Live<1>** to show its properties at any time.

> ⚠ **Tip:** Click the  icon in the Scene Tree to view Media Asset properties at any time

5. Select the required **Keying Mode** (see Keying Mode).



6. Click **Tree**.

7. Add a Geometry to the Container.

To Feed Live Video as DVE

> **Note:** If a DVS board is installed, DVE is simulated through the software.

1. Click on **Built Ins**.
2. Click on **Media Assets** (MA).
3. <Right-click> on the required Live Channel.
4. Select **Add Live<1> as DVE**.
   - The **DVE** button in the Media Asset Manager (Scene Tree panel) is automatically selected

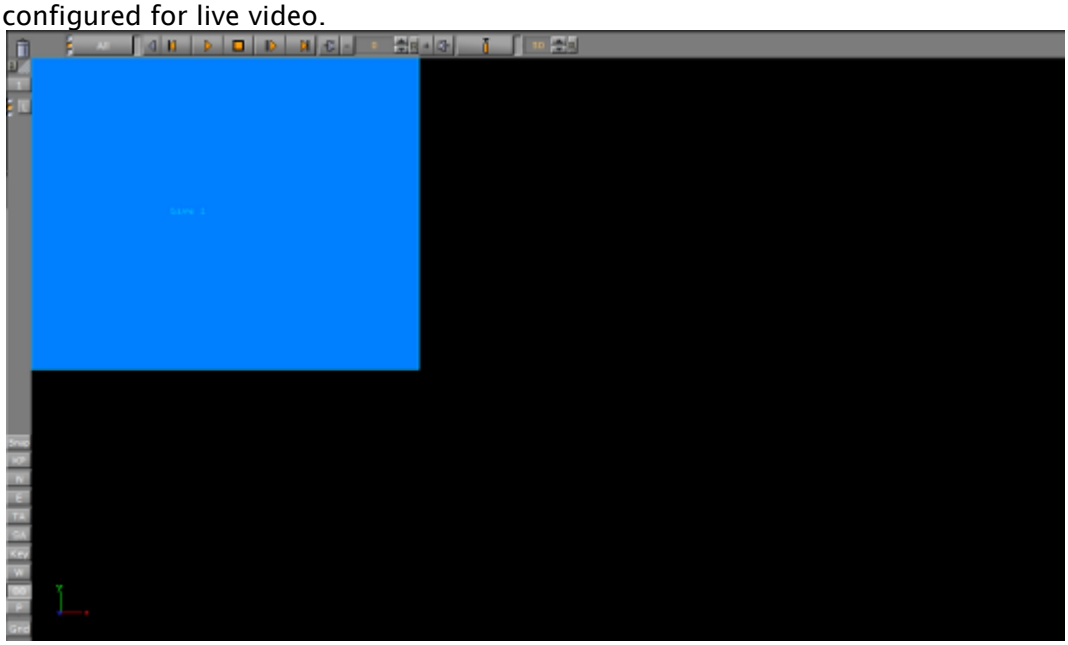
> **Tip:** The **Live<1>** Channel can be made active or inactive. Toggle the active/ inactive button (1).

5. Click on **Live<1>** Channel, in the Media Asset Manager (Scene Tree panel) **DVE** view, to open its properties panel.

> **Tip:** Click the  icon on the top of the Scene Tree to view Video settings, at any time.

6. Adjust the parameters in the DVE control panel, as required.

7. The Scene Editor shows a blue rectangle, which shows that the live input channel is correctly configured for live video.



7.3.7 Stream Media Asset

Video Streams represent IP-based live Video streams (IP Engine). They can be used either as DVE or Texture Effect.

Stream Media Assets visualize a video being streamed into the engine. A detailed functionality description is in the Media Asset Panel.

Live Feed from a Video Stream

Video from a video stream can be applied through either DVE or Texture.

To play a Video Stream, make sure that a Stream channel is active (see the **Video Input** page in the **Configuring Viz** section of the Viz Engine Administrator Guide).

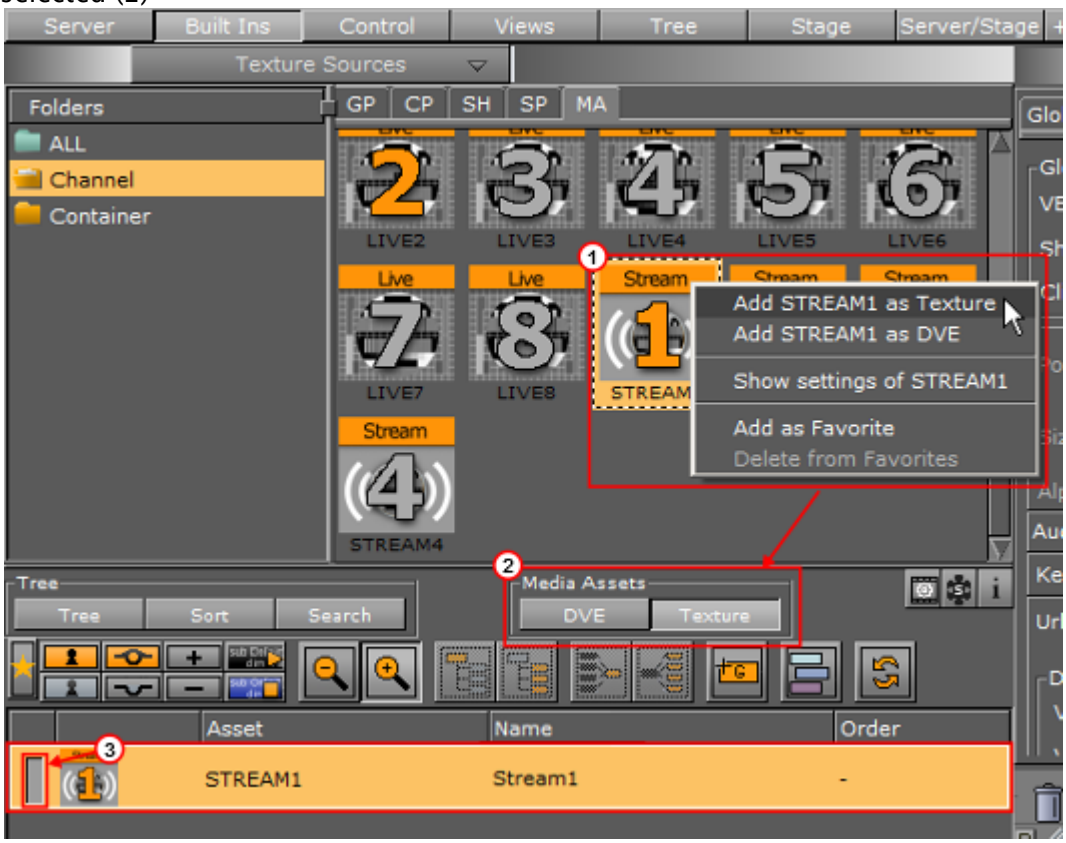
This section details the following procedures:

- To Feed a Video Stream as Texture
- To Feed a Video Stream as DVE
- To Remove a Video Stream

To Feed a Video Stream as Texture

1. Click on **Built Ins**.
2. Click on **Media Assets** (MA).
3. <Right-click> on a Stream Channel, and select **Add Stream<1> as Texture** (1).

- A new container is created in the Scene Tree and the Stream Channel is added to it as a property
- The **Texture** button in the Media Asset Manager (Scene Tree panel) is automatically selected (2)



> ⚠ **Tip:** The **Stream<1>** Channel can also be dragged to the Scene Tree, where it is added as a Texture in a new Container.

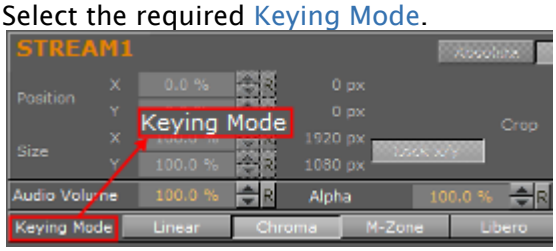
> ⚠ **Tip:** The **Stream<1>** Channel can be made active or inactive. Toggle the active/inactive button (3).

4. Click on **Stream<1>** Channel, in the Media Asset Manager (Scene Tree panel) **Texture** view, to open its properties panel.

> ⚠ **Tip:** Also <right-click> the **Stream<1>** icon in its Container. Select **Show settings of Stream<1>** to show its properties at any time.

> ⚠ **Tip:** Click the  icon on the top of the Scene Tree to view Media Asset properties at any time

5. Select the required Keying Mode.

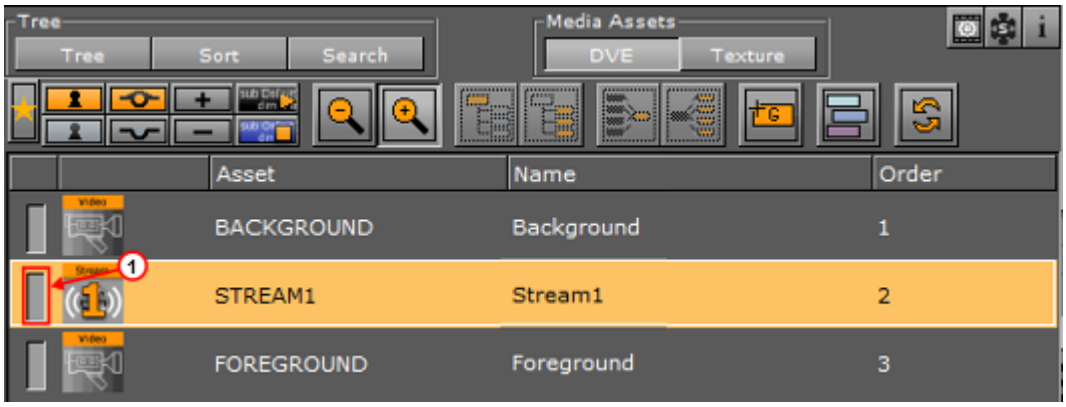


6. Enter the stream **URL**.
7. Click on **Load**.

> **Note:** To load a new Video Stream, enter the new URL in the **URL** box. This will automatically clear the old Video Stream.

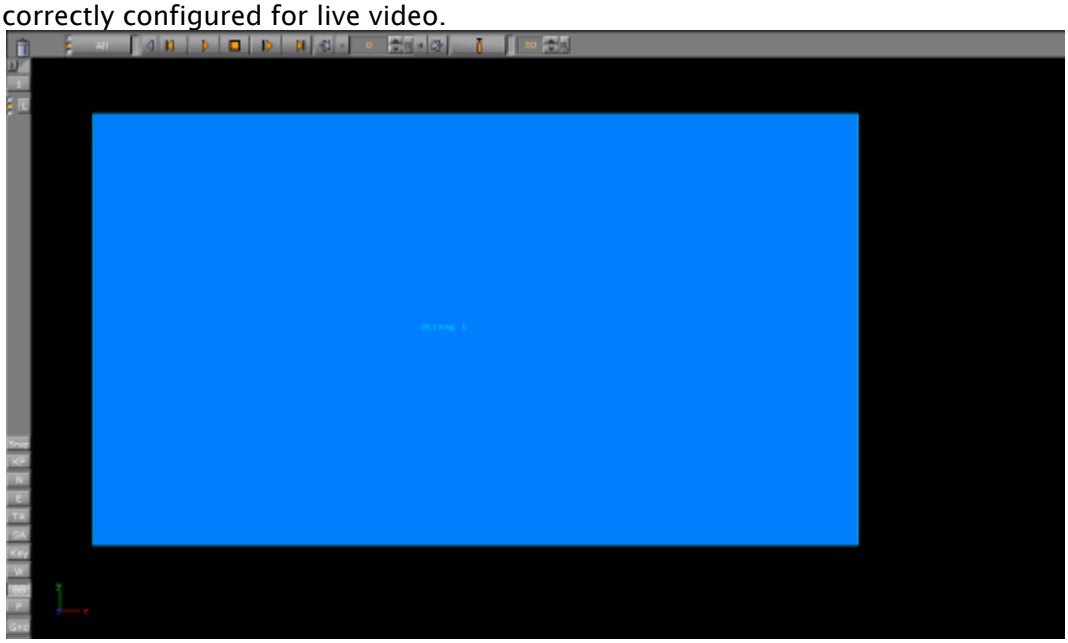
To Feed a Video Stream as DVE

1. Click on **Built Ins**.
2. Click on **Media Assets** (MA).
3. Right click on the required Stream Channel.
4. Select **Add Stream<1> as DVE**.
   - The **DVE** button in the Media Asset Manager (Scene Tree panel) is automatically selected.



> **Tip:** The **Stream<1>** Channel can be made active or inactive. Toggle the active/ inactive button (1).

The Scene Editor shows a blue rectangle, which shows that the live input channel is correctly configured for live video.



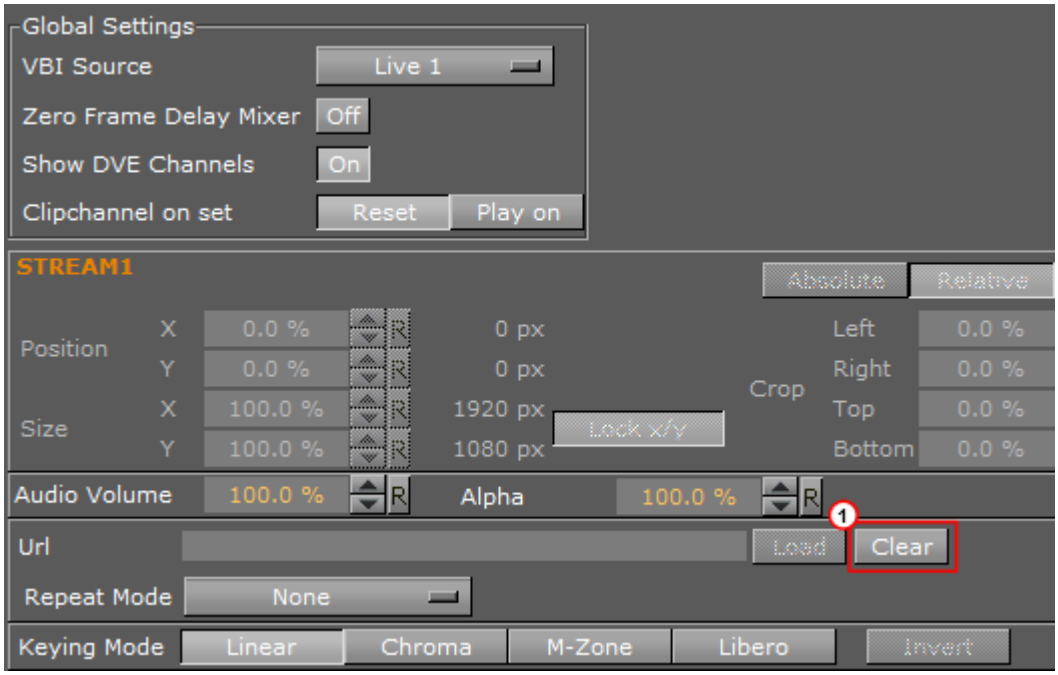
5. Click on **Stream<1>** Channel, in the Media Asset Manager (Scene Tree panel) **DVE** view, to open its properties panel.

> ⚠ **Tip:** Click the  icon on the top of the Scene Tree to view Video settings, at any time

6. Adjust the parameters in the DVE control panel, as required.
7. Enter the stream **URL**.
8. Click on **Load**.

To Remove a Video Stream

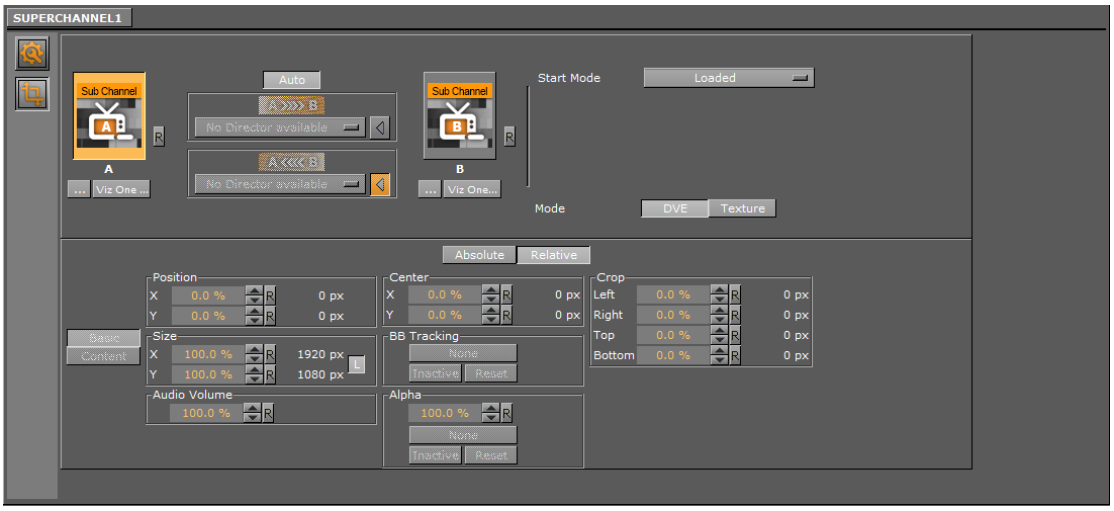
- Click on the **Clear** button (1).



7.3.8 Super Channels

Super Channels provide a generic tool to create and group transitions between various types of assets in Viz Artist. They are an abstraction mechanism to represent, access and apply transitions between the various types of media assets. Within the Super Channel, you assign two media assets, A and B, to provide event based transitions between them. Internally, the Super Channel then utilizes the channel type required of the corresponding media asset to represent the underlying content. Images use Image Channels, scenes use GFX Channels, clips use Clip Channels, streams use the Stream Channels, and live feeds use Live Channels. There are eight Super Channels available that can be used simultaneously.

⚠ Super Channels are designed to work only in DVE mode, not Texture mode.

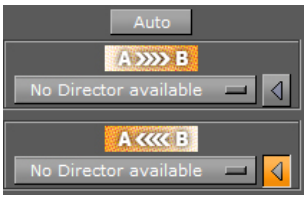


Here, you find information on the following:

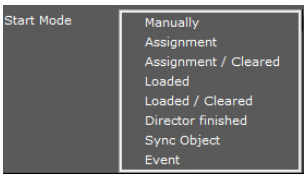
- Super Channel Settings
- Super Channel Context menu
- Super Channel Transformation
- Sub Channels
- Sub Channel Settings
- Enforced Background Loading

Super Channel Settings

Sub Channel (A, B): Drag scenes, images or clips to the **Sub Channel** icon. Click **Viz One** to load a clip from your configured Viz One system, or load a local clip by clicking the **Browse for a clip** button.



Transition: Sets the direction of the transition, from sub channel **A to B**, or from **B to A**. Disable `Auto` to select an existing Director to run the transition with, or `None` to disable the transition.



Start Mode: Selects what will trigger the Super Channel transition:

- **Manually:** Sets user manual switching from sub channel A to B, or vice versa, by selecting the sub channel in Viz Artist or through external commands.
- **Assignment:** Triggers transition when you assign a new asset to inactive sub channel.
- **Assignment / Cleared:** Triggers transition asset assignment, or when you clear the sub channel.
- **Loaded:** Triggers transition when an asset is loaded. This is useful for Super Channels that display images when image background loading is active in the Viz Engine configuration.
- **Loaded / Cleared:** Triggers transition when an asset is loaded, or when you clear the sub channel.
- **Director finished:** Triggers transition once a given director has finished execution.
- **Sync Object:** Drag a Super Channel to the **Sync objects** area. Sync objects provide a way to synchronize execution of Super Channel transitions between multiple Super Channels. Sync objects monitor all linked Super Channels until all their inactive sub channels have been loaded. The sync object then calls `onAllChannelsArmed` to trigger the transition for all Super Channels managed by the sync object. You can add as many Super Channels to a sync object as are required.
- **Event:** Allows you specify the Scene, the sender (Animation from Stage or from Super Channel) and the Event (on Started, on Stopped, onDirectorStarted, onDirectorStopped).

Mode: Selects the mode for using internal channels. This only affects hardware based channels, e.g. Live1 or Clip1. In **DVE** mode, the internally used channel for a subchannel will be used in DVE mode, all properties of the subchannel are copied to the DVE layer. In **Texture** mode, the internally used channel is used in texture mode and an ImageChannel is used to show the texture.

⚠ **Sync Object Example:**

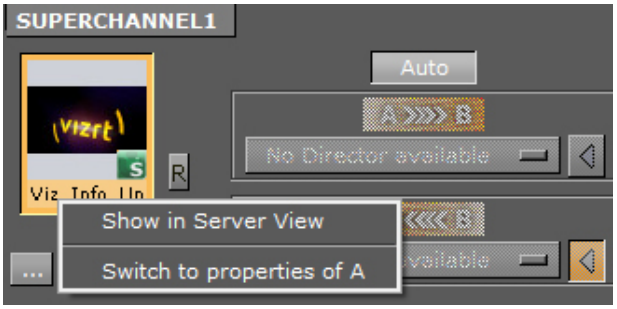
Add two Super Channels to the scene. Then set **Start Mode** of Super Channel 1 to **Sync Object** and drag Super Channel 2 into the list of linked channels. Notice how this links Super Channel 1 and Super Channel 2 to each other. The transition from the active to the inactive **Sub Channel** starts as soon as both inactive Sub Channels have fired **onAssigned** and **onLoaded**. Assign an image to sub channel **B** of both Super Channels to achieve this. You can do this manually in Viz Artist, or via external commands:

```
MAIN_SCENE*Super Channel*<CHANNEL_ID>*<A|B> SET IMAGE <UUID>
```

In case a sync object is active, but you do not want to change the assets on **allsyncedSuper Channels**, you still have to call the assign command on all Sub Channels. Otherwise, the sync object will not fire, and thus the transition will not trigger. However, re-assigning the already assigned Media Asset is completely free in terms of performance, as no internal reloading takes place. As such, this represents a viable option to trigger the `requiredonAssignedevent`.

Super Channel Context menu

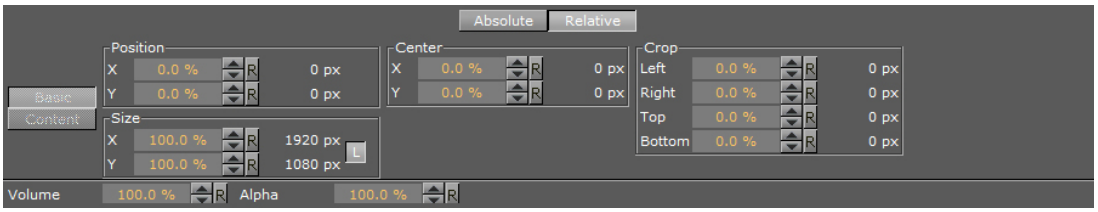
The channel context menu provides quick access to the properties of the scene, as well as to quick-select it in the database.



- **Show in Server View**: Selects and highlights the scene in the server menu and highlighted.
- **Switch to properties of A/B**: Provides direct access to the transformation of the selected Sub Channel.

Super Channel Transformation

This section is used to animate the whole Super Channel, including both sub channels. Super Channel animations are independent of the sub channel animations. This means that you can have a transition animation between multiple Super Channels, which is often required for content on Videowalls.

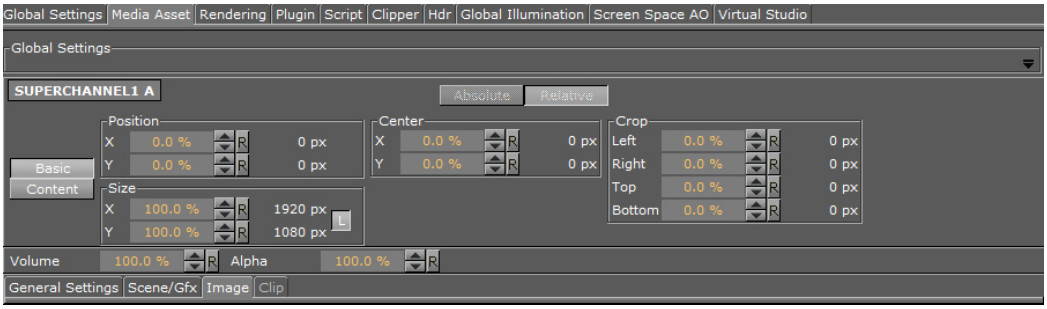


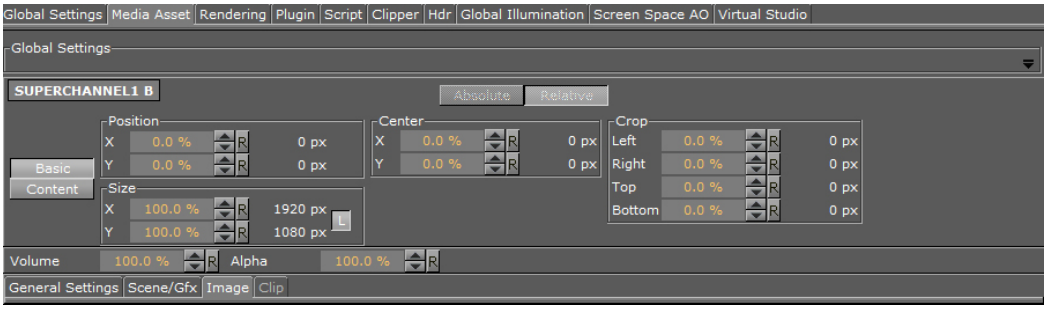
You can animate certain Super Channel properties, such as **Position**, **Size**, or **Alpha**, to define custom transitions between one or more Super Channels. A detailed description of the parameters can be found in the Media Asset Panel.

> ⚠ **Note:** Super Channels deal with rendering content assigned to the two sub channels, A and B, in the current main scene. Thus, they represent an intuitive and easy to use mechanism for transitioning between the sub-channels A and B.

Sub Channels

A Super Channel is composed of two Sub Channels, A and B. Here, you can see the sub channels A and B of Super Channel 1:

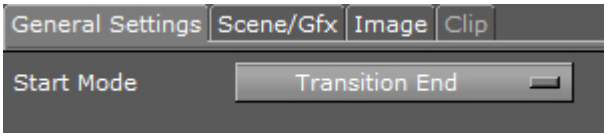




You can animate certain Sub Channel properties, such as **Position**, **Size**, or **Alpha**, to define custom transitions between the two Sub Channels. A detailed description of the parameters can be found in the Media Asset Panel. Click the text label, for example, **Sub Channel 1A**, in the Sub Channels upper right corner to switch between all Super Channels and Sub Channels in your scene.

⚠ **Note:** To utilize background loading in combination with Super Channels, enable it globally in Viz Engine. The dedicated `SETBGL` commands used with containers, `<ObjectID>*TEXTURE*IMAGE SETBGL <UUID>`, do not work with Super Channels.

Sub Channel Settings

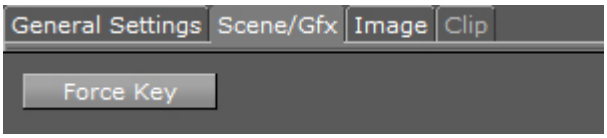


Here you can select how to trigger the Sub Channel animation. The Start Mode options are:

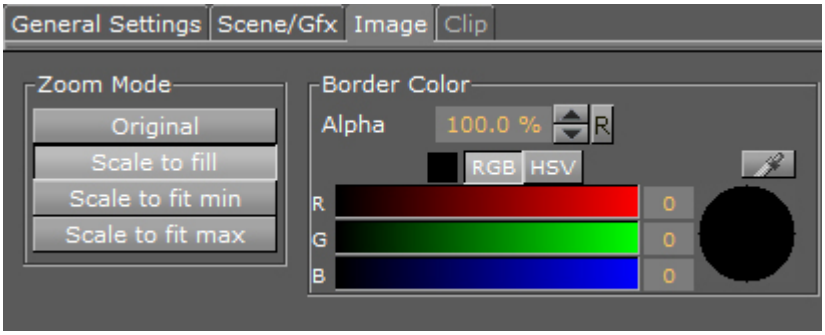
- **Manually:** Sets user manual switching from subchannel A to B or vice versa by clicking the corresponding subchannel in Artist or through external commands.
- **Transition Start:** Triggers on transition start.
- **Transition End:** Triggers on transition end.
- **Loaded:** Triggers transition once a newly assigned asset has been loaded. This is of special interest for Super Channels displaying images, with image background loading activated in the Viz Engine configuration.
- **Event:** Allows you specify the Scene, the sender (Animation from Stage or from Super Channel) and the Event (on Started, on Stopped, onDirectorStarted, onDirectorStopped). An event is fully defined by its event name, the identifier/name of the sender, and its

arguments. A new internal event system was introduced in Viz Engine 3.9. This allows certain objects to fire and listen to internal events. Amongst those objects are:

- **Super Channels or their sub channels**
- **Stage**
- **Scripts**



- **Force Key:** Allows you to force a full-screen key of your loaded scene, even there is no key function applied to it.



- **Image:** Allows you to control the Zoom mode and the border color of an loaded Image.

⚠ **Event Example: An event is fully defined by its Eventname, the identifier/name of the Sender and its Arguments.**

Eventname:
Sender:
Arguments:

```
onDirectorFinished
#1004*STAGE
director1
```

Query available/registered events using the following commands:

```
EVENT_POOL*REGISTERED_OBJECTS GET
EVENT_POOL*REGISTERED_EVENTS GET <$objectname>
```

Example

```
EVENT_POOL*REGISTERED_EVENTS GET #427*STAGE
```

⚠ **Note:** To obtain detailed information about status changes on the various Super Channels and subchannels enter command:

```
EVENT_POOL*DEBUGGING SET 1
```

You will then find additional status information related to the Super Channels and their Sub Channels in the Viz Engine's console output.

Enforced Background Loading

Similar to generic images in the scene tree, images for superchannels can be loaded using background loading even if background loading is disabled in the config. To enforce background loading, use the following command when loading an image:

VizEngine commands

```
$(scene)*SUPERCHANNEL*$(id)*$(SUBCHANNEL) SETBGL IMAGE $(image_path_or_uuid_or_url)
example:
MAIN_SCENE*SUPERCHANNEL*2*A SETBGL IMAGE <31E74EAF-E4B1-254C-B50987E7DF19182F>
```

See Also

- Event Pool

7.4 Playback Of Media Assets

The playback of Media Assets depends on the machine and video board configuration. There are three Media Asset playback methods:

- Video Clip Playback
- Live Video Feeds
- Live Feed from a Video Stream

Media Assets are applied to a Scene either as DVE or as a Texture.

Before video can be played, the **Video Input** settings must be configured in Viz Configuration (see the Viz Engine Administrator Guide). These settings determine the frame rate (PAL/NTSC) of the whole video system.

Video clips transfered from Viz One can be played if the clip's container and codec combination is supported by the currently installed video card and the corresponding clip player (Matrox clip player for Matrox video card, DirectShow player for all other boards).

These video cards are supported for video playback:

- DVS
- Bluefish
- NVIDIA Quadro Digital Video Pipeline
- Matrox

The Matrox DSX LE2<n> (see the release notes for supported <n> versions) does not have any input, codec support or compositor. Furthermore, for DSX LE3, X.mio, X.mio2 and X.mio2 Plus Matrox cards video inputs and clip channels can be disabled in the **Matrox** section of Viz Configuration (see the Viz Engine Administrator Guide).

⚠ **IMPORTANT!** The most basic rule for working with the video version of Viz Engine is to always have a valid genlock attached to the video board, no matter which board is installed.

Without the correct genlock setup there is no guarantee that the Engine will function correctly (see **Video Output** in Viz Configuration (see the Viz Engine Administrator Guide)).

Media Asset parameters are modified in the Media Asset Panel Tab in the Scene Settings.

To control Media Assets from an external control application, see the **Viz Command Interface Documentation** located in the *<Viz Installation Folder>*.

This section contains the following topics:

- Media Asset Application
  - DVE
  - Texture

7.4.1 Media Asset Application

Media Assets are applied in Viz Artist either as Digital Video Effect (DVE) or as a Texture:

DVE

The playback of external video through **DVE** does not have any impact on rendering performance, since it is handled solely on the board.

When DVE is used, the graphics scene and the channel are composited on the video card (Matrox X.mio) and an SDI monitor is needed to show the video channel. There is also a limited set of 2D control options available in the design and animation of the Viz Artist scene (position, size, crop and alpha).

Chroma keying is not supported for channels set to DVE. The **DVE Alpha** setting for **GFX Channels** allows Media Assets as DVE to be blended, however, setting alpha for nested DVEs is currently not supported.

Texture

The playback of external video through **Texture** will have an impact on rendering performance, since the live video needs to be transferred to the graphics board.

When a video is applied as a Texture, the video is part of the graphics.

When more control of a video is needed, for example, to map the video onto a 3D object in a Scene, then Texture mode can be used. The video would then have the same control parameters as an image in Viz Artist. However, the video may now affect rendering performance, since it needs to composite the video into the render window.

7.5 Video Clips

This section details where video clips are stored, the transfer and download from Viz One and video clip playback.

This section contains the following topics:

- Video Clip Storage

- Playback of Clips with VBI
- Video Clip Playback

7.5.1 Video Clip Storage

This section details where video clip can be stored and accessed for use in Viz Artist.

This section contains the following topics:

- Hard Drive Video Clips
- Viz One Video Clips
- Video items
- Dual Channel and Trio One Box Configuration
- Variables used in file-name expansion

Hard Drive Video Clips

All Viz Artist/Engine machines, as well as Vizrt clients with preview, for example, Viz Trio or Viz Weather, come with a mapped video drive (D:). This drive is configured for faster access and speeds. This is where all video clips should be stored.

> ⚠ **IMPORTANT!** It is not recommended to store video clips on the C:\ drive, as this is where the operating system is installed.

A second or third video clip directory can be added in **Local Settings** of Viz Configuration (see the Viz Engine Administrator Guide), which can be used for:

- A Dual Channel and Trio One Box Configuration, or
- With any Viz Artist/Engine configuration for the selection of a secondary video clip directory if a directory fails.
  If the file path for a video clip is set to '<clip_root>', and the first directory falls over the same video file will be searched for and used, from the next directory in the Clip directory list.
  If, when multiple Clip directories are selected, the **Clip Name** box is used to search for a video, the search will default to the last used directory.

Viz One Video Clips

Video clips can also be transferred and downloaded from a Viz One. To do this Viz One must be integrated with a Viz Engine. See the **Viz One** page in the **Configuring Viz** section of the Viz Engine Administrator Guide and the **Viz One Deployment Guide**.

Video items

Video clips are also stored in the Graphic Hub (see Working with Video Items).

Dual Channel and Trio One Box Configuration

When more than one directory path is selected, for example, D:\AEClips; D:\DecoClips, etc., in the **Clip Name** box change the file path to '<clip_root>'. The file name is then searched for, in the selected directories (selected in the Viz Configuration), one after the other, for stored video clips with the same name.

⚠ **Example:** Search for a video clip named 'next_tuesday.avi' in D:\AEClips\AFL\Promos. In the **Clip Name** box change 'D:\AEClips\AFL\Promos' to '<clip_root>' The filename shown in Viz Artist should be <clip_root>\next_tuesday.avi.

Variables used in file-name expansion

Variable	Definition
<base_scene_name>	Expands to the scene name
<absolute_scene_name>	Expands to the complete path of the loaded scene
<absolute_scene2_name>	Expands to the complete path of the second loaded scene
<clip_root>	Expands to the clip_root
<hostname>	Expands to the hostname
<if0>	Expands to the IPV4 network address of the first network interface
<if1>	Expands to the IPV4 network address of the second network interface
<if2>	Expands to the IPV4 network address of the third network interface

7.5.2 Playback of Clips with VBI

VBI (Vertical blanking interval) is the interval of time between the last line of a given field/frame and the beginning of the next field/frame, during which the incoming data stream is not shown on a monitor.

Video clips which contain VBI information, can be played back in two ways:

- One: Use a video clip which contains the embedded VBI information, or

- Two: Use a separate video clip which contains the VBI data. This separate video clip must be named [clipname]_vbi.avi and conform to the size limitations detailed in the procedure below

To Playback Clips with VBI in a Separate File

⚠ **Note:** The links in this procedure all refer out to the Viz Engine Administrator Guide.

1. Make sure that VBI is set to On for both the Clip Channel (Video Input: Clip Input) and the Video Output (Matrox).
2. For each clip channel, make sure that the Video Input and Output Format resolutions are the same.
   The VBI clip contains only VBI information. The size of the clip must match the maximal total lines for VBI in this resolution:
   - NTSC: 32
   - PAL: 34
   - 720p: 19
   - 1080i: 30
   - 1080p: 35

Using VBI embedded in a full size clip is not supported as separate clip. This is only supported when played as the main clip.

7.5.3 Video Clip Playback

Video clip playback should be used sparingly. The size and formats of some video clips can affect the Viz Engine render performance. To build a Scene with video clips (in Viz Artist that supports clip playback) there are also three plug-in options that can also be used:

- Image Clip
- MoViz
- SoftClip

This section contains the following topics:

-
  - Video Clip Playback General Guidelines and Considerations
  - To Play a Video Clip as Texture
  - To Play a Video Clip as DVE
  - Video Clip Animation in the Stage
  - To Animate a Video Clip
  - To Play a Single Clip in the Stage with Stage Loop
  - To Play Multiple Clips in One Channel
  - To Play Multiple Clips in One Channel with Stage Loop
  - To Play Single and/or Multiple Clips in Multiple Channels
  - Video Clip Queue
  - Video Clip Queue Commands
  - Example Clip Queue commands:

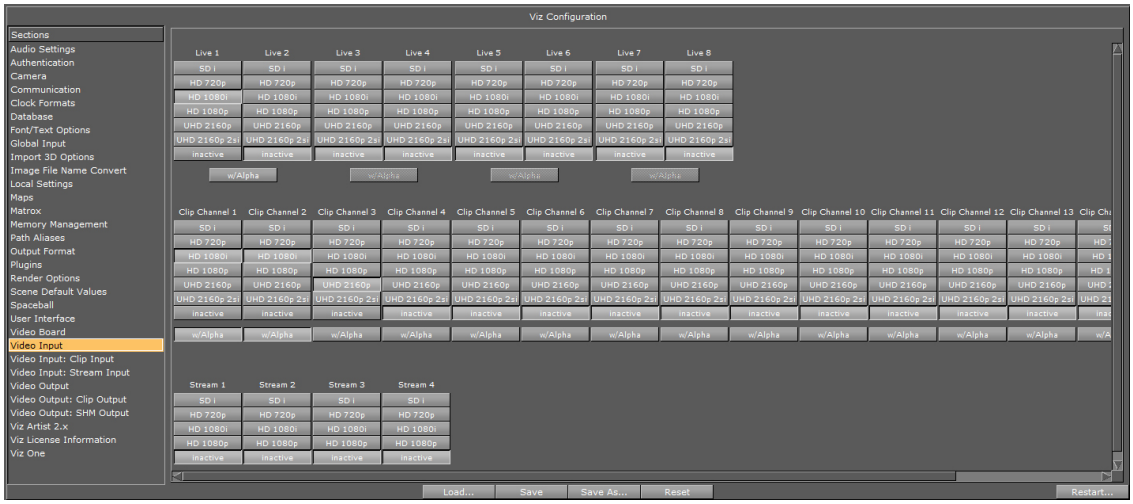
Video Clip Playback General Guidelines and Considerations

For video clip playback methods, consider these general guidelines:

Use-case	Playback Method
If no 3D effects or animation control - and only limited 2D effects - on the clips are needed	To Play a Video Clip as DVE
For full screen (e.g. background) clips	To Play a Video Clip as DVE
If video must be mapped onto a texture in the scene or requires minor adjustments	Video Clip Animation in the StageVideo Clip Animation in the Stage
For non-standard image format/resolution/codec	SoftClip or MoViz plug-in.
If more than two small clips are required	SoftClip or MoViz plug-in.
For low resolution and short animations	Image Clip plug-in.

⚠ **IMPORTANT!** Always test the performance of the different solutions and work with the solution that gives the best quality versus performance ratio. A performance test should always take place on a machine with the same specifications as those used for production playback.

- The selected video clip must have the same frame rate as the clip channel. You can mix different **Output Formats** and **Video Input Formats** with different resolutions, but you must make sure to use either progressive or interlaced for Input and Output. You can play, for example, a 720p clip with an Output format of 1080p, but you can not mix and 720p with a 1080i format:
  - Video Input: Clip Channel 1: HD 1080i
  - Output Format: 1080i



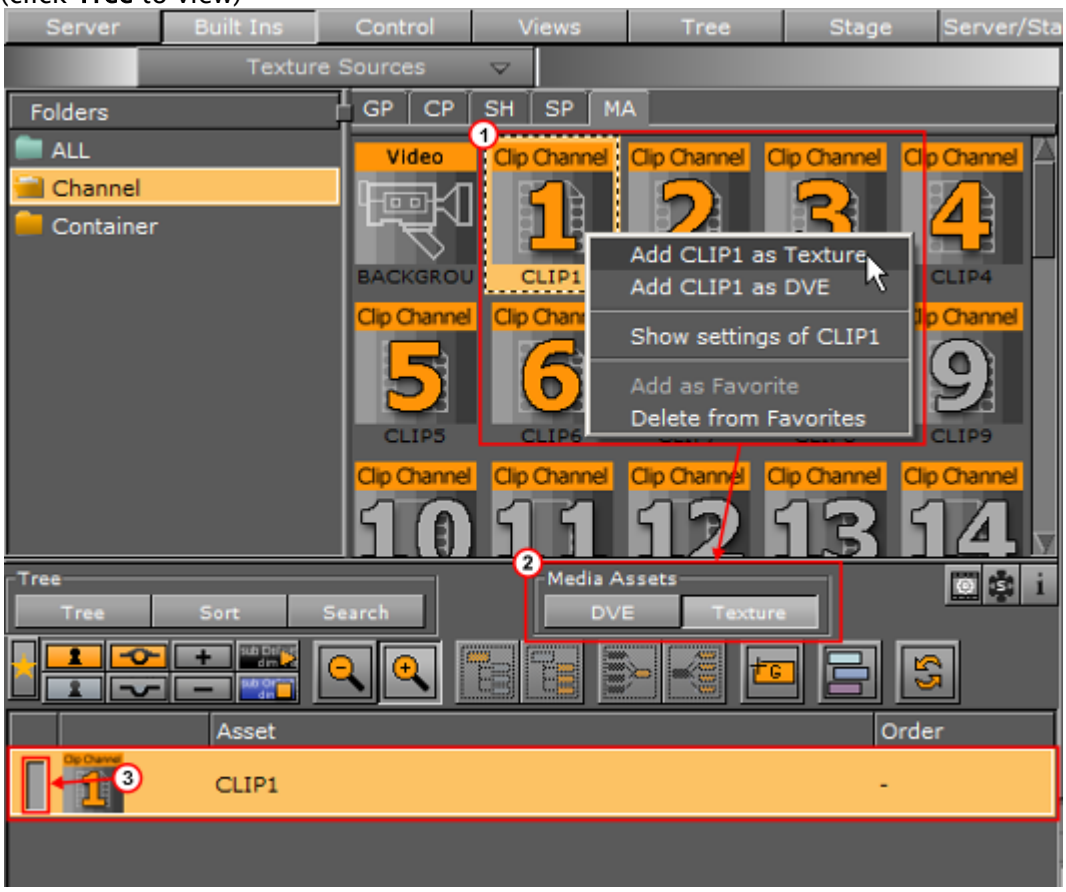
Each available Clip Channel can be configured for independent resolutions, providing the designer quick access to a range of options. These settings are configured in the Video Input section in Viz Config. Please refer to the **Video Input** page in the **Configuring Viz** section of the Viz Engine Administrator Guide.

> ⚠ **Tip:** The Media Asset Manager provides a useful tool-tip when hovering the mouse pointer over **Clip Channel** media assets, displaying the resolution configured for the Clip Channel in use.

- Make sure that the correct container and codec is selected (see Supported Codecs).
- Video clip playback can be applied through either DVE or Texture.
- To be able to use a Live input channel, a Clip channel or a Stream channel, they first have to be set to active in **Video Input** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).
- If alpha is required for a video clip, click **Config** and configure the alpha in **Video Input** (see the **Configuring Viz** section of the Viz Engine Administrator Guide). Enable **w/Alpha** when alpha is required in video clips. Either two clips, or one clip that contains both the fill and key, are required. If there is only one clip, that contains the fill (without a key), the key will be created dynamically and be fully opaque. For the naming convention of two clips, the alpha/key clip must have the same name with an attached `_key` before the file extension, this means if the main/fill clip is named `example.mxf`, the alpha/key clip must be named as follows: `example_key.mxf`.
- The Matrox Clip Player supports TDIR (Time Delay Instant Replay), which enables Viz Engine to play back clips while they are still being written by Viz One. When a clip is played, with TDIR support enabled, looping is not enabled for this clip. Viz Engine determines automatically if it needs to set TDIR support **on**, or if the clip is already fully loaded.
- BG Graphics are always the lowest in the stack, even below zero, and FG graphics are assigned a default of 100. Please note that priority can not be assigned to BG Graphics.

To Play a Video Clip as Texture

1. Click on **Built Ins**.
2. Click on **Media Assets** (MA).
3. <Right-click> on a Clip Channel, and select **Add Clip<1> as Texture** (1).
   - The **Texture** button in the Scene Tree panel (Media Asset Manager) is automatically selected (2)
   - The selected Clip Channel is added to the Scene Tree and a new Container is created (click **Tree** to view)



> ⚠ **Tip:** The **Clip<1>** Channel can be made active or inactive. Toggle the active/inactive button (3).

> ⚠ **Tip:** The **Clip<1>** Channel can also be dragged to the Scene Tree, where it is added as a Texture in a new Container.

4. Click on **Clip<1>** Channel, in the Scene Tree panel (Media Asset Manager) Texture view, to open its properties panel.

> ⚠ **Tip:** <Right-click> the **Clip<1>** icon in its Container. Select **Show settings of Clip<1>**.

> **Tip:** Click the icon in the Scene Tree to view Media Asset properties at any time

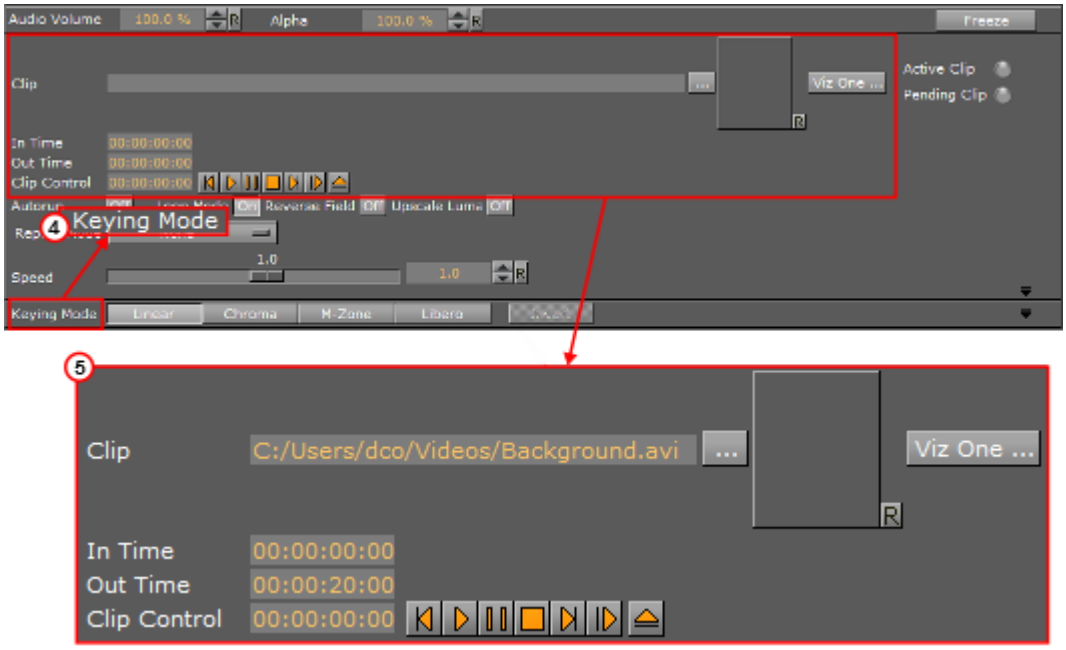
5. Select the required **Keying Mode** (4 (see Keying Mode)).
6. Add a video clip:
   - In the **Clip Name** (5) line, click ... , or
   - Drag a Video Clip item from the Server Panel, or
   - Click Viz One ... , and select a video clip to import

> **Note:** To transfer and download videos from Viz One, Viz One must be integrated and configured with Viz Artist/Engine.

7. Configure the video clip playback as required.
8. Click (5 (Play)) to test the video clip.

> **Note:** Matrox Card Only: A green 'traffic light' shows that the video clip has loaded successfully. A gray 'traffic light' shows that no video clip has been loaded. A red 'traffic light' shows that there was an error when the video clip was loaded.

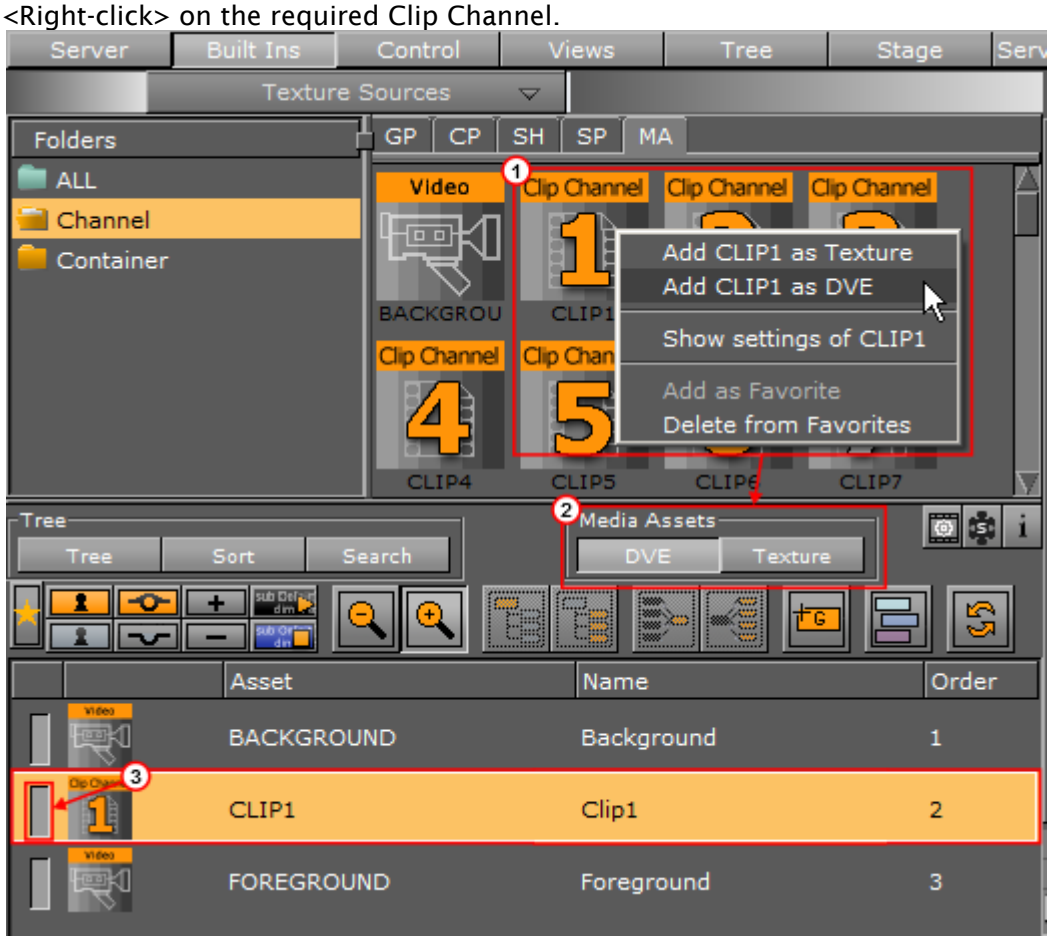
> **Note:** Click the expand button to show addition information. If the clip loaded successfully, clip details are shown. If the clip did not load successfully, additional errors are shown.



To Play a Video Clip as DVE

1. Click on **Built Ins**.

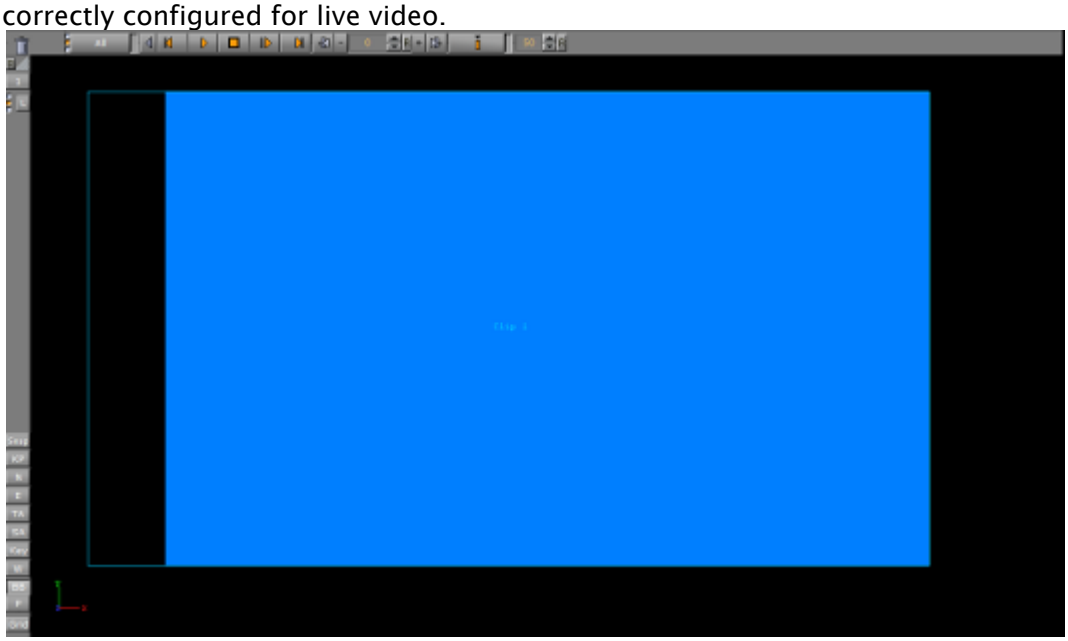
2. Click on **Media Assets** (MA).
3. <Right-click> on the required Clip Channel.



4. Select **Add Clip<1> as DVE** (1).
   - The **DVE** button in the Scene Tree panel (Media Asset Manager) is automatically selected (2).

> ⚠ **Tip:** The **Clip<1>** Channel can be made active or inactive. Toggle the active/ inactive button (3).

The Scene Editor shows a blue rectangle, which shows that the live input channel is correctly configured for live video.



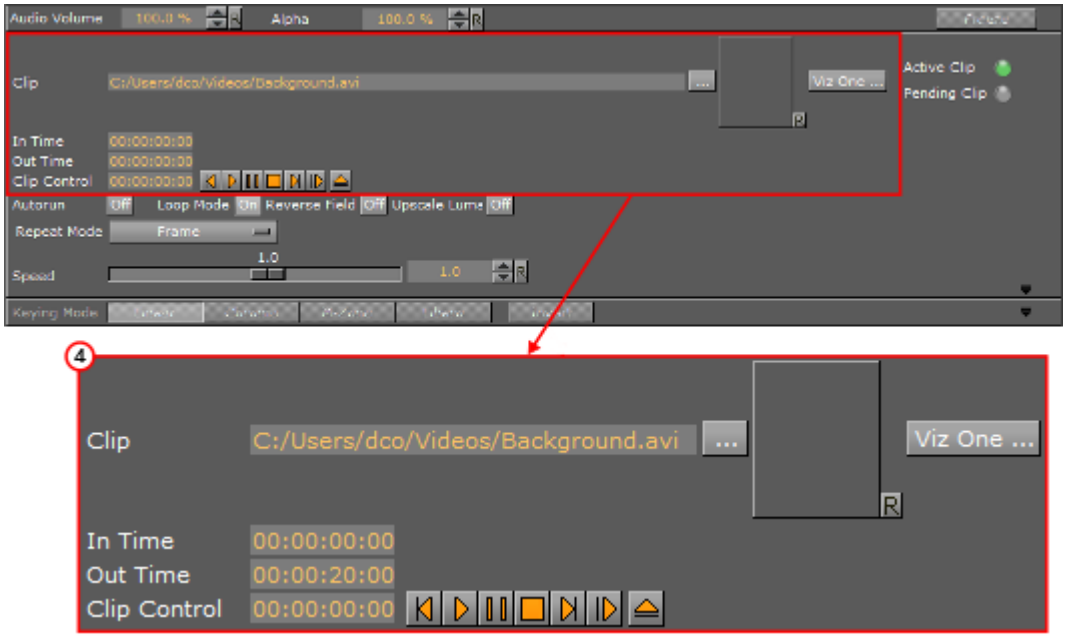
5. Click on the **Clip<1>** Channel to open its properties panel.
6. Adjust the parameters in the DVE control panel, as required.
7. Add a video clip:
   - In the **Clip Name** (5) line, click ... , or
   - Drag a Video Clip item from the Server Panel, or
   - Click Viz One ... , and select a video clip to import

   > ⚠ **Note:** To transfer and download videos from Viz One, Viz One must be integrated and configured with Viz Artist/Engine.

8. Configure the video clip playback as required.
9. Click (4 (Play)) to test the video clip.

> ⚠ **Note:** Matrox Card Only: A green 'traffic light' shows that the video clip has loaded successfully. A gray 'traffic light' shows that no video clip has been loaded. A red 'traffic light' shows that there was an error when the video clip was loaded.

> ⚠ **Note:** Click the expand button to show addition information. If the clip loaded successfully, clip details are shown. If the clip did not load successfully, additional errors are shown.

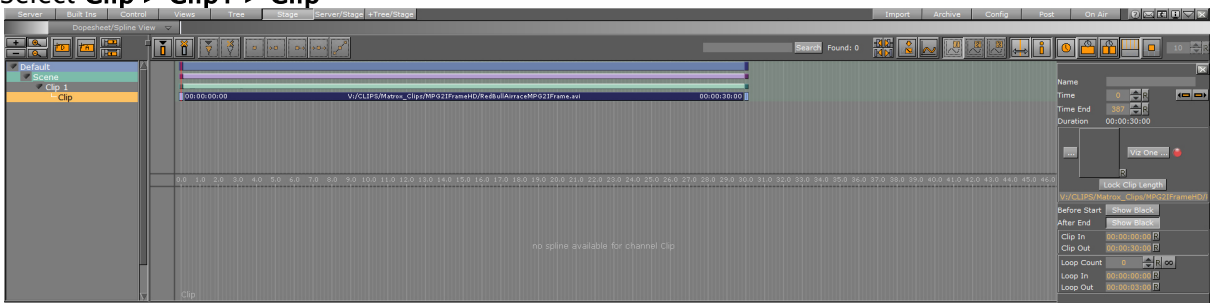


Video Clip Animation in the Stage

Video can also be played out as part of an animation. This is useful when video clips are required to interact with animation.

To Animate a Video Clip

1. Before a video clip can be animated, it must first be applied as a Texture or DVE. See:
   - To Play a Video Clip as Texture
   - To Play a Video Clip as DVE
2. Click **Stage**.
3. Click the New Director button.
4. Right-click on the Director.
5. Select **Clip > Clip1 > Clip**

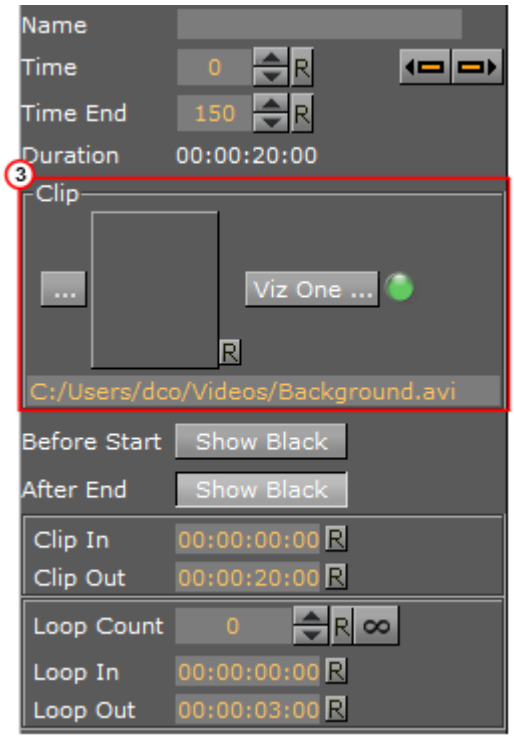


6. In the Clip1 channel, click in the dark blue area (2). The Clip Key Frame Editor opens on the far right.
7. Add a video clip (3):
   - In the **Clip Name** (5) line, click  , or
   - Drag a Video Clip item from the Server Panel, or

- Click Viz One ... , and select a video clip to import
- Click Lock Clip Length to specify a certain time/length for your clip. This is useful when loading clips from an external control application via Viz One. In this case, the lock clip length will act as a placeholder, but not change the clip length automatically after loading.

> ⚠ **Note:** To transfer and download videos from Viz One, Viz One must be integrated and configured with Viz Artist/Engine.

> ⚠ **Note:** Matrox Card Only: A green 'traffic light' shows that the video clip has loaded successfully. A gray 'traffic light' shows that no video clip has been loaded. A red 'traffic light' shows that there was an error when the video clip was loaded.



> ⚠ **Note:** This does not necessarily have to be the clip enabled in the previous procedures. It can be any video clip that Viz Artist can play.

> ⚠ **IMPORTANT!** If there is a conflict, the last clip selected, whether in the Stage or the Scene Settings, will be the one to play, with overriding effect.

8. Configure the Clip Key Frame Editor as required.

To Play a Single Clip in the Stage with Stage Loop

1. Set a clip in the stage as described in To Animate a Video Clip.
2. Click **Clip**, in the Stage Tree.
3. Set the clip to loop a specific number of times or infinite.

To Play Multiple Clips in One Channel

1. Add a clip to the stage as described in To Play a Single Clip in the Stage with Stage Loop.
2. Set the time indicator to a position after the first clip Key Frame
3. Click  to add a Key Frame.
4. Select a clip for the new Key Frame (see To Animate a Video Clip).
   Show black rules for pauses between clips:

Black Rules	Frames Shown
Both clips NS	Frame 1 of Clip 2 will be shown
Clip 1 AE, Clip 2 NS	Frame 1 of Clip 2 will be shown
Clip 1 NS, Clip 2 BS	Last frame of Clip 1 will be shown
Clip 1 AE, Clip 2 BS	Black will be shown
Key:NS: Nothing setBS: Show black before start setAE: Show black after end set	

If **Pending** is Active (which is recommended) subsequent clips are loaded into the pending clip player and are swapped at the correct time. This feature enables the clips to concatenate seamlessly. However, when pending support is set to Inactive, black frames will be shown until the clip is ready to be played (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

To Play Multiple Clips in One Channel with Stage Loop

1. Set multiple clips in the stage as described in To Play Multiple Clips in One Channel.
2. Set the clips to loop. Click the corresponding entry in the Stage Tree.
   All Key Frames including the pauses in between and before will be taken into account during play back, for example, pause - clip 1- pause - clip 2 will loop as such.

To Play Single and/or Multiple Clips in Multiple Channels

This is a combination of the use cases described above, with the clips used on several channels.

Video Clip Queue

Video clips can also be queued, providing automatic playback of a series of clips. If no clip is playing at the moment of clip queue creation, playback of the first clip in the queue will start immediately. All other clips are played in the sequence they were added to the list. By default, every clip needs to be played out manually, although it is not required to set their names again. This behavior can be overridden by clicking the **Autorun** button, setting it to **On**.

The current state of the auto-play flag is stored with every queued clip, and can thus be used on demand.

Commands (n is the clip channel):

Video Clip Queue Commands

Command	Description
`[...]*CLIPIN*n*NAME CUE name in out`	Adds clip with `name` to the queue, using `in` and `out` as trim in and out points. The parameters `in` and `out` are optional and can be omitted.
`[...]*CLIPIN*n*QUEUE LIST`	Lists all the clips currently queued.
`[...]*CLIPIN*n*QUEUE INSERT name in out position`	Inserts clip name using `in` and `out` at `position` in the queue.
`[...]*CLIPIN*n*QUEUE REPLACE name in out position`	Replaces the clip at `position` with the new one using `name`, `in` and `out`.
`[...]*CLIPIN*n*QUEUE DELETE position`	Deletes the clip at `position` in the queue.

Command	Description
`[...]*CLIPIN*n*QUEUE MOVE source target`	Moves the clip from position `source` to position `target` in the queue.
`[...]*CLIPIN*n*QUEUE FLUSH`	Removes any and all clips from the play-list.

Example Clip Queue commands:

1. `RENDERER*VIDEO*CLIPIN*n*NAME CUE`
2. `MANI_SCENE*VIDEO*CLIPIN*n*NAME CUE`

See Also

- For Dual Channel and Trio OneBox set up see Dual Channel and Trio One Box Configuration.
- Supported Codecs

7.5.4 Video Clip Playback Considerations

There are two methods to playback Video Clips:

- Playback directly in a Clip Channel (see Video Clip Playback)
- Playback as a Key Frame in the Stage (see Video Clip Animation in the Stage)

In both methods the Media Assets all share the same available Clip Channels (1 to 16).

A Clip Channel can only have one video clip applied to it. If Clip Channel1 is applied to a Container, and is also applied to a Director in the Stage, the same video clip will be shown in the Container and Stage, and the whichever is set last will have priority. When the same Clip Channel is used in both the Scene Settings and the Stage, the main problem is that, whatever is changed in one place is reflected in the other. For example, when the out time of the clip is changed in the Scene Settings (which means the content of the player itself are changed) these changes affect the clip playback in the Stage as well, because it is the same player. In most if not all cases you do not want this to happen. To have Clip Channels in both the Scene Tree and the Stage use two different Clip channels (i.e. Clip1 and Clip2), or do not use both methods in one Scene.

Layers

All Scenes, in all layers, also share the same Clip Channels.

If a Scene is loaded in the Main Layer with a video clip on Clip Channel 3, and a Scene is also loaded to the Back Layer with a different video clip, but also on Clip3, both Scenes/Layers will play the last loaded video clip. The one from the Back Layer in this example.

Clip Players

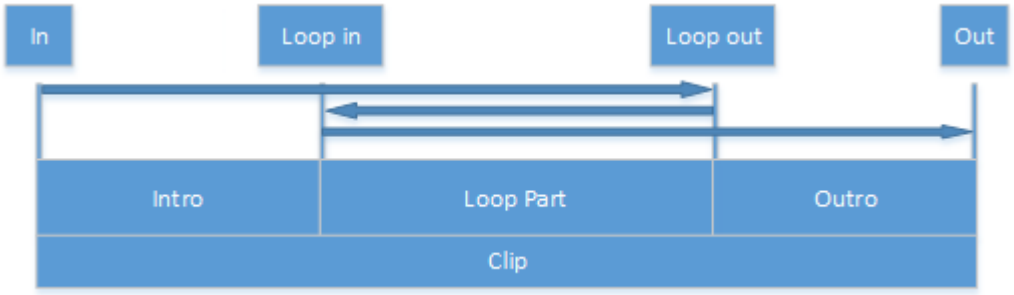
Each Clip Channel holds at least one Clip Player. If pending Clip Player support is enabled for the Clip Channel, an additional Clip Player is used for play back. Every Clip Player uses up additional system memory. But seamless transition from one video clip to another Video clip is only possible with pending Clip Player support enabled.

This section contains the following topics:

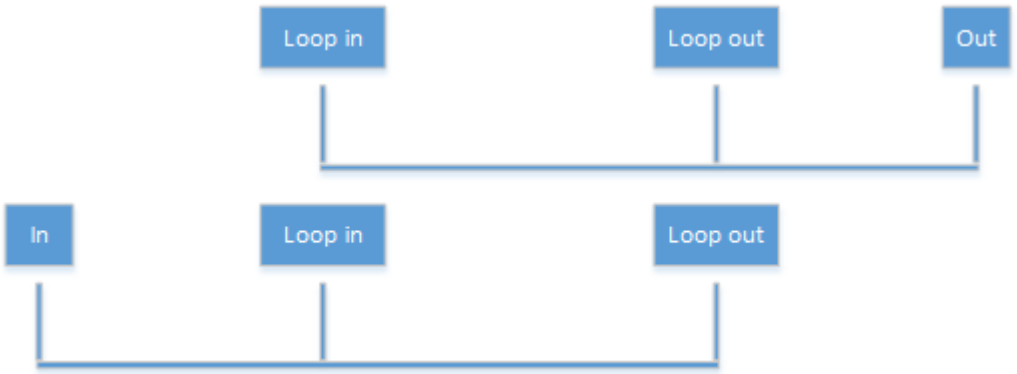
- Layers
- Clip Players
  - 3 and 4-Point Loops in Clip Player
  - Example: 4-Point Loop Use
  - Parameters for 4-Point Loops in the Stage Editor
- General, Mixed and Special Modes
- Video Clip Playback Use Cases
- Clip Handling
- Settings
- Scripting and Plug-in Interface
- IO Matrix

3 and 4-Point Loops in Clip Player

A 4-Point Loop describes a clip that has at least a loop part and an intro or an outro or both. To define a 4-Point Loop, the loop in and out time code and the number of times the loop part should be repeated, possibly infinite, are required.



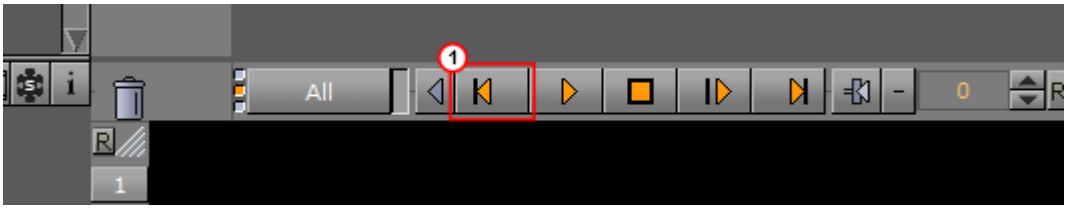
⚠ **Note:** A 3-Point Loop is a clip which has a loop part and only an intro or an outro (see below).



The 4-Point Loop features are only supported on video version of Viz Artist and require Matrox hardware to work, i.e. X.mio, X.open or DSX LE.

⚠ **Note:** Other supported video hardware use the DirectShow clip player and do not support 4-point-loop clips.

⚠ **IMPORTANT!** The two clip players need to be configured correctly for the 4-Point Loop to work correctly (the animation in the Stage must be set to the beginning). Click on the **Jump to beginning of animation** button (1) to set the two clip players, to the beginning of the animation.



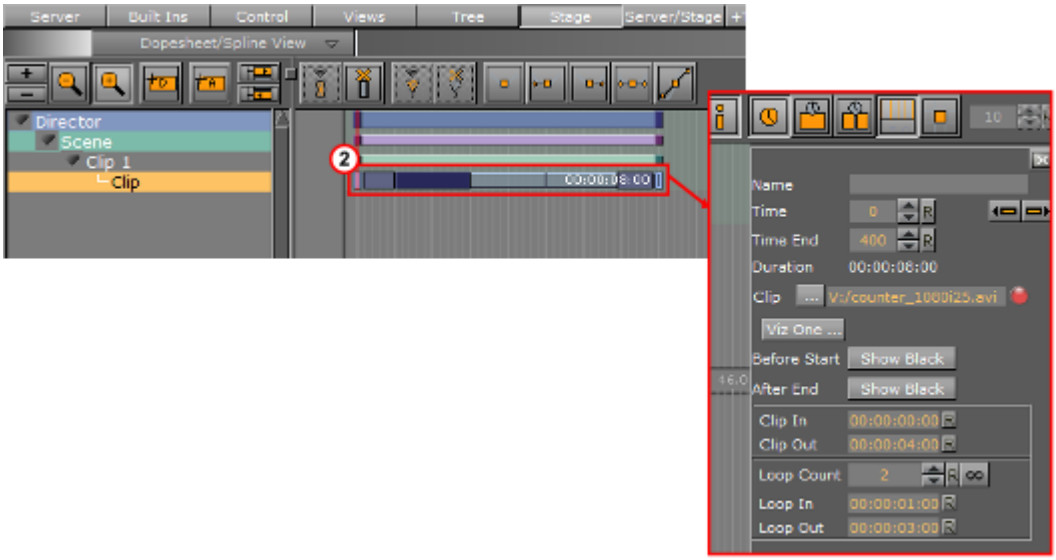
Example: 4-Point Loop Use

4-Point Loops can useful in a wide variety of cases where you need to loop a video segment. Perhaps you do not know in advance exactly how many times you will want to loop the video. For example, let us assume the video clip is a spinning globe animation. You know you have an intro, but how long the globe should spin, is dependent of the time another scene item is shown, say a video of the anchor speaking. When the anchor-video is done, you will want the spinning globe to exit the scene.

Parameters for 4-Point Loops in the Stage Editor

Additional fields have been added to the clip setting pane where loop in, loop out and loop count can be set as shown in the image below. No additional command is needed to start or stop the loop.

Click on the Clip channel (2) to view the 4-Point Loop parameters.



During scene-design the values of the 4-Point Loop are recalculated automatically when the Key Frames are changed by dragging them in the director tree. Any changes made in the entry fields will be reflected in the time line pane to show the real-time length of the clip.

When the start or the end frame of the clip is dragged it will snap to the nearest possible time code to have an integer loop count.

General, Mixed and Special Modes

This section contains the following topics:

- Play Clips with Resolutions Different from Channel
- Frame Accurate Clip Playback

Play Clips with Resolutions Different from Channel

It is possible to play clips even if the resolution is different from the one set for the channel. However, there are some restrictions to that:

- Can only play clips from the same genlock family
- The clip's resolution must be of a regular broadcast format
  For detailed information on which combinations are possible check the appendix.

Frame Accurate Clip Playback

If the clip, in the stage, needs to be played frame accurately, these features need to be set to active:

- First, the pending clip playback support has to active. If not the clips will not be ready when needed (see the **Video Input: Clip Input** page in the **Configuring Viz** section of the Viz Engine Administrator Guide).
- Additionally the Frame Accurate Viz Commands (FAVC) need to be set to active in **Global Properties** (see the **Communication** page in the **Configuring Viz** section of the Viz Engine Administrator Guide).

- And the Command Execution Field Dominance should be set to **Odd Retrace Counter** (see Communication -> Global Properties in Viz Configuration)
  When using external control applications it may be necessary to tweak the FAVC Bias.

For a scene containing a clip in the stage to play frame accurately it is necessary to use the `SCENE*CUE` command, which preloads the first clip Key Frame.

Video Clip Playback Use Cases

With the Clip Channel Media Asset it is possible to play images as DVE or Texture with the Matrox clip player. Supported image containers are jpg and png. The images have to be available on a physical disk for this feature to work. One advantage of an image played as a clip, is that the same features available for a video clip can be used on the image.

This section contains information on the following common clip playback use cases:

- - Playing Overlay Clips (Not Full Screen)
  - Clip with Audio (Not Full Screen)
  - Clip with Audio (Full Screen)
  - Clip with Audio (Full Screen and 3D Space Transformation)

Playing Overlay Clips (Not Full Screen)

Playing a clip which is not full screen in size, like a spinning logo, an insert snipe or a special effect without audio. These are known as overlay clips. To play overlay clips it is recommended to use the SoftClip clip player. Even though SoftClip can play clips up to HD 1080, it should be mainly used for overlay clip playback. For full screen playback there is a more efficient clip player available in Viz. The advantage of the SoftClip clip player is that it can play clips in any pixel size, aspect ratio or frame rate. SoftClip is based on the VFW (Video for Windows) AVI clip playback workflow on Windows.
SoftClip is able to play all installed VFW (Video For Windows/AVI) codecs on your computer. There are dozens of codecs available, each with it's own advantages and drawbacks. There is no general rule which codec to use (see Supported Codecs), but it is recommended to use the different types of Matrox codecs (depending on whether the alpha channel is necessary or not), avi uncompressed or the freely available HuffYuv codec. SoftClip supports RGBA playback if the codec is able to store the alpha channel; avi uncompressed supports alpha, also the Matrox Mpg2iFrame&alpha codec and the HuffYuv codec, just to mention a few.

A typical use case for the SoftClip player would be:

- The playback of a spinning looping logo with alpha channel. The size of this clip could be around 350x350 pixels
- The playback of an explosion overlay clip with alpha. The size is around 500x500 pixels
- The playback of a snipe/insert with alpha. The size is around 600x150 pixels

> ⚠ **Note:** The SoftClip player is not intended for full screen playback.

Clip with Audio (Not Full Screen)

Playing a clip with audio which is not full screen in size, like a strap of an advertiser, a 3D object transition or an effect or a reveal of an item only.

To play clips which need audio, but do not need to be played back in full screen mode, use MoViz. MoViz is very similar to SoftClip, with the main difference being that it supports audio. MoViz is based on the DirectX clip playback workflow on Windows, which means it uses a different library and codec filters than the SoftClip plug-in.

If you install a VFW codec for avi playback like for SoftClip, it is not guaranteed that your clip will also play in MoViz, since MoViz is DirectX-based. In general it is very helpful to install the free package of the main concept codecs, since with this package you get dozens of ffdshow filters for DirectX for free and this will raise the compatibility with which you will be able to play regular avi files via MoViz.

To use the full potential of MoViz you should encode clips with ffd instead of avi. If you want to render clips into ffd, as an example from Adobe After Effects, there will be a new section showing up in the render section named ffd, besides QuickTime and Video for Windows, TGA single frames and others. In the ffd settings you can select the necessary format, like dv, mpg and many others.

A typical use case for the MoViz player would be:

- The playback of a insert advertiser with audio on the bottom of your screen. The size of this clip would be something around 720x150 pixels.
- The playback of a 'clip universe': the animated Viz camera is flying through space and passes clips.

> ⚠ **Note:** The MoViz player is not intended for full screen playback!

Clip with Audio (Full Screen)

Playing a clip with audio in full screen mode, but only using it as a background in my Viz scene. The clip does not need to be rotated in 3D space.

To play full screen background clips it is recommended to use the Matrox clip channels. The dimensions and frame rate of these clips must be either HD 1080, HD 720, PAL or NTSC. Any free pixel size, aspect ratio or frame rate is not supported on the Matrox channels.

If the clip is 2D, meaning there is no transformation/rotation in 3D space needed, then you should set the Matrox clip channel to DVE. In DVE mode you get the maximal clip quality achievable.

When using DVE mode, do not forget to apply a key function to your Viz items or switch the renderer to autokey, otherwise they will not show in front of the clip (see Keying Mode).

If you want to use an alpha channel on this clip, use the Matrox Mpg2iFrame&alpha codec and configure the clip channels in Viz Configuration (see the Viz Engine Administrator Guide) correctly (activate alpha). Also here it is recommended that you use the Matrox AVI codecs, but the Clip channels support many more formats than just the Matrox codecs (see Supported Codecs).

A typical use case for the DVE setting would be:

- The playback of a clip at it's best quality, with or without any overlaying graphics.

> ⚠ **Note:** Using the Matrox clip channels for playback is the most efficient clip playback for full screen clips.

Clip with Audio (Full Screen and 3D Space Transformation)

Playing a clip with audio in full screen mode and transforming it in 3D space.

To play full screen clips it is recommended that you use the Matrox clip channels. The dimensions and frame rate of these clips must be either HD 1080, HD 720, PAL or NTSC. Any free pixel size, aspect ratio or frame rate is not supported on the Matrox channels.

To be able to transform the clip in 3D space you must set the clip channel to Texture. Compared to DVE mode, the clip is slightly more blurred, but this is essential to avoid noise on the clip texture when moved or rotated in 3D space.

If you want to use an alpha channel on this clip, use the Matrox Mpg2iFrame&alpha codec and configure your clip channels in Viz Configuration (see the Viz Engine Administrator Guide) correctly (activate alpha). Also here it is recommended that you use the Matrox AVI codecs, but the Clip channels support many more formats than just the Matrox codecs (see Supported Codecs).

A typical use case for the Texture setting would be:

- The playback of a clip in 3D space with transformations on rotation, scaling and position.
- The playback of a clip on a 3D object.

> ⚠ **Note:** Using the Matrox clip channels for playback is the most efficient clip playback for full screen clips.

Clip Handling

This section contains the following topics:

- Slow Motion Playback
- Fast Forward and Fast Backward
- Flushing Clip Channels

Slow Motion Playback

A shuttle rate between -1.0 and 1.0 will result in the clip playing slower than normal, where negative values result in a backwards playback.

The actual frame to play is determined by the internal frame counter which is increased on every grabbed frame.

For a frame rate of 0.3 this would be:

Internal Counter	Actual Frame	Internal Counter	Actual Frame
Internal 0	Frame 0	Internal 0.3	Frame 0
Internal 0.6	Frame 0	Internal 0.9	Frame 0
Internal 1.2	Frame 1	Internal 1.5	Frame 1

Internal Counter	Actual Frame	Internal Counter	Actual Frame
Internal 1.8	Frame 1	Internal 2.1	Frame 2

Fast Forward and Fast Backward

A shuttle rate <-1.0 or >1.0 results in the clip playing faster than normal, where negative values result in a backwards playback.

The calculation of the next Frame to be delivered is done the same way as described above using the set shuttle rate.

Flushing Clip Channels

The FLUSH command can be used to clear the active clip player and reset all parameters to their default value. All valid clip frames still in the pipeline will be cleared as well and black will be sent.

If the pending clip player also needs to be cleared, send FLUSH to the PENDING command channel.

Settings

This section contains the following topics:

- Repeat Mode
- Reverse Fields
- Loop Mode
- Auto Run

Repeat Mode

Determines how clip playback should behave when paused:

- **None**: Black will be played out
- **Frame**: The last frame is shown, which shows both fields
- **Field**: Only the last field is shown, copied to first and second field
  This setting is only relevant if the clip's scan mode is interlaced.

Reverse Fields

This setting is used for smooth playback when playing interlaced clips in reverse. Enabling reverse fields swaps the playback of first and second field during reverse playback.

This setting applies only to interlaced clips.

Loop Mode

If on, the clip loops infinitely between clip in and out times.

Auto Run

If on, the clips start immediately after being loaded and delivering the first frame.

Scripting and Plug-in Interface

This section contains the following topics:

- Scripting
- Plug-ins

Scripting

The scripting interface refers to the plug-in interface and has therefore the same limitations as the plug-ins have.

To use a clip channel in scripting, first retrieve a channel from the system:

```
oClipChannel as ClipChannel = system.GetClipChannel( iChannel )
```

This object can now be used to control the clip channel:

```
oClipChannel.SetClipName( "clip.avi" )
oClipChannel.LoopMode = true
oClipChannel.Play(0)
```

The scripting interface to clip channel allows for the same use cases as the plug-in interface does.

Plug-ins

The plug-in interface provides almost all functions available in the Command Interface. The only limitation here is, all functions refer to the current scene. The use cases are the same as described in 3 and 4-Point Loops in Clip Player. Stage use cases cannot be addressed with the plug-in interface.

IO Matrix

This section contains the following topics:

- Source PAL
- Source 720p
- Source 1080i

Source PAL

Output DVE	Video In				Output Texture	Video In		
	PAL	720p	1080i			PAL	720p	1080i
PAL	+	-	-		PAL	+	-	-
720p	-	-	-		720p	-	-	-
1080i	+	-	-		1080i	+	-	-

Output DVE	Clip In				Output Texture	Clip In		
	PAL	720p	1080i			PAL	720p	1080i
PAL	+	-	-		PAL	+	-	-
720p	-	+	-		720p	+	+	-
1080i	+	+	+		1080i	+	+	+

Source 720p

Output DVE	Video In				Output Texture	Video In		
	PAL	720p	1080i			PAL	720p	1080i
PAL	-	-	-		PAL	-	+	-
720p	-	+	-		720p	-	+	-
1080i	-	+	-		1080i	-	+	-

Output DVE	Clip In				Output Texture	Clip In		
	PAL	720p	1080i			PAL	720p	1080i
PAL	+	-	-		PAL	+	+	-
720p	-	+	-		720p	+	+	-
1080i	+	+	+		1080i	+	+	+

Source 1080i

Output DVE	Video In				Output Texture	Video In		
	PAL	720p	1080i			PAL	720p	1080i
PAL	-	-	-		PAL	-	-	+
720p	-	-	-		720p	-	-	+
1080i	-	-	+		1080i	-	-	+

Output DVE	Clip In				Output Texture	Clip In		
	PAL	720p	1080i			PAL	720p	1080i
PAL	+	-	-		PAL	+	-	+
720p	-	+	-		720p	+	+	+
1080i	+	+	+		1080i	+	+	+

See Also

- The Stage for Animation
- Key Frame Editors
- Channel Editor

7.5.5 Transfer Clips From Viz One

This section details how to transfer and download video clips from a configured Viz One.

To transfer and download a video clip from Viz One, Viz One must be integrated and configured with Viz Artist/Engine (see the **Viz One** and **Authentication** pages in the **Configuring Viz** section of the Viz Engine Administrator Guide and the **Viz One Deployment Guide**).

To Transfer Viz One Clips

1. Do the procedure To Play a Video Clip as Texture or To Play a Video Clip as DVE. When, in the procedure, a video clip is to be added click .
2. In the **Viz One Uri** drop down box, select a Viz One to import from.
3. Enter a search parameter (Viz One files are not shown until an entry is made).
4. Locate the required file. A preview of the clip is shown in the browser window.
5. Click **Transfer**. The video clip is transferred from Viz One and downloaded to the local video clip library (see Hard Drive Video Clips).

During transfer, the **Transfer** button (5) in the **Viz One Browser** will be inactive. To add several clips from Viz One simultaneously, add more Clip Channels. When correctly configured, the traffic light icon to the right of the drop-down menu will be green (2). If the Viz One integration has been incorrectly configured, the tool-tip for the traffic light icon will display a problem description.

When set to **On**, the transfer status for each clip channel will show as a progress bar in the **Viz One Transfer Status** pane. This pane's visibility depends on the **Mediaftp (File Transfer)** and **Fsmon (File System Monitor)** services being installed and running, and that the **Viz One (MessageQueue Server)** is correctly configured. Please refer to the Viz Engine Administrator Guide for detailed information on how to configure and install these services.

7.6 Keying Mode

There are four types of keying modes built into Viz Artist, which can be used with Media Assets applied as Texture:

- Linear Key
- Chroma Key
- M-zone Key
- Libero Key

7.6.1 Linear Key

Linear key is the standard keying technique, and can be used for most applications. This is a logical key (enabled by the Key plug-in):

- If a video or clip channel is applied to an object it is treated the same way as if it would be a static image.

- If it is used as a background image it is the first item to be drawn and all objects in the scene are drawn on top of it regardless of the key settings in the global section of scene settings.
- If it is set as the foreground image it will be the last item to be drawn and all other objects will only be visible in the parts of the screen where the key/alpha of the video is not 100% (opaque)

7.6.2 Chroma Key

Chroma key is mostly used with a virtual studio.

In chroma keying mode the key/alpha channel for the video source is generated for each field or frame. When used in a virtual studio, the video source should be dragged into the foreground image. This way items in the scenes without a key will be drawn first followed by the video source and last items with key information. This drawing order allows for semi transparent objects to be drawn in front of the talent.

The chroma keyer can also be used to cutout people or objects which were filmed in front of a blue or green screen for head shots or product placement. For sports applications where only one color needs to be keyed out the chroma keyer can also be used.

⚠ **Note:** Each input channel has a freely configurable chroma keyer.

7.6.3 M-zone Key

M-zone key is often used in outdoor sporting events broadcasts.

The multi zone keyer prepares the input video to be keyable by the chroma keyer plug-ins in the scene tree. It is typically used in sports productions where more than one color needs to be keyed out. The video source needs to be dragged into the background image in global settings. An unlimited number of Chroma Keyer plug-ins can be created in the scene, each keying out a different color.

7.6.4 Libero Key

If Viz Libero Clips are to be played, the Keyer must be set to Libero. This way the chroma key settings from the Viz Libero Clip will be used. Make sure that the clip comes with the Camera ID (`.cid`), Tracking Data (`.trk`) and Chroma Key Settings(`.col`) files.

7.6.5 Keying Best Practices

This section contains information on some best practices when Keying.

- **Take advantage of the self-configuration**
  Before you start to play with the different parameters, let the system configure by itself. Select a section of the image, which contains in best case all color tones, you want to key. If

you have no foreground objects, you can set the whole input video as section for the self-configuration.

- **Modify the keying color**
  Put your real foreground objects into the Scene. Mostly your objects will still have a blue edge after the self-configuration. To eliminate it, increase the Outer and Inner UV-Radius. When increasing the Inner UV-Radius have a look at opaque foreground objects in the video. It is perfectly configured if it cuts off also a little bit of the object. Have a look at tiny things, like hair or transparent objects and modify the Outer UV-Radius to get a smooth fade out at the edges of these objects.
- **Desaturate the blue/green edges**
  Up till now you have probably had tiny edges around people or objects which are close to the background color. If you enable color correction by setting it to blue or green, the color will be replaced.
- **Add highlights and shadows**
  By modifying the add highlights and add shadows, borders of objects will show, that you have keyed before by setting the inner and outer radius.
- **Play around with it**
  You can either try to improve the keyer by increasing or decreasing the values, or you just play around with it.

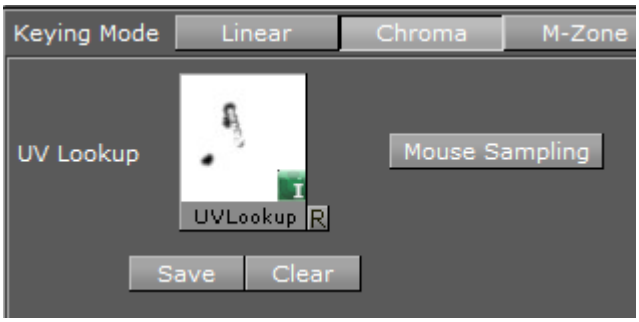
7.6.6 Keying Mode Configuration

The Keying mode parameters are split into six panels.

This section contains the following topics and procedures:

- UV Lookup
- U-Color/V-Color
- Color Correction
- Restore Dark Colors/Shadows
- Restore Bright Colors
- Sample
- To Configure Keying Mode

UV Lookup



- **UV Lookup:** Drag a UV Look up image to the place holder. Click R to reset the UV Lookup image box

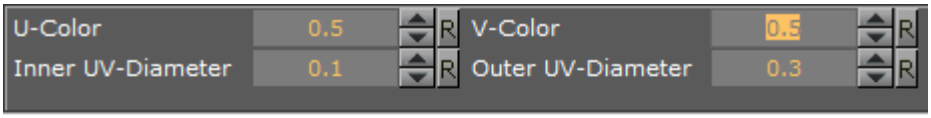
  ⚠ **Note:** A UV lookup image is a luminance image of size 256x256.

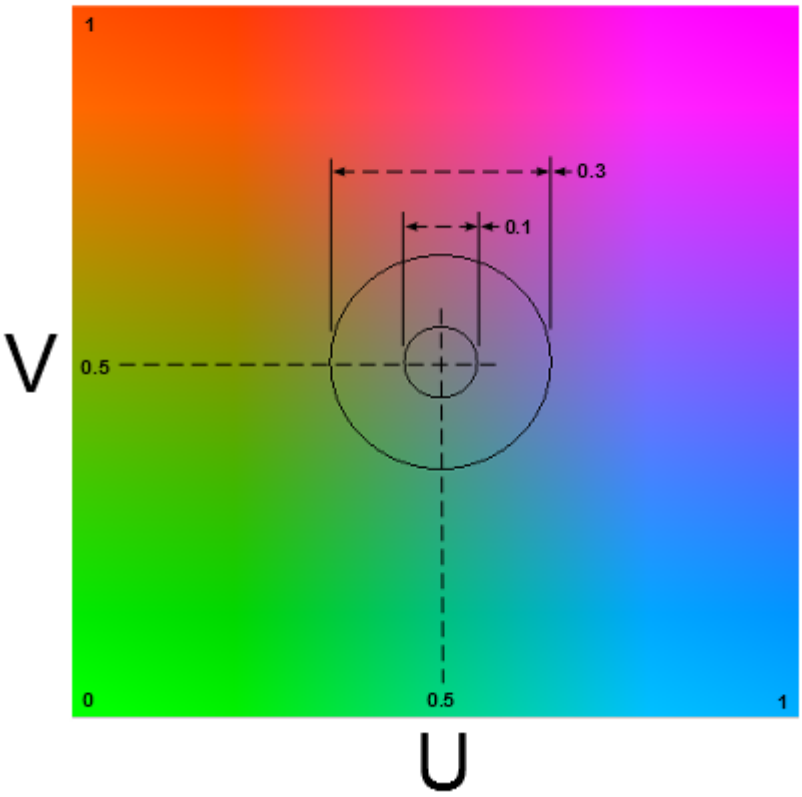
- **Mouse Sampling:** Collects the chroma information out of a video signal, at the position of the mouse pointer.
  Hold the left mouse button and move the mouse around to add the colors at the mouse pointer location to the chroma key table. To remove the colors from the chroma key table, hold down <Alt> while repeating this procedure.

  ⚠ **IMPORTANT!** Chroma Key picking is not supported if the same input source is used in different scenes loaded as layers within a scene, or if **GPU Direct** is enabled in Viz Config.

- **Save:** Saves a modified UV LookUp image
- **Clear:** Removes the current UV LookUp texture from the current scene (no colors are removed from the image)

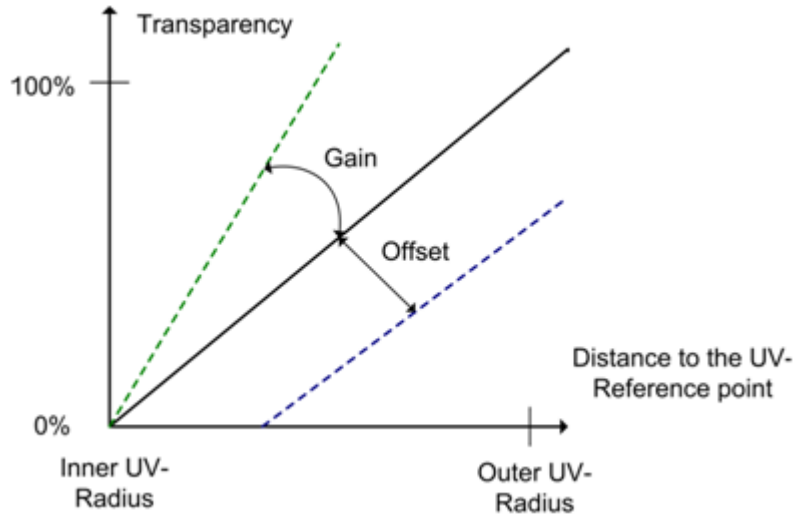
U-Color/V-Color





- **U-Color** and **V-Color**: Sets the background color in the YUV color space. The point set by these two parameters is the reference for the next parameters, which determine the colors to remove.
- **Inner UV-Diameter/ Outer UV-Diameter:** Removes all colors between the point defined by the **U and V-Color** and the **Inner UV-Diameter** completely. For all colors between the **Inner UV-Diameter** and the **Outer UV-Diameter**, the transparency will be set to a value between 0%

and 100%.



Color Correction

Color Correction is especially helpful to remove chroma bleed.

- **Color Correction**: Blue or green colors can be removed completely from the image:
  - **Off:** Turns off Color Correction.
  - **Blue:** Activates the parameters for **Cyan** and **MagentaGain**.
  - **Green:** Activates the parameters for **Cyan** and **YellowGain**.
    High **Gain** values for all these parameters mean that nearly all tones of this color will stay unchanged. Low **Gain** values signify that this tone will be removed completely from the image.
- **Brightness**: Sets the gain value of the Chroma Key area as required, in relation to the background. Useful, for instance, for tuning the video going through the chroma keyer if the light in the studio changes in a virtual studio setup. Value 1 means keep the original intensity, below 1 means darker, above 1 brightens.

Restore Dark Colors/Shadows

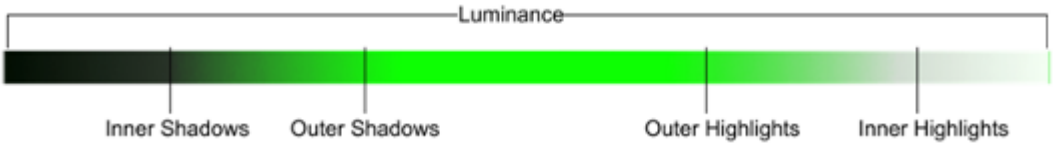
- **Restore Dark Colors/Shadows**: When set to **Active** all pixels with a luminance lower than complete restoration to the key/alpha will be make the pixel 100% opaque.
  A pixel with a luminance between **Complete Restoration to** and **Zero Restoration from** the key value is interpolated between 0 and 100%. This key is compared with the results of the Chroma Key and the higher value will be used. The key value of a pixel where the luminance is higher than **Zero Restoration from** will stay unchanged. **Gain** can be used to darken dark

colors even more, if it is set below 1.0. **Gain** is only applied to pixels with luminance between 0 and **Zero Restoration To**.

> ⚠ **Tip:Gain** is useful in a virtual studio, where the set design uses dark colors in a shadow, but when seen through the camera the shadow is lighter, in color, than the floor. A darker shadow will make the final composition more believable.

Restore Bright Colors

- **Restore Bright Colors**: When set to **Active** all pixels with a luminance higher than complete restoration to the key/alpha will be making the pixel 100% opaque.
  For a pixel with a luminance between **Complete Restoration to** and **Zero Restoration from**, the key value is interpolated between 0 and 100%. This key is compared with the results of the Chroma Key and the higher value will be used. The key value of a pixel where the luminance is lower than **Zero Restoration from** will stay unchanged. **Gain** can be used to make bright colors show even brighter. **Gain** is only applied to pixels with a luminance between 1 and **Zero Restoration To**.



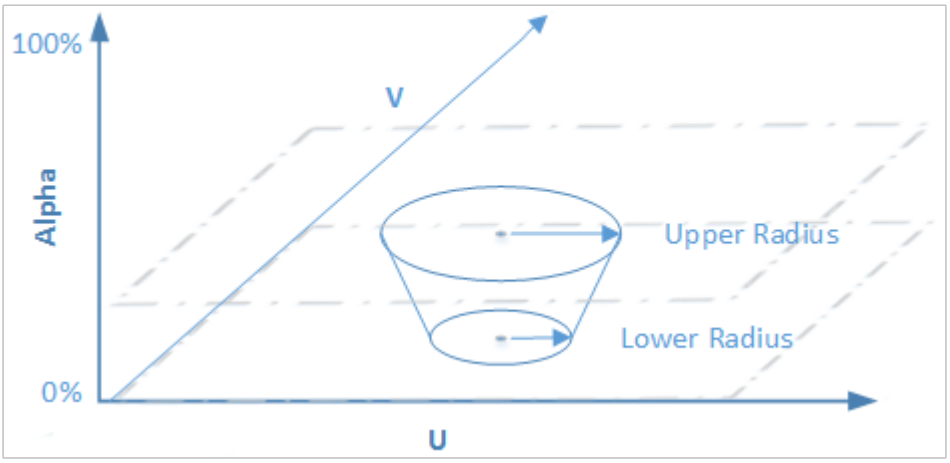
Sample

The **Sample** functions define a section in the video input, which can make the Chroma Key results more accurate.

Press **Sample Now** to start the self-configuration.

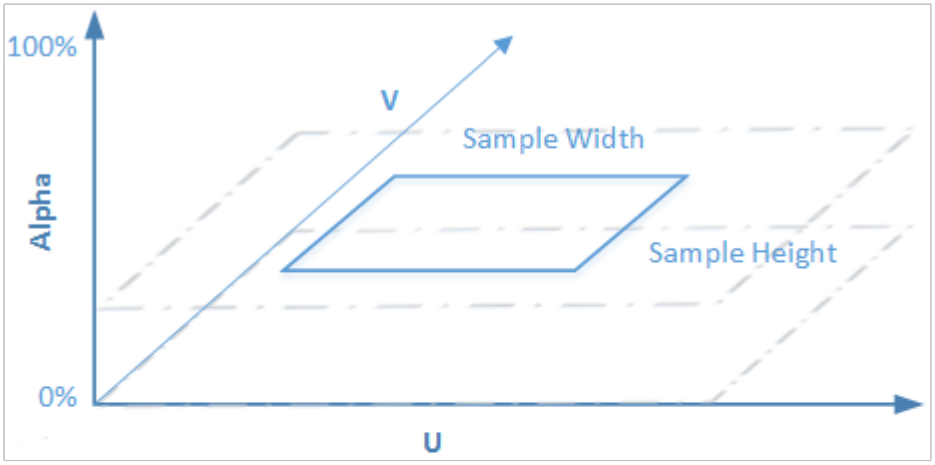
- **Sample With UV Lookup**:
  - Sets the **Sample X/Y** coordinates: The x/y position on the screen to select the picking color. Not used.

- Sets the **Upper Radius/Lower Radius**.



- **Sample Without UV Lookup**:
  - Sets the **Sample X/Y** coordinates: The x/y position on the screen to select the picking color.

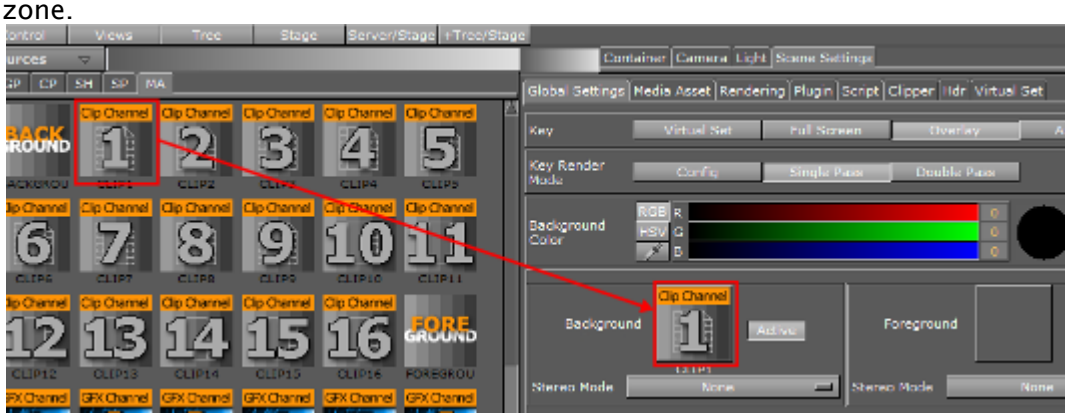
- Sets the **Sample Width/Height**.



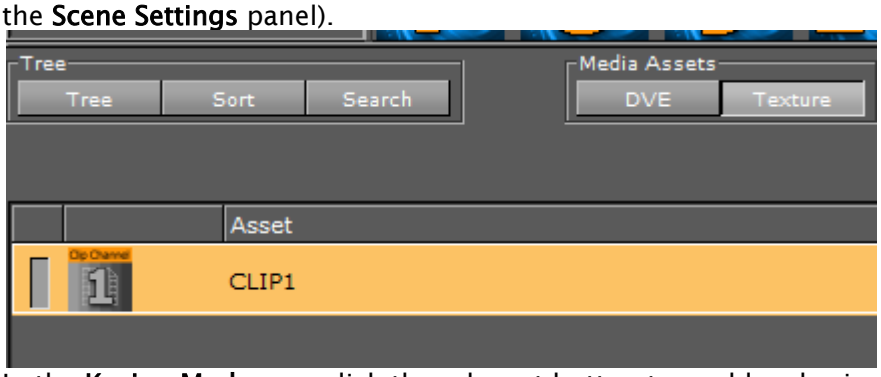
To Configure Keying Mode

1. Click **Built Ins**.
2. Click the **Media Assets** tab.
3. In the Properties panel click the **Scene Settings** tab
4. Click the **Global Settings** tab.

5. Select a Media Asset and drag it onto the **Background Image** or **Foreground Image** drop-zone.



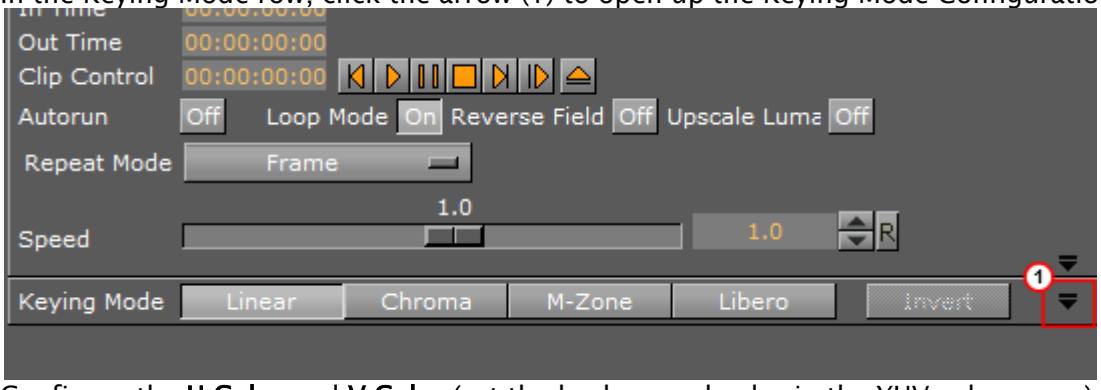
6. Click on the Media Asset in the Media Asset Manager (this opens the **Media Asset** tab in the **Scene Settings** panel).



7. In the **Keying Mode** row, click the relevant button to enable a keying method. Options are:
   - Linear Key
   - Chroma Key
   - M-zone Key
   - Libero Key

> ⚠ **Note:** To use the M-zone key, the Chroma Keyer plug-in must also be used. Drag the Chroma Keyer plug-in to the container which holds the video texture in the Scene Tree.

8. In the Keying Mode row, click the arrow (1) to open up the Keying Mode Configuration.



9. Configure the **U-Color** and **V-Color** (set the background color in the YUV color space). The point set by these two parameters is the reference for the next parameters, which determine the colors to remove.

All colors between the point defined by the **U-Color** and **V-Color** and the **Inner UV-Diameter** will be removed completely, and all colors between the **Inner UV-Diameter** and the **Outer UV-Diameter** the transparency will be set to a value between 0% and 100%.

10. Configure the **Sample X/Y** coordinates and the **Sample Width/Height** (**Upper Radius/Lower Radius** with UV Lookup). These parameters define a section of the background in the video-input. Press **Sample Now** to start the self-configuration.
11. Configure **Color Correction**. This can help to completely remove the blue or green colors from the image.
12. Configure the Luminance Control:
    - **Restore Dark Colors/Shadows**: All pixels with a luminance lower than complete restoration to the key/alpha will be make the pixel 100% opaque.
    - **Restore Bright Colors**: All pixels with a luminance higher than **Complete Restoration to** the key/alpha will be making the pixel 100% opaque.

7.7 Seamless Input Channel Switcher

This feature works as a software enabled video and audio crossbar (a coordinated switching system). In earlier versions of Viz Engine every video channel was bound to one physical video in a channel. The use of another physical input was only possible if the setting in the Viz Configuration was changed.

Seamless input channel switching changes this binding seamlessly, even during play out. It is also possible to use more inputs than the number of licensed channels, for example, if a video card is equipped with an X.RIO card. However, only the number of licensed channels can be used simultaneously. At the moment the switching is only possible with commands.

⚠ **Note:** Seamless switching is available for X.mio, X.mio2 and X.RIO cards

The initial setting is defined by the configuration. In run time change the used video input channel with the commands listed here:

- **Change the Input Channel**
  The command, `RENDERER*VIDEO*VIDEOIN*1*LEASE SET B`, sets video channel 1 to use physical input B as its source. If used during play out, the change happens at the next frame. If you try to lease an invalid channel or a channel that is currently used by another Viz Engine input, you will get Invalid as a result.
- **Check the Input Channel**
  The command, `RENDERER*VIDEO*VIDEOIN*2*LEASE GET`, checks which physical input is currently mapped to Viz Engine's channel 2.
- **Release the Input Channel**
  The command, `RENDERER*VIDEO*VIDEOIN*1*LEASE RELEASE`, releases the currently captured physical input channel.
- **Get the Free Physical Inputs**
  The command, `RENDERER*VIDEO*VIDEOIN*3*FREE_DEVICES GET`, lists the free physical inputs as a comma separated list (e.g. A, C, E, F).

7.8 Supported Codecs

This section details the supported Codecs for all supported Video Cards used by Vizrt.

7.8.1 Matrox Codecs

The Matrox X.mio series cards all support a selection of different codecs for both SD and HD. Certain codecs require certain card classes. Depending on the card class, a license upgrade may be required to extend the range of codecs supported by the installed video card. The Matrox video cards have a built-in license dongle used by the Matrox codecs to determine licensing rights.

- For the Matrox X.mio, X.mio2 and X.mio2 Plus video cards, the following classes apply:
  - 6000 class: SD codecs only. No D10
  - 8000 class: SD, HD, D10 and D12 codecs
  - 8500 class: SD, HD, D10 and D12 codecs, and Apple ProRes playback capabilities
- For the Matrox X.mio3 and DSX LE 4 video cards, the following classes apply:
  - 100 class: No clip playback
  - 500 class: SD and HD clip playback
  - 550 class: SD and HD clip playback, including support for Apple ProRes codecs

  All codecs are implemented in the software.

7.8.2 QuickTime Codecs for the Viz Artist/Engine 64-bit Platform

Although QuickTime is natively supported by the Matrox clip player, which is available with any Matrox video card or Matrox X.Open dongle, the QuickTime player cannot be used as fallback device on a 64-bit Viz Engine. A 64-bit Viz Engine requires Matrox hardware to utilize QuickTime codecs for clip playback and post-rendering with MatroxFileWriter and ClipOut channel.

The following QuickTime codecs are supported:

- RLE (animation): (Playback only)
- H264
- Apple ProRes
- HDV
- XDCam
- DVCPro
- DNxHD - Requires Matrox X.mio2, X.mio2 Plus or X.mio3 Video board

⚠ **IMPORTANT!** The Codec DNxHD requires its own license, issued by Vizrt.

Matrox cards support the following codec types and formats:

- Matrox Codecs
- QuickTime Codecs for the Viz Artist/Engine 64-bit Platform
- PAL Codecs

    - MPEG2 IFrame
    - MPEG2 IBP
    - MJPEG Lossless
    - DVCAM
    - DVCPRO
    - XDCAM
    - H264
    - ProRes
    - Uncompressed
- NTSC Codecs
    - MPEG2 IFrame
    - MPEG2 IBP
    - MJPEG Lossless
    - DVCAM
    - DVCPRO
    - XDCAM
    - H264
    - ProRes
    - Uncompressed
- 720p50 and 720p59.94 (60M) Codecs
    - MPEG2 IFrame
    - MPEG2 IBP
    - DVCPRO HD
    - HDV
    - XDCAM HD
    - DNxHD
    - AVC-Intra
    - H264
    - ProRes
    - Uncompressed
- 1080i25 and 1080i29.97 (30M) Codecs
    - MPEG2 IFrame
    - MPEG2 IBP
    - DVCPRO HD
    - HDV
    - XDCAM HD
    - DNxHD
    - AVC-Intra
    - H264
    - ProRes
    - Uncompressed
- 1080p50 and 1080p59.94 (60M) Codecs
    - MPEG2 IFrame
    - ProRes
- 2160p UHDTV Codecs

- MPEG2 IFrame
- ProRes
- XAVC
- QuickTime Writer plug-in Codecs
- DVS and Bluefish Codecs

7.8.3 PAL Codecs

MPEG2 IFrame

Container	Codec	Features	Audio
.avi	MPEG2-IFrame422720x576	Alpha: noBitrate: 10-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2-IFrame4224720x576	Alpha: yesBitrate: 10-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

MPEG2 IBP

Container	Codec	Features	Audio
.avi	MPEG2 IBP SD 4:2:0, itemary, Program, Transport720x576	Alpha: noBitrate: 1-15Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2 IBP SD 4:2:2, itemary, Program, Transport720x576	Alpha: noBitrate: 5-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf	MPEG2 IBP SD 4:2:0, itemary, Program, Transport720x576	Alpha: noBitrate: 1-15Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf	MPEG2 IBP SD 4:2:2, itemary, Program, Transport720x576	Alpha: noBitrate: 5-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

MJPEG Lossless

Container	Codec	Features	Audio
.avi	MJPEG Lossless720x576	Alpha: noBitrate: 160Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

DVCAM

Container	Codec	Features	Audio
.avi	DV/DVCAM 4:2:0720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.dv	DVCAM 4:2:0720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.mov	DVCAM 4:2:0720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	DVCAM 4:2:0720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 4 ch: 16 in 16bit; 48kHz

DVCPRO

Container	Codec	Features	Audio
.avi	DVCPRO 4:1:1720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	DVCPRO 50 4:2:2720x576	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Container	Codec	Features	Audio
.dv	DVCPRO 4:1:1720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.dv	DVCPRO 50 4:2:2720x576	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.mov	DVCPRO 4:1:1720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	DVCPRO 50 4:2:2720x576	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	DVCPRO 4:1:1720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	DVCPRO 50 4:2:2720x576	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf Panasonic P2 (OP-Atom)	DVCPRO 4:1:1720x576	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No Audio
.mxf Panasonic P2 (OP-Atom)	DVCPRO 50 4:2:2720x576	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No Audio

XDCAM

Container	Codec	Features	Audio
.mxf XDCAM (OP1a)	D10 (IMX) 4:2:2720x608	Alpha: noBitrate: 50Bitdepth: 10VBI: requiredTimecode: required	AES3- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz

H264

Container	Codec	Features	Audio
.mov	H.264/ AVC720x576	Alpha: noBitrate: VBRBitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

ProRes

Container	Codec	Features	Audio
.mov	ProRes 422720x576	Alpha: no Bitrate: 42Bitdepth: 8VBI: N/ ATimecode: N/A	No Audio- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	ProRes 422 HQ 720x576	Alpha: noBitrate: 63Bitdepth: 8VBI: N/ ATimecode: N/A	No Audio- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Uncompressed

Container	Codec	Features	Audio
.avi	Uncompressed YUVU 4:2:2720x576	Alpha: noBitrate: 160Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	Uncompressed YUAVUA 4:2:2:4720x576	Alpha: yesBitrate: 240Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

7.8.4 NTSC Codecs

MPEG2 IFrame

Container	Codec	Features	Audio
.avi	MPEG2-IFrame422720x480	Alpha: noBitrate: 10-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2-IFrame4224720x480	Alpha: yesBitrate: 10-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

MPEG2 IBP

Container	Codec	Features	Audio
.avi	MPEG2 IBP SD 4:2:0, itemary, Program, Transport720x480	Alpha: noBitrate: 1-15Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2 IBP SD 4:2:2, itemary, Program, Transport720x480	Alpha: noBitrate: 5-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	MPEG2 IBP SD 4:2:0, itemary, Program, Transport720x480	Alpha: noBitrate: 1-15Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	MPEG2 IBP SD 4:2:2, itemary, Program, Transport720x480	Alpha: noBitrate: 5-50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

MJPEG Lossless

Container	Codec	Features	Audio
.avi	MJPEG Lossless720x480	Alpha: noBitrate: VBRBitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

DVCAM

Container	Codec	Features	Audio
.avi	DV/DVCAM 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.dv	DV/DVCAM 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.mov	DVCAM 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	DVCAM 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 4 ch: 16 in 16bit; 48kHz
.mxf Panasonic P2 (OP-Atom)	DV/DVCAM 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No Audio

DVCPRO

Container	Codec	Features	Audio
.avi	DVCPRO 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	DVCPRO 50 4:2:2 720x480	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.dv	DVCPRO 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.dv	DVCPRO 50 4:2:2720x480	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.mov	DVCPRO 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	DVCPRO 50 4:2:2720x480	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf	DVCPRO 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf	DVCPRO 50 4:2:2720x480	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf Panasonic P2 (OP-Atom)	DVCPRO 4:1:1720x480	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ATimecode: N/A	No Audio

Container	Codec	Features	Audio
.mxf Panasonic P2 (OP-Atom)	DVCPRO 50 4:2:2720x480	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No Audio

XDCAM

Container	Codec	Features	Audio
.mxf XDCAM (OP1a)	D10 (IMX) 4:2:2720x512	Alpha: no Bitrate: 30, 40, 50Bitdepth: 10VBI: requiredTimecode: required	AES3- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz

H264

Container	Codec	Features	Audio
.mov	H.264/AVC720x480	Alpha: noBitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

ProRes

Container	Codec	Features	Audio
.mov	ProRes 422720x486	Alpha: noBitrate: 42 Bitdepth: 8 VBI: N/A Timecode: N/A	No Audio- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	ProRes 422 HQ720x486	Alpha: no Bitrate: 63 Bitdepth: 8 VBI: N/A Timecode: N/A	No Audio- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Uncompressed

Container	Codec	Features	Audio
.avi	Uncompressed YUVU 4:2:2720x480	Alpha: noBitrate: 160Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	Uncompressed YUAVUA 4:2:2:4720x480	Alpha: yesBitrate: 240Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

7.8.5 720p50 and 720p59.94 (60M) Codecs

MPEG2 IFrame

Container	Codec	Features	Audio
.avi	MPEG2-IFrame4221 280x720	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2-IFrame4224 1280x720	Alpha: yesBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

MPEG2 IBP

Container	Codec	Features	Audio
.avi	MPEG2 IBP HD 4:2:0, itemary, Program, Transport1280x720	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Container	Codec	Features	Audio
.avi	MPEG2 IBP HD 4:2:2, itemary, Program, Transport1280x720	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	MPEG2, IBP HD 4:2:0, itemary, Program, Transport1280x720	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	MPEG2, IBP HD 4:2:2, itemary, Program, Transport1280x720	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

DVCPRO HD

Container	Codec	Features	Audio
.avi	DVCPRO HD1280x720	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.dv	DVCPRO HD1280x720	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.mov	DVCPRO HD1280x720	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	DVCPRO HD1280x720	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf Panasonic P2 (OP-Atom)	DVCPRO HD1280x720	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ATimecode: N/A	No Audio

HDV

Container	Codec	Features	Audio
.mov	HDV1280x720	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

XDCAM HD

Container	Codec	Features	Audio
.mov	XDCam EX1280x720	Alpha: noBitrate: 25CBR, 35VBRBitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	XDCam HD 4221280x720	Alpha: noBitrate: 18-35Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mp4	XDCam EX1280x720	Alpha: noBitrate: 25CBR, 35VBRBitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf XDCAM (OP1a)	XDCAM HD, IBP HD 4:2:0,itemary, Program, Transport1280x720	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz- 4 ch: 16 in 16bit; 48kHz
.mxf XDCAM (OP1a)	XDCAM HD422, IBP HD 4:2:2, 1280x720	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ ATimecode: N/A	PCM:- 8 ch: 24 in 32bit; 48kHz

DNxHD

Container	Codec	Features	Audio
.mxf (OP-Atom)	DNxHD 145, 220, 220x1280x720	Alpha: noBitrate: 100Bitdepth: 8, 10VBI: N/ ATimecode: N/A	No Audio
.mxf (OP1a)	DNxHD 145, 220, 220x1280x720	Alpha: noBitrate: 100Bitdepth: 8, 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	DNxHD 4:2:21280x720	Alpha: noBitrate: 36, 145, 220Bitdepth: 8; 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

AVC-Intra

Container	Codec	Features	Audio
.mxf (OP1a)	AVCIntra Class 501280x720	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	AVCIntra Class 1001280x720	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf Panasonic P2 (OP-Atom)	AVCIntra Class 501280x720	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ ATimecode: N/A	No Audio
.mxf Panasonic P2 (OP-Atom)	AVCIntra Class 1001280x720	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ ATimecode: N/A	No Audio

H264

Container	Codec	Features	Audio
.mov	H.264/AVC1280x720	Alpha: noBitrate: VBRBitdepth: 8; 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mp4	XDCAM EX1280x720	Alpha: noBitrate: VBRBitdepth: 8; 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

ProRes

Container	Codec	Features	Audio
.mov	I-Frame 4:2:21280x720	Alpha: noBitrate: 100 (LT), 147, 220 (HQ)Bitdepth: 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	I-Frame 4:4:4:41280x720	Alpha: yesBitrate: VBRBitdepth: 8, 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Uncompressed

Container	Codec	Features	Audio
.avi	Uncompressed YUVU 4:2:21280x720	Alpha: noBitrate: 700Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	Uncompressed YUAVUA 4:2:2:41280x720	Alpha: yesBitrate: 1050Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

7.8.6 1080i25 and 1080i29.97 (30M) Codecs

MPEG2 IFrame

Container	Codec	Features	Audio
.avi	MPEG2-IFrame42219 20x1080	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2-IFrame42241 920x1080	Alpha: yesBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

MPEG2 IBP

Container	Codec	Features	Audio
.avi	MPEG2 IBP HD 4:2:0, itemary, Program, Transport1920x1080	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2 IBP HD 4:2:2, itemary, Program, Transport1920x1080	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	MPEG2, IBP HD 4:2:0, itemary, Program, Transport1920x1080	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	MPEG2, IBP HD 4:2:2, itemary, Program, Transport1920x1080	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

DVCPRO HD

Container	Codec	Features	Audio
.avi	DVCPRO HD1920x1080	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.dv	DVCPRO HD1920x1080	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz
.mov	DVCPRO HD1440x1080	Alpha: noBitrate: 100Bitdepth: 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP-Atom)	DVCPRO HD1440x1080	Alpha: noBitrate: 100Bitdepth: 10VBI: N/ ATimecode: N/A	No Audio
.mxf (OP1a)	DVCPRO HD1440x1080	Alpha: noBitrate: 100Bitdepth: 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

HDV

Container	Codec	Features	Audio
.mov	HDV1440x1080	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

XDCAM HD

Container	Codec	Features	Audio
.mov	XDCam EX1920x1080	Alpha: noBitrate: 25CBR, 35VBRBitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Container	Codec	Features	Audio
.mov	XDCam HD1440x1080	Alpha: noBitrate: 35Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	XDCam HD 4221920x1080	Alpha: noBitrate: 18-35Bitdepth: 8VBI: N/ ATimecode: N/A	PCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mp4	XDCam EX 1920x1080	Alpha: no Bitrate: 25CBR, 35VBR Bitdepth: 8 VBI: N/ A Timecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	XDCAM HD,IBP HD 4:2:0,itemary ,Program,Transport1920x1080	Alpha: noBitrate: 25Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 16 in 16bit; 48kHz- 4 ch: 16 in 16bit; 48kHz

DNxHD

Container	Codec	Features	Audio
.mxf (OP-Atom)	DNxHD 145, 220, 220x 1920x1080	Alpha: noBitrate: 100Bitdepth: 8, 10VBI: N/ ATimecode: N/A	No Audio
.mxf (OP1a)	DNxHD 145, 220, 220x1920x1080	Alpha: noBitrate: 100Bitdepth: 8, 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	DNxHD 4:2:21920x1080	Alpha: noBitrate: 36, 145, 220Bitdepth: 8; 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

AVC-Intra

Container	Codec	Features	Audio
.mxf (OP-Atom)	AVCIntra Class 501440x1080	Alpha: noBitrate: 50Bitdepth: 10VBI: N/ATimecode: N/A	No Audio
.mxf (OP-Atom)	AVCIntra Class 100 1920x1080	Alpha: noBitrate: 100Bitdepth: 10VBI: N/ATimecode: N/A	No Audio
.mxf (OP1a)	AVCIntra Class 501920x1080	Alpha: noBitrate: 50Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mxf (OP1a)	AVCIntra Class 1001920x1080	Alpha: noBitrate: 100Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

H264

Container	Codec	Features	Audio
.mov	H.264/AVC1920x1080	Alpha: noBitrate: VBRBitdepth: 8; 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mp4	XDCAM EX1920x1080	Alpha: noBitrate: VBRBitdepth: 8; 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

ProRes

Container	Codec	Features	Audio
.mov	I-Frame 4:2:21920 x1080	Alpha: noBitrate: 100 (LT), 147, 220 (HQ)Bitdepth: 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	I-Frame 4:4:4:4192 0x1080	Alpha: yesBitrate: VBRBitdepth: 8, 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Uncompressed

Container	Codec	Features	Audio
.avi	Uncompressed YUVU 4:2:21920x108 0	Alpha: noBitrate: 800Bitdepth: 8VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	Uncompressed YUAVUA 4:2:2:41920x10 80	Alpha: yesBitrate: 1200Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

7.8.7 1080p50 and 1080p59.94 (60M) Codecs

MPEG2 IFrame

Container	Codec	Features	Audio
.avi	MPEG2-IFrame42219 20x1080	Alpha: noBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

Container	Codec	Features	Audio
.avi	MPEG2-IFrame42241 920x1080	Alpha: yesBitrate: 50-300Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

ProRes

Container	Codec	Features	Audio
.mov	I-Frame 4:2:2 1920x1080	Alpha: noBitrate: 147, 220Bitdepth: 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	I-Frame 4:4:4:4 1920x1080	Alpha: yesBitrate: VBRBitdepth: 8, 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

7.8.8 2160p UHDTV Codecs

MPEG2 IFrame

Container	Codec	Features	Audio
.avi	MPEG2-IFrame42238 40x2160	Alpha: noBitrate: 900-1200Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.avi	MPEG2-IFrame42243 840x2160	Alpha: yesBitrate: 900-1200Bitdepth: 8VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

ProRes

Container	Codec	Features	Audio
.mov	I-Frame 4:2:2 3840x2160	Alpha: noBitrate: VBRBitdepth: 10VBI: N/ ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz
.mov	I-Frame 4:4:4:4 3840x2160	Alpha: yesBitrate: VBRBitdepth: 8, 10VBI: N/ATimecode: N/A	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

XAVC

Container	Codec	Features	Audio
.mxf (OP1a)	H.264 level 5.2,3840x2160	Alpha: noBitrate: 960Bitdepth: 8, 10, 12VBI: yesTimecode: yes	No AudioPCM:- 2 ch: 24 in 32bit; 48kHz- 4 ch: 24 in 32bit; 48kHz- 8 ch: 24 in 32bit; 48kHz- 16 ch: 24 in 32bit; 48kHz

⚠ **IMPORTANT!** Due to its huge amount of required CPU power, it is not recommended to use this codec. If is to be used, only use the V210 surface format.

7.8.9 QuickTime Writer plug-in Codecs

The QuickTime Writer plug-in supports these QuickTime codecs in the following formats:

Name	Options
Animation	Alpha: optional
BMP	Alpha: no
Cinepak	Alpha: no
Component Video	Alpha: no
DV - PAL	Alpha: no

Name	Options
DV/DVCPRO - NTSC	Alpha: no
DVCPRO - PAL	Alpha: no
Graphics	Alpha: no
H.264	Alpha: no
JPEG2k	Alpha: optional
Motion JPEG A	Alpha: no
Motion JPEG B	Alpha: no
MPEG-4 Video	Alpha: no
Photo - JPEG	Alpha: no
Planar RGB	Alpha: optional
PNG	Alpha: optional
Sorenson Video	Alpha: no
Sorenson Video 3	Alpha: no
TGA	Alpha: optional
TIFF	Alpha: optional
Uncompressed	Alpha: optional
Video	Alpha: no

7.8.10 DVS and Bluefish Codecs

The DVS and Bluefish Video cards support only the **Microsoft DirectShow Filter Graph** framework to render video clips. This enables play, for example, of MPEG files or streams from a server.

This means that DVS and Bluefish Video cards can play everything that can be played out by the Windows Media Player, so it depends on what codecs/DirectShow filters are installed on the machine.

For more information on how to use video clips see Video Clip Playback Considerations and Video Clips, in particular, the SoftClip and MoViz plug-ins.

7.9 Advanced Issues With Video Codecs

Many codecs have specific limitations and will only work on certain types of input. It is not possible to list all things to be aware of, but here are some:

- 24-bit uncompressed video is valid but rare. Some programs may have difficulty in reading such files. Try creating an uncompressed video using RGBA to create a more common 32-bit uncompressed video.
- Not all codecs support both 24-bit and 32-bit video. You may need to change between RGB and RGBA to get a specific codec to work.
- Some codecs will work with RGBA but will nevertheless store the video as 24-bit.
- If you plan to use an RGBA video in a program that can handle alpha, you need to make sure that you have KEYFunctions on containers you want to have visible. Otherwise the alpha channel will be completely blank, thus making the whole video totally transparent.
- Some codecs require a handle to the top level application window. The Viz Artist plug-in interface is unable to supply such a handle. As a result the configuration windows of some of the codecs will fail to show. Also, some codecs will crash unless they get a handle.
- Files created in raw uncompressed mode can display artifacts when played back in Windows Media Player. This is some oddity of the Media Player. The AVI video file contains pure RGB data, but the Media Player dithers it (often using only 16 colors) for some unknown reason. As said, the video file is correct and you should be able to import it into other video editing applications without any problems.
- The free Matrox Video for Windows (VFW) software codecs let you render and play back Matrox .avi files without having Matrox video hardware installed. The current release as of this writing (2015/1 codec version 2.0) supports both Windows 7 and Windows 8.1. Visit the Matrox Support site for more information and to download the codec installer: http://www.matrox.com/video/en/support/windows/vfw_software_codecs/. On machines with a Matrox board and driver version 7.5 and higher, you do not need to install VFW-Codecs since they *are included in the driver*. On machines with a Matrox board and drivers before version 7.5, there's always a matching VFW-Codecs pack available with the driver. However **do not** install Matrox VFW Codecs or any non-matching VFW-Codecs pack on a machine with a Matrox board and drivers already installed. The following codecs are included in Matrox's free codec pack:
  - Matrox DV/DVCAM
  - Matrox DVCPRO
  - Matrox DVCPRO50
  - Matrox DVCPRO HD
  - Matrox MPEG2 I-frame in SD and HD resolutions, with or without alpha
  - Matrox Uncompressed in SD and HD resolutions, with or without alpha
  - Matrox Offline HD (playback only)
  - Matrox M-JPEG and HDV (playback only)

8 Lights

Lights are used to illuminate objects in a Scene.

How lights and materials are used in a Scene makes a big difference in the appearance of a rendered Scene. Materials define how light reflects off a surface, while direct light and ambient light levels define the light that is reflected.

When materials are used in a scene, lights are also required. A Material needs to reflect light to be visible. Without light, just the ambient part of the material is visible, which means that the object is rendered almost black with default settings.

Closely related to lights are shadows. Objects that have a light-source applied on them will cast a shadow in real life. Shadow mapping is a process by which shadows are added to 3D scenes in Viz Artist for more realistic images, see Shadow Maps.

This section contains information on the following topics:

- Types Of Light
- Light Editor
- Light Visualization
- Light Source Animation
- Shadow Maps

8.1 Types Of Light

To understand what effect light has on the surface of rendered 3D objects, three types of light can be defined:

- Local Light
- Spot Light
- Infinite Light

When using lights, it is important to understand the effect each type of light creates on the surface of a rendered 3D object.

8.1.1 Local Light

Local lights have color and position within a Scene, but no single direction. They give off light equally in all directions (isotropic light source). A light bulb is a good example of a local light.

8.1.2 Spot Light

Spotlights have color, position, and direction in which they emit light. Light emitted from a spot light is made up of a bright inner cone and a larger outer cone, with the light intensity diminishing between the two.

8.1.3 Infinite Light

Infinite lights have color and direction. They emit parallel light. This means that all light generated by infinite lights travels through a Scene in the same direction. Imagine a directional light as a light source at near infinite distance, such as the sun.

8.2 Light Editor

The Light editor makes it possible to modify the light settings in a Scene. A Scene can have up to eight active, simultaneous, lights.

The Light Editor consists of two panels:

- **Left panel** (1): Shows a list of all eight editable lights.
- **Right panel** (2): Shows the parameters for the currently selected light.

This section contains information on the following topics:

- Light Editor Left Panel
- Light Editor Right Panel
- Light Editor Procedures

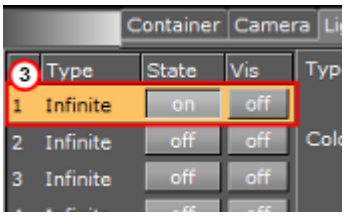
8.2.1 Light Editor Left Panel

The Light Editor left panel shows a list of all eight editable lights, and their current status.

- **Number**: The number of each available light, from 1 to 8.
- **Type**: The type of light used, which is selected in the Light Editor Right Panel.
- **State**: Sets the light On or Off.
- **Vis**: Sets the visualization of a light On or Off (see Light Visualization):
  - **On**: The visualization of a light can be seen in the Scene Editor (and the Scene Editor Views).
  - **Off**: The visualization of a light cannot be seen in the output window.

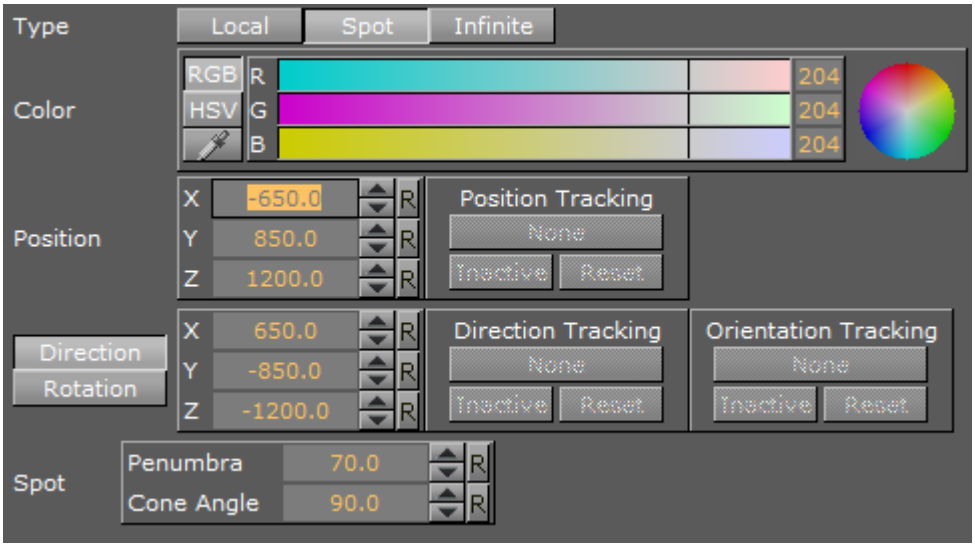
⚠ **Note:** The Bounding Box (BB (see Scene Editor Buttons)) must be set to **On**. If it is set to **Off** no light visualizations can be seen in the Scene Editor or Scene Editor Views.

⚠ **Note:** If the Bounding Box is set to **On**, the visualization of a light, which **State** is set to **On** and is highlighted in the GUI (3), is always visible in the Scene Editor.



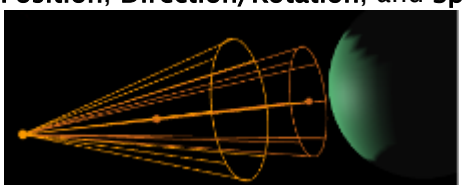
8.2.2 Light Editor Right Panel

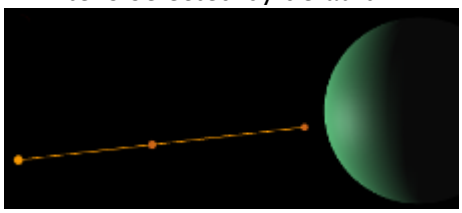
The Light Editor right panel shows the editable parameters for the currently selected light:



- **Type:** Select which Light type to use:
  - **Local:** See Local Light for light type details. The Local light has properties for **Color** and **Position**.

    
  - **Spot:** See Spot Light for light type details. The Spot light has properties for **Color**, **Position**, **Direction/Rotation**, and **Spot** (Penumbra and Cone Angle).

    
  - **Infinite:** The Infinite light has properties for **Color**, **Position**, and **Direction/Rotation**. Infinite is selected by default.

    
- **Color**: Sets the color of the light.
- **Position**: Sets the position of the light along the X, Y, and Z axis.
- **Direction**: Defines the direction vector for **Spot** or **Infinite** light.
- **Rotation: Sets the values to rotate the Spot** or **Infinite** light sources.
- **Tracking**: A light can be configured to track the Position, and Direction or Orientation of a Container (see To Set a Light to Track Containers):
  - **Position Tracking:** The position of the light will be the same as the target Containers position.

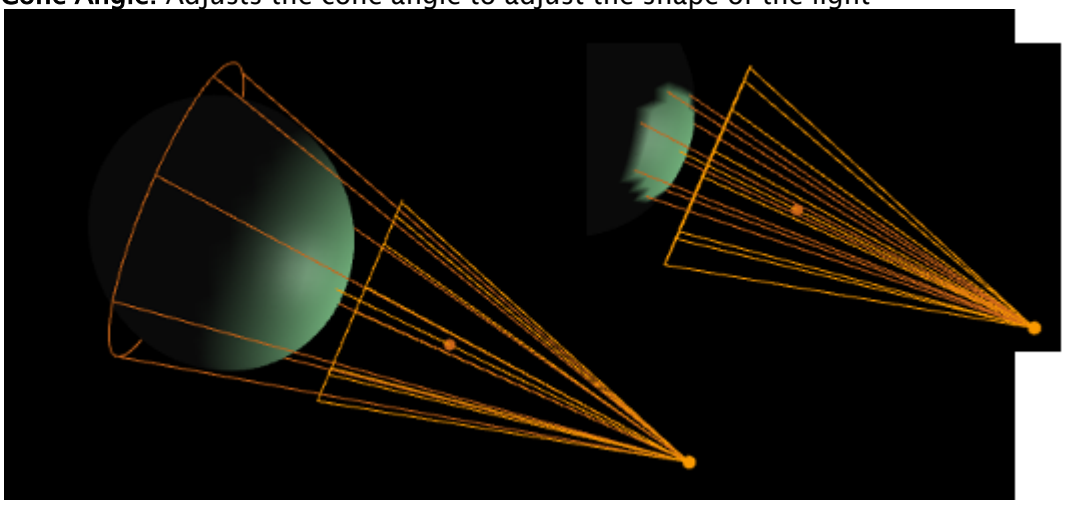
- **Direction Tracking**: The direction of the light will be updated to always point towards the current position of the target container.
- **Orientation Tracking:** The rotation (orientation in space) of the light will be the same as the target Containers rotation (orientation in space).

> **Note:** Direction and Orientation tracking cannot be used at the same time. The results would contradict each other (both of them result in a potentially different rotation value for the container).

The following parameters can also be adjusted for tracking with light:

- **Active/Inactive**: Drag a Container to the drop zone, to automatically activate light tracking. To stop tracking, click the Active/Inactive button.
- **Reset**: Disables light tracking.

- **Spot** (available when **Spot** is selected):
  - **Penumbra:** Sets the concentration of the light (penumbra) within the light cone of the Spot light. When set to zero, the whole light cone has the same intensity. If set greater than zero, the intensity decreases away from the center
  - **Cone Angle:** Adjusts the cone angle to adjust the shape of the light



> **Note:** A Material item must be used to show light effects.

8.2.3 Light Editor Procedures

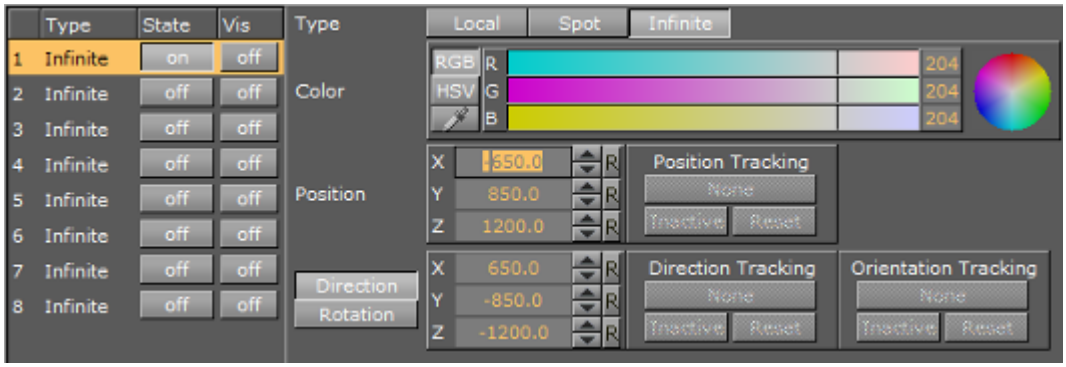
This section contains information on the following topics:

- To Apply Light in a Scene
- To Set a Light to Track Containers
- To Copy the Properties of One Light to Another Light

To Apply Light in a Scene

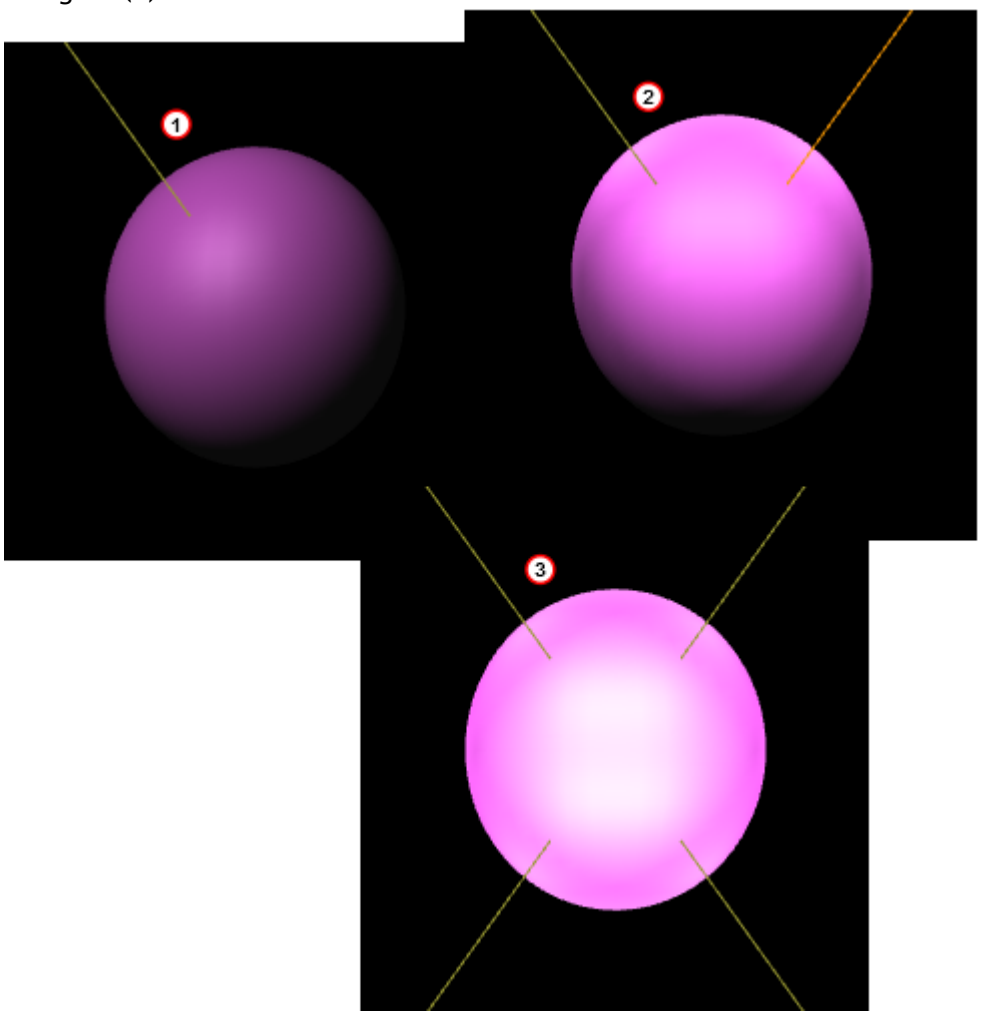
1. Click **Built Ins**.
2. Click the Geom plug-ins tab.
3. Drag an object into the Scene Tree.
4. Add a material to the same Container.
5. Click the **Light** tab to view the Light Editor.

> ⚠ **Note:** Light number 1 is ON by default.



6. In the left panel, click any or all of the lights in the **State** column to show different light perspectives:
   - One light (1)
   - First four lights (2)

- All lights (3)

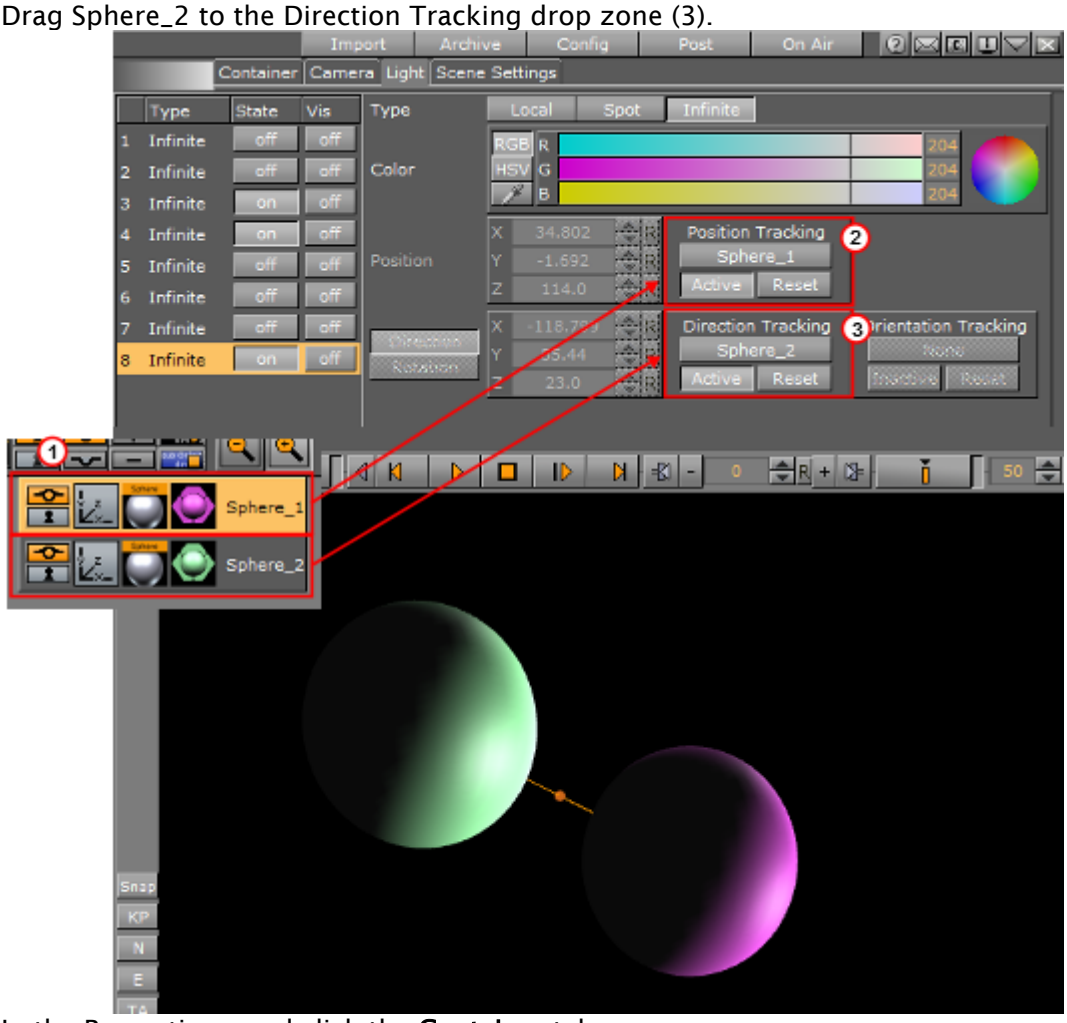


To Set a Light to Track Containers

A light can be configured to track containers in both position and direction (see **Tracking** in the Light Editor Right Panel)

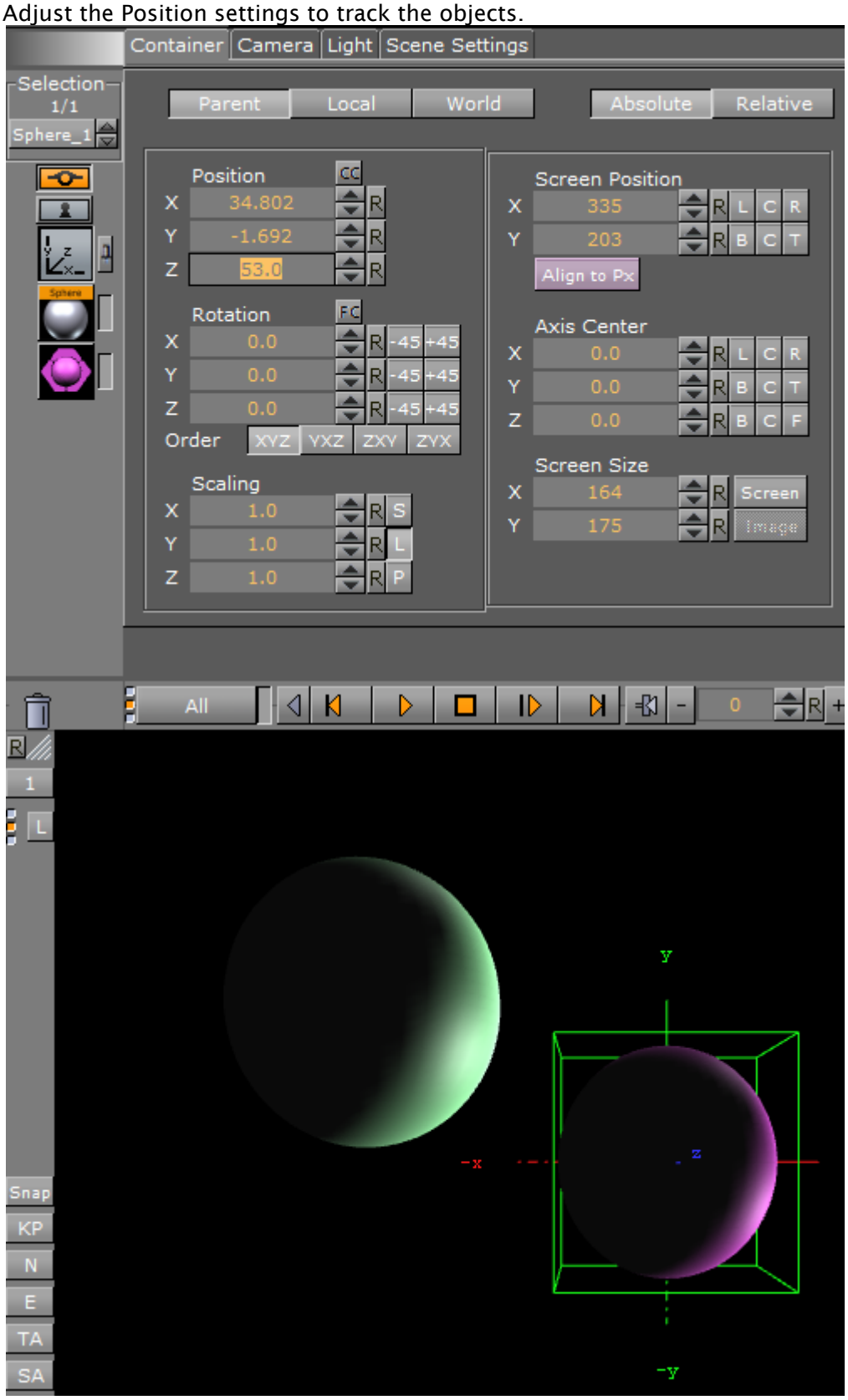
1. In the Scene Tree create two Objects (1), one to track position (Sphere_1) and one to track rotation (Sphere_2).
2. In the right panel select the **Light** tab.
3. Drag Sphere_1 to the Position Tracking drop zone (2).

4. Drag Sphere_2 to the Direction Tracking drop zone (3).



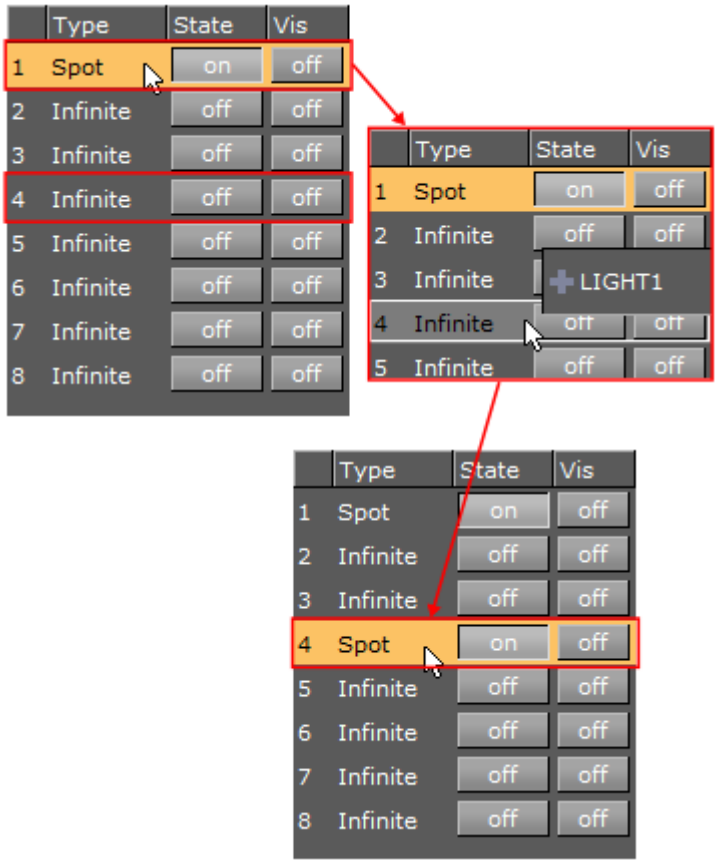
5. In the Properties panel click the **Container** tab.

6. Adjust the Position settings to track the objects.



To Copy the Properties of One Light to Another Light

1. Select the light for the source properties.
2. Drag it onto the target light.



The target light will inherit all properties belonging to the source light.

See Also

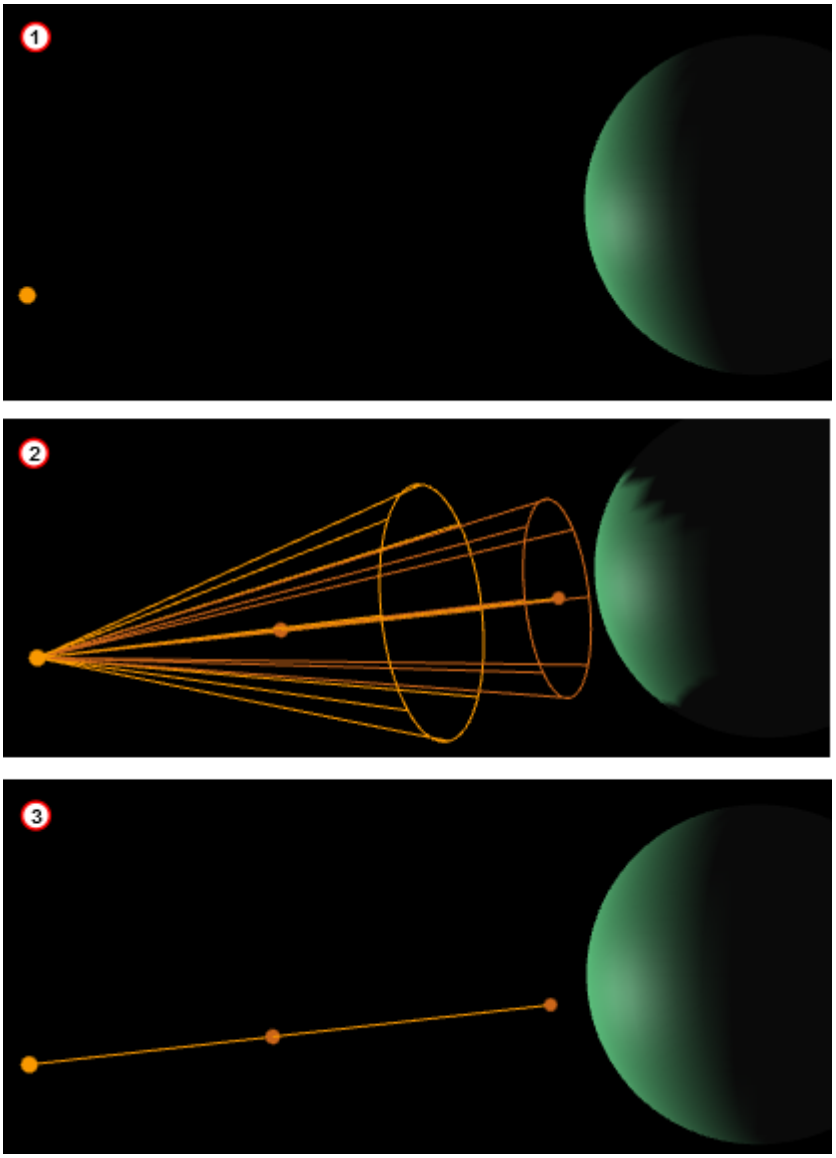
- Types Of Light
- Light Visualization
- Light Source Animation
- Lighting Container plug-in

8.3 Light Visualization

The position, direction or rotation of a light source can be set and modified in the Light Editor or directly in the Scene Editor or the Scene Editor Views, by selection of handles, which are part of the light visualization.

In the Light Editor, a light can be set to **On** or **Off** (its **State**), and its visualization (editing parameters) can also be set **On** or **Off**.

A Local Light is visualized as a single point (1), and the Spot Light (2) and Infinite Light (3) are visualized as a line with three handles. The **Spot** light also has a wire-frame box that shows the penumbra and light cone angle (the scope of the light).



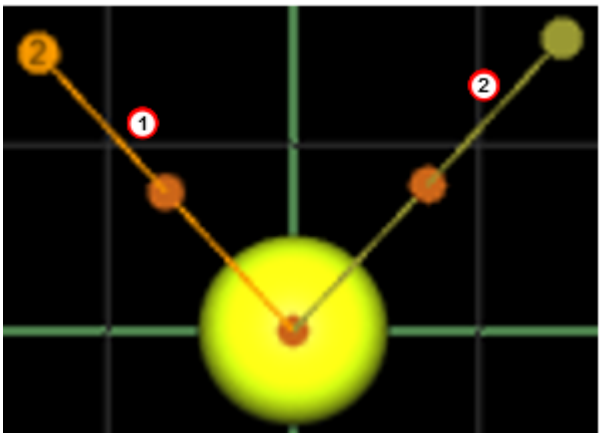
If the **State** of a light is set to **On**, and it is also highlighted in the Light Editor, its visualization will be shown in the Scene Editor (if **BB** is clicked (see Scene Editor Buttons)) and the Scene Editor Views.

To view the visualization of all lights, whose **State** is set to **On**, set **Vis** for each light to **On**.

	Type	State	Vis
1	Infinite	on	on
2	Infinite	on	off
3	Infinite	off	on

If all available light visualizations are visible, click on a light in the Light Editor to select its visualization in the Scene Editor and the Scene Editor Views.

A selected light visualization shows orange (with its light source number) (1). An unselected light visualization shows green (2), with two handles shown in orange.



Each light visualization has handles, which are used to modify the light direction in the Scene Editor and Scene Editor Views.

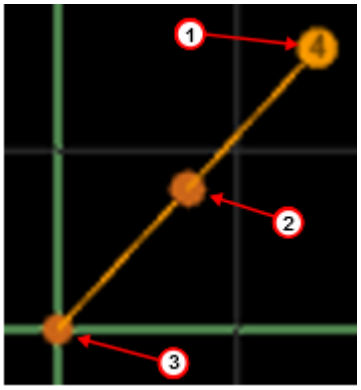
8.3.1 To Modify the Light Direction with Light Handles

1. Open the Light Editor.
2. Select a light, which has its **State** set to **On.**
   The visualization of the selected light will show in the Scene Editor.

   > ⚠ **Note:** To view the visualization of all active lights set **Vis** to **On** for each light.

3. Click on a light visualization handle and drag where required.
   Each light visualization has handles, which are used to modify the light direction in the Scene Editor and Scene Editor Views.



- **Handle** (1): The light source. Drag the handle to set its position
- **Handle** (2): Drag the whole light without altering the proportionality of position and direction
- Handle (3): The end point. Drag the handle to set the direction and distance of the light

---

8.4 Light Source Animation

All light source parameters can be animated in the same way as all other animations:

1. Set a value in the Light editor.
2. Add a Key Frame.
3. Change the value.
4. Add a Key Frame again.

Animations can also easily be created in one of the orthogonal views. There, the light sources show as symbols, and by repositioning them and adding Key Frames, animation sequences are created. For more information, see Light Visualization.

More advanced light source animations are normally created with the light source tracking feature, see To Set a Light to Track Containers.

For more general animation information, see Create Animations.

8.5 Shadow Maps

Shadow maps in Viz Artist produces more realistic images, but shadows can be resource intensive to render in real-time so care should be taken to consider the effects of using shadows on various graphics hardware.

The concept of shadows in a scene is closely related to how you see shadows in real life: When lights are applied to an object the object will cast a shadow, and the intensity and direction of the light source as well as the distance from the light source to the object that the lights hit will affect the shadow. The darkness, intensity and smoothness of the shadow will affect the quality of the rendered scene.

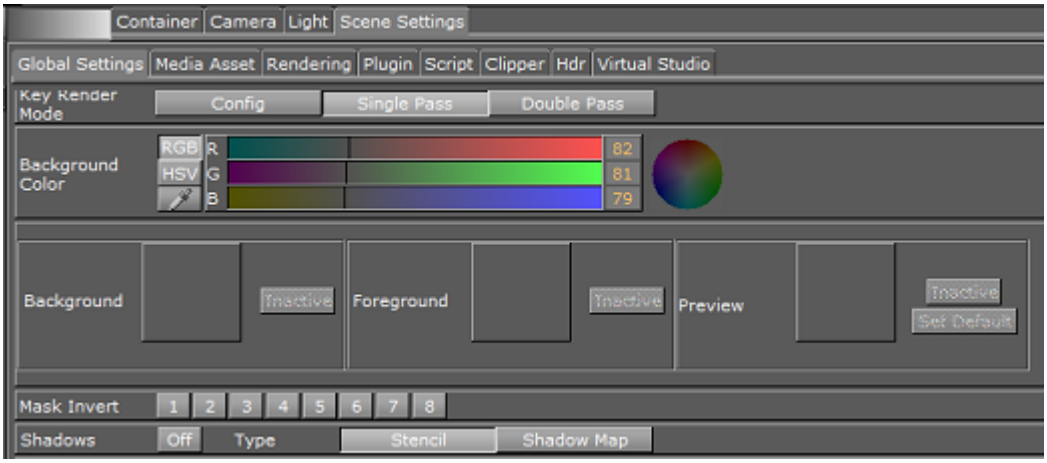
This section contains information on the following topics:

- About Shadow Maps
- To Use Shadows in a Scene

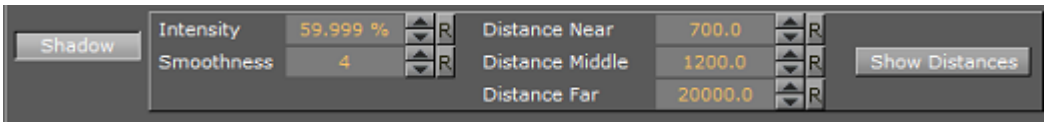
8.5.1 About Shadow Maps

There are two principal methods to show shadows in Viz Artist:

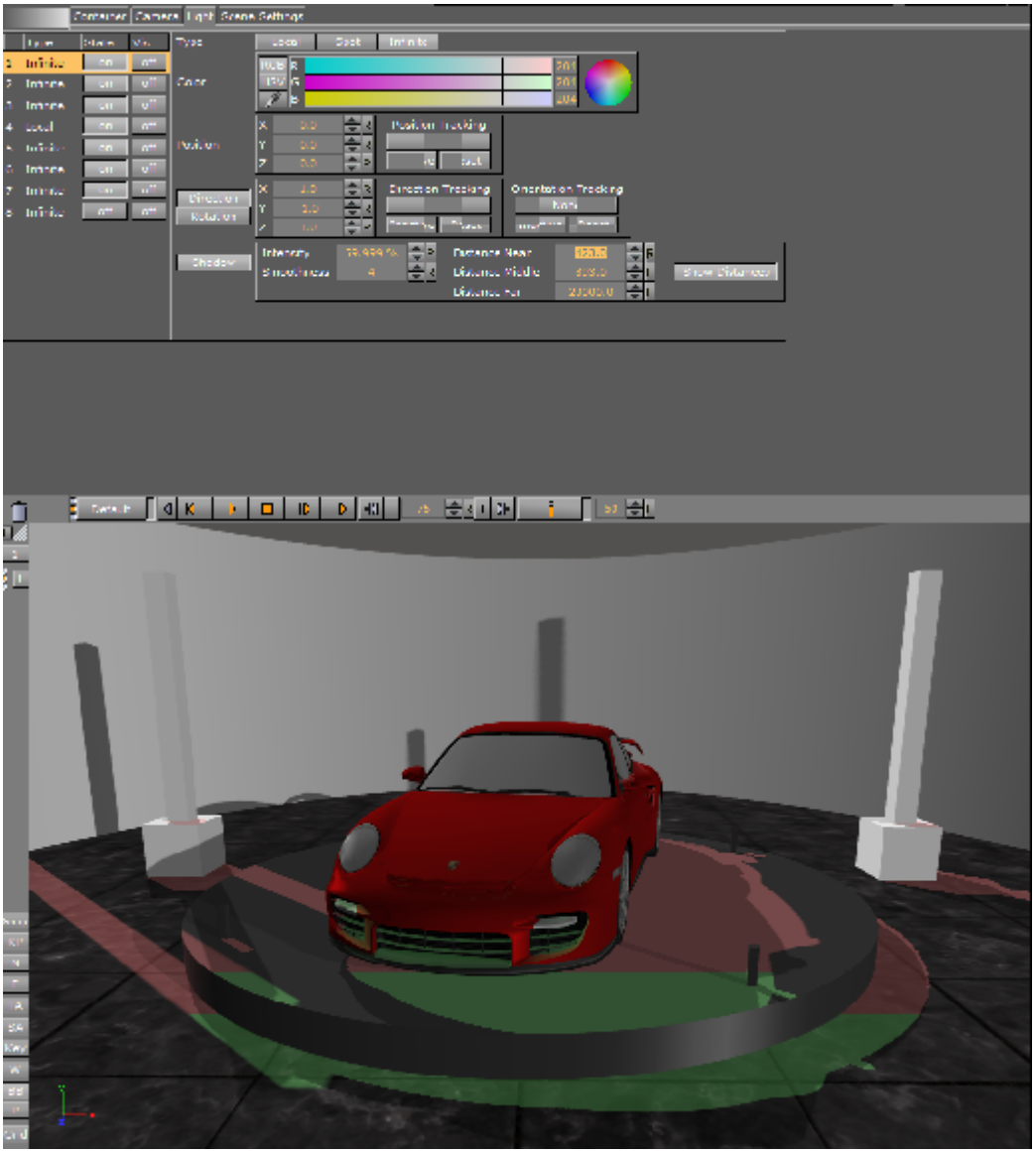
- **Shadow Maps**: More advanced and better looking shadows, but uses more graphic resources. Often called soft shadows. Uses cascaded shadow maps
- **Stencils**: Simpler, not so photo-realistic, but uses less graphic resources
  You enable shadow maps and select type in *Scene Settings > Global Settings*:



Select Light from the menu to show and change the shadow map properties:



- **Intensity**: Sets the intensity of the shadow effect, in percent. Controls the alpha value, simply said it makes the shadows darker.
- **Smoothness**: Determines how smooth the edges and the shadow itself blend with the surroundings given a light-source. Makes shadows smoother the farther away a receiver is from the shadow caster. Possible values: 1 to 10.
  The different layers of the shadow map are grouped into *Near*, *Middle* and *Far* slices or areas. The default values for near, middle and far would normally be fine to use as is, but if required you can change them:
- **Distance Near**: Determines the distance, in viz units, of the near layer of the shadow map.
- **Distance Middle**: Determines the distance, in viz units, of the middle layer of the shadow map.
- **Distance Far**: Determines the distance, in viz units, of the far layer of the shadow map.
- **Show Distances**: Shows a color-map of the "slices" (areas) affected by the shadow maps. When designing a scene, use the "Show Distances" on/off switch to see a color-visualization of how the shadow map will affect each slice of the shadow map. Areas affected by *Near* are shown in green, *Middle* in red and Far in magenta as indicated in the image below:



8.5.2 To Use Shadows in a Scene

1. Go to *Light* and select the light source that should cast a shadow. More than one caster is possible but this comes with a noticeable performance hit for every additional shadow casting light. After selecting the light, on the property panel click the button that says Shadow to change property values for shadows, if required.
2. From *Built Ins > Container Plugins* select the **Caster** and **Receiver** plug-ins and drop them on containers to show shadows. The layer parameters correspond to light sources, so you can have two lights cast shadows but containers can only cast or receive shadows from one of the light sources.

> ⚠ **Note:** The shadow mapping technique can be used together with lens-deformation as well, see Rendering Panel.

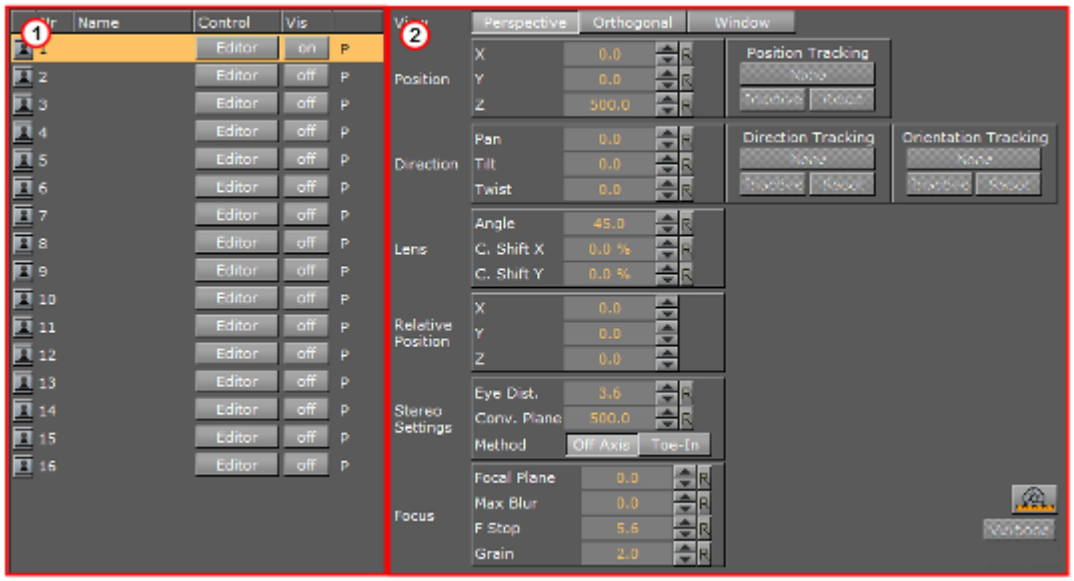
9 Cameras

The camera which shows the Viz Artist Scene can be controlled much the same way that a real camera is controlled. Viz Artist cameras are used by assigning two tracking objects. The first is the camera itself, and the second is the object being tracked.

This section contains information on the following topics:

- Camera Editor
- Stereo Settings
- Stereoscopy Best Practices
- Stereoscopic Output Using Shutter Glasses
- Change Camera Parameters in Orthogonal Views
- Track Objects with a Camera
- Receive Tracking Data from a Real Camera
- Copy Properties from one Camera to Another
- Camera Selection
- Camera Animation
- Advanced Lens Distortion

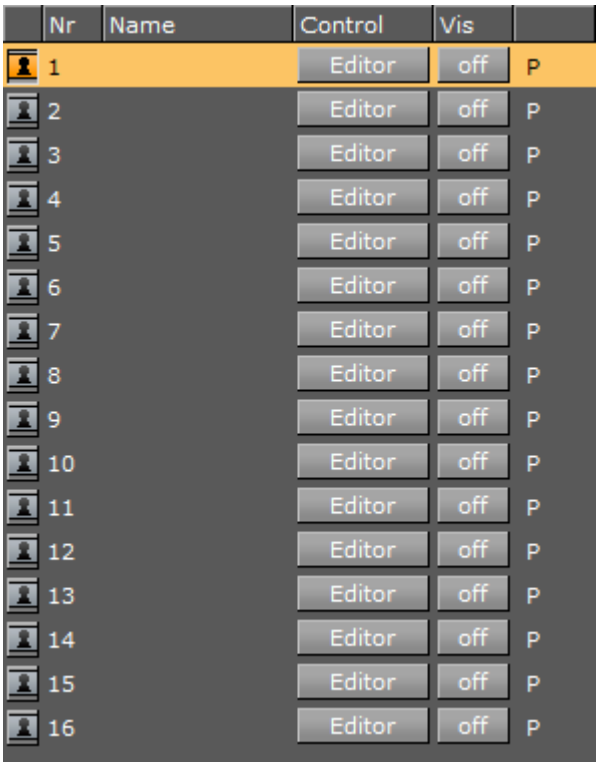
9.1 Camera Editor



The Camera editor makes it possible to modify the properties for the 16 editable cameras that can be used in Viz Artist. To open the Camera Editor, select the **Camera** tab in the Properties panel.

- Camera Editor Left Panel (1): Shows a list of all 16 editable cameras and the current view of each camera
- Camera Editor Right Panel (2): Shows the parameters for the currently selected camera

9.1.1 Camera Editor Left Panel



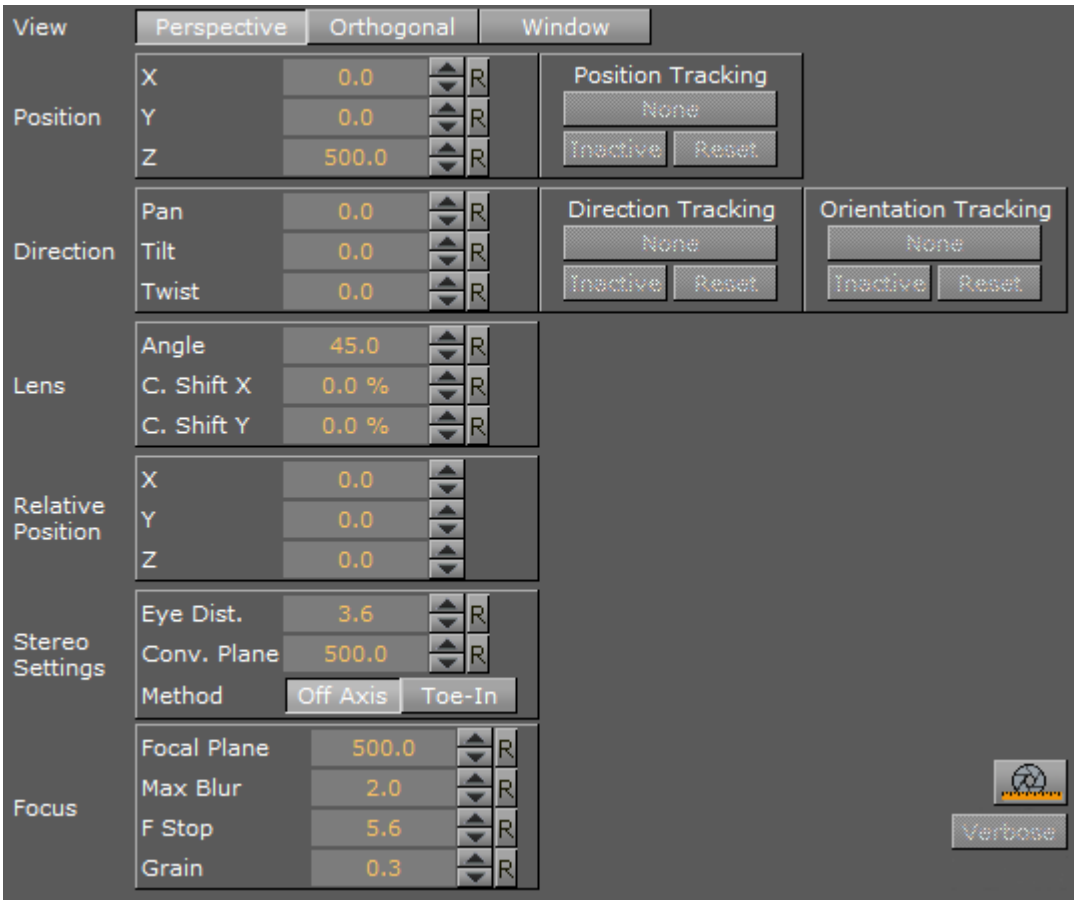
- **Lock/Unlock**: Locks and unlocks a camera when the Keyhole is clicked. Locked cameras are indicated by an orange icon. Except for shortcuts that switch to other cameras, the Scene Editor keyboard shortcuts are disabled for locked cameras. The properties of a locked camera can still be edited through manipulating it in the Scene Editor Views, or by changing the camera parameters in the Camera Editor Right Panel.
- **Nr**: Shows the camera number, from 1 to 16. Selecting a camera will automatically set the viewing perspective in the Scene Editor to that of the selected camera.
- **Name**: Shows the camera name. Right-click or press <F2> to assign a new camera name.
- **Control**: Toggles between:
  - **Remote:** Receives tracking data from a real camera when enabled. Remote cameras are not supported for **GFX Channel** media assets.
  - **Editor:** Configures camera settings when enabled.
- **Vis**: Shows the status of the camera:
  - **Off**: The camera will not be shown in the six predefined orthogonal views. To see a camera in a predefined orthogonal view can help to change the properties of a camera, by dragging the parameters in those views.
  - **On**: Shows the camera.
  - **Inf**: The camera will be shown with infinite lines. This is a good aid for visualizing the scope of a camera that is located far away.
- **View**: Shows the current Camera Editor Left Panel view of each camera. Options are:
  - **P:** Perspective

- **O**: Orthogonal
- **W**: Window

9.1.2 Camera Editor Right Panel

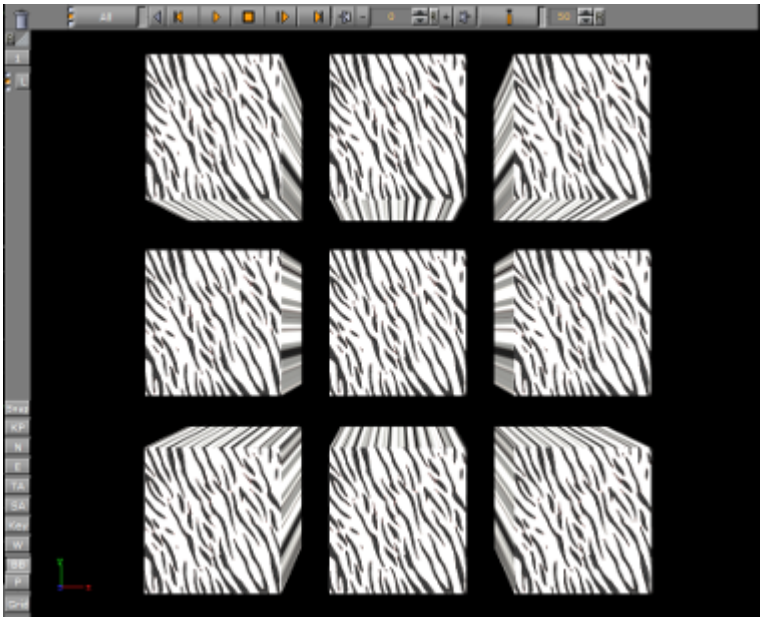
There are three camera views available, and the corresponding parameter options depend on the selected view. The views are described below, and their parameters are explained in the following pages:

- Parameters for Perspective View
- Parameters for Orthogonal View
- Parameters for Window View



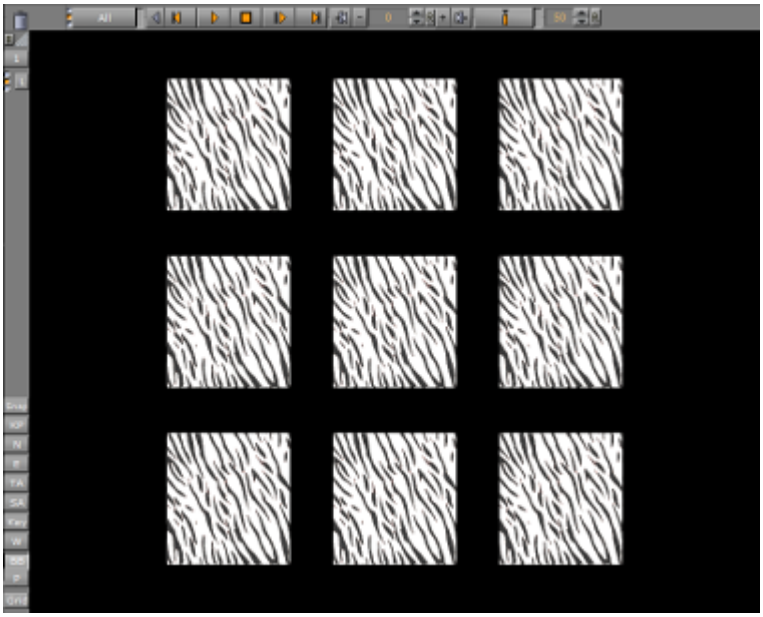
Perspective View

The basic view of a virtual camera is a normal depth perspective, showing three dimensions (see Parameters for Perspective View).



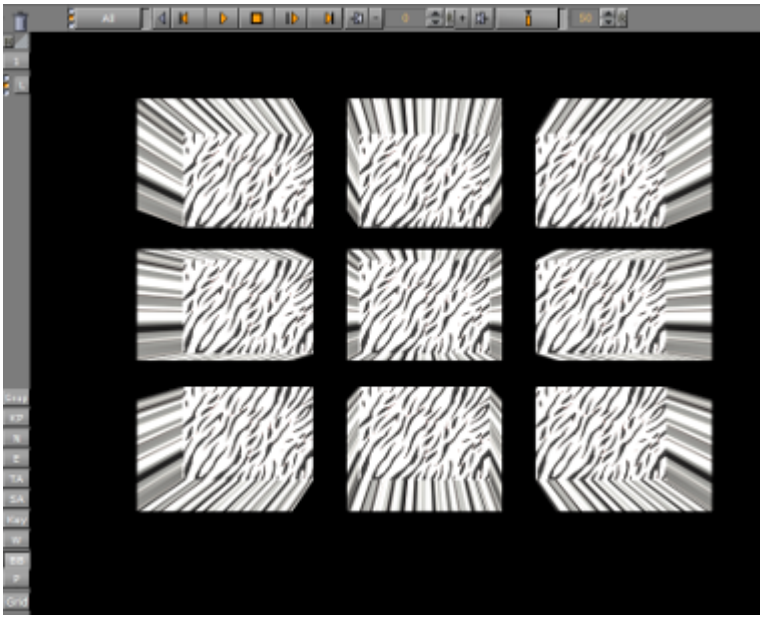
Orthogonal View

Shows a camera view without perspective, so that objects will not change their size when being moved towards or away from the camera (see Parameters for Orthogonal View).



Window View

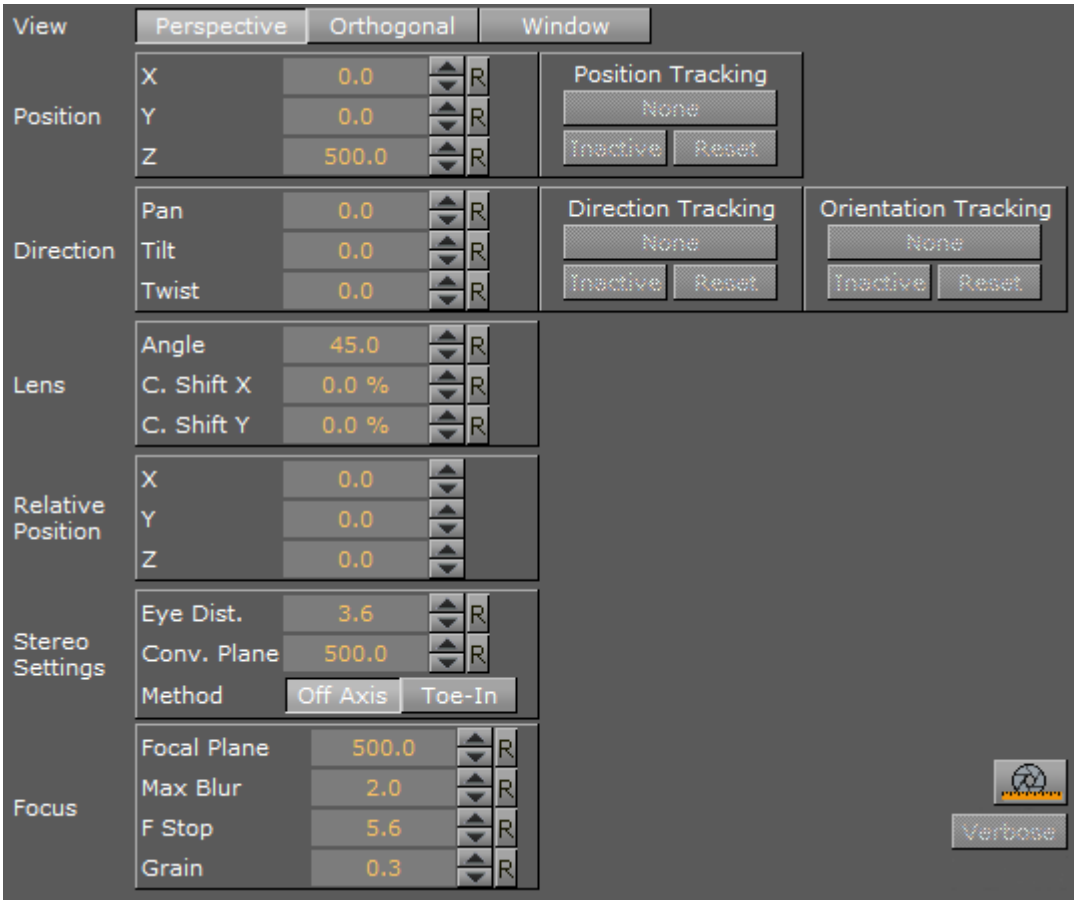
Shows the camera view like looking through a real window. The graphics, through a virtual camera, are adapted according to the real camera movement to produce this behavior. Set parameters for the physical position and rotation of the display unit (see Parameters for Window View).



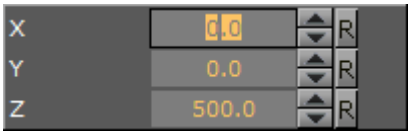
See Also

- Parameters for Perspective View
- Parameters for Orthogonal View
- Parameters for Window View
- Change Camera Parameters in Orthogonal Views

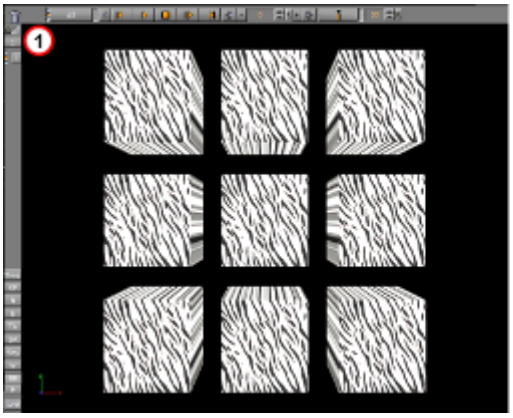
Parameters for Perspective View

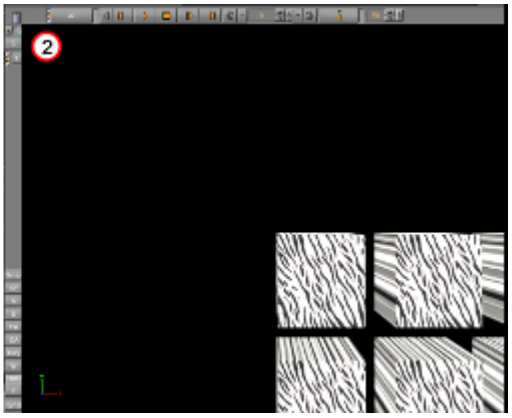


Position

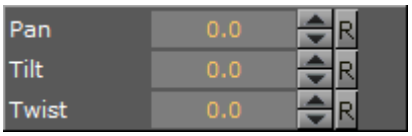


Set the position of the camera along the X, Y, and Z axis. Default setting (1) and modified setting (2)

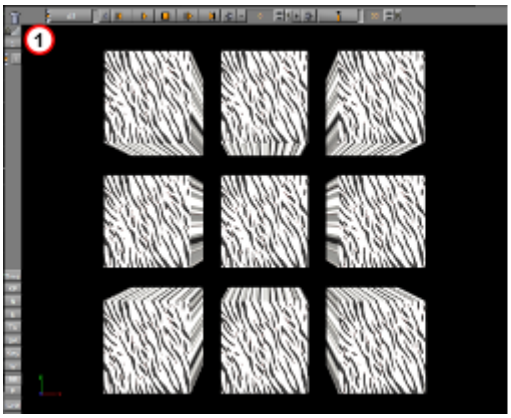


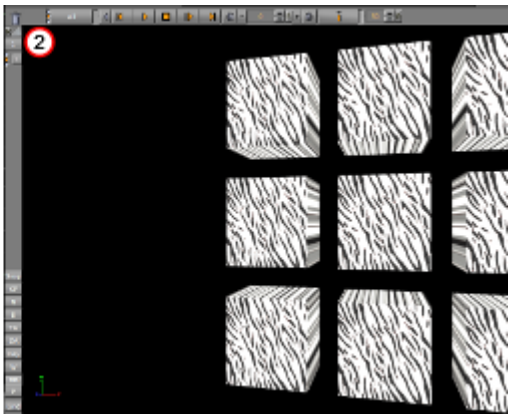


Direction

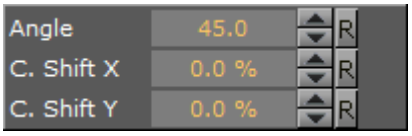


Set the values for Pan, Tilt, and Twist. Pan set to 0 (1). Pan set to 15 (2)



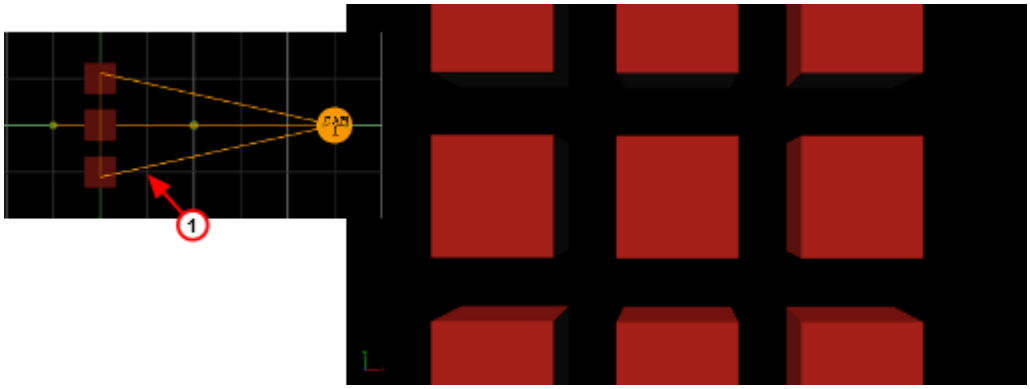


Lens



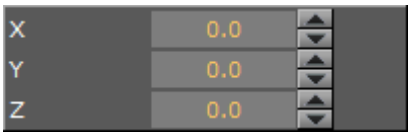
- **Angle**: Sets the Camera 'field of view' angle (1)
- **C. Shift X**: Shifts the center of the lens on the X axis.

- **C. Shift Y**: Shifts the center of the lens on the Y axis.



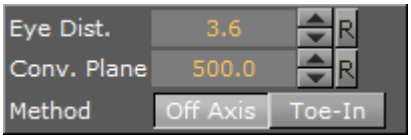
Relative Position

Set the position of the camera along the X, Y, and Z axis, relative to a real camera. Entering a value in for X, Y and Z axis for the Relative Position changes the actual camera position values, but relative to its orientation.



Stereo Settings

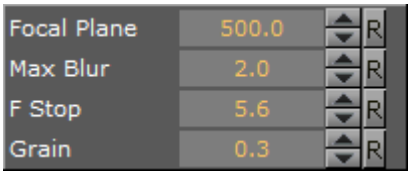
Available if set in **Camera Properties** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).



- **Eye Dist.:** Fine-tunes for individual users. A higher value for will result in more depth whereas a lower value will flatten the image.
- **Conv. Plane:** A higher value for the zero parallax distance means that objects will come closer to the viewer, a lower one will push the graphics to the back.
- **Method:**
  - **Off Axis:** Creates a distortion-free stereo pair. This is the preferred option.
  - **Method Toe-In:** The cameras point slightly inwards towards each other so that the lens axes converge at a single point (“vergence point”). Objects in the same plane as the vergence point will show to be close to the plane of the image, with other objects seeming to be behind or in front of this point.
    These values must be set during production in accordance with the convergence values used for the cameras.

Focus

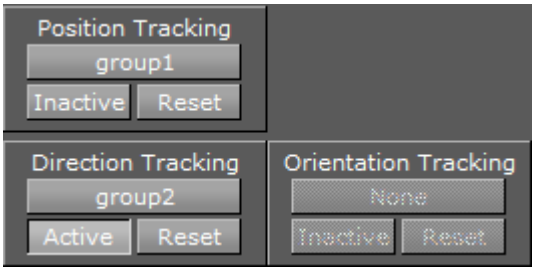
Focus settings are available if the **Depth of Field** parameter is set to **Active**.



- **Focal Plane:** Defines the distance between the camera and objects which are in focus.
- **Max Blur**: Sets the size of maximum blur furthest from the camera. For example, a value of 20 signifies that a pixel is enlarged to a size of 20 pixels.
- **F Stop**: Sets the F-Stop value (relative aperture): The lens model used by the depth of field algorithm mimics a real lens behavior
- **Grain**: Defines the blur level.

Tracking

A camera can be set up to track the Position, and Direction or Orientation of a Container.

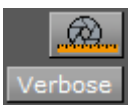


- **Position Tracking:** The position of the camera will be the same as the target containers position.
- **Direction Tracking**: The rotation of the camera will be updated to always point towards the current position of the target container.
- **Orientation Tracking:** The rotation (orientation in space) of the camera will be the same as the target containers rotation (orientation in space).

> ⚠ **Note:** Direction and Orientation tracking cannot be used at the same time. The results would contradict each other (both of them result in a potentially different rotation value for the container).

- **Active/Inactive:** Drag a container to the drop zone to automatically enable camera tracking. To disable tracking, click the **Active/Inactive** button.
- **Reset**: Disables camera tracking.

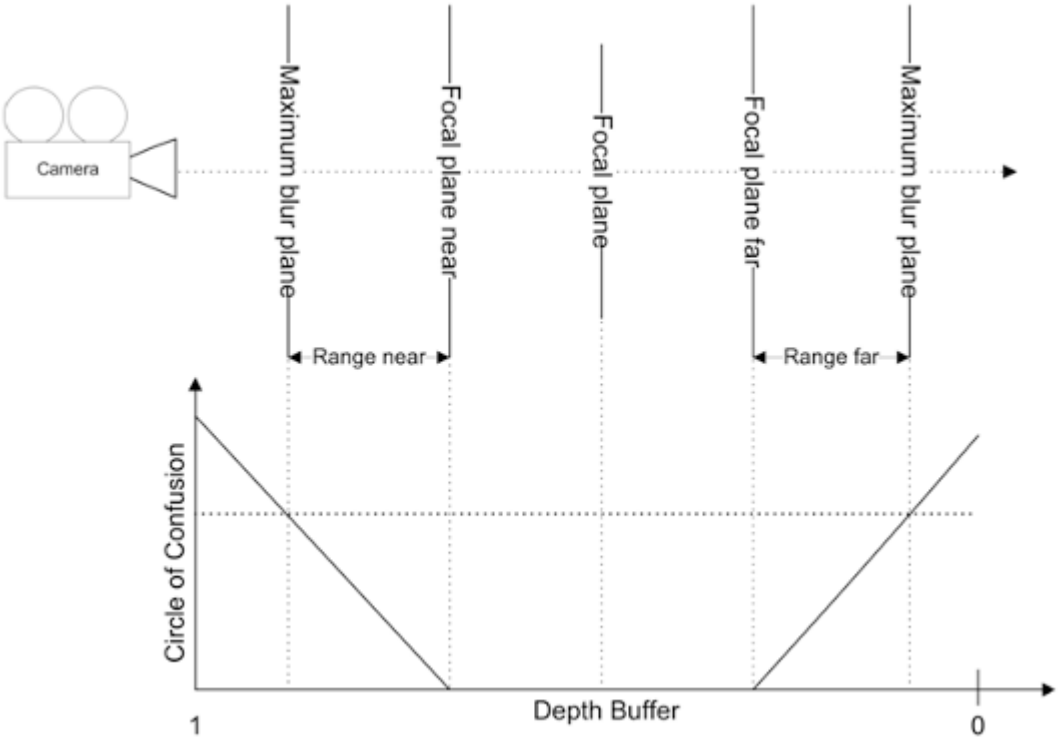
Miscellaneous Buttons



- **Lens File Calibration Editor**: Opens the Lens File Calibration Editor.
- **Verbose**: Linked to cameras in virtual studio. For more information about the Virtual Studio option see the Virtual Studio documentation.

Avoid Artifacts

To avoid artifacts, avoid using maximum blur levels which are higher than 30 pixels. A blur radius of 30 pixels creates rather blurry results which should be sufficient. It is possible to increase this maximum radius, but this would also increase the rendering time.



Set the clipping planes in the range you really need for the scene. So if your scene has only a depth of 200, set the far clipping plane to 250 or a similar value. Since the depth of field shader takes the depth from the depth buffer, this causes more accurate results.

Try to avoid high blur levels for objects before the focal plane, as this can cause unwanted effects.

You can increase performance, but artifacts would show for blur levels lower than 30px.

See Also

- Track Objects with a Camera
- Parameters for Orthogonal View
- Parameters for Window View

Parameters for Orthogonal View

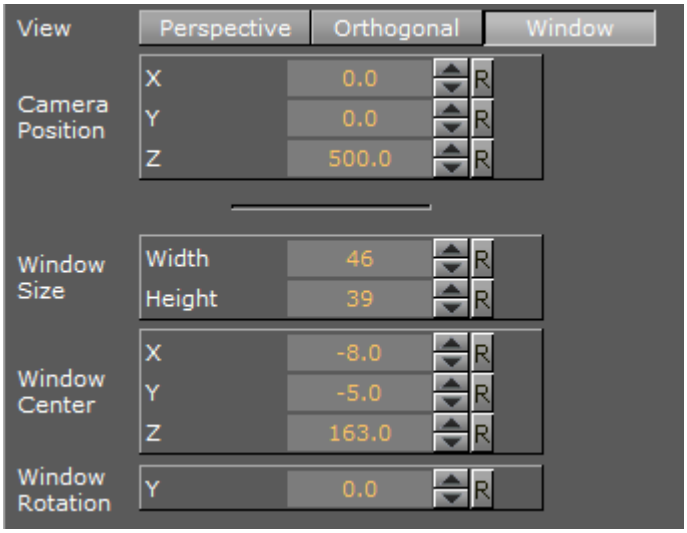
The parameters for Orthogonal view are:

- **Position**: Sets the position of the camera along the X and Y axis.
- **Focus:** See Parameters for Perspective View.

See Also

- Parameters for Perspective View
- Parameters for Window View

Parameters for Window View

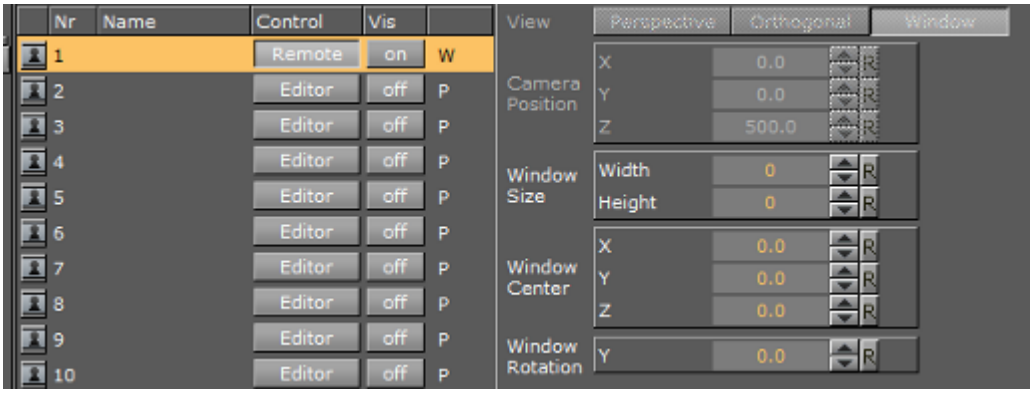


- **Camera Position:** Sets the position of the camera along the X and Y axis.

  > ⚠ **Note:** When Control is set to Remote, Camera Position is not available and the Tracking system controls the Camera Position.

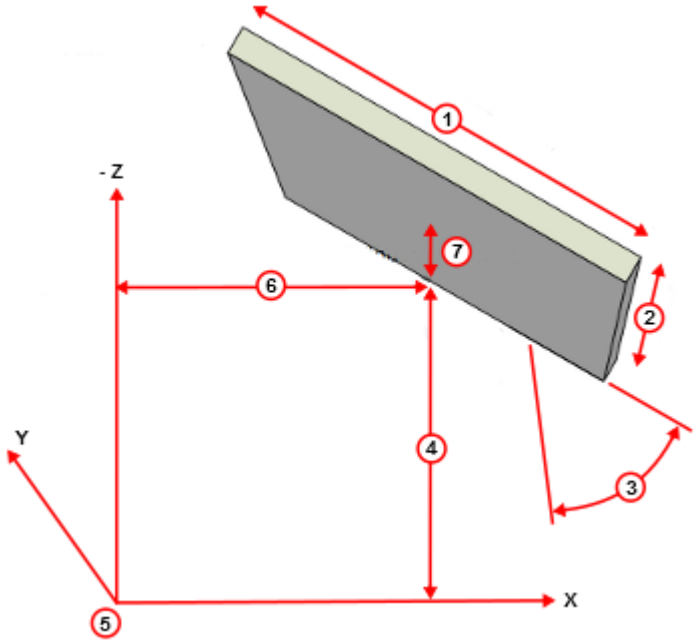
- **Window size:** Sets the window size (set to the size of the display unit).
- **Window center:** Sets the position of the window from the zero position of the Tracking system.
- **Window rotation:** Rotates the window in the Y axis.

Configure a Virtual Window



1. Set the **View** to **Window**.
2. Set the **Control** to **Remote**.
3. Set **Window Size** (1 and 2). The width and height of the video wall, screen or monitor.
4. Set **Window Center** (4, 6 and 7). The distance of the screens center point to the zero point of the studio, defined in the studio setup.

5. Set **Window Rotation** (3).The rotation (y axis) angle of the video wall, screen or monitor, in relation to the coordinate system of the studio.



Item	Description
1	Window Width
2	Window Height
3	Window Rotation
4	Window Center
5	Zero Point of VSS
6	Window Center X
7	Window Center Y

See Also

- Parameters for Perspective View
- Parameters for Orthogonal View

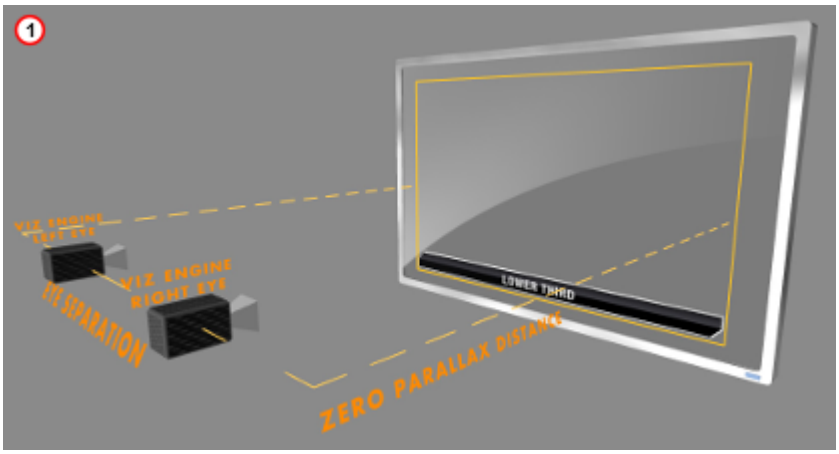
9.2 Stereo Settings

Any Viz Artist animation that was created for 2D broadcast can also be rendered as a stereo pair.

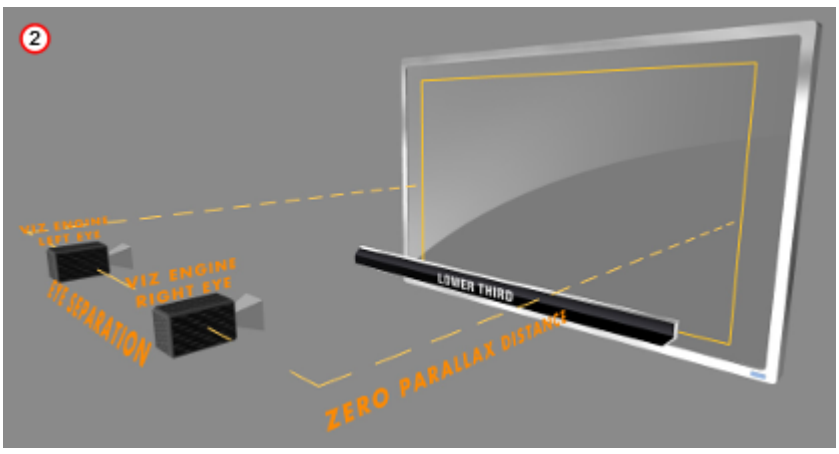
There are two parameters which define how each image is rendered. Both parameters are part of the camera configuration, but they can be changed remotely on the fly as well.

Zero parallax means that the pixels for the left and the right eye are identical. An object which would be positioned at zero parallax distance from the camera would be identical for the left and right eye. The eye separation defines the distance of the cameras for the left and right eye. When looking at the drawing the cameras are separated by the same amount as the eyes of the viewer. A different scaling might be used in the scene and the eye separation must be adjusted accordingly.

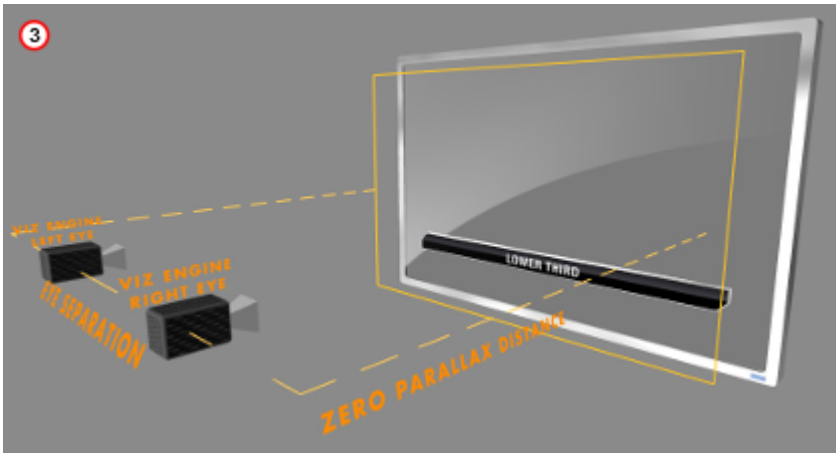
If the scene is built in the same scale as the monitor, with the camera at the same distance from the object as the viewer, and the eyes separated at the same amount an object positioned at the zero parallax, the distance shows to be flat with the screen's surface (1):



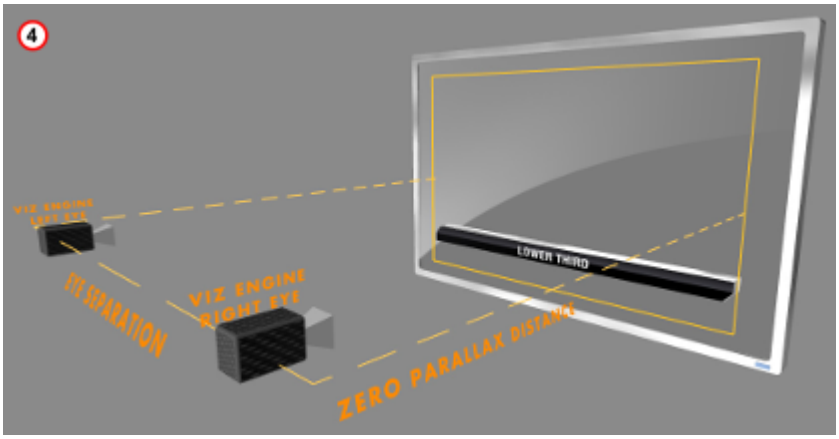
If the zero parallax distance is increased the object moves to the front (2):



If the distance is decreased the object moves away (3):



The eye separation defines how flat or deep an object shows. If the eye separation is increased, the scene shows deeper (4):



A very large eye separation parameter results in an unreadable 3D representation, as the brain cannot mix the images for the left and right eye anymore. Zero eye separation results in a flat representation, like a 2D image.

See Also

- Stereo Settings in Parameters for Perspective View
- Stereoscopy Best Practices

9.3 Stereoscopy Best Practices

The following is a list of what to do and not to do to get the best results.

9.3.1 What to do

- Think of it as a window you are looking through.
- Make sure that single items which form a new item, like rows of a table, are all at the same distance and not scaled to show the same size in 2D.

- Make sure that all graphics which belong to one show are positioned roughly the same distance away from the camera. This way you do not need to adjust the convergence parameters all the time.
- With alioscopy, keep the floating objects not too far from the 0 plane, as this will make them have artifacts.
- For best effect have objects animate slowly forward and then fade to 0 alpha before hitting the screen edge.
- With alioscopy, for text that is supposed to be readable, keep it big. Dropshadow is a very good effect to enhance the sense of depth. 3DS productions use HD signals.
- For a 24" screen the best viewing distance is about 2.8 meters. For a 40" it is about 4.4 meters.
- With alioscopy, look at the scene from the top view and space things out like in a theatre. Depth and various objects in different Z positions will make it more believable: foreground, middle ground and background.
- Wobble effects are fine as long as they do not exaggerate the effect, otherwise you get the paper bag effect. This again depends on how far or close to the camera the object is positioned.

9.3.2 What not to do

- Do not use transparency where it goes over video. It's OK if a transparent graphics object is over another graphical item if the distance is not too large.
- Do not use too much depth. Try to keep the items separated not too far from each other. The Z space available for graphics is limited.
- Do not create animations which come in from the side, especially when the final result will be floating in front of the screen (this is very disturbing).
- Do not use moving flairs or effects.
- Do not have any objects pulsating forth and back in Z space as a part of a loop as this will lead to the paper bag effect.

See Also

- Stereo Settings
- Stereoscopic Output Using Shutter Glasses

9.4 Stereoscopic Output Using Shutter Glasses

It is very easy to set up a working 3D environment for a designer to start creating stereoscopic scenes. The demands on the hardware are moderate.

9.4.1 Hardware Requirements

- Any workstation
- Any NVIDIA Quadro graphics card with a 3-pin DIN connector
- NVIDIA 3D vision or equivalent
- Monitor with a refresh rate >= 100Hz

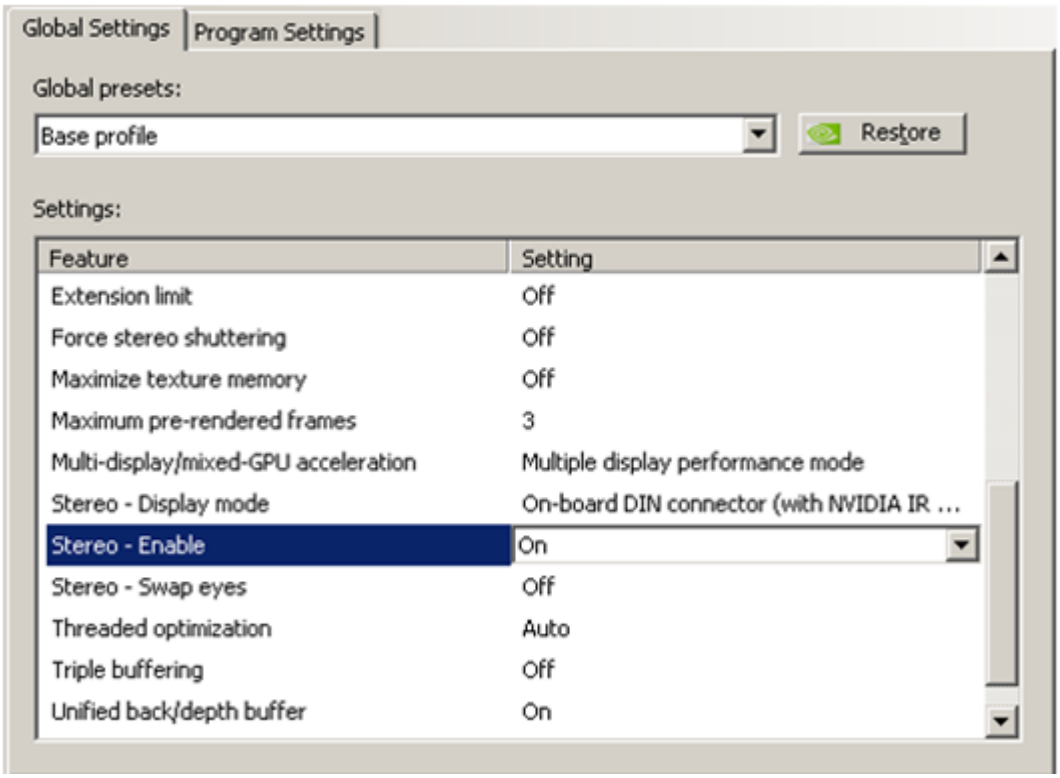
9.4.2 Software Requirements

- Viz Artist/Engine 3.3 or later
- The display must be set to a refresh rate >= 100Hz

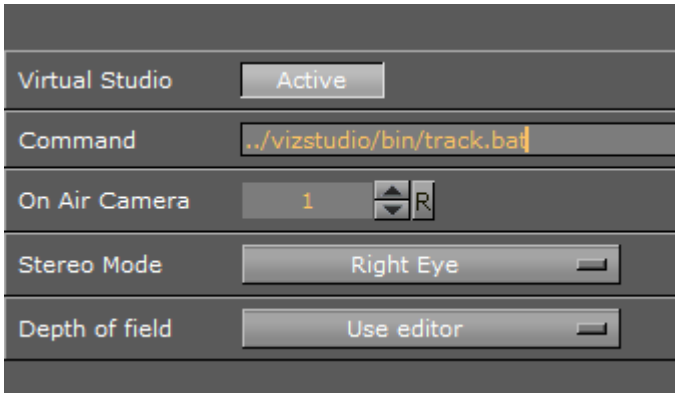
9.4.3 To Set Up a 3D Working Environment

1. In the Windows video board 3D settings, Set **Stereo - Enable** to **On**
2. Select the appropriate **Stereo - Display mode** for each stereo device.

> ⚠ **Note:** For NVIDIA's 3D Vision this is 'On-board DIN connector'.



3. In **Camera** settings (see the **Configuring Viz** section of the Viz Engine Administrator Guide) set **Stereo Mode** to **Quad Buffered**.



Also it is possible to use an autostereoscopic monitor that supports sub-fields or side-fields, such as Miracube's c190 series. The drawback with these displays is that you need to go on-air to get a full-screen image, which is halved in resolution.

See Also

- Camera Editor
- Stereo Settings
- Stereoscopy Best Practices
- Change Camera Parameters in Orthogonal Views
- Track Objects with a Camera
- Receive Tracking Data from a Real Camera
- Copy Properties from one Camera to Another
- Camera Selection
- Camera Animation

9.5 Change Camera Parameters In Orthogonal Views

Camera parameters can be modified in the predefined orthogonal views.

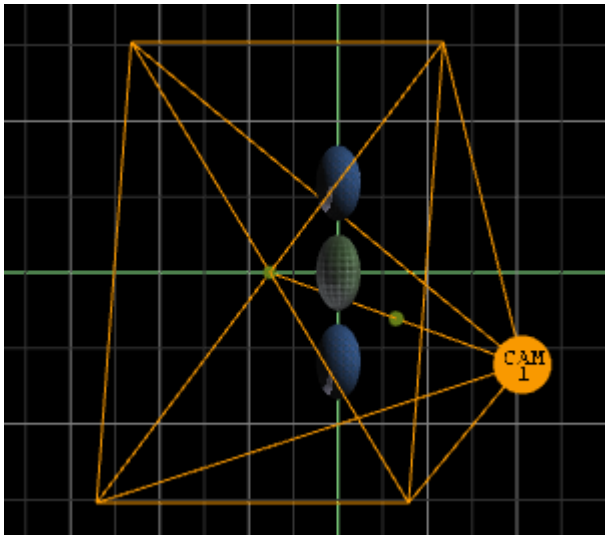
This section contains information on the following procedures:

- To View the Predefined Orthogonal Views
- To Change the Camera Views

9.5.1 To View the Predefined Orthogonal Views

- Click **Views**.
  The camera is shown as an orange circle (3) in the predefined orthogonal views. Two small, green, circular handles (1 and 2) make it possible to modify the position along the X, Y and

Z axis of the camera, and also the values for pan and tilt.



9.5.2 To Change the Camera Views

- Drag handle (1): Change the pan and tilt while the position of the camera stays fixed.
- Drag handle (2): The camera (3) and handle (1) will be dragged along, so that pan and tilt will not be altered.
- Drag the camera (3): Change the position of the camera while handle (1) stays fixed. The values for pan and tilt will also change.

See Also

- Camera Editor
- Stereo Settings
- Stereoscopy Best Practices
- Stereoscopic Output Using Shutter Glasses
- Track Objects with a Camera
- Receive Tracking Data from a Real Camera
- Copy Properties from one Camera to Another
- Camera Selection
- Camera Animation

9.6 Track Objects With A Camera

A virtual camera can be set up to track Containers both regarding position and direction. Position Tracking means that the position of the camera will be in the center of the container. Direction Tracking direction means that the camera will look at the center of the container, for example throughout an animation.

There are three options to configure the parameters view, The options depend on the number of tracking objects:

- **Zero Tracking Objects:** Only Position X and Y can be modified.

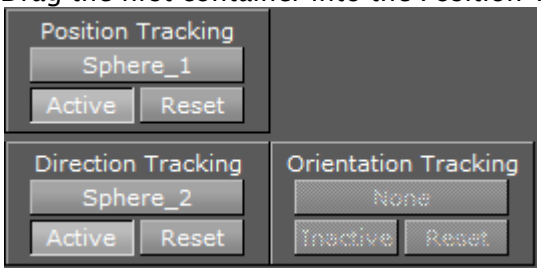
- **One Tracking Object:** When a Container is dragged to the Position Tracking drop zone, all the parameters are grayed out. In the Properties Panel, switch from **Camera** to **Container** to modify Positions X, Y and Z.
- **Two Tracking Objects:** Only the Lens and Relative Position parameters can be modified.

9.6.1 To Track an Object

1. In the Scene Tree create two Objects, one to track position (Sphere_1) and one to track rotation (Sphere_2).

   > ⚠ **Tip:** It is better to use objects with a low polygon count, like a rectangle, as a tracking object. High polygon tracking objects, such as a sphere, make the scene heavy and slow.

2. Drag the first container into the Position Tracking drop zone.



3. Drag the second container into the Direction or Orientation Tracking drop zone.

   > ⚠ **Note:** Direction and Orientation tracking cannot be used at the same time. The results would contradict each other (both of them result in a potentially different rotation value for the container).

4. Modify the camera parameters:
   - Select the Position and, or, Direction tracking object in the Scene Tree, and change the position and/or direction in the Container properties (select **Container** from the Properties Panel).

See Also

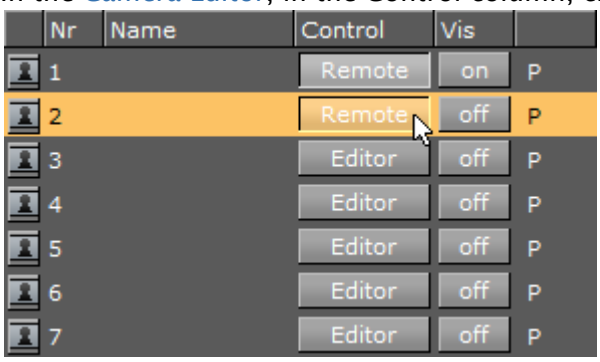
- Camera Editor
- Stereo Settings
- Stereoscopy Best Practices
- Stereoscopic Output Using Shutter Glasses
- Change Camera Parameters in Orthogonal Views
- Receive Tracking Data from a Real Camera
- Copy Properties from one Camera to Another
- Camera Selection
- Camera Animation
- **Tracking** in Camera Editor Right Panel

9.7 Receive Tracking Data From A Real Camera

In virtual studio scenarios, as the real cameras moves, the virtual camera should move in the same manner.

9.7.1 To Receive Tracking Data from a Real Camera

- In the Camera Editor, in the Control column, click the **Editor** button to set it to **Remote**.



  For more information about the Virtual Studio option, see the Virtual Studio documentation.

See Also

- Camera Editor
- Stereo Settings
- Stereoscopy Best Practices
- Stereoscopic Output Using Shutter Glasses
- Change Camera Parameters in Orthogonal Views
- Track Objects with a Camera
- Copy Properties from one Camera to Another
- Camera Selection
- Camera Animation

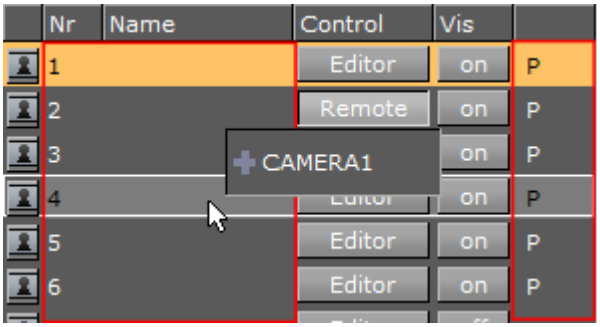
9.8 Copy Properties From One Camera To Another

The properties applied to one camera can be copied and applied to another camera.

9.8.1 To Copy Camera Properties

- In the Camera Editor, click the source camera and drop it on the target camera.
  The properties of the source camera will be copied onto the target camera.

⚠ **Note:** Click on or drop the source camera in the areas shown below.



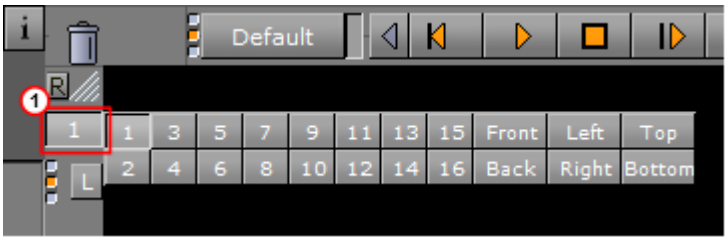
See Also

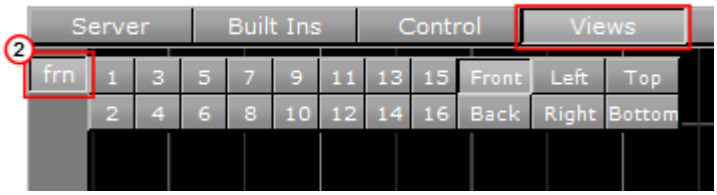
- Camera Editor
- Stereo Settings
- Stereoscopy Best Practices
- Stereoscopic Output Using Shutter Glasses
- Change Camera Parameters in Orthogonal Views
- Track Objects with a Camera
- Receive Tracking Data from a Real Camera
- Camera Selection
- Camera Animation

9.9 Camera Selection

A Scene can be viewed from various camera angles and views, with 16 editable cameras and six predefined orthogonal views available.

In the Scene Editor and Scene Editor Views the Camera Selection button (1 and 2) shows which camera or view is currently selected and used. Click the button to open the Camera Selection panel.



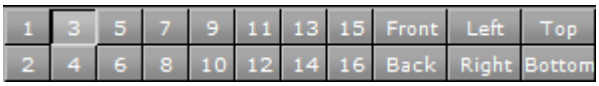


Each Camera can also be selected or deselected in the Camera Editor.

This section contains the following topics:

- Camera Selection Panel Properties
- Camera Selection in the Scene Editor Views

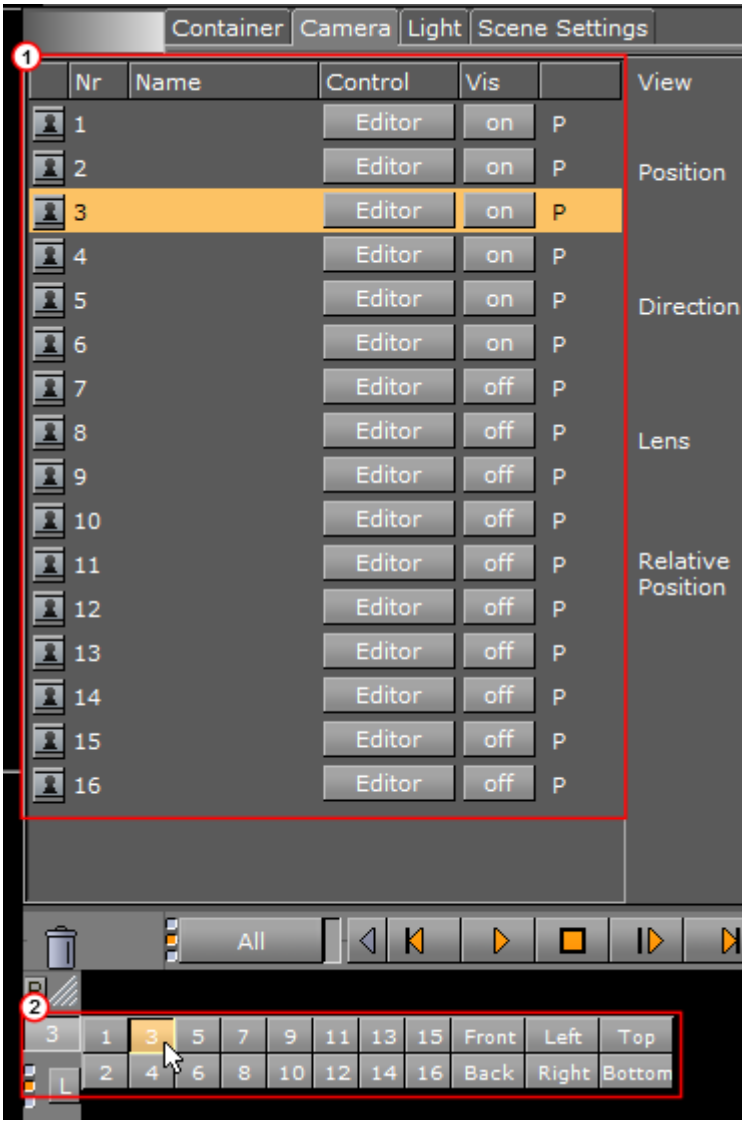
9.9.1 Camera Selection Panel Properties



Note: The position and direction of the 16 editable cameras can be defined in the Camera Editor Panel.

- **1-16**: Selects one of the 16 different cameras.
- **Front**: Shows output through orthogonal camera in front of the scene.
- **Back**: Shows output through orthogonal camera behind the scene.
- **Left**: Shows output through orthogonal camera at the left side.
- **Right**: Shows output through orthogonal camera at the right side.
- **Top**: Shows output through orthogonal camera above the scene.
- **Bottom**: Shows output through orthogonal camera below the scene.

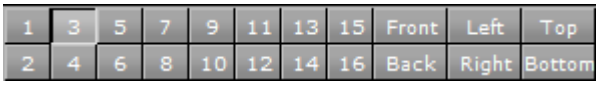
A Scene can be viewed from different camera angles and views in the Scene Editor. You can select different angles or views using the Camera Editor (1) or the Camera Selection panel (2). Camera Selection is displayed in the Scene Editor.



To Select a Different Camera Angle or View

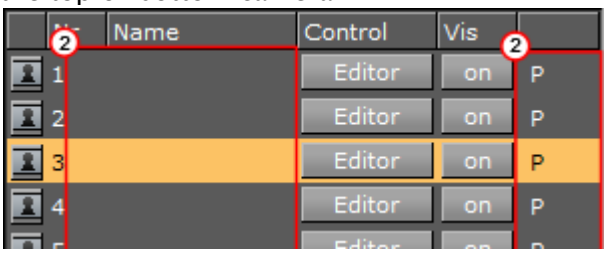
1. Click the Camera Selection button (1), or open the Camera Editor.

2. Select a Camera angle or view:
   - In the Camera Selection panel select a camera (1 to 16) or a camera view. To close the Camera Selection panel, click the Camera Selection button again.



or

- To select a camera in the Camera Editor, 1) click on the required camera (2), 2) use the keyboard up/down arrows, or 3) use the keyboard Page Up/Page Down keys to go to the top or bottom camera.

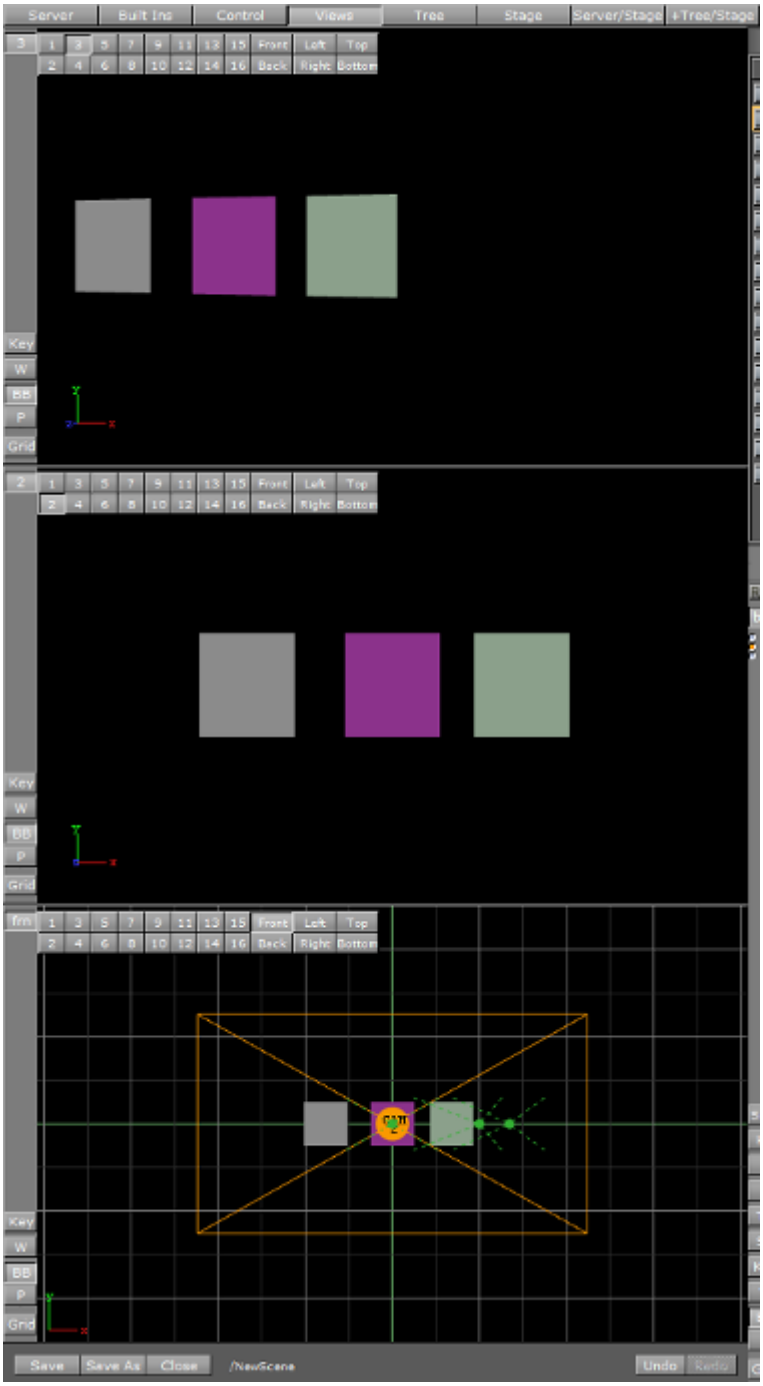


or

- With the mouse cursor over the Scene Editor use the keyboard or numeric keypad.

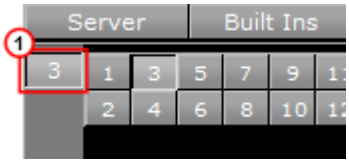
9.9.2 Camera Selection in the Scene Editor Views

A Scene can also be viewed from different camera angles and views, in the Scene Editor Views.

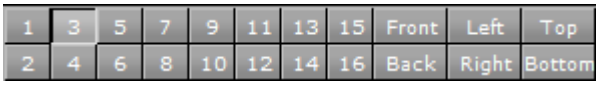


To Select a Different Camera Angle or View

1. In each view click the Camera Selection button (1), as required.



2. Select a Camera angle or view:
   - In the Camera Selection panel select a camera (1 to 16) or a camera view.

     or
   - With the mouse cursor over a Scene View, use the keyboard or numeric keypad.

9.10 Camera Animation

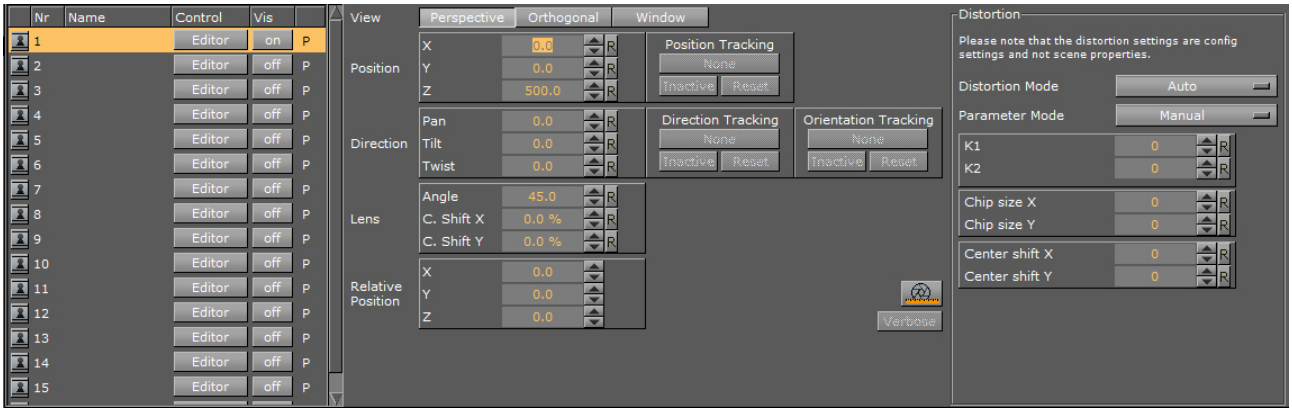
All camera parameters can be animated in the same way as Create Animations in general. The process is roughly as follows:

- Set a value in the Camera editor
- Add a Key Frame
- Change the value
- Add a Key Frame again

  Animations can also easily be created in one of the orthogonal views. There the cameras show as symbols and by repositioning them and adding Key Frames, animation sequences are created. For more information, see Change Camera Parameters in Orthogonal Views .

More advanced camera animations are normally created with the camera tracking feature, as in Track Objects with a Camera.

9.11 Advanced Lens Distortion



Here, you find the following information:

- Important changes compared to previous versions
- Distortion algorithms
- Parameter providers
- Configuration file flags
- Lens compatibility mode

Lens distortion replaces the lens deformation, and was introduced with Viz Artist and Engine 3.9. You can now select the distortion algorithm and distortion parameters provider for each individual

camera. There is no longer a need for multiple configuration flags, such as `render_scale` and Virtual Studio.

The distortion algorithm defines how the rendered image is distorted, and the parameters are the input to the distortion algorithm. Typical algorithms expect “k1”, “k2”, “centershift_x”, “centershift_y”, “chipsize_x”, “chipsize_y”, etc.

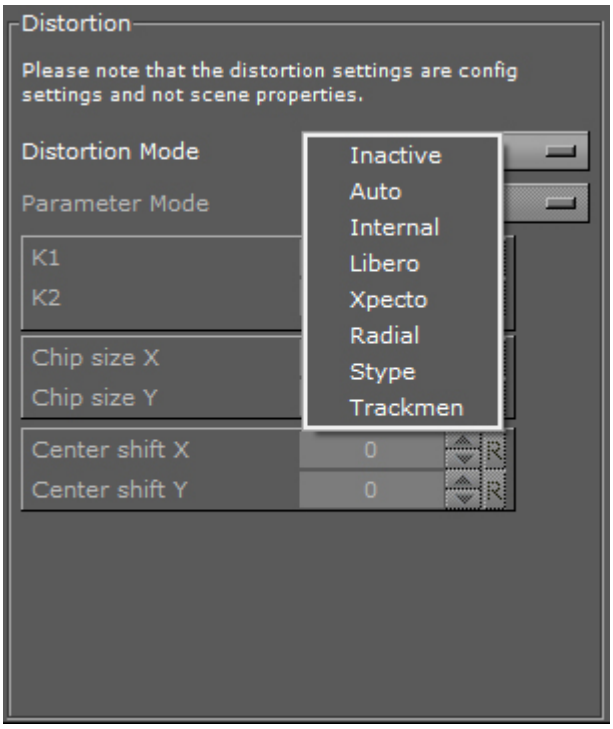
9.11.1 Important changes compared to previous versions

- Activation of lens distortion does not depend on other settings, such as `render_scale` or entering Virtual Studio mode.
- You can explicitly select the algorithm and parameter source per camera, also in Viz Artist. There is an **auto** mode, which attempts to determinate the correct algorithm and parameter mode based on the available data.
- Now called **Lens Distortion** instead of **Lens Deformation**.
- `render_scale` is now correctly applied to FieldOfView. There is no longer a “jump” when enabling or disabling post processing (**PP**) in the Scene Editor.

> **Technical note**: In previous versions, FOV was adapted when data was received from the external camera. The received value was already modified to compensate the `render_scale` setting. This could cause wrong results when the setting was not considered for rendering, such as when disabling post processing effects with the **PP** button. In 3.9.0, the FOV is untouched and modified on the fly while rendering if necessary.

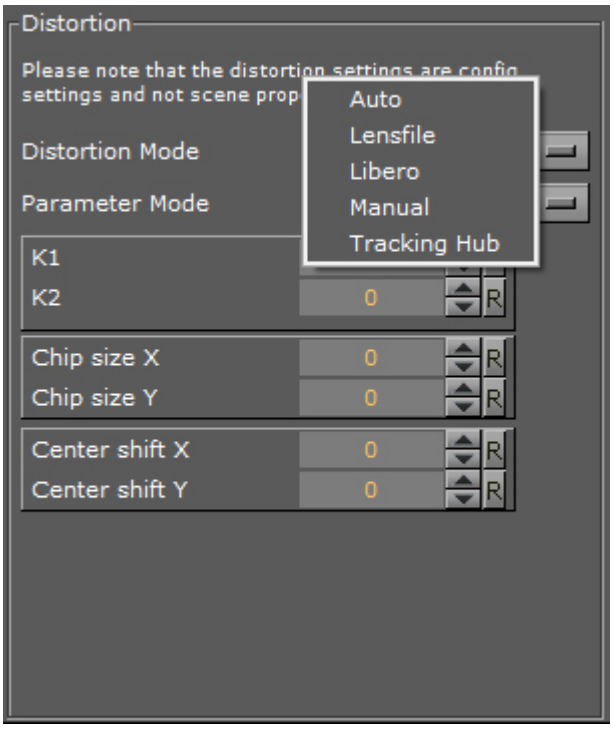
- Previous versions rejected distortion values if the distortion was too strong, if pixels from the outside of the rendered area would have been moved inside the visible area. The `render_scale` defines how much of the border around the visible area should be rendered, to be able to “move” parts from outside to the inside - think of the barrel distortion! If distortion values were too high, and non-available pixels from the outside would have been distorted “into the screen”, it simply used the last valid distortion parameters instead. This could lead to confusion, as the distortion did not use the real values. Now, the distortion uses the values provided, even if some invisible areas show up on the borders. If you see unexpected black or transparent areas around the border, consider increasing the `render_scale`. You can do this by editing the entry for **render_scale** in the configuration file. The default value is `0`, but you can change it as an example to `1.2` or even higher.
- New distortion algorithms:
  - **Radial:** A classic algorithm.
  - **Stype** and **Trackmen**: Implemented according to specifications from tracking data providers.

9.11.2 Distortion algorithms



Auto	Automatically selects one of the algorithms below, depending on available data.
Inactive	-
Internal	The internal distortion algorithm used in 3.8.x, when reading data from lens file.
Libero	The distortion algorithm used in 3.8.x when connected to Viz Libero or Viz Arena.
XPECTO	The distortion algorithm used in 3.8.x with external data.
Radial	Radial lens distortion, based on the Brown-Conrady model. See https://en.wikipedia.org/wiki/Distortion_(optics)#Software_correction.
Stype	Lens distortion as expected by Stype.
Trackmen	Lens distortion as expected by Trackmen.

9.11.3 Parameter providers



Auto	Automatically selects one of the providers below, depending on the received data.
Lensfile	Reads zoom/focus from external and calculates all other parameters from lens file.
Libero	Reads **K1/K2** from Viz Arena or Viz Libero.
Manual	Uses parameter from the configuration. See `MAIN*CONFIGURATION*CAMERA*1*DISTORTION*PARAMS*MANUAL SET/GET name value`
Tracking Hub	Reads parameters from Tracking Hub.

9.11.4 Configuration file flags

```
## Possible distortion modes: 0=INACTIVE, 1=AUTO, 2=INTERNAL, 3=XPECTO, 4=LIBERO,
5=RADIAL, 6=STYPE, 7=TRACKMEN

## Possible distortion param modes: 0=AUTO, 1=MANUAL, 2=LENSFILE, 3=Tracking Hub,
4=LIBERO


camera1_distortion_mode = 0

camera1_distortion_param_mode = 0

camera1_distortion_param_manual = {"centershift_x": 0.000000, "centershift_y":
0.000000, "chipsize_x": 8.800000, "chipsize_y": 5.200000, "k1": 0.000000, "k2":
0.000000}

camera2_distortion_mode = 0

camera2_...


# lens compatibility mode, see note

use_lens_compatibility_mode=0
```

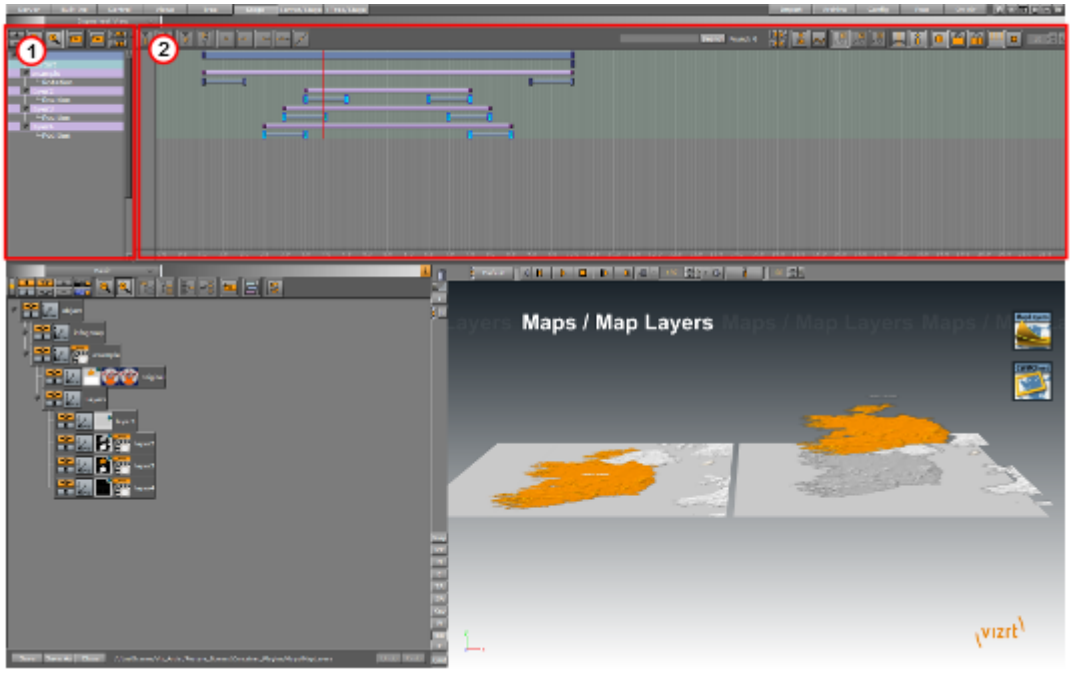
9.11.5 Lens compatibility mode

⚠ **Important!** For new lens calibrations, always use `use_lens_compatibility_mode=0`.

Until Viz Artist and Engine 3.8.2, the result of the lens deformation was affected by the **hd_halfheight** setting. This was corrected in 3.8.3, and the lens deformation now generates the same result for `hd_halfheight=1` and `hd_halfheight=0`. However, if you have lens files that were calibrated with `hd_halfheight=1` using an older version, the result will now differ. In this case, enable `use_lens_compatibility_mode`.

10 The Stage For Animation

The Stage is the interactive user interface to create animations.



The Stage is divided in two:

- The Stage Tree Area (1): Shows a representation of all animated items in the current scene.
- The Stage Editor (2): Shows the animated items as bars along the time-line.

> ⚠ **Tip:** To resize the two areas, drag the vertical separator.

The Stage can be viewed from three different views. In the Main Menu Left click on either:

- **Stage**
- **Server/Stage**
- **+Tree/Stage**

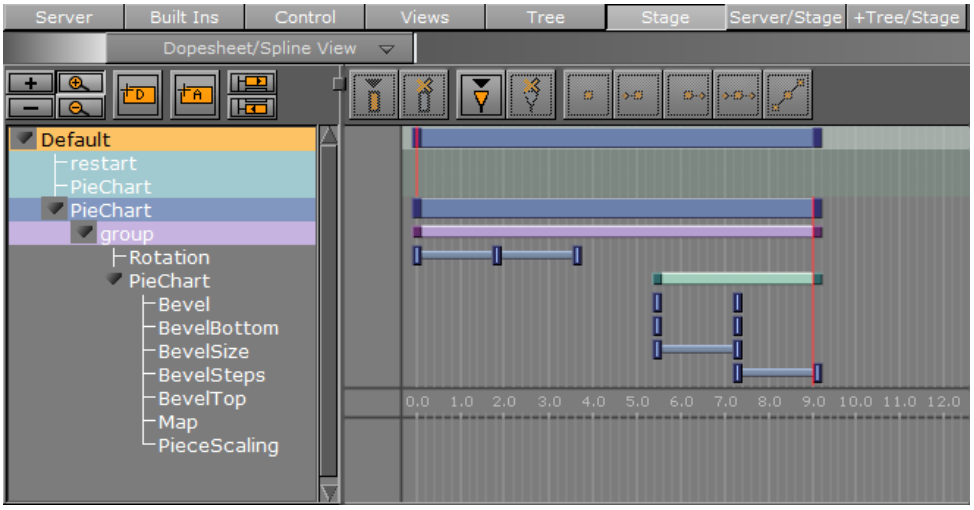
This section provides a detailed description of the following Stage for Animation topics:

- Stage Tree Area
- Stage Editor
- Time-line Editor
- Time-line Marker
- Artist Director Control Panel
- Director Editor
- Master Clip
- Actor Editor
- Channel Editor
- Dopesheet Editor

- Spline Editor
- Stage Object Editor
- Key Frame Editors
- Event Editor

10.1 Stage Tree Area

The Stage Tree area shows a representation of all animated items in the current scene.

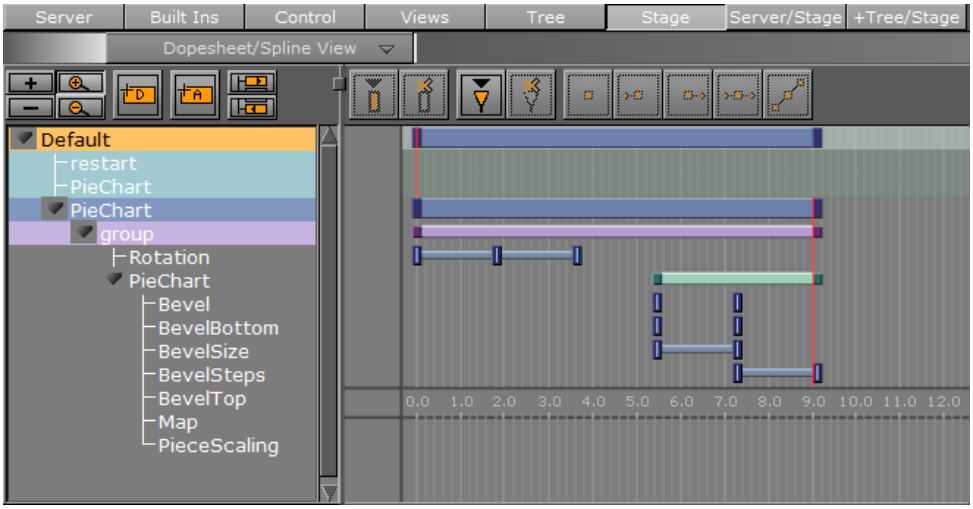


This section contains information on the following topics:

- Stage Tree
- Stage Tree Bar
- Stage Tree Menu

10.1.1 Stage Tree

The Stage Tree shows an overview of all Directors, Actors, and Channels that are used to create the animations in the current scene.



Expand or Collapse Stage Tree Branch

Click on the arrow to the left of the item to expand or collapse a branch.

- **Expand a Stage Tree Branch**: Click the (Expand) icon
- **Expand all Stage Tree Branches**: Click the top (Expand) button or press <Alt> and click the Expand icon to the left of the branch name
- **Collapse a Stage Tree Branch**: Click the (Collapse) icon to the left of the branch name
- **Collapse all Stage Tree Branches**: Click the top (Collapse) button

10.1.2 Stage Tree Bar

The Stage Tree bar is located above the Stage Tree.

The Stage Tree bar contains these options:

- **Expand / Collapse:** Expands or collapses all branches in the Stage Tree.
- **Zoom Out / In:** Scales the Stage to provide a better overview of complex animations. Scales the Stage back to standard display.
- **Add New Director:** Adds a new Director to the Stage Tree. The icon can also be dragged to add a new Director on a specific position.
- **Add New Action:** Adds a new action to a Director.
- **Jump to Next / Previous Animation:** Selects the next or previous channel in the Stage Tree if an Actor (Container) has multiple animation channels placed on various Directors.

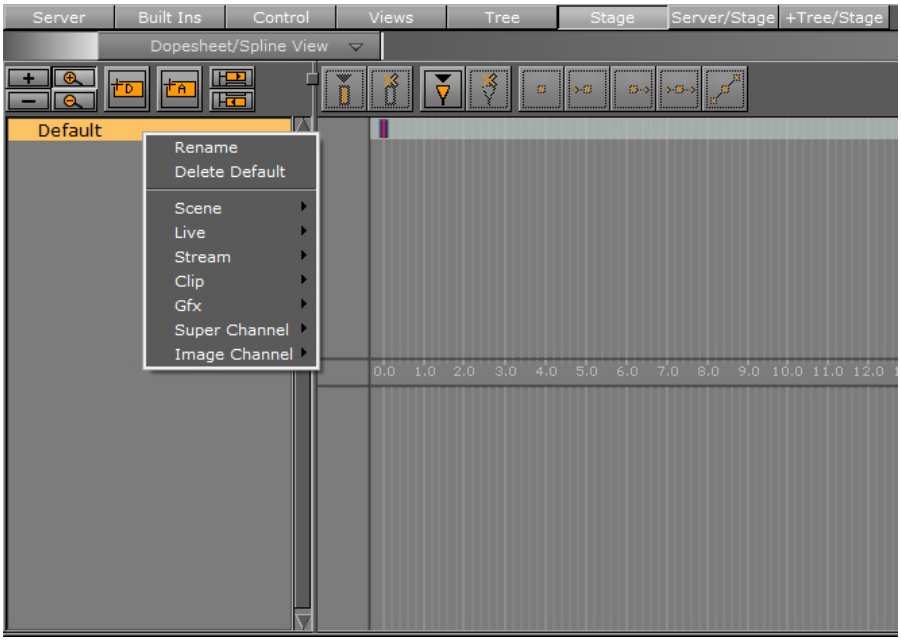
10.1.3 Stage Tree Menu

This section contains information on these topics:

- Directors
- Actors
- Channels
- Action Channels

Directors

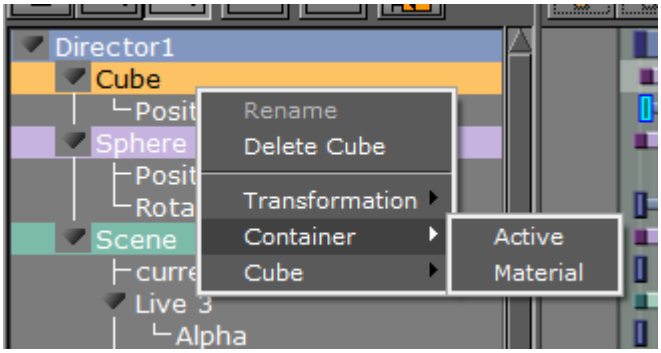
Right-click a Director to open a context menu. For more information about Directors, see Directors.



- **Rename:** Renames the Director.
- **Delete Director:** Deletes the Director.
- **Scene:** Adds a camera channel for camera cuts.
- **Live:** Adds any of the 8 Live VideoInputs
- **Stream:** Adds an IP Stream Channel.
- **Clip:** Adds a Clip Channel.
- **GFX:** Adds a GFX Channel.
- **Super Channel:** Adds a Super Channel.
- **Image Channel:** Adds an Image Channel.

Actors

Right-click an Actor in the Stage Tree to open its context menu (see also Actors).



- **Rename:** Not available.
- **Delete Actor:** Deletes the actor.
- **Transformation:**
  - **Position:** Creates a position channel for the actor, and automatically adds two default Key Frames.
  - **Rotation:** Creates a rotation channel for the actor, and automatically adds two default Key Frames.
  - **Scaling:** Creates a scaling channel for the actor, and automatically adds two default Key Frames.

    ⚠ **Note:** If a transformation channel is disabled (grayed out), the channel is already available for the actor in this Director.

- **Container:**
  - **Active:** Creates an active channel for the actor, and automatically adds two default Key Frames. The active status can be modified in the Key Frame Editors. If an active Key Frame is set to Value=**Off**, the whole actor (container) including all animations will be hidden in the Scene Editor. The action remains hidden until another Key Frame shows in the time-line with Value=**On**.
  - **Material**: Creates a material channel for the actor, and automatically adds two default Key Frames. The material can be modified in the Key Frame Editors.

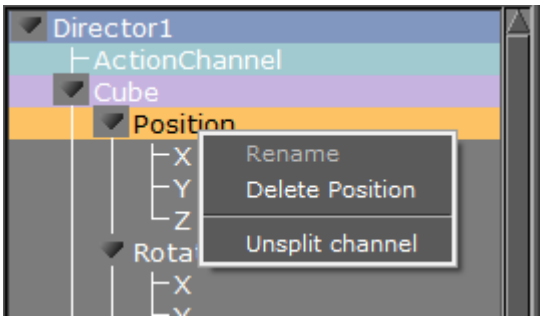
    ⚠ **Note:** The material option is only available if a material is added to the container.

- **Texture:** This option varies according to the item that is available on the actor.

  ⚠ **Example:** If an image is added to the container, various texture options become available. If a built-in geometry is added, for example a cube, various cube options become available. If a font is added to the container, various text options become available. For details about options for the various items, see the respective editor sections, for example Image Editor and Fontstyle Editor.

Channels

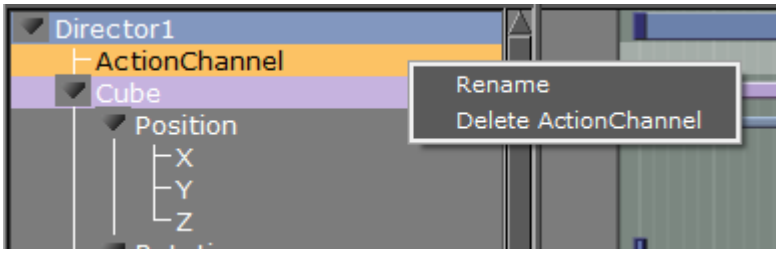
Right-click a channel, in the Stage Tree, to open the channel context menu (see also Channels).



- **Rename:** Not available.
- **Delete Channel:** Deletes a channel.
- **Split/Unsplit Channel:** Shows or closes Transformation Channel parameters.

Action Channels

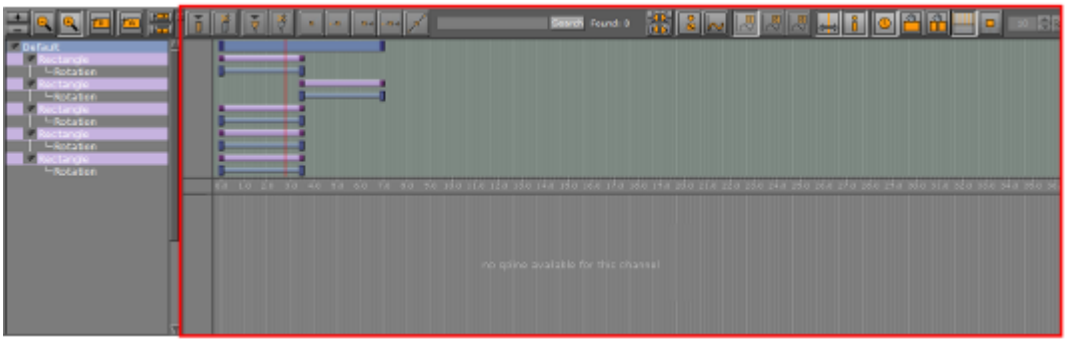
Right-click an Action Channel, in the Stage Tree, to open the Action Channel context menu (see also Action Channels).



- **Rename:** Makes it possible to rename an Action Channel.
- **Delete Action Channel:** Deletes an action channel.

10.2 Stage Editor

The Stage Editor area shows the animated items as bars and splines along the time-line.

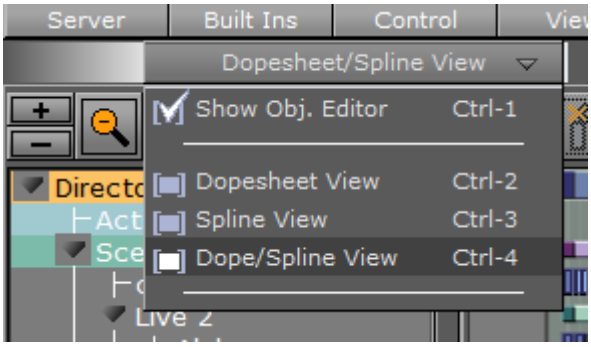


This section contains information on the following topics:

- Stage Editor Menu
- Stage Editor Bar

10.2.1 Stage Editor Menu

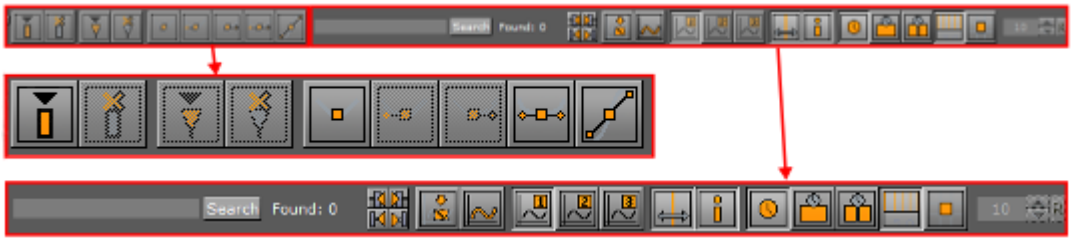
Above the Stage Tree Area is a menu that defines the appearance of the .Stage Editor v3.11.



- **Show Obj. Editor:** Enables/disables the Stage Object Editor at the right side of the Stage.
- **Dopesheet View**: Shows only the Dopesheet Editor.
- **Spline View**: Shows only the Spline Editor.
- **Dope/Spline View**: Shows both the Dopesheet Editor and Spline Editor.

10.2.2 Stage Editor Bar

The Stage Editor bar is located above the Dopesheet Editor/Spline Editor.

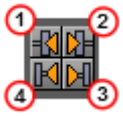


-  **Add Key Frame:** Adds a Key Frame at the current time/frame/field (see Key Frames).
-  **Delete Key Frame:** Deletes the selected Key Frame.
-  **Add Stop/Tag:** Adds a stop/tag Event to the selected Director at the current time/frame/field.
-  **Delete Stop/Tag:** Deletes the selected stop/tag Event.
-  **Linear Left and Right:** Sets the left and right spline handles in the selected Key Frame to linear.

-  **Smooth Left and Linear Right:** Sets the spline handles in the selected Key Frame to smooth left and linear right.
-  **Linear Left and Smooth Right:** Sets the spline handles in the selected Key Frame to linear left and smooth right.
-  **Smooth Left and Right:** Sets the left and right spline handles in the selected Key Frame to smooth.
-  **Handles Tangential:** Sets the left and right spline handles in the selected Key Frame to tangential.
- 

Search: Searches the name of a Key Frame.
- 

Items Found: Shows how many items were found.
-  **Search Results**: The arrow buttons can be used to search for the first, next, previous and last Key Frame matching the search criteria:

  - 1: **Previous** Key Frame
  - 2: **Next** Key Frame
  - 3: **First** Key Frame
  - 4: **Last** Key Frame
-  **Zoom to Standard:** Zooms the Dopesheet Editor and Spline Editor to standard view.
-  **Zoom to Whole Spline:** Zooms the Dopesheet Editor and Spline Editor to fit the animation time-line of the current Channel.
-  **Show Spline 1, 2, or 3:** Shows spline 1, 2, or 3 in the Spline Editor. The splines refer to animations performed on various axes. If only one axis is animated, only the spline 1 option is available.

  ⚠ **Example:** When animating a rotation, each spline describes the rotation around one axis. Spline 1 shows the rotation around axis X, spline 2 shows axis Y, and spline 3 shows axis Z.

-  **Animate Time-line Marker:** Moves the Time-line Marker along the Dopesheet Editor and Spline Editor while playing the animation when enabled.

> ⚠ **Tip:** When computing performance should be reduced to a minimum, this option should be disabled.

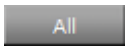
-  **Show Info:** Shows time-line information as a tool tip when holding the pointer over an item in the Dopesheet or Spline Editor, for example a Key Frame, when enabled.
-  **Seconds/Frames/Fields:** Switches the time scaling between seconds, frames, and fields. The time scaling is available in the bar separating the Dopesheet and Spline Editor.
-  **Show Grid:** Shows a grid of vertical lines in the Dopesheet and Spline Editor, making it easier to position items accurately along the time-line when enabled.
-  **Snap to Grid:** Shows the snap to grid feature in the Dopesheet and Spline Editor when enabled. This feature shows vertical lines at the field interval defined in the grid width option. By using this feature, Key Frames and other items can be dragged precisely along the time-line, as it is not possible to drop items between the grid lines.

> ⚠ **Example:** If the snap to grid feature is enabled and the grid width is set to 10, it is only possible to drag Key Frames to field 10, 20, 30, and so on. If a Key Frame should be placed at field 23 while the snap to grid feature is enabled, this must be manually entered in the Key Frame Editors.

-  **Grid Width:** Defines the width of the snap to grid in fields.

10.3 Time-Line Editor

The Time-line editor is located above the Scene Editor. It gives various options to control animations.

-  **Layer Indicator**: Shows the current layer, as one of three vertical indicators, in the Director Control Panel. The three layers are Front, Main and Back.
-  **Director:** Defines the selected (active) Director and opens the Director Control Panel.

> ⚠ **Note:** The settings in the Director Control Panel are not saved with the scene. If only one Director is selected, the button will be labelled with the Director name. If multiple directors are selected, the button will be labelled *User Defined*. If all directors are selected, the button will be labelled *All*.

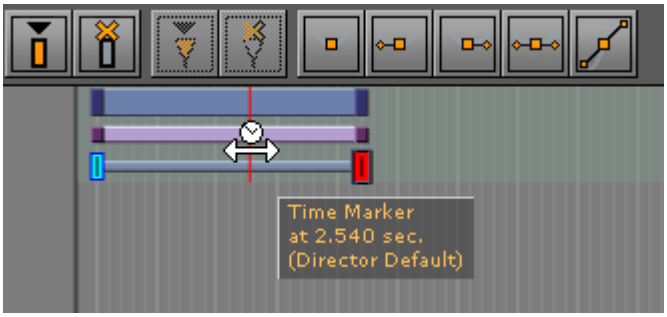
-  **Lock Selection**: Activates the Director(s) of the selected container in the scene tree to be active in the stage when enabled. When disabled, locks the selected Director(s) so that no matter where else you click in the scene tree, only the selected Director will be controlled by the buttons in the Time-line Editor.
-  **Reverse:** Plays the animation in reverse when enabled.
-  **Go to Start:** Jumps the time-line to the start of the current Director(s).
-  **Start Animation:** Plays the current Director(s) in the current layer from the beginning.
-  **Stop Animation:** Stops the currently playing animation at the current point in the time-line. Right-clicking stops the currently selected Director or directors.

  > ⚠ **Note:** The animation is automatically stopped when it reaches the end of the time-line, or a stop point along the time-line.

-  **Continue Animation:** Continues the animation from where it was stopped.
-  **Go to End:** Jumps to the end of the animation.
-  **Key Frame Previous:** Jumps to the previous Key Frame of the selected container in the scene tree.
-  **Time-line:** Shows the current position on the time-line in fields of the input box .
-  **Jump to Next Time-line Position:** Jumps to the next position in the Time-line.
-  **Key Frame Next:** Jumps to the next Key Frame of the selected container in the scene tree.
-  **Set Key:** Creates a Key Frame at the defined time-line.
-  **Lock Time-line Position**: Jumps the time-line automatically after a Set Key operation by the defined Increment Value (see below) when activated. When deactivated, the time-line does not jump after a Set Key operation.
-  **Increment Value:** Defines the step value (in fields from the input) box to be incremented when creating a Key Frame.

10.4 Time-Line Marker

The time-line marker defines the current position along the time axis (X) in the Dopesheet and Spline Editor. The time-line marker is shown as a thin red vertical line.



A separate time-line marker is created for each Director in the Stage Tree. Dragging a time-line marker shows the animation in the Scene Editor accordingly.

If the Show Info option is enabled in the Stage Editor Bar, a tool tip with information about the current time and belonging Director will show when holding the pointer over the time-line marker.

If the Animate Time-line Marker option is enabled in the Stage Editor Bar, the time-line marker is animated in the Dopesheet while playing the animation.

This section contains information on the following procedures:

- To Manually Move the Time-line Marker
- To Move the Time-line Marker a Set Amount of Fields
- To Move the Time-line Marker a Set Amount of Fields with Shortcuts

10.4.1 To Manually Move the Time-line Marker

- Drag the time-line marker along the time-line in either the Dopesheet Editor or Spline Editor
  or
- Click an empty area of the Dopesheet Editor or Spline Editor.

> ⚠ **Note:** In complex animations with many directors, channels, and actors, there might not be an empty area to click. If this is the case, press <Alt> while clicking.

10.4.2 To Move the Time-line Marker a Set Amount of Fields

- In the Dopesheet Editor, enter the time (in fields) in the **Time-line** input box
  or

1. Enable the Stage Object Editor.
2. Select the Director in the Stage Tree or Dopesheet Editor.
3. In the Director Editor that shows, enter the time (in fields) in the **Time-line** text box.
   or

- Drag the Time-line in the Scene Editor Time-line editor 

.

10.4.3 To Move the Time-line Marker a Set Amount of Fields with Shortcuts

- **Move One Field at a Time:**
  - In the Dopesheet Editor or Director Editor click the **Time-line** up/down buttons, or
  - Press <Ctrl> and the left or right arrow key on the keyboard
- **Move More Than One Field at a Time:**

> ⚠ **Note:** The movement interval in fields, is format dependent.

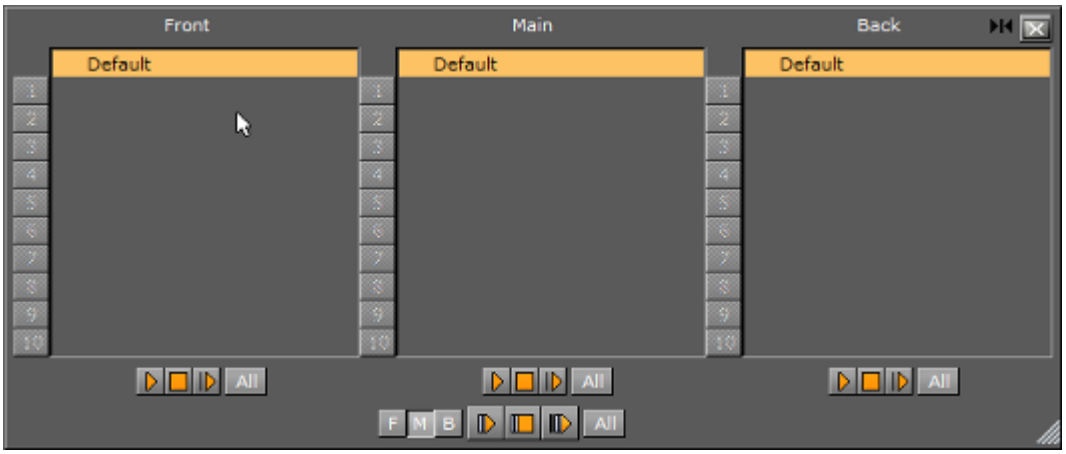
  - **Press the left or right arrow key on the keyboard**. This, for example, in 1080i50 format, moves the time-line marker 5 fields, left or right, and in 1080i60 the movement is 6 fields, left or right
  - **Press <Alt> and the left or right arrow key on the keyboard**. This, for example, in 1080i50 format, moves the time-line marker 50 fields, left or right, and in 1080i60 the movement is 60 fields, left or right

---

10.5 Artist Director Control Panel

Click the **Director** option in the Time-line Editor to open the **Director Control** panel.

The **Director Control** panel lists all available Directors of the selected layer. To see all layers (Front, Main and Back) click the expand icon ◀▶ .



It is possible to select one or more directors in the list to play them individually.

> ⚠ **Note:** There are two sets of control buttons; one set for controlling individual directors in the selected layer and one set for controlling the all directors in all layers.

- **Front:** Lists the directors belonging to the front layer scene.
- **Main:** Lists the directors belonging to the main scene.

- **Back:** Lists the directors belonging to the back layer scene. This button is only enabled if a back layer has been defined in the Layer Manager.
- **All:** Selects all directors in the list when activated.
- **Start Animation/Play Layer:** Plays the current Director(s) of all layers or the selected layer from the beginning (see Basic Animation Functions).
- **Stop Animation/Stop Layer:** Stops the currently playing animation at the current point on the time-line for all layers or the selected layer. Right-clicking stops only the selected directors (see Basic Animation Functions).

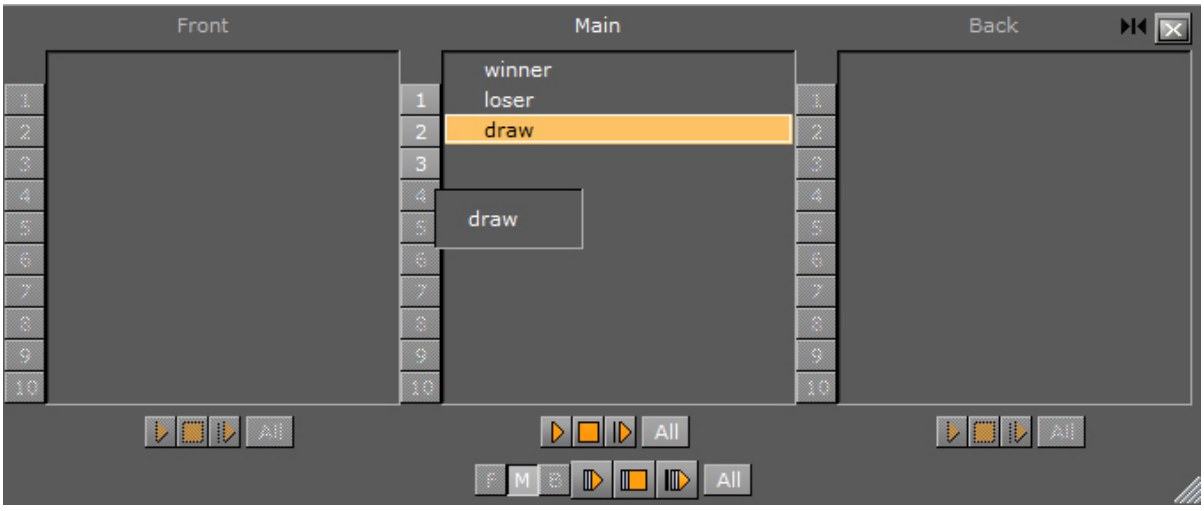
> ⚠ **Note:** The animation is automatically stopped when it reaches the end of the time-line, or a stop point along the time-line.

- **Continue Animation/Continue Layer**: Continues the animation of all layers or the selected layer to the next point on the time-line.
- **[0-10]**: You can save up to ten combinations of selected directors per layer by selecting them and dragging them onto one of the numbers. Afterward, when you click a number, the combination of saved directors is then selected.

> ⚠ **Note:** These selections are on a per-layer basis and also cannot be saved.

- **F, M, B**: Enables the layers that should be controlled using the Play Stop and Continue Layer buttons.

10.5.1 Working with the Director Control Panel



When multiple Directors in the selected layer are available, you can select and drag them onto the preset/numbers bar in the **Director Control** panel to call up different presets at any time.

To do so you need to:

1. Select at least one Director (or multiple directors with **Shift+click**)
2. Drag the selection on to one of the numbers in the **Director Control** panel.

For example, in the image above, *winner* can be dragged onto slot **1**, *loser* to slot **2** and *draw* to slot **3**.

To then control other combinations you need only drag the selected Container on to a number and a new preset is defined. In this example, you can select *winner* and *draw* and drag them on to a *new number*. Then, whenever this number is clicked, the predefined selection is highlighted/ selected and can be triggered via the **Director Control** buttons.

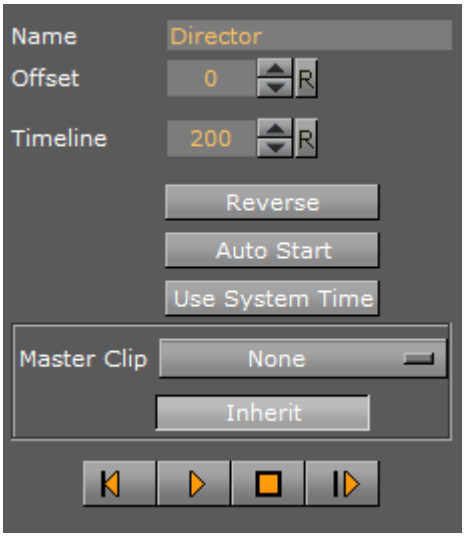
10.6 Director Editor

The Director editor shows details about the selected Director.

This section contains the following:

- Director Editor Properties

10.6.1 Director Editor Properties



- **Name:** Defines the name of the Director.
- **Offset**: Sets the offset in fields for the whole Director from the start of the time-line. This is the easiest way to shift all animations within the Director.
- **Time-line:** Defines the time-line of the Director in fields (each Director has a separate time-line). Changing the value in this box will move the Time-line Marker in the Time-line Editor, and the other way around.
- **Reverse**: Plays animations in this Director backwards when enabled.
- **Auto Start:** Starts the animation automatically when it is loaded from the database when enabled.
- **Use System Time:** Plays seamless looping directors, if you have multiple scenes with the same background animation. As an example, if you have multiple scenes for elections but each of the scenes has the same background animation/Director and you set the Use System Time to **Active** for this Director, it will play the background without dropouts or breaking the

loop while loading the other scenes, as long as they have the same Director where the system time is activated.

- **Master Clip**: Renders an animation, defined in the Stage, based on the actual frame of a played clip. This clip can either be set in Video Clip Playback Considerations or within a separate Director in the Stage:
  - **Drop-down box**: Select, None, Clip 1 to 16. Select the Clip Channel that contains the clip, which will control the animation
  - **Inherit**: When set to **on**, the set Master Clip Channel will be passed to all sub directors.
  - See Master Clip
-  **Go to Start:** Jumps to the start of the time-line.
-  **Start Animation:** Starts the animation from the beginning.
-  **Stop Animation:** Stops the animation at the current time-line.

> ⚠ **Note:** The animation is automatically stopped when it reaches the end of the time-line, or a stop point along the time-line.

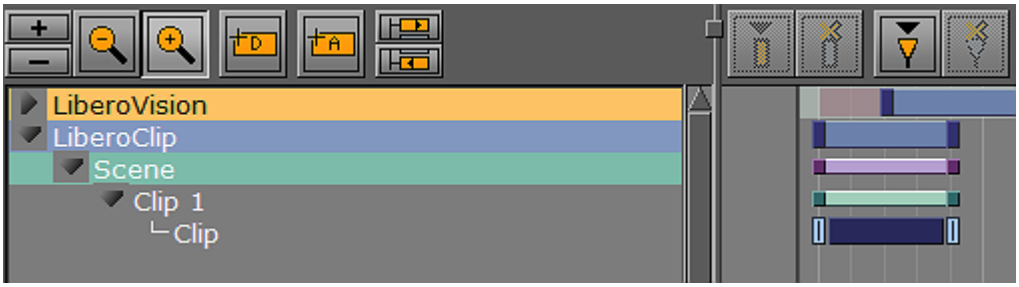
-  **Continue Animation:** Continues the animation after it has been stopped.

10.7 Master Clip

The Master Clip feature is particularly useful when you need to tie graphic animations to a specific location in a video clip. Using the director properties to tie a master clip to a director in the Director Editor will make sure that the stage and the clip are in sync. This is to make sure, that when the clip plays the frame at time 3:12 for example, you will also get the animation of the graphic rendered at 3:12.

It is essential that you have at least two top-level directors in a scene using the Master Clip feature. One contains the clip only, and the second all other scene items. The second director can also contain several sub-directors. For the animations in the second director to be bound to the clip it is necessary to set the Master Clip to the clip channel containing the clip for this director. If there are sub-directors, which should also respect the clip as the animation controller, the button *Inherit* in the Director Editor must be checked.

A simple scene using the Master Clip feature could be constructed like this:



⚠ **Note:** Scrubbing in the controlled director is not possible when using the Master Clip feature as the timing is set by the clip. Scrubbing is only supported in the director containing the clip.

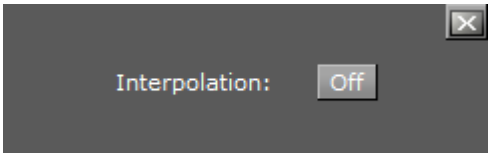
10.8 Actor Editor

The Actor Editor shows details about the selected Actor. The available actions may vary, depending on the actor selected.

- **Offset:** Sets the offset in fields for the whole actor from the start of the time-line. This is the easiest way to shift all animations within the actor.

10.8.1 Actor Editor for Integer Channels

Integer channels are channels for objects that have numeric values written without a fractional component, as opposed to floating point values.



- **Interpolation**: General setting for all integer animation channels. If interpolation is switched off, no interpolation of the integer values happens when animating from one key-frame to another. The value changes only when a key-frame is passed when running the animation. The default setting is `Off`.

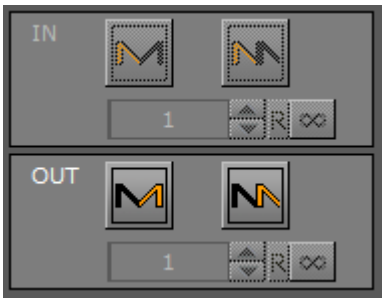
10.9 Channel Editor

The Channel editor shows details about the selected Channel.

This section contains information on the following topics:

- Common Channel Editor Properties
- Position Channel Editor Properties
- Action Channel Editor Properties

10.9.1 Common Channel Editor Properties



> ⚠ **Note:** Not available for four Point Loop clips.

IN:

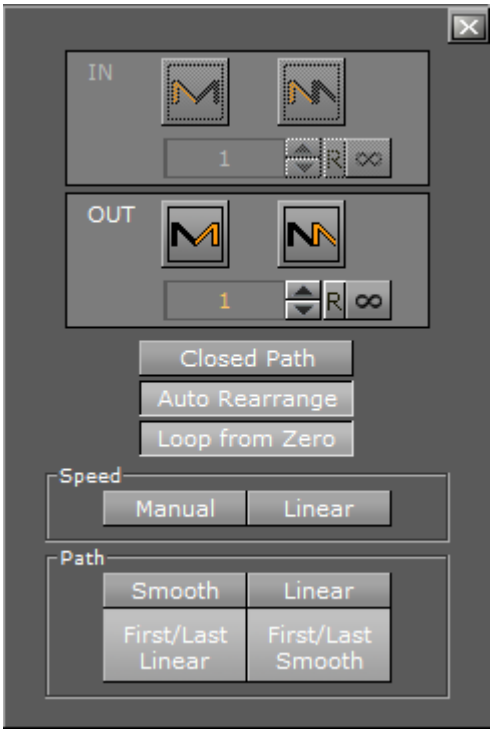
- **Swing In:** Enables swing in.
- **Loop In:** Enables loop in.
- 10 R **Number of Swings/Loops:** Defines the number of swings/loops in.
- ∞ **Infinite:** Sets number of swings/loops in to infinite (from the start of the animation).

OUT:

- **Swing Out:** Enables swing out.
- **Loop Out:** Enables loop out.
- 10 R **Number of Swings/Loops:** Defines the number of swings/loops out.
- ∞ **Infinite:** Sets the number of swings/loops out to infinite (animation never ends).

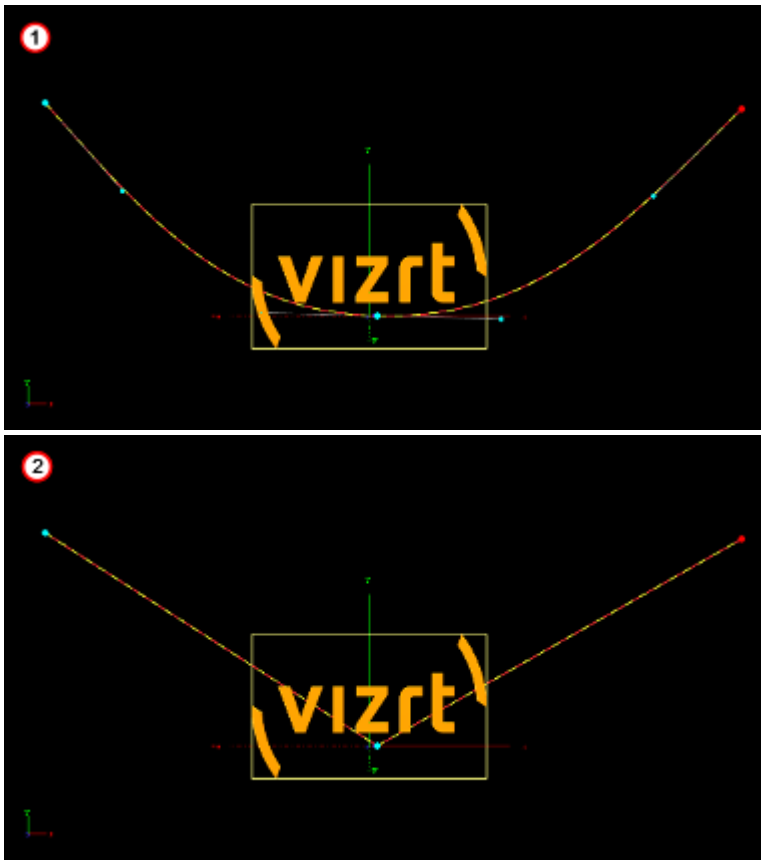
10.9.2 Position Channel Editor Properties

The **Position Channel** editor has, in addition, these extra parameters that can be edited:



- **Closed Path**: Forces the animation path to go from the last Key Frame to the first when activated. When deactivated, the animation path will remain open.
- **Auto Rearrange**: Ensures that the order of the Key Frames in the Dopesheet editor is the same as in the Scene Editor so that the animation only ever continues forward. When disabled, if you change the order of the Key Frames in the Dopesheet editor, the spline will remain the same, but the animation may return from the previously last Key Frame to the newly added Key Frame now in last position. By default, this setting is enabled.
- **Loop from Zero**: Includes a gap in the loop (like 2.x behavior), if there is a gap before an animation loop, when enabled.
- **Speed:**
  - **Manual**: Locks all Key Frames. There is no linear speed with this option. The duration between each Key Frame can be different, depending on the timing value and the transformation values between the different Key Frames. The result is that you will not have a continuous animation speed.
  - **Linear**: Unlocks all Key Frames. The Key Frames between the first and last Key Frames are locked. If you change the value of either the first or last Key Frame, or any of the Key Frames in between, the timing will change automatically for each Key Frame. The result is an animation with a continuous speed between all Key Frames.
- **Path:**
  - **Smooth** (1): Sets the animation path to be smooth.
  - **Linear** (2): Sets the animation path to be linear.
  - **First/Last Linear**: Sets the animation to be linear for the first and last Key Frame, and smooth for the rest.
  - **First/Last Smooth**: Sets the animation to be smooth for the first and last Key Frame, and linear for the rest.

> ⚠ **Note:** These settings are global for the whole animation in question.



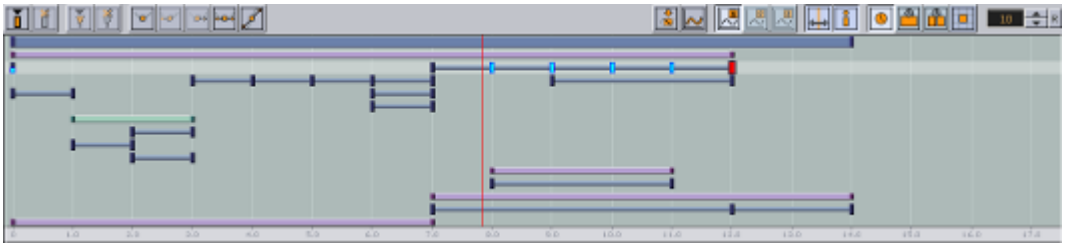
10.9.3 Action Channel Editor Properties

The Action Channel editor has, in addition, these extra parameters that can be edited:

- **Always Execute Action at 0:** If an action is set to execute at 0 on the time-line, press this button to make sure that the action is executed at 0. If this button is not pressed (inactive) the action may not work.

10.10 Dopesheet Editor

The Dopesheet editor shows the items in the Stage Tree as bars along the time-line.



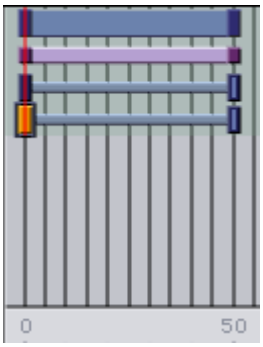
The items are presented vertically in the Dopesheet editor according to the structure in the Stage Tree. The horizontal scaling represents the time value.

> ⚠ **Tip:** The time value can be set to either seconds, frames, or fields from the Stage Editor bar.

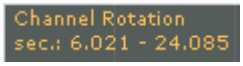
The Time-line Marker defines the current time, and the Scene Editor shows the animation accordingly.

The Grid and Snap to Grid options that can be enabled in the Stage Editor make it possible to position items in the Dopesheet editor more accurately.

The image below shows the Dopesheet editor with **Snap to Grid** option enabled with a field width of five:



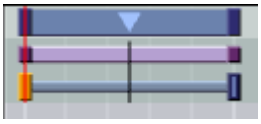
If the Show Info option is enabled in the Stage Editor, a tool tip with time information will show when the pointer is held over an item in the Dopesheet editor:



The Dopesheet editor can contain bars for various stage items: Directors, Actors, Channels, etc. Similar to the Stage Tree, Directors are shown with a gray blue color, Container Actors with light purple, while channels and actions are shown with a gray color.

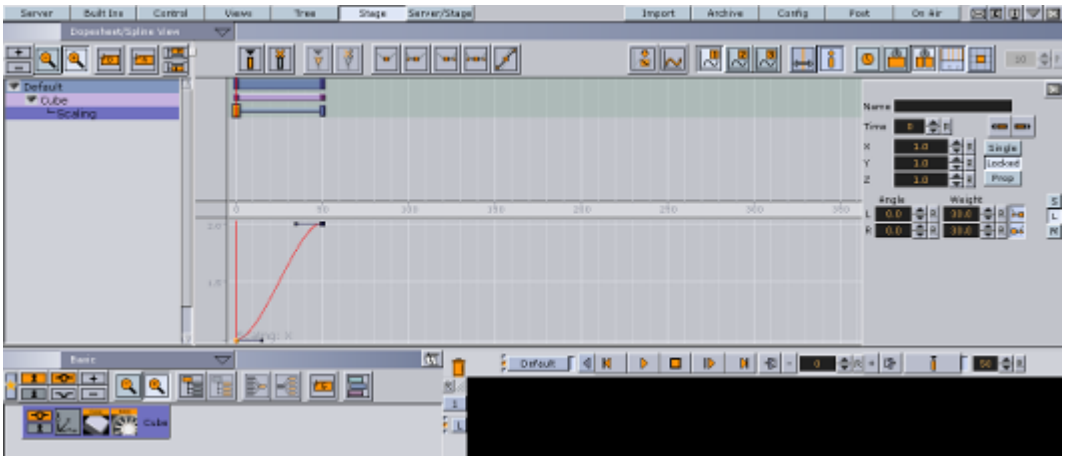
To build up an animation, Key Frames and stops/tags can be added to the Scene. A Key Frame defines a property at a given time. Multiple Key Frames belonging to the same channel are connected in the Dopesheet editor with one or more bars.

The image below shows an animation where a stop/tag has been added between two Key Frames:



A stop/tag can be added to a Director to end or pause an animation at a given time.

⚠ **Example:** If the scaling of a cube is 1.0 at field 0 and 2.0 at field 50, one Key Frame will be created at field 0 and another at field 50. A bar will connect the two Key Frames in the Dopesheet editor. See the image below:



If the Stage Object Editor is enabled, clicking an item in the Dopesheet editor will open the corresponding editor.

⚠ **Example:** Clicking a Director will open the Director Editor, clicking a Key Frame will open the Key Frame Editors, and so on.

10.10.1 To Enable/Disable the Dopesheet Editor

- From the Stage Editor Menu, select **Dopesheet View**, or
- Hold the pointer over the Stage and press <Ctrl+2>.

10.10.2 To Enable the Dopesheet Editor Together with the Spline Editor

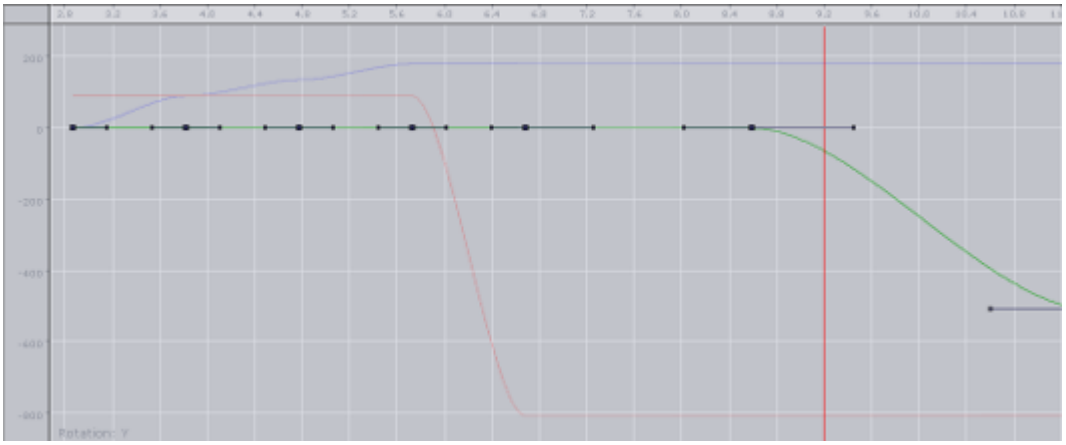
- From the Stage Editor Menu, select **Dope/Spline View**, or
- Hold the pointer over the Stage and press <Ctrl+4>.

10.10.3 To Resize the Height or Width of the Dopesheet Editor

- **Height**: Drag the horizontal separator between the Dopesheet editor and Spline Editor.
- **Width**: Drag the vertical separator between the Stage Tree Area and Stage Editor.

10.11 Spline Editor

The Spline editor shows the Key Frames on a Channel as splines along the time-line.



The Key Frames are presented vertically in the Spline editor according to the properties they possess. The horizontal scaling represents the time value.

> ⚠ **Tip:** The time value can be set to either seconds, frames, or fields from the Stage Editor Bar.

The Time-line Marker defines the current time, and the Scene Editor shows the animation accordingly.

It is possible to drag Key Frames in the Spline Editor both horizontally (changing the time) and vertically (changing the value) at once.

This section contains information on the following topics:

- Spline Editor Procedures
  - To Enable/Disable the Spline Editor
  - To Enable the Spline Editor Together with the Dopesheet Editor
  - To Resize the Height or Width of the Spline Editor
  - To Switch Between the Various Splines
  - Example of axes X, Y, and Z in the Spline editor
  - To Move Key Frames in the Spline Editor
- Editing Handles
- Position in the Spline Editor
- Rotation in the Spline Editor
- Scale in the Spline Editor

- Material in the Spline Editor
- Additional Best Practices for the Spline Editor
    - Example of Key Frame and handle tool tips

10.11.1 Spline Editor Procedures

To Enable/Disable the Spline Editor

- From the Stage Editor Menu select **Spline View**, or
- Hold the pointer over the Stage and press <Ctrl+3>.

To Enable the Spline Editor Together with the Dopesheet Editor

- From the Stage Editor Menu select **Dope/Spline View**, or
- Hold the pointer over the Stage and press <Ctrl+4>.

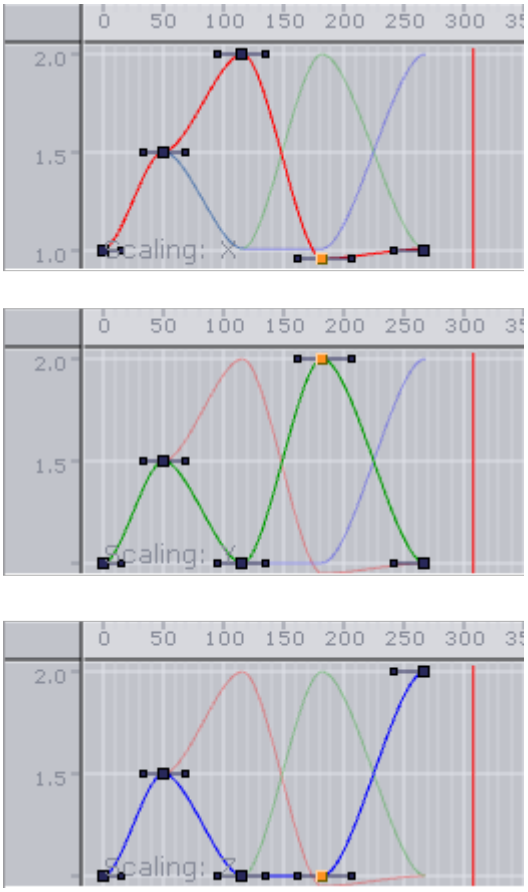
To Resize the Height or Width of the Spline Editor

- **Height**: Drag the horizontal separator between the Dopesheet editor and Time-line Marker.
- **Width**: Drag the vertical separator between the Stage Tree Area and Stage Editor.

To Switch Between the Various Splines

- Use the Show Spline 1,2, and 3 buttons from the Stage Editor Bar, or
- Use the keyboard shortcuts <Ctrl+Z>, <Ctrl+X> and <Ctrl+C>

Example of axes X, Y, and Z in the Spline editor



To Move Key Frames in the Spline Editor

- Select the required Key Frame and drag it around.
- To change only the position on the time-line or the value, press <Shift> to lock the axes. Keep <Ctrl> pressed while you drag the Key Frame if you want to snap to grid along the time-line.
  You can move a Key Frame in the spline editor, so either the position on the time-line (left and right) or the value stored (up and down) in the Key Frame will be changed.

10.11.2 Editing Handles

As each Key Frame on the spline has two handles, moving them is the most easy way to edit the curve. To edit a handle left-click it and drag it around. You will see that the spline will reflect your movement.

By default you move the handles proportional this way. This means if you change the handle on one side, the one on the other side will move too. If you want to only edit the handle on one side, you need to left-click the required handle. Keep the left button pressed and press <Ctrl>. If you

drag the handle, only the one you drag right now will move and the other one will stay. This “single handle mode” will be enabled for the Key Frame as long as you press <Ctrl>.

Furthermore you can use the buttons in the main menu to edit the handles at a Key Frame. To do so select the Key Frame you want to change the handles in and select the action in the main menu. For ease of use multiple Key Frames can be selected.

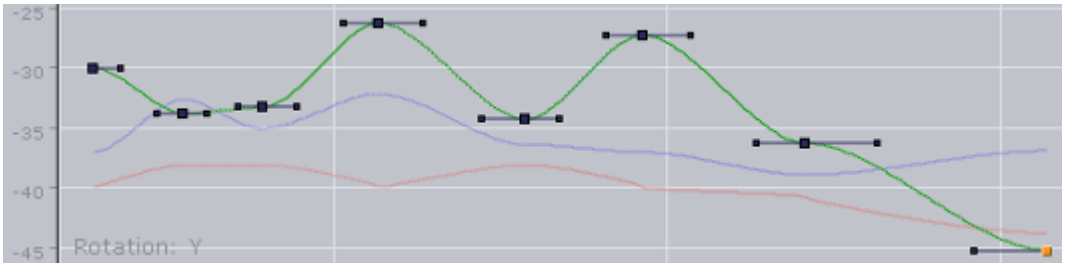
-  Set left and right spline handles in selected Key Frame to linear.
-  Set spline handles in selected Key Frame to smooth left and linear right.
-  Set spline handles in selected Key Frame to linear left and smooth right.
-  Set left and right spline handles in selected Key Frame to smooth.
-  Set left and right spline handles in selected Key Frame to tangential.

10.11.3 Position in the Spline Editor

If you animate a position, there are much more considerations to make.

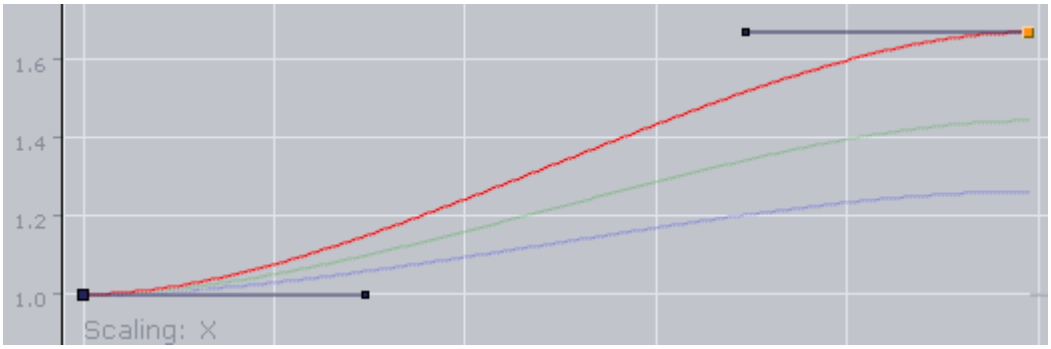
10.11.4 Rotation in the Spline Editor

If you have animated the rotation, you can rotate about up to all three axes at the same time. In the spline editor three different splines will be shown, one for every axes. The red one is for the X-axis, the green one for the Y-axis and the blue one for the Z-axis. For ease of use in the lower left corner you will find indicated which axis is currently selected.



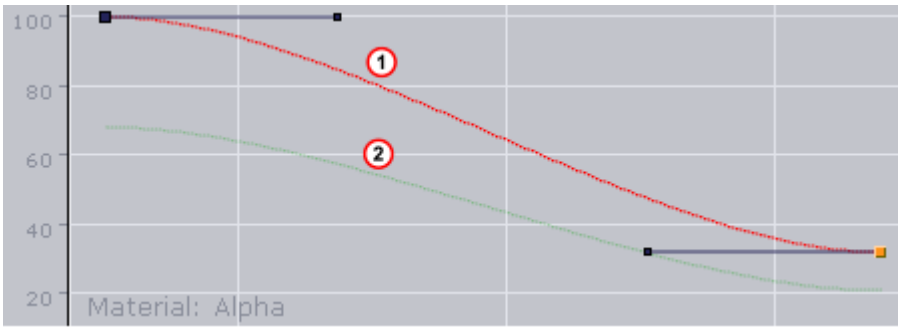
10.11.5 Scale in the Spline Editor

If you have animated the scale, you can scale along up to all three axes. In the spline editor three different splines will be shown (when **Prop** or **Single** is selected), one for every axis. The red one is for the X-axis, the green one for the Y-axis and the blue one for the Z-axis. For ease of use in the lower left corner you will find indicated which axis is currently selected.



10.11.6 Material in the Spline Editor

If the material has been animated, in the spline editor two different splines will be shown. The red one is for the alpha value, the green one for the shininess. For ease of use in the lower left corner you will find shown which parameter is currently selected.



10.11.7 Additional Best Practices for the Spline Editor

Normally for each channel only one spline is shown in the spline editor. But there are animation channels where more splines are available for editing. As you can only edit the Key Frames/ handles of one spline, you must use the number keys in the Stage Editor Bar to select the required spline.

Pressing `<Shift>` while moving a Key Frame in the Spline Editor enables you to drag it only horizontally or vertically, but not both directions at the same time. Pressing `<Ctrl>` while moving a Key Frame will enable the Snap to Grid function, allowing the Key Frame to be placed only at certain time-line intervals. For more information (see Stage Editor Bar).

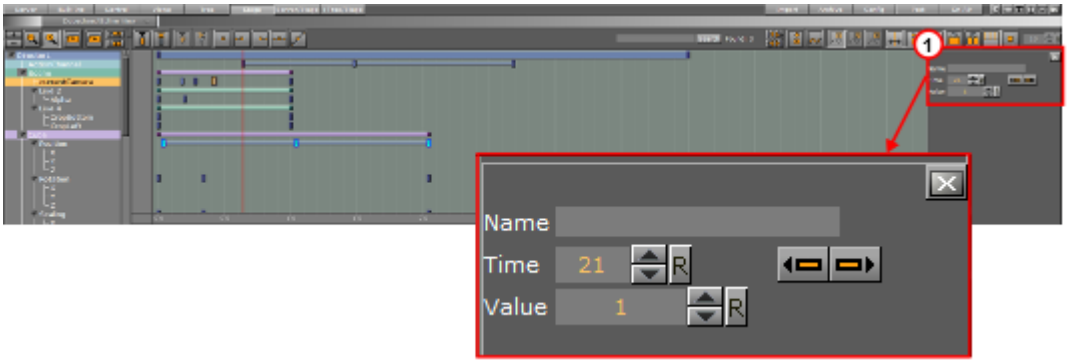
When animating certain channels, such as scaling or rotation, up to three splines will be shown in the Spline editor, each describing the state along one axis, either X, Y, or Z. For a better overview, different colors are used; red for X-axis, green for Y-axis, and blue for Z-axis.Every spline contains various Key Frames with handles.

Example of Key Frame and handle tool tips

If the Show Info option is enabled in the Stage Editor Bar, placing the pointer over a Key Frame or handle will show a tool tip with status information about time-line and value.

10.12 Stage Object Editor

The Stage Object editor (1) is positioned at the right side of the Stage Editor. The editor has different views depending on the particular editor to be shown.

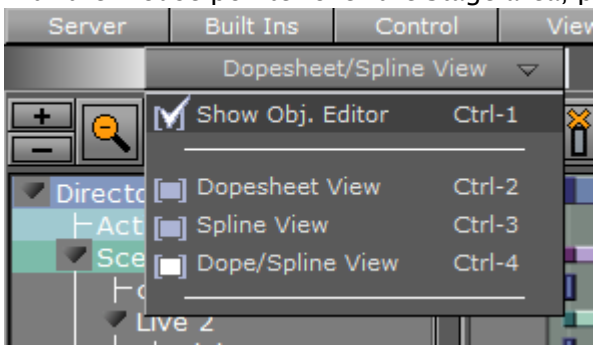


⚠ **Example:** The Director Editor shows information about directors, the Actor Editor shows information about actors, the Channel Editor shows information about channels, the Actor Editor shows information about actions, the Key Frame Editors shows information about Key Frames, the Event Editor shows information about stops/tags, and so on.

Clicking an object in the Dopesheet Editor or selecting an item from the Stage Tree will open the corresponding editor in the Stage Object Editor.

10.12.1 To Enable/Disable the Stage Object Editor

- From the Stage Editor Menu, select **Show Obj. Editor**, or
- With the mouse pointer over the Stage area, press <Ctrl+1>.



10.13 Key Frame Editors

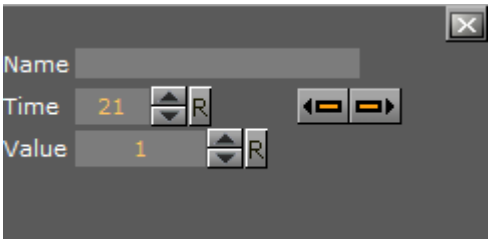
The Key Frame editors show details about the selected Key Frame. To view the Key Frame editor click on a Key Frame.

Make sure that the Stage Object Editor is enabled.

This section contains information on the following topics:

- Common Key Frame Editor Properties
- Locked and Unlocked Key Frames
- Action Key Frame Editor Properties
- Position Key Frame Editor Properties
- Rotation Key Frame Editor Properties
- Scale Key Frame Editor Properties
- Material Key Frame Editor Properties
- Clip Key Frame Editor Properties
- Adding Camera Events Using Key-frames

10.13.1 Common Key Frame Editor Properties



- **Name:** Defines the name of the Key Frame. This name is used for external commands or scripts.
- **Time:** Defines where along the time-line (in fields) the Key Frame is placed.
-  **Previous/Next buttons:** Selects the previous or next Key Frame along the time-line if the property has more than one Key Frame.
- **Value:** The rest of the Key Frame editor options vary according to which property is being animated.

> ⚠ **Example:** For scaling, it is possible to define the size of an object in direction X, Y, and Z.

10.13.2 Locked and Unlocked Key Frames

By default the Key Frames in the position animation are locked. They store information about the position of the container along the X-, Y- and Z-axis, as well as the time the Key Frame will be passed by in the animation.

An unlocked Key Frame stores only information of the position along X-, Y- and Z-axis, without the information at which time the Key Frame should be passed. To reflect this, in the object editor for a locked Key Frame the **Time** is grayed out. Furthermore, as the spline editor shows time and path-percentage, the unlocked Key Frames will not be shown there. In the animation editor unlocked Key Frames will be shown at half height of locked Key Frames. Viz Artist calculates the time the Key Frame will be passed to fit with the handles set in the next locked Key Frame before and after the unlocked Key Frame.

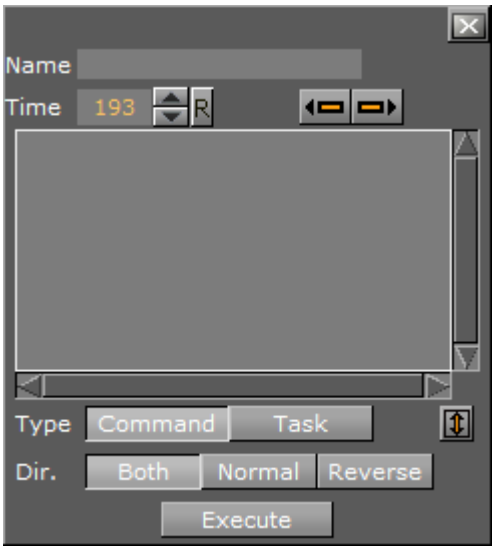
To better understand the difference between locked and unlocked Key Frames, create a simple position animation. You will see the container moving with different speeds between the Key

Frames. Next unlock all Key Frames and play the animation once more. The container will move evenly during the whole animation.

Use an unlocked Key Frames to define the path without editing the time for every Key Frame. This can save time.

10.13.3 Action Key Frame Editor Properties

In addition to the common key frame editor properties, the Action Key Frame Editor defines the Action Channel. The action will work when the Director time-line equals the Action Channel Key Frame.



- **Value:** Executes the command or OS task entered here.
- **Type:** Initalizes action Key Frames for two main types of actions:
  - **Command:** Use Viz Artist internal commands to execute a number of automated tasks in the program (for Viz Artist internal commands documentation go to: *<viz install folder>/Documentation/CommandInterface/index.html*)
  - **Task:** Enter a task command. Any relevant command which can be started from a command line window (i.e., cmd.exe) can be executed as a task. If the path to the executable file contains spaces, set the complete path in quotation marks, for example, `"C:\program files\microsoft office\office\winword.exe"`

    ⚠ **Tip:** Multiple commands can be executed in the Action Key Frame Editor. The commands must be terminated by a ; (semi-colon) and on a new line. For example: `RENDERER*TREE*$sometext*GEOM*TEXT SET Hello World; MAIN_SCENE*STAGE*DIRECTOR*$Default GOTO_TRIO $B $A;`

- **Dir:** Sets which way, forwards or backwards, the animation should go to trigger the Action Key Frame:
  - **Both:** The Key Frame will be triggered when the animation goes forward and in reverse.
  - **Normal:** The Key Frame will be triggered when the animation goes forwards.
  - **Reverse:** The Key Frame will be triggered when the animation goes backwards.

- **Execute:** Starts the action.

Examples of Commands for Action Key Frames

Here are some examples of Viz Artist internal commands, which can be used in the Action Key Frame Editor to create actions:

- **THIS_SCENE**: The created action acts on this Scene only.
- **THIS_EDITOR**: The created action acts on this Editor only.
- **THIS_DIRECTOR**: The created action acts on this Director only.
- **THIS_PARENT_DIRECTOR**: The created action acts on this Parent Director only.
  An example to invoke a script function:

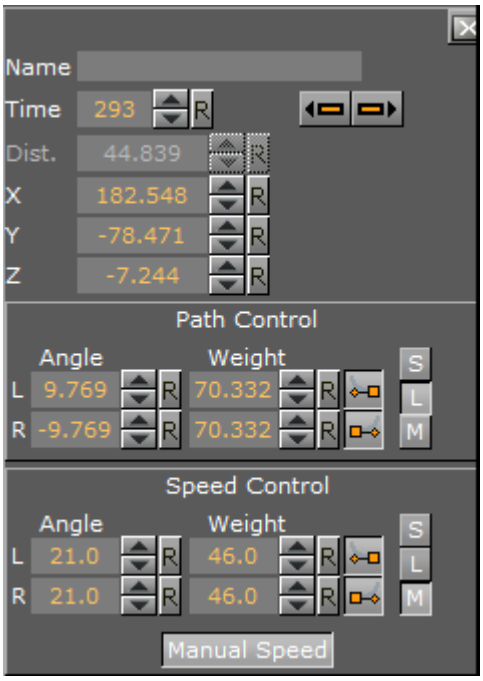
```
THIS_SCENE*TREE*$[name of the container]*SCRIPT INVOKE [your function name/sub routine name in the script]
```

An example to set a plug-in value:

```
THIS_SCENE*TREE*$[name of the container where the plug-in resides]FUNCTION[name of plug-in]*[name of UI item of the plug-in, i.e the button or the drop down list] SET [value]
```

10.13.4 Position Key Frame Editor Properties

In addition to the Common Key Frame Editor Properties, a Position Key Frame contains information about the position of a Container at a specific time. The parameters of Position Key Frames are:



- **Distance**: Defines the distance of a selected Key Frame (value in percent) between the first and last Key Frame of the animation (information only)

- **X:** Defines the position along the X axis.
- **Y:** Defines the position along the Y axis.
- **Z:** Defines the position along the Z axis.

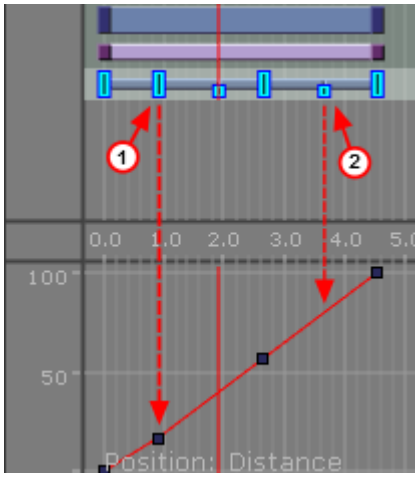
Path Control

- **Angle:** Defines the angle of the handles for the path.
- **Weight:** Defines the weight of the handles for the path.
-  **Smooth Left/Right:** Allows the handles for the path to be smooth or linear (see Channel Editor).
- **S / L / M**: Defines the way handles for the path should be modified:
  - **S (Single)**: Only the left or right parameter can be adjusted.
  - **L (Locked):** Left and right parameters are locked in their current position and adjustment changes both left and right parameters together.
  - **M (Mirror):** The left and right parameters are mirrored when adjusted

Speed Control

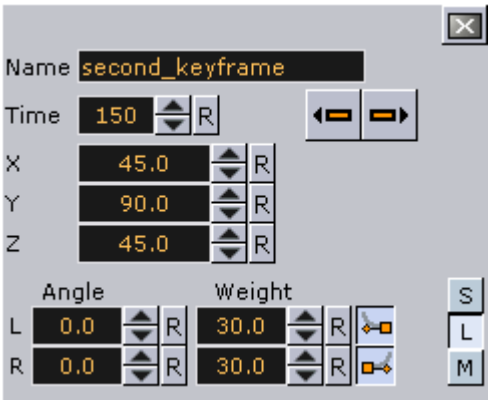
- **Angle:** Defines the angle of the handles for the speed.
- **Weight:** Defines the weight of the handles for the speed.
-  **Smooth Left/Right:** Allows the handles for the speed to be smooth or linear.
- **S / L / M**: Defines the way handles for the speed should be modified:
  - **S (Single)**: Only the left or right parameter can be adjusted.
  - **L (Locked):** Left and right parameters are locked in there current position and adjustment changes both left and right parameters together.
  - **M (Mirror):** The left and right parameters are mirrored when adjusted
- **Manual Speed**: Toggles the Key Frame Active or Inactive:
  - **Active:** Unlocks Speed Control (1). The speed between the previous, selected, and the next Key Frame is calculated by the position and the speed control of this Key Frame in the time-line.
  - **Inactive:** Locks Speed Control (2). The speed between the previous and the next Key Frame is calculated linear between those two and cannot be modified. If the value of either the previous or next Position Key Frame is changed, or any of the Position Key Frames, the timing will be adjusted automatically for each locked Key Frame. The

result is an animation with a constant speed between all locked Key Frames.

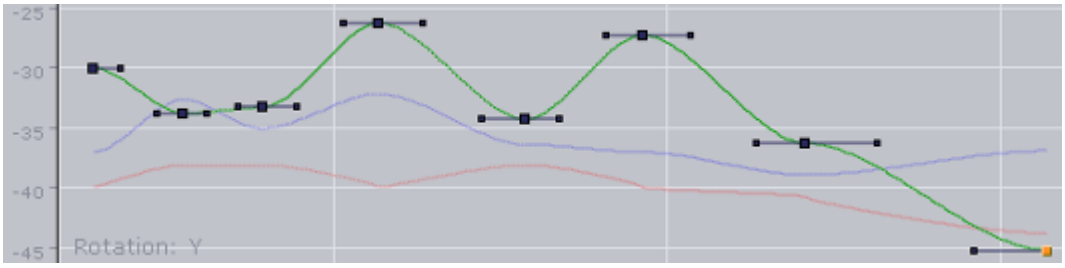


10.13.5 Rotation Key Frame Editor Properties

A rotation Key Frame contains information about the container rotation. In addition to the default options in the .Key Frame Editors v3.11, for a rotation Key Frame it is possible to define specific rotation options.

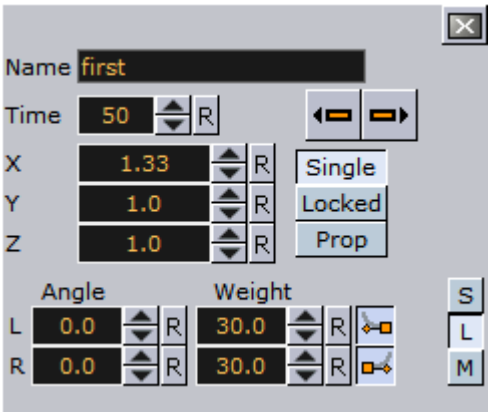


- **X:** Defines the rotation along the X axis.
- **Y:** Defines the rotation along the Y axis.
- **Z:** Defines the rotation along the Z axis.
- **Angle:** Defines the angle of the handles for the path or speed.
- **Weight:** Defines the weight of the handles for the path or speed.
- **Smooth Left/Right:** Allows the handles for the path to be smooth.
- **Single (S)**, **Locked (L)** and **Mirror (M):** Defines the way handles for the path or speed should be modified.
  Furthermore in the spline editor three different splines will be shown, one for every axes. The red one is for the X-axis, the green one for the Y-axis and the blue one for the Z-axis. As you can only edit the currently selected spline, you must use three spline buttons at the top of the Dopesheet Editor (or shortcuts <Ctrl+Z>, <Ctrl+X> and <Ctrl+C>) to select the spline you want to edit. For ease of use in the lower left corner you will find indicated which axis is currently selected.



10.13.6 Scale Key Frame Editor Properties

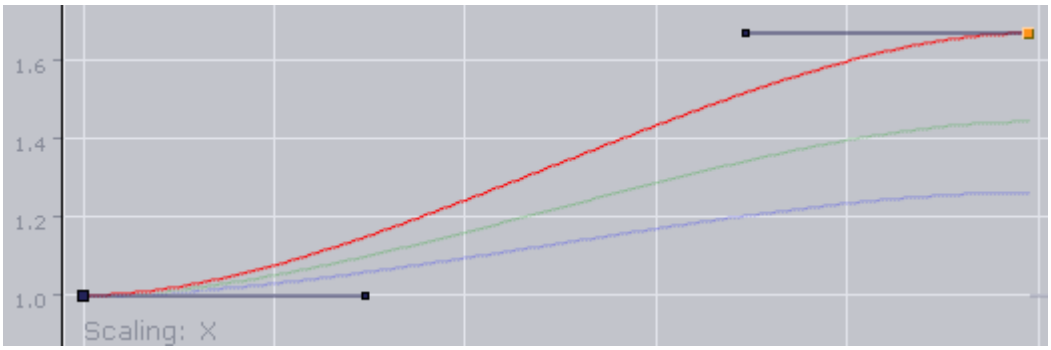
Scale Key Frames have to do with the scaling of the object in the animation. For a scale Key Frame, in addition to the default options in the .Key Frame Editors v3.11, it is possible to define the scaling along the X, Y, and Z axes in the three input boxes.



If you have animated the scale, you can scale along up to all three axes. To change all three scale values, in the object editor you will find an input box for **X**, one for **Y** and one for **Z**. There you can change if the values should be **Locked** or if you change them proportional (**Prop**) or each **Single** value.

Furthermore in the spline editor three different splines will be shown (when **Prop** or **Single** is selected), one for every axis. The red one is for the X-axis, the green one for the Y-axis and the blue one for the Z-axis. As you can only edit the currently selected spline, you must use three spline buttons at the top of the Dopesheet Editor (or shortcuts <Ctrl+Z>, <Ctrl+X> and <Ctrl+C>) to select the spline you want to edit. For ease of use in the lower left corner you will find indicated which axis is currently selected.

Scale axis, X, Y and Z

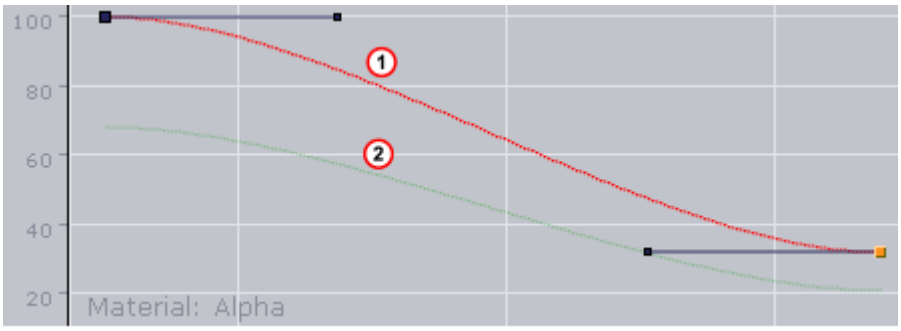


10.13.7 Material Key Frame Editor Properties

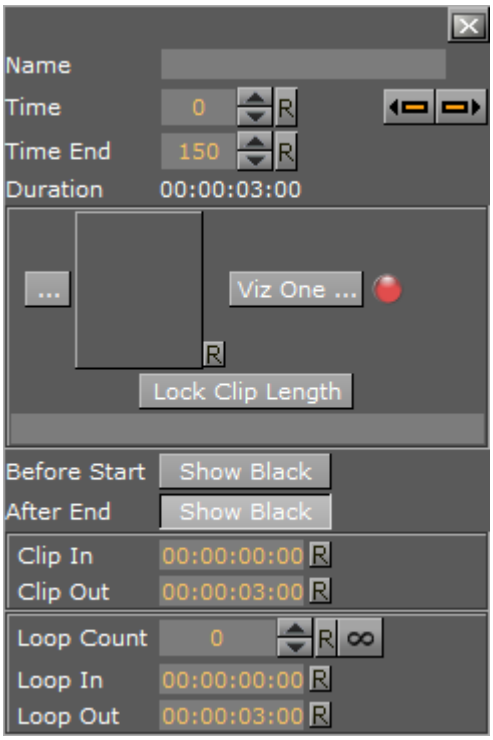
With a material Key Frame, in addition to the default options in the .Key Frame Editors v3.11, it is possible to define color.

If a material has been animated, the object editor for the Key Frames shows a Material Editor . This allows you to conveniently the change the material.

Furthermore in the spline editor two different splines will be shown. The red one is for the alpha value (1), the green one for the shininess (2). As you can only edit the Key Frames/handles of one spline, you must use the number keys in the Stage Editor Bar to select the required spline. For ease of use in the lower left corner you will find indicated which parameter is currently selected.



10.13.8 Clip Key Frame Editor Properties



- **Name**: Sets a name for the Clip Key Frame.
- **Time/Time End**: Sets the start and end time, in fields, of the Key Frame.
- **Duration**: Sets the time duration of selected clip Key Frame as defined by **Time/TimeEnd.**
- **Clip**: Selects a clip to use when the ... is clicked.
- **Viz One**: Selects a clip from Viz One (see Transfer Clips From Viz One).
- **Lock Clip Length**: Locks automatic length adjustment. Keyframe will maintain length for set clips. Useful when exporting the scene after making adjustments to the clip length in the stage, as this will keep the clip length as designed, and not revert to the length of the clip file.
- **Before Start:** Sets what should happen before the clip starts to play within the animation. If the button is **disabled**, it shows the in-frame. If the button is **enabled**, it shows black until the animation gets to the clip.
- **After End:** Sets what should happen after the animation leaves the clip Key Frame. If the button is **disabled**, it shows the out-frame. If the button is **enabled**, it shows black.
- **Clip In/Out**: Sets the time the clip comes in and goes out.
  Example:

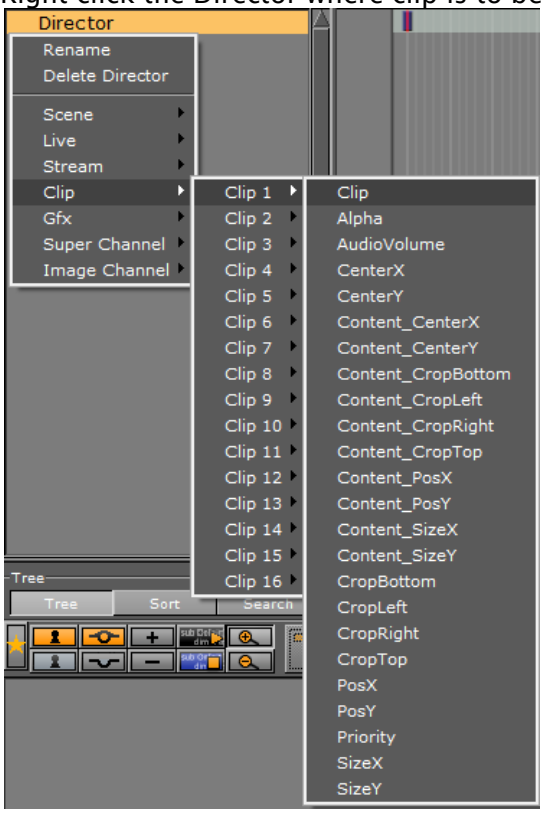
- Time = 50
- Time End = 250
- Clip Length = 4 sec (50i clip)
- Clip In = 00:00:01:00
- Clip Out = 00:00:03:00

  The clip playback starts 1 second after the Director starts. The clip starts at frame 25 and

ends at frame 75. Because the Key Frame duration is twice as long the clip part will be played twice.

To Add a Clip to the Stage

1. Right-click the Director where clip is to be added.

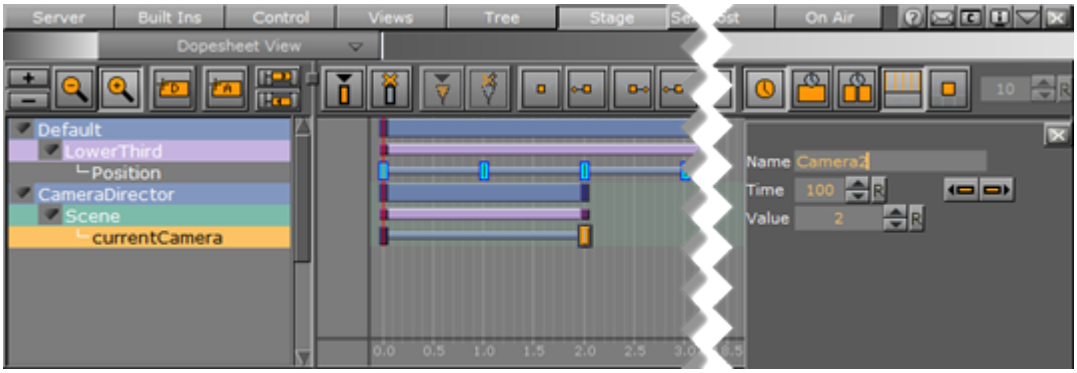


2. Select **Clip <number>** from the **Clip** menu, then choose **Clip**.
   - To open the clip Key Frame editor, click the Clip Key Frame in the Dopesheet.

10.13.9 Adding Camera Events Using Key-frames

The Stage Editor also allows the designer to change the active camera in a scene by using key-frames.

To Add a Basic Camera Shift Event



1. In the **Stage Tree**, right-click a Director and select **currentCamera** from the context menu. This will add a channel called **Scene** with **currentCamera** as an actor, with two key-frames.
2. Select the first key-frame. In the Actor Editor, enter a descriptive **Name** and a **Value** for the initial camera position. The default value is `1`, which translates into Camera 1.
3. Select the second key-frame. Enter a descriptive **Name** and a **Value** for the second camera position in the Actor Editor.

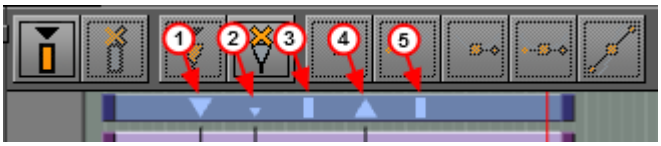
10.14 Event Editor

The Event Editor shows details about the selected Stop, Tag, Pause, Local Stop or Next Scene Events.

- **Name:** Sets a name for the Event. This name is used for external commands or scripts.
- **Time:** Sets where in the time-line (in frames) the Event is placed.
-  **Previous/Next buttons:** Moves to the previous or next Event along the time-line if a Director has more than one Event.
- **Type**: Defines the type of Event used:
  - **Stop (1):** Stops the animation until continue animation is clicked. A stop applies for the Director and all subdirectors.
  - **Tag (3):** Shows a position along the time-line. This Event type does not stop the animation.

- **Local Stop (2)**: Stops the animation until continue animation is clicked. A local stop only applies for the Director it is placed on, not its subdirectors.
- **Pause (4):** Stops the animation for a specified amount of frames (set in **Length**).
- **Next Scene (5):** Marks a position where a Scene ends. The next Scene will start automatically.

> ⚠ **Note:** Viz Weather has an automatic play list option where the user does not need to start the next Scene. The Next Scene Event starts the next Scene automatically when the current Scene reaches the Next Scene Tag.



- **Length**: Sets the length (frames) of a **Pause**
- **Dir: Set the direction where the Event is valid:**
  - **Both:** The Event is valid to both normal (forwards) and reverse directions.
  - **Normal:** The Event is only valid when the animation direction is normal (foward).
  - **Reverse:** The Event is only valid when the animation direction is reversed.

11 Create Animations

When an animation is created, the state of an object (for example its position, rotation, or scaling) is defined at various points in time. Key Frames are used to define these states, and the animation system interpolates between them to create a continuous animation.

This section contains information on the following topics:

- Directors
- Actors
- Channels
- Action Channels
- Key Frames
- Basic Animation Functions
- Create a Basic Animation
- Create an Advanced Animation
- Advanced Animation Functions
- Create an Over the Shoulder Scene
- Create a Stand-alone Scene
- Create Transition Effects

11.1 Directors

To create an animation, there must be at least one Director in the Stage Tree. Directors are used to group a certain set of Stage items, such as Actors and Channels. It is possible to create Directors at the root level of the Stage Tree, or as Sub-Directors.

A Director can hold as many items as required, but to improve usability it is recommended to group certain items in different Directors, especially when working with complex animations. In addition to improved overview in the Stage Tree, multiple Directors make it possible to show only the animations contained within each Director. This opens the possibility to, for example, have multiple animation channels on one object, driven by multiple Directors.

Each Director can also hold controlling items such as stop points and actions that only affect the animations within the Director (see Basic Animation Functions).

⚠ **Note:** The first Director created in the Stage is always called **Default** Director. Any subsequent Directors are called **Director** (rename as required).

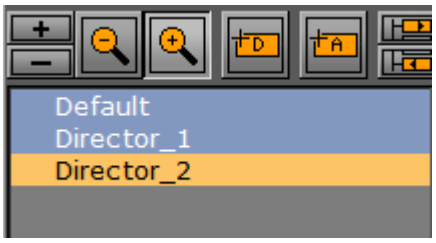
Most actions on Actors are done using Directors. Actions affecting the time-line are performed directly in the Dopesheet Editor. Working with Directors in the Dopesheet editor enables the Director Editor.

This section contains information on the following topics:

- Directors in the Stage Tree
- Directors in the Dopesheet Editor
- Events in a Director

11.1.1 Directors in the Stage Tree

Directors are shown in the Stage Tree with a gray blue background color.



To Create a Director

> ⚠ **Note:** The first Director created in the Stage is always called **Default** Director. Any subsequent Directors are called **Director** (rename as required).

- Click the  (Add Director) button

  > ⚠ **Note:** The new Director is added under the currently selected Director. If no Director is selected then its added at the end of the director tree.

  Or

- Drag the  (Add Director) icon to the Stage Tree:
  Directors that are dragged to the Stage Tree can be positioned in various ways, and create either root or Sub-Directors.

-  **Sub-Director**: Drop the  icon at the right side of a Director to create a Sub-Director

-  **Root Director Below/Above**: Drop the  icon at the left side of a Director to create a root Director at the same hierarchy level, either below or above the selected Director

To Move a Director

- Drag one or more Directors to a different place in the Stage Tree hierarchy

To Copy a Director

- Press <Ctrl> and drag one or more Directors to a different place in the Stage Tree hierarchy

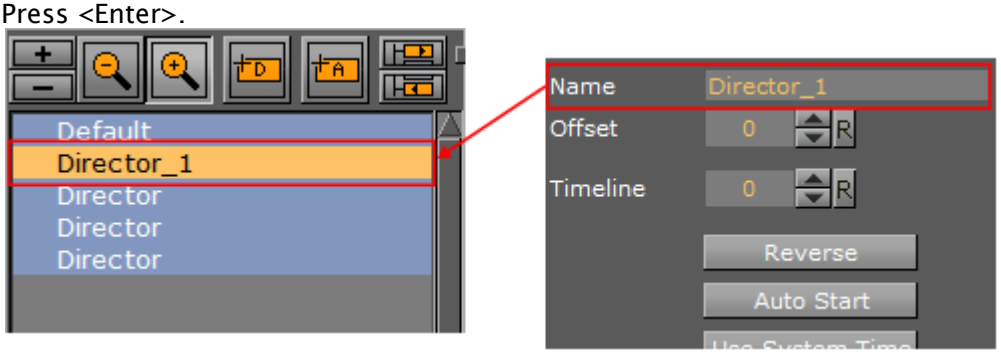
To Rename a Director

> ⚠ **Note:** Special characters, such as space, are not allowed in Director names.

1. Right-click a Director.
2. In the context menu, select **Rename**.
3. Type a new name.
4. Press <Enter>.

Or:

1. Select the required Director in the Stage Tree.
2. Open the Director Editor.
3. Type a descriptive name in the **Name** text box.
4. Press <Enter>.



To Delete a Director

1. Right-click a Director in the Stage Tree.
2. In the context menu, select **Delete Director**.

Or:

- Drag the Director, or multiple Directors, to the trash can

Or:

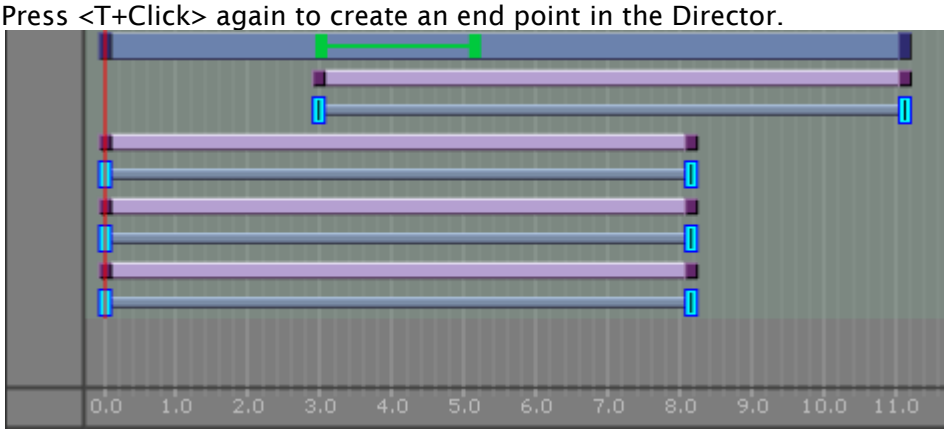
- Select a Director, or multiple Directors, and press <Delete>

11.1.2 Directors in the Dopesheet Editor

Directors are shown in the Dopesheet Editor with a gray-blue color.

To Mark a Region Within a Director

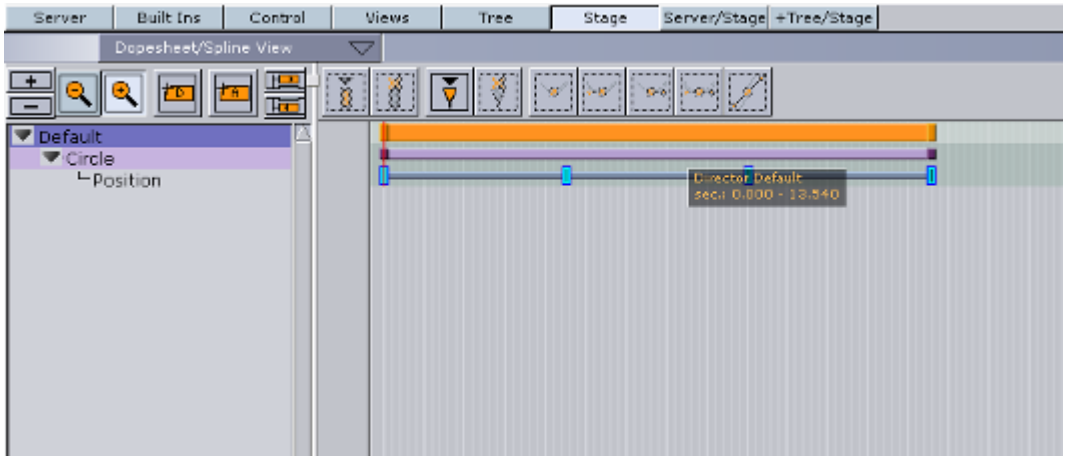
1. In the Dopesheet Editor, press <T+Click> to create a start point in a Director.
2. Press <T+Click> again to create an end point in the Director.



Drag the green start and end points to lengthen or shorten the selected region. In the Director only the Key Frames inside this region are affected.

To Move a Director in the Time-line

- Click on the Director and drag to a different time.



To Speed Up or Slow Down a Director

- Click on the start or the end of the Director.
- Drag the start or end point to change the Director speed:
  - **Speed up:** Lengthens the Director.
  - **Slow Down:** Shorten. the Director.

11.1.3 Events in a Director

Events can be added to a Director to add control to an animation. Events that can be added to the Director are:

- **Stop:** Stops the animation until continue animation is clicked. A Stop applies for the Director and all Sub-Directors
- **Tag:** Shows a position along the time-line. This Event type does not stop the animation
- **Local Stop**: Stops the animation until continue animation is clicked. A local stop only applies for the Director it is placed on, not its Sub-Directors.
- **Pause:** Stops the animation for a specified amount of frames.
- **Next Scene:** Marks a position where a Scene ends. The next Scene will start automatically.

> ⚠ **Note:** Viz Weather has an automatic play list option where the user does not need to start the next Scene. The Next Scene tag starts the next Scene automatically when the current Scene reaches the Next Scene Tag.

To Create a New Event

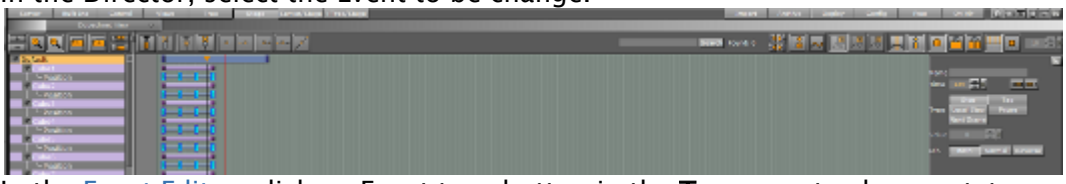
1. Click on a Director in the Stage Tree.
2. Set the Time-line Marker to the point where the new Event is to be added.
3. Add an Event:
   - Click the  (Add Stop/Tag) button, or
   - Press <S> and click in a Director where the new Event is to be added.

   > ⚠ **Note:** The default new Event is a **Stop** Event.

4. If required, change the Event type.

To Change an Event Type

1. Enable the Event Editor in the Stage.
2. In the Director, select the Event to be change.



3. In the Event Editor, click an Event type button in the **Type** area to change status.
4. If **Pause** is selected: In the Value box, define the number of frames that the animation should pause for, before it continues.

To Move an Event along the Time-line

1. Select one or more Events.
2. Drag the selected Events along the Director Time-line.

> ⚠ **Note:** To select more than one Event press and hold <Ctrl> and click on each required Event.

or

1. Enable the Event Editor in the Stage.
2. Select an Event in the Director.
3. Move the selected Event:
   - In the Event Editor, enter the new position (in frames) in the **Time** input box, or
   - Click in the **Time** input box and drag left or right.

> ⚠ **Note:** This moves only the last selected Event in the Director.

To Name an Event

1. Enable the Event Editor in the Stage.
2. Select an Event in the Director Time-line.
3. In the Event Editor, enter a descriptive name in the **Name** text box.
4. Press <Enter>.

> ⚠ **Note:** Special characters, such as space, are not allowed in Event names.

To Delete an Event

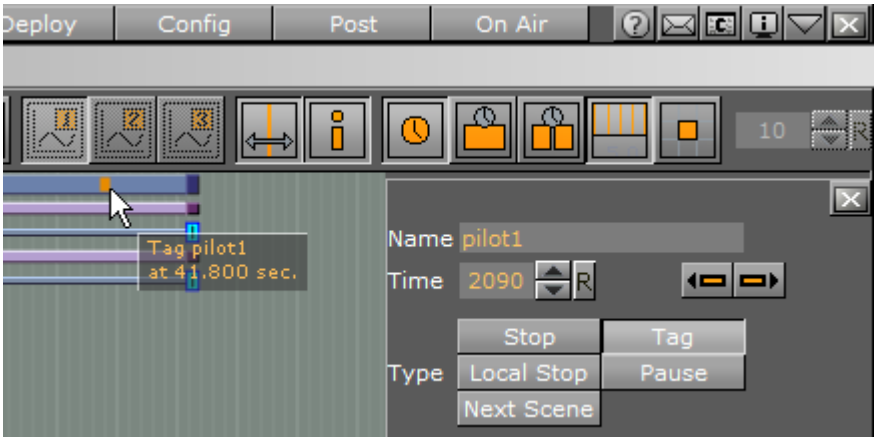
1. Select the Event to be deleted.
2. Click the  (Delete Stop/Tag) button.

or

1. Select one or more Events.
2. Press <Delete>.

To Create a Tag (Preview Point) for External Applications

(for example Viz Trio and Viz Pilot)

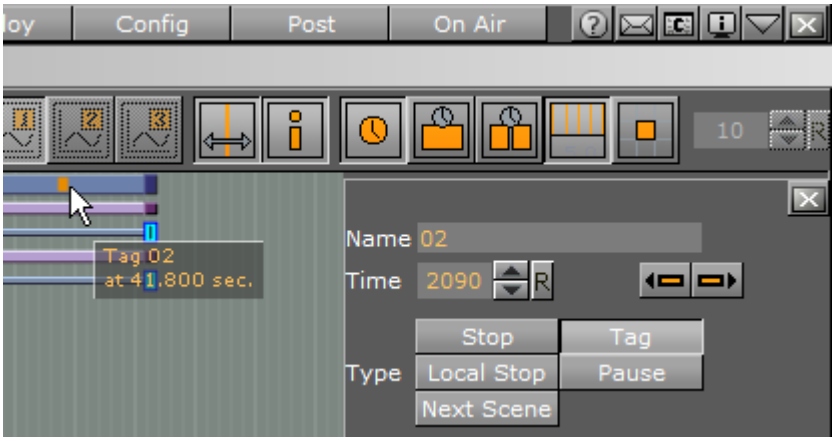


1. Open the Stage

2. Select the default Director.
3. Position the Time Marker at the point where the Scene is to be previewed.
4. Click the  (Add Stop/Tag) button to add an Event.
5. In the Event Editor, enter `pilot1` (case sensitive) in the **Name** text box. The name **pilot1** is the Tag name requested by control application, and is also the default preview point for an Object scene in Transition Logic-enabled scenes. For labeling more Tags, or preview points, continue with pilot2, pilot3, etc. Make sure `Tag` is selected in the **Type** area.

> ⚠ **Note:** Custom Tag Events can be used as well. However, those Tags Events are only selectable within the Viz Pilot Newsroom client.

To Create a Tag (preview point) for Viz Trio



1. Open the Stage, and select the default Director.
2. Position the Time Marker at the point where the Scene is to be previewed.
3. Click the  (Add Stop/Tag) button to add an Event.
4. In the Event Editor enter the **Name** as **01**, **02**, **03**, etc. according to the value of the corresponding Field identifiers (01, 02, 03, etc.).
    - Setting these points will, when a tab field in Viz Trio is selected (01, 02, 03, etc.), preview the Scene in the local Viz Trio preview window.

Preview Callback Events

> ⚠ **IMPORTANT!** This event must only be triggered for the local preview in the External Control Application, and not for any external preview or program engines.

Viz Engine supports a script event called **OnPreview** upon loading a scene in an External Control Application, such as Viz Trio or Viz Pilot. This gives the designer additional capabilities for when the scene is loaded in the local preview, as it can be used to show or hide parts of the scene to make it clearer for the operator what will be shown later in the animation, or to write some instructions for the operator, while the scene should be taken on air without changes.

It may also be used to compensate for unavailable capabilities on a preview Engine. For example, if a scene plays a video in the background, which a VGA mode preview Engine cannot play, a back plate can be shown in the preview with a snapshot of the video. This will give the operator an idea of how the scene will look like when taken on air. It could also be used as an optimization, to save the local preview engine from loading large video files.

⚠ **Note:** Preview Callback Event support requires Viz version 3.8.1 or higher.

The **OnPreview** script event can be used in a scene or container script. If adding the script as a container script, make sure it is added to the same container as the Control Object plug-in. The script can also be loaded as both a scene and a container script, in which case both scripts will be invoked when the scene is loaded.

In Transition Logic scenes, the function can be placed in the scene script of the Master scene, or as a container script on the Control Object container in one or more of the Object scenes. In Transition Logic scenes, the scene script of the Master scene, *and* the container script on the root Control Object container of all Object scenes, is invoked.

A simple **OnPreview** script event handler could be like:

```
sub OnPreview(doPreview as Integer)

if doPreview then

println("4015 Container OnPreview")

end if

end sub
```

See Also

- Transition Logic
- Scripting

11.2 Actors

A Container is represented as an Actor in the Stage Tree. The Actor carries the same name in the Stage Tree as the corresponding Container in the Scene Tree. If the Container is renamed, the Actor will reflect the change.

If an Actor or Actor channel is clicked on it will also show the relevant Container (in the Scene tree) and object (in the Scene Editor).

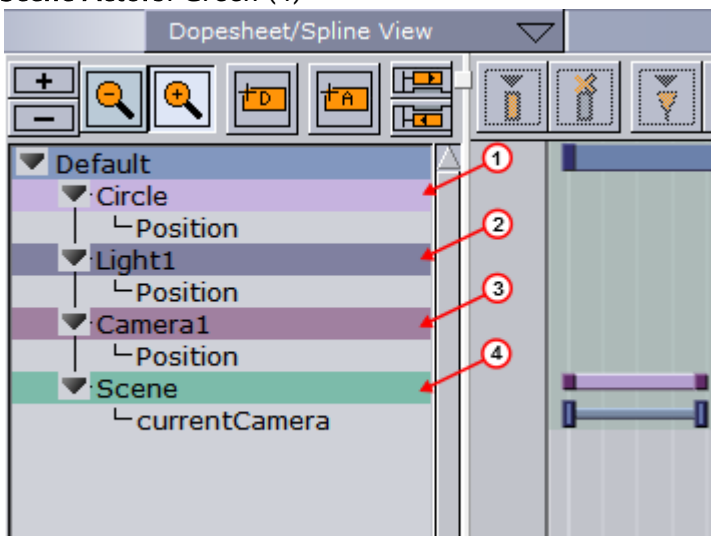
⚠ **Note:** If multiple Actors or Actor Channels are selected this is also shown in the Scene Tree and Scene Editor.

When adding an animation to a Container, the corresponding Actor and channel will automatically show in the Stage Tree inside a Director. The Actor for a Container holds the information of all animated channels: Scene, Camera, or Light.

As it is possible to place different animation channels of an Actor in separate Directors, the same Actor can be available in several places in the Stage Tree.

Each Actor type is represented by a different color:

- **Container Actors**: Light purple (1)
- **Light Actors**: Dark gray (2)
- **Camera Actors**: Dark purple (3)
- **Scene Actors**: Green (4)

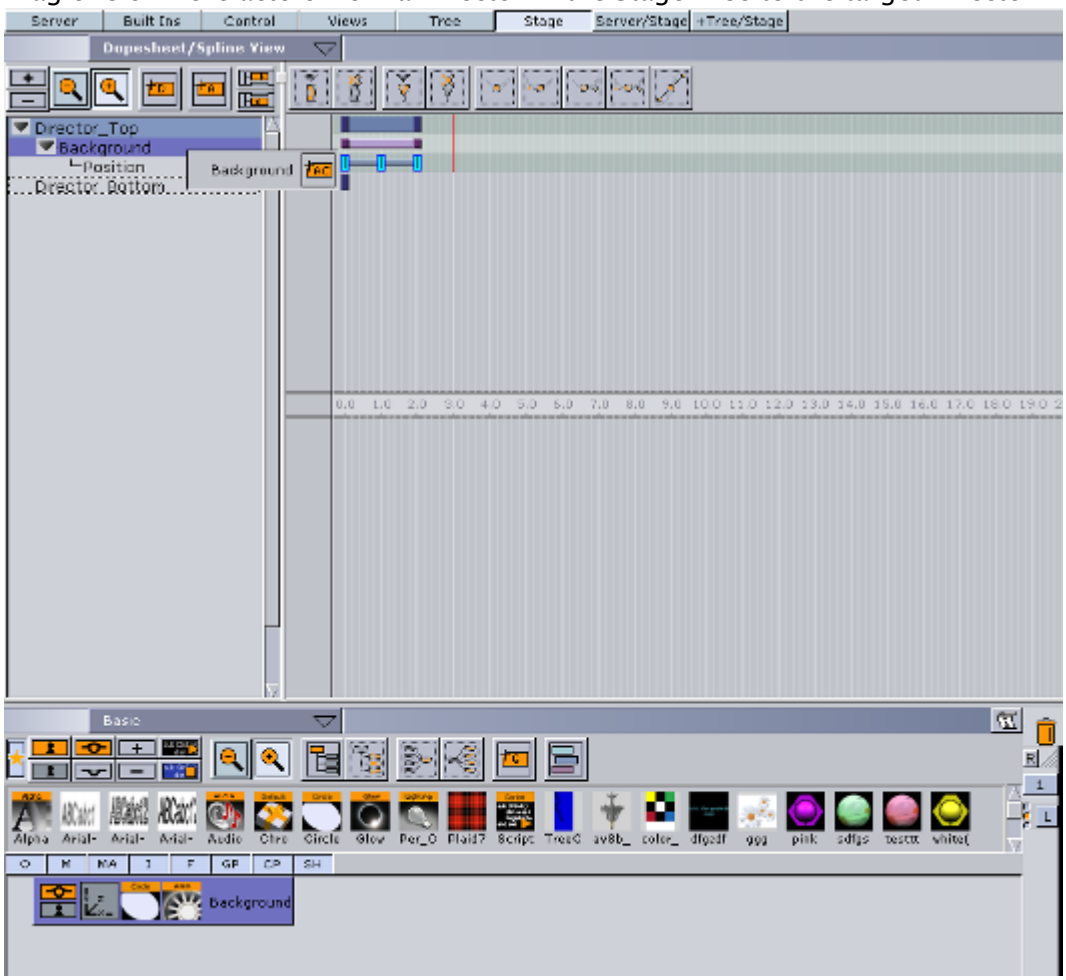


Most actions on Actors are done in the Stage Tree. Actions which affect the time-line are done directly in the Dopesheet editor. Working with Actors in the Dopesheet editor enables the Actor Editor.

11.2.1 Actors in the Stage Tree

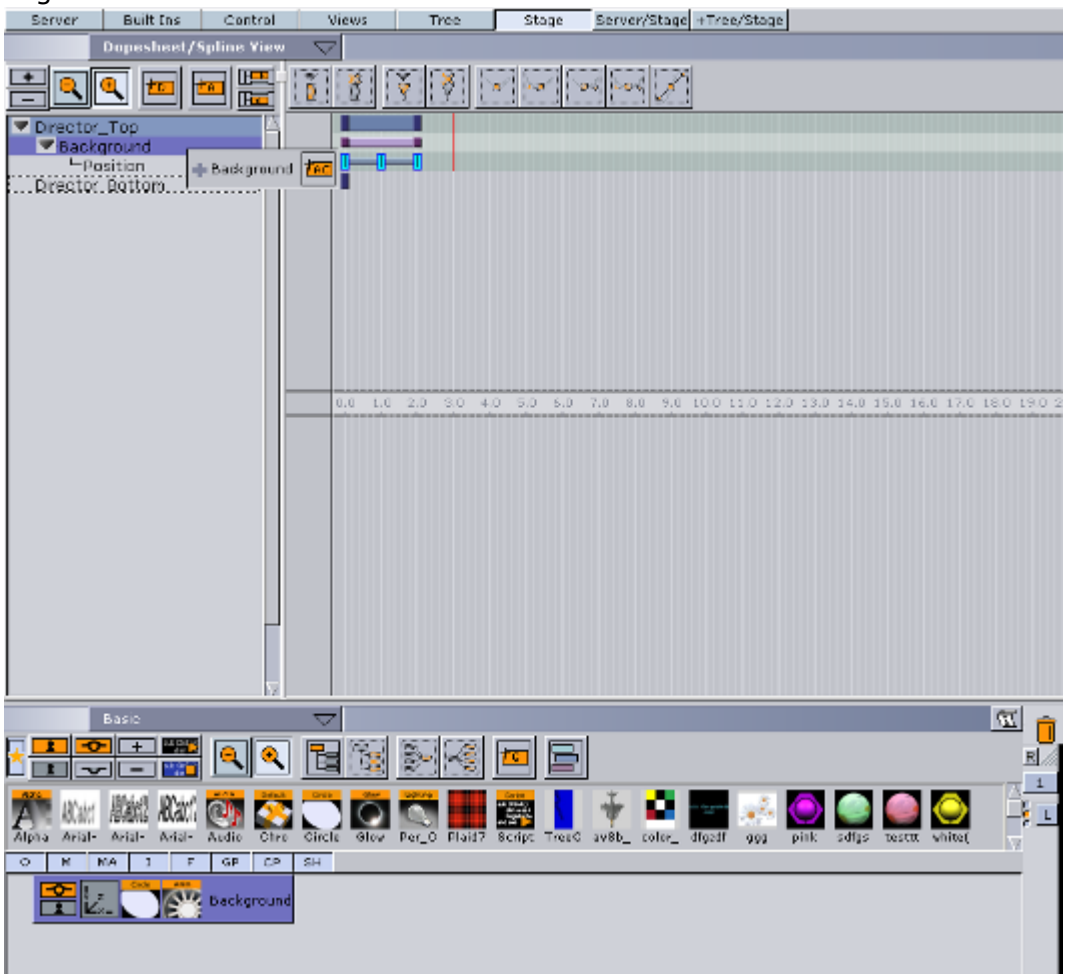
To Move an Actor

- Drag one or more actors from a Director in the Stage Tree to the target Director.



To Copy an Actor

- Press <Ctrl> while dragging one or more actors from a Director in the Stage Tree to the target Director.



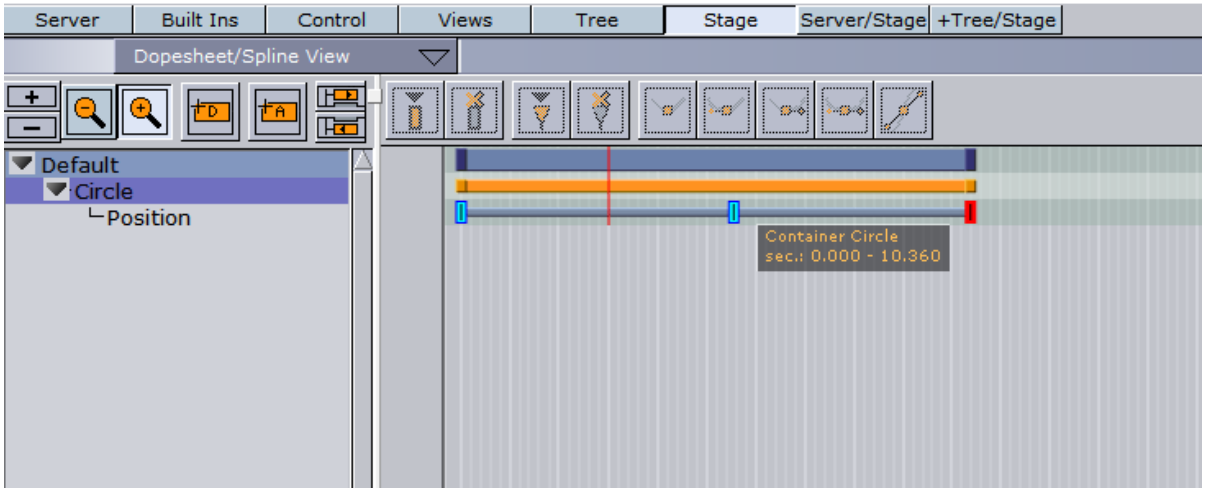
To Delete an Actor

- Right-click an actor in the Stage Tree and select **Delete Actor**.
  or
- Drag the actor to the trash can.

11.2.2 Actors in the Dopesheet Editor

To Move an Actor in the Time-line

- Drag the actor to a different time.



To Speed Up or Slow Down an Actor

- Click on the start or the end of the Actor.
- Drag the start or end point to change the Actor speed:
  - **Speed up:** Lengthens the Actor.
  - **Slow Down:** Shortens the Actor.

To Delay all the Animations in an Actor

1. Enable the Actor Editor.
2. Select an actor.
3. In the Actor Editor, set the **Offset** value (in fields).
   Also, it is possible to move the actor (horizontally along the time-line) including all its animations directly in the Dopesheet Editor, although this will not actually add the offset value.

11.3 Channels

A Channel stores the animated properties of an Actor.

Separate Channels are created for Position, Rotation, Scaling, and so on. The Channel is named automatically according to the property being animated. It is not possible to rename a Channel.

Key Frames are placed along the Channel. Key Frames hold information about the value of the property at the time they are placed on the time-line. If Key Frames are not connected by a Channel in the Dopesheet Editor, the information about the property is identical, and there is no animation of this property between these Key Frames.

Most actions on Channels are done in the Stage Tree. Actions which affect the time-line are done directly in the Dopesheet editor. Working with channels in the Dopesheet editor enables the Channel Editor.

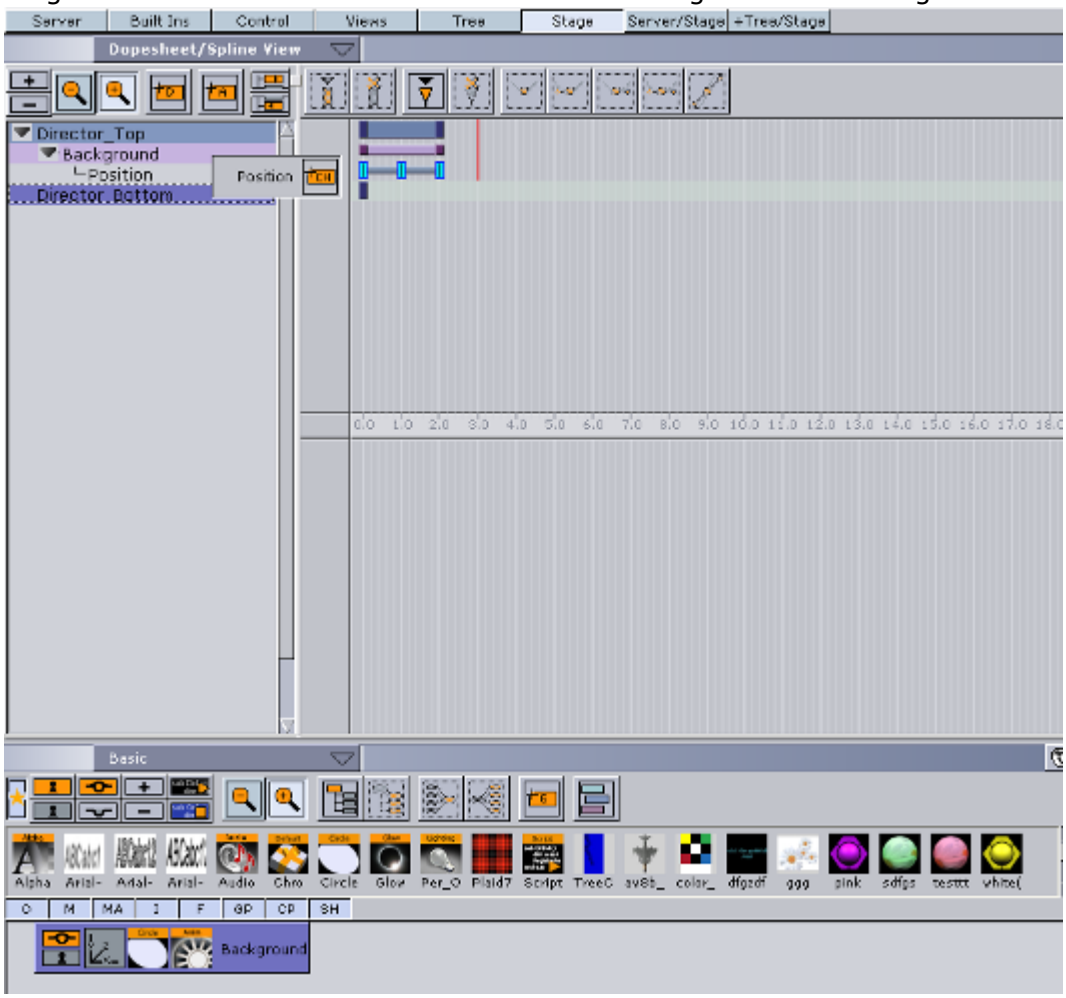
This section contains the following information:

- Channels in the Stage Tree
- Channels in the Dopesheet Editor

11.3.1 Channels in the Stage Tree

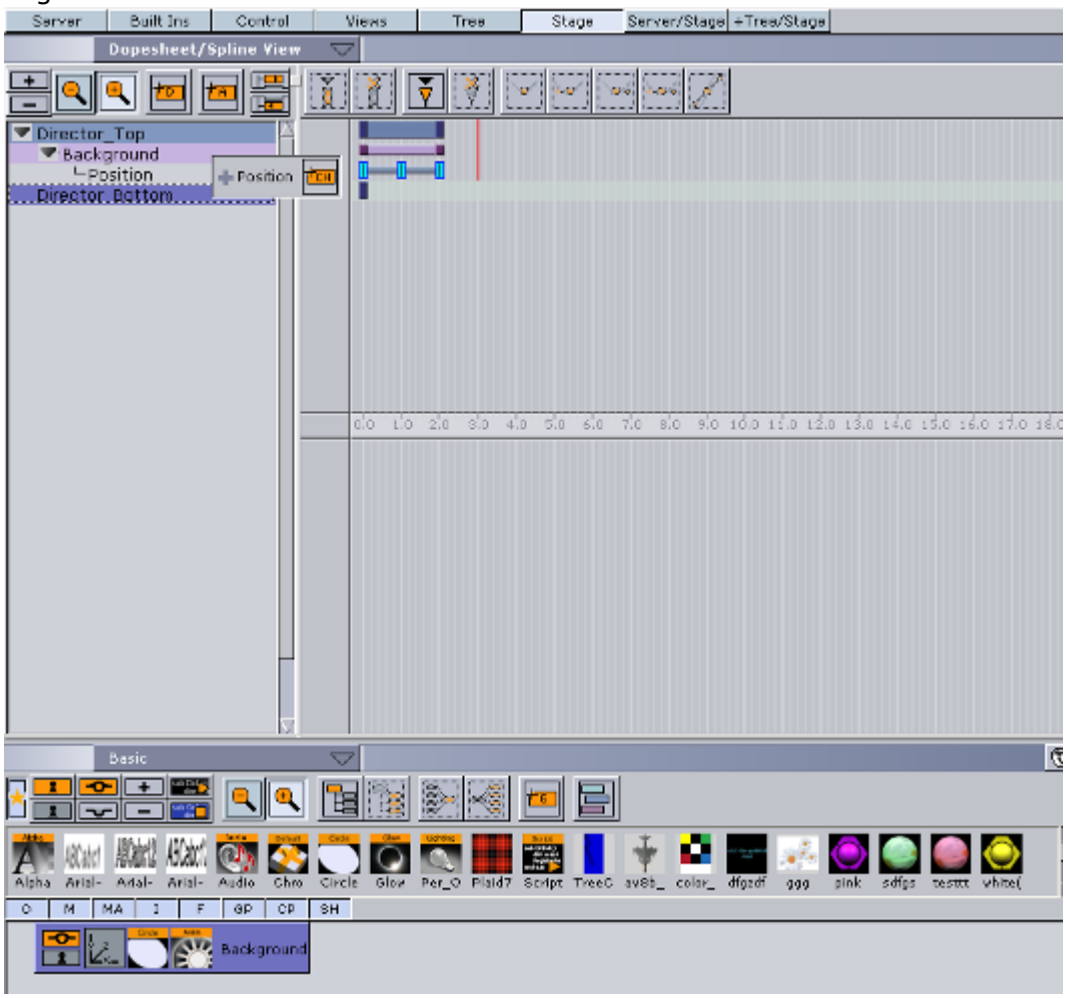
To Move a Channel

- Drag one or more channels from a Director in the Stage Tree to the target Director.



To Copy a Channel

- Press <Ctrl> while dragging one or more channels from a Director in the Stage Tree to the target Director.



To Select a Channel

- Click a channel in the Dopesheet Editor, or
- Click one or more channels in the Stage Tree.

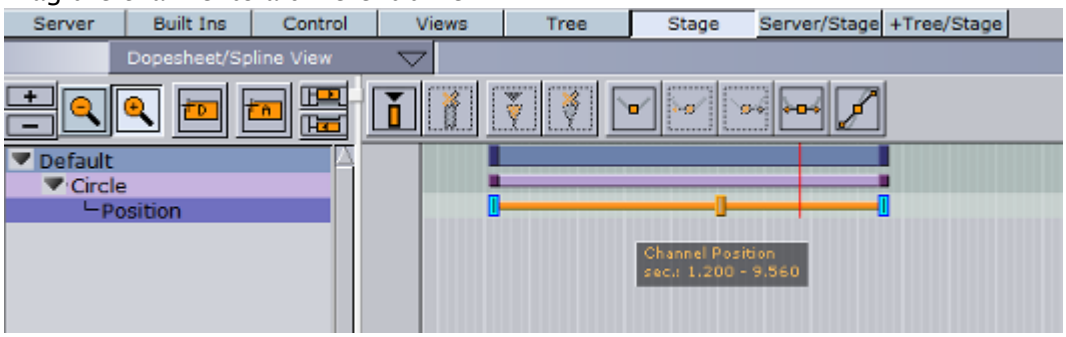
To Delete a Channel

- Right-click a channel in the Stage Tree and select **Delete Channel**
  or
- Drag the channel to the trash can.

11.3.2 Channels in the Dopesheet Editor

To Move a Channel in the Time-line

- Drag the channel to a different time.



To Speed Up or Slow Down a Channel

- Click on the start or the end Key Frame of the Channel.
- Drag the start or end Key Frame to change the Channel speed:
  - **Speed up:** Lengthens the Channel.
  - **Slow Down:** Shortens the Channel.

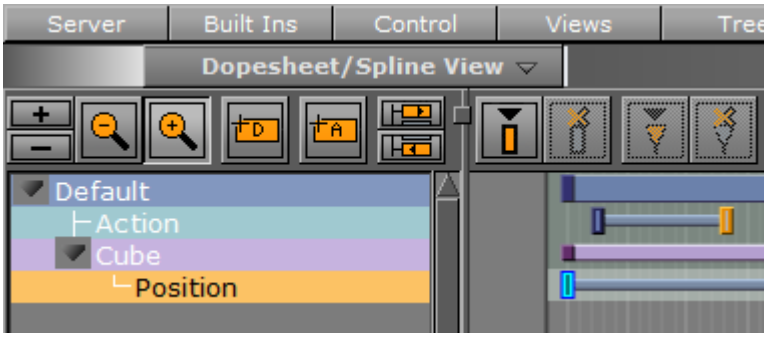
11.4 Action Channels

Actions Channels are control items which, when an Action Key Frame is added, do special tasks when the time line passes them.

Regular Channels are typically basic actions that can be created through the use of the Viz Artist GUI. For example, the rotation of a cube can be defined in the Transformation Editor of the cube container.

For more special scenarios, such as audio animations, it may not be possible to create the animations based on the standard options. In such cases it is necessary to create actions with custom scripts that suit the situation.

Action Channels are colored cyan in the Stage Tree.



Most procedures on Action channels are done in the Stage Tree. Procedures which affect the time-

line are done directly in the Dopesheet Editor. Working with Action channels in the Dopesheet Editor enables the Action Channel Editor.

11.4.1 Action Channels in the Stage Tree

To Create an Action Channel

The created Action Channel will also require an Action Key Frame.

1. Select a Director.
2. Click  (Add New Action) in the **Stage Tree** Bar.
   or:

- Drag the  icon in the Stage Tree Bar to a Director.

To Move an Action Channel

- Drag an action channel from a Director to the target Director

To Copy an Action Channel

- Press <Ctrl> while dragging an action channel from a Director to the target Director

To Rename an Action Channel

1. Right-click an Action channel and select **Rename**.
2. In the text box that shows, type a new descriptive name.
3. Press <Enter>.

> ⚠ **Note:** Special characters, such as space, are not allowed in action channel names.

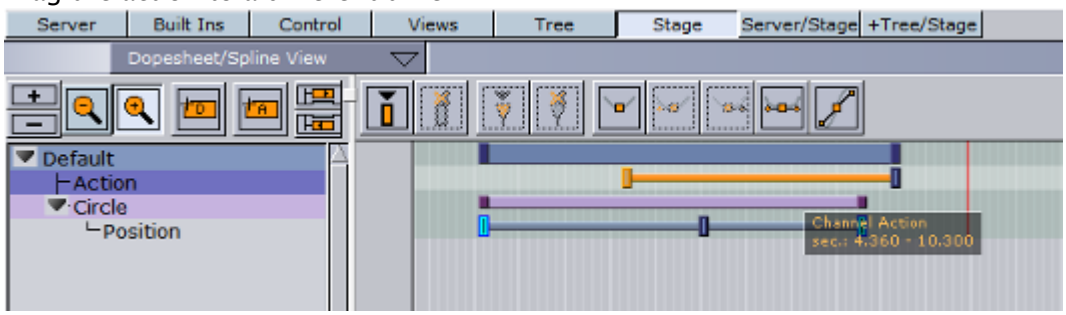
To Delete an Action Channel

- Right-click an action channel in the Stage Tree and select **Delete Action Channel**.
  or
- Drag the action channel to the trash can.

11.4.2 Action Channels in the Dopesheet Editor

To Move an Action Channel in the Time-line

- Drag the action to a different time.



To Speed Up or Slow Down an Action Channel

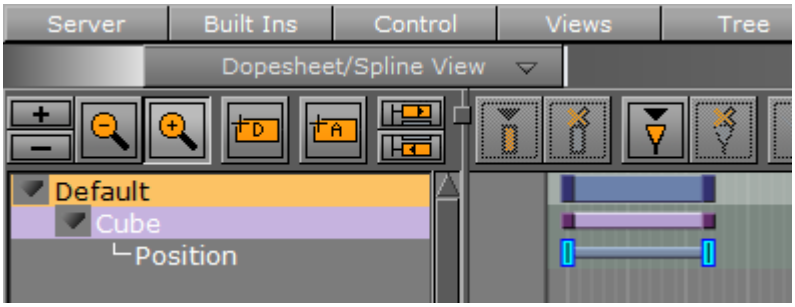
- Click on the start or the end Key Frame of the Section Channel.
- Drag the start or end Key Frame to change the Action Channel speed:
  - **Speed up:** Lengthens the Action Channel.
  - **Slow Down:** Shortens the Action Channel.

11.5 Key Frames

A Key Frame stores information about a property at a given time. Key Frames can be added to Channels and Action channels. If two Key Frames next to each other hold different values, the Key Frames are connected with a bar in the Dopesheet Editor.

When selecting a Key Frame in the Dopesheet editor or the Spline Editor, the corresponding Key Frame in the other editor will be selected too. Working with Key Frames in the Dopesheet enables the Key Frame Editors.

⚠ **Example:** The position of a cube can be at X=0 at field 0 and X=100 at field 50. In such a scenario, two Key Frames will be created, one at field 0 and one at field 50. Since the position of the cube is different in the two Key Frames, a bar is added between them, indicating that an animation takes place when the time-line moves from Key Frame A to Key Frame B:



This section contains information on the following procedures:

- To Add a Key Frame
- To Select One or More Key Frames
- To Move One or More Key Frames
- To Move a Key Frame to a Specific Time
- To Copy a Key Frame
- To Rename a Key Frame
- To Delete a Key Frame
- Reverse the Order of Selected Key Frames
- To Reverse the Order of Key Frames

11.5.1 To Add a Key Frame

There are four ways to create a Key Frame:

1. Select the required **Channel** or **Action Channel** in the Stage Tree.
2. Set the Time-line Marker to when the Key Frame should be added.
3. Click  (Add Key Frame).
   Or:

1. Select the required **Channel** or **Action Channel** in the Stage Tree.
2. Set the Time-line Marker to when the Key Frame should be added.
3. Set the Key Frame:
   - Click  in the Time-line Editor, or
   - With the mouse pointer over the Scene Editor, press <Enter>.
     Or:

1. Select the required **Channel** or **Action Channel** in the Stage Tree.
2. Set the Time-line Marker to when the Key Frame should be added.
3. In the Spline Editor, make sure to hold the pointer above the spline at the required time-line location (the spline should turn orange if doing this correctly).
4. Click the spline.

11.5.2 To Select One or More Key Frames

- Click on a Key Frame to select it. This will clear the selection of any previously selected Key Frames.
- Ctrl-Click on a previously selected Key Frame to clear the selection of it.

- Ctrl-Click on Key Frames to have more than one selected.
- Shift-Click on a segment between two Key Frames to select both.
- Ctrl-Click on a segment to select all Key Frames.

> ⚠ **Note:** When several Key Frames are selected, actions like move or delete will affect all selected Key Frames.

11.5.3 To Move One or More Key Frames

1. Select one or more Key Frames.

> ⚠ **Tip:** You can also hold down the <S> key and use the rubber band to select multiple Key Frames.

2. Drag the Key Frames along the time-line in the Dopesheet Editor.
   Or:

1. Select one or more Key Frames.
2. Drag the Key Frames along the time-line (horizontally) in the Spline Editor.

> ⚠ **Note:** Dragging a Key Frame vertically in the Spline Editor will change the value of the Key Frame, not the time-line value.

11.5.4 To Move a Key Frame to a Specific Time

1. Enable the Key Frame Editors.
2. Select a Key Frame.
3. In the Key Frame editor, enter the time (in fields) in the **Time** input box.

11.5.5 To Copy a Key Frame

1. Select one or more Key Frames.
2. Press <C> while dragging the Key Frames to the new position in the Dopesheet Editor.

11.5.6 To Rename a Key Frame

1. Enable the Key Frame Editors.
2. Select a Key Frame.
3. In the Key Frame editor, enter a descriptive Key Frame name in the **Name** text box, and press <Enter>.

> ⚠ **Note:** Special characters, such as space, are not allowed in Key Frame names.

11.5.7 To Delete a Key Frame

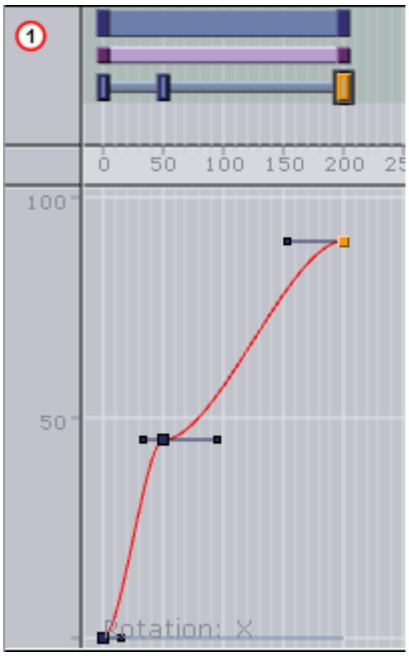
1. Select one or more Key Frames.
2. Click the  (Delete Key Frame) button.
   Or:

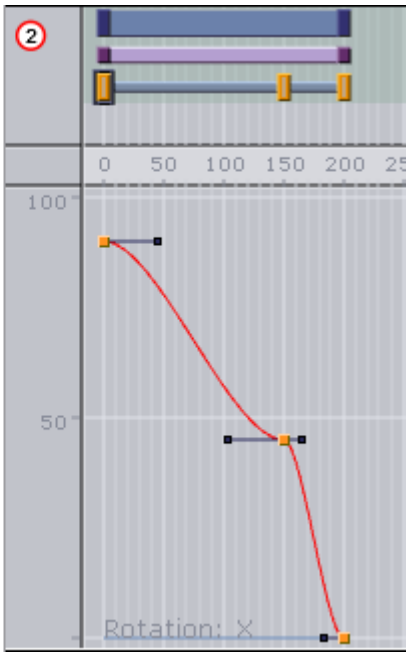
1. Select one or more Key Frames.
2. Press <Delete>.

11.5.8 Reverse the Order of Selected Key Frames

Reversing the order of Key Frames is like mirroring the Key Frames.

⚠ **Example:** Image 1 (below) illustrates a rotation animation, with Key Frames placed at fields 0, 50, and 200. The rotation value at field 0 is X=0, at field 50 X=45, and at field 200 X=90. After the Key Frames have been rotated, image 2 illustrates the animation with Key Frames placed at fields 0, 150, and 200. The rotation value at field 0 is X=90, at field 50 X=45, and at field 200 X=0.





11.5.9 To Reverse the Order of Key Frames

1. Select two or more Key Frames in the Dopesheet Editor.
2. While holding the mouse pointer over the Stage, Press <Ctrl+R>.

11.6 Basic Animation Functions

This section contains information on the following procedures:

- To Play All Animations in a Director from the Start of the Time-line
- To Play Multiple or All Animations in a Scene
- To Play an Animation in Reverse
- To Stop a Director
- To Stop an Animation
- To Stop all Animations
- To Continue an Animation
- To Jump to Start of an Animation
- To Jump to the End of an Animation
- To Delay Animations in a Director with Offsets
- Scale Animations
- To Scale Animations in a Director
- To Scale a Region Within a Director

11.6.1 To Play All Animations in a Director from the Start of the Time-line

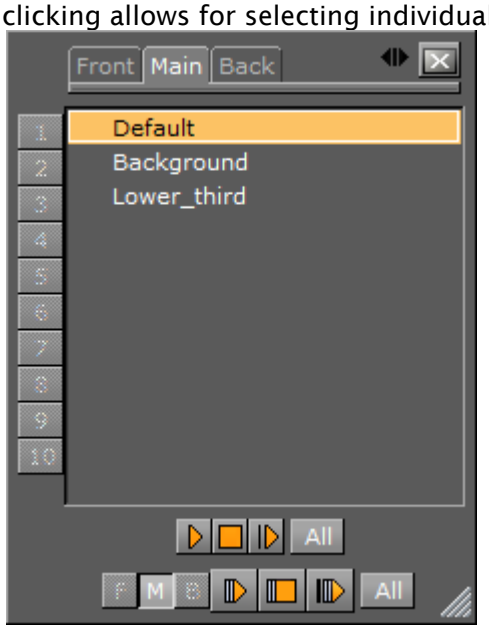
1. Select the required Director.
2. Either
   - Click the **Start Animation** button in the Time-line Editor.
   - When the **Stage** is active, pressing <Enter> will start any animations on the selected director(s) from the first frame. Pressing <Space> will pause or resume play-out.
   - Click the **Start Animation of Selected Layers** button from the Director Control Panel.

11.6.2 To Play Multiple or All Animations in a Scene

When working with a scene that contains more than one director, the current director will always be selected by default in the scene editor preview window. To preview all animations in a multi-director scene, all directors can be started simultaneously:

1. Open the Director Control Panel.

2. Click the **All** button, or hold Shift and click to select the range of directors. Holding Ctrl and clicking allows for selecting individual directors instead of a range.



3. To start the animation(s), either
   - Click the **Start** button in the Time-line Editor.
   - Click the **Play layer** button in the Director Control Panel.
   - Click the **Start animation of selected layers** button in the Director Control Panel. This will also start the selected animations in the other layers, if they are available.
   - When the **Stage** is active, pressing <Enter> will start any animations on the selected director(s) from the first frame. Pressing <Space> will pause or resume play-out.

11.6.3 To Play an Animation in Reverse

1. In the Time-line Editor, click the  (Reverse Play) button.
2. Either
   - Click the **Start** button in the Time-line Editor.
   - Click the **Play layer** button in the Director Control Panel.
   - Click the **Start animation of selected layers** button in the Director Control Panel. This will also start the selected animations in the other layers, if they are available.
   - Press <Shift+Enter> when the Stage is active

11.6.4 To Stop a Director

1. Select one or more Directors.
2. Either

- In the Time-line Editor, right-click the **Stop** button.
- Click the **Stop** button from the Director Editor.
- Press <SPACE> while the Stage is active.

11.6.5 To Stop an Animation

- In the Time-line Editor, click the **Stop** button.
  It is also possible to add one or more stop points to a Director, so that each time the time-line reaches a certain frame, the animations stop automatically.

11.6.6 To Stop all Animations

1. Enable the Director Control Panel
2. Select all directors and all layers.
3. Click the **Stop animation of selected layers** button. This stops all animations in all layers.

11.6.7 To Continue an Animation

1. Select the required Director.
2. Either:

- Click the **Continue Animation** button in the Time-line Editor
- While in the Stage, press <SPACE>
- Click the **Continue Animation** button in the Director Editor.

11.6.8 To Jump to Start of an Animation

1. Select the required Director.
2. Click the **Go to Start** button in the Time-line Editor, or press <Ctrl+Home>.

11.6.9 To Jump to the End of an Animation

1. Select the required Director.
2. Click the **Go to End** button in the Time-line Editor, or press <Ctrl+End>.

11.6.10 To Delay Animations in a Director with Offsets

1. Enable the Director Editor and select the required Director.
2. In the Director editor, set the **Offset** value (in fields).
   It is also possible to move the Director, including all its animations, horizontally along the time-line in the Time-line Editor. This will not add the offset value.

11.6.11 Scale Animations

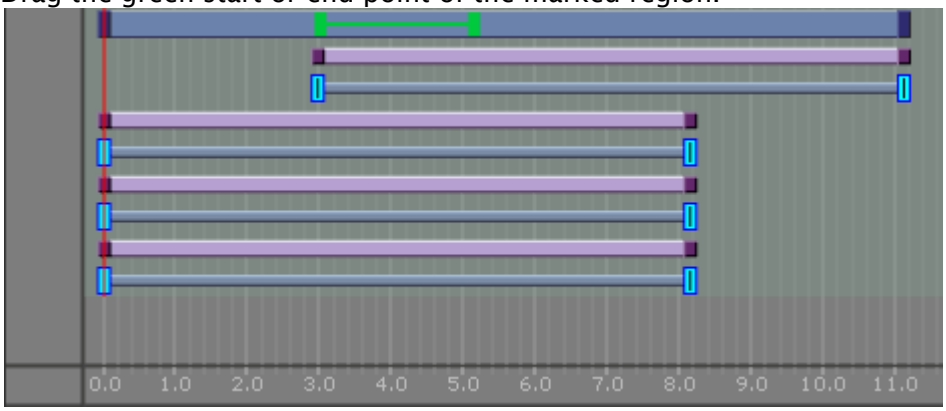
Scaling a Director is the easiest way to change the total running time of the animations in the Director. Scaling is done proportionally and includes all Sub-Directors.

11.6.12 To Scale Animations in a Director

- Drag the start or the end point of a Director in the Dopesheet Editor. It is not possible to drag the start point beyond the offset value.

11.6.13 To Scale a Region Within a Director

1. Mark a region within a director.
2. Drag the green start or end point of the marked region.



11.7 Create A Basic Animation

This section details how to create a basic animation in a Scene.

11.7.1 Animation Task

Create a Scene with single a Cube Geometry. Animate the Cube to move right to left.

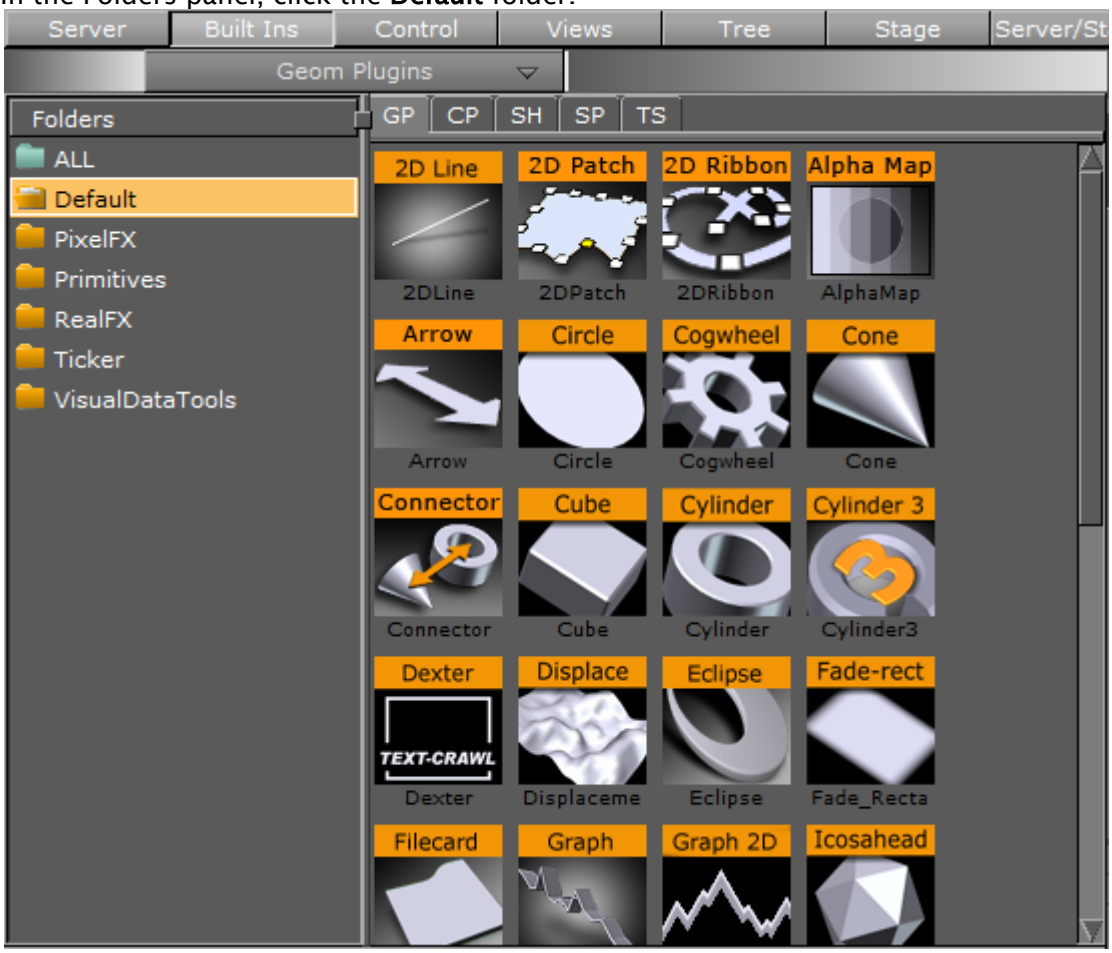
This use case contains the following procedures:

- Animation Task
- To Create the Basic Animation Scene
- To Create a Basic Animation
- To Create Key Frames with the Transformation Editor

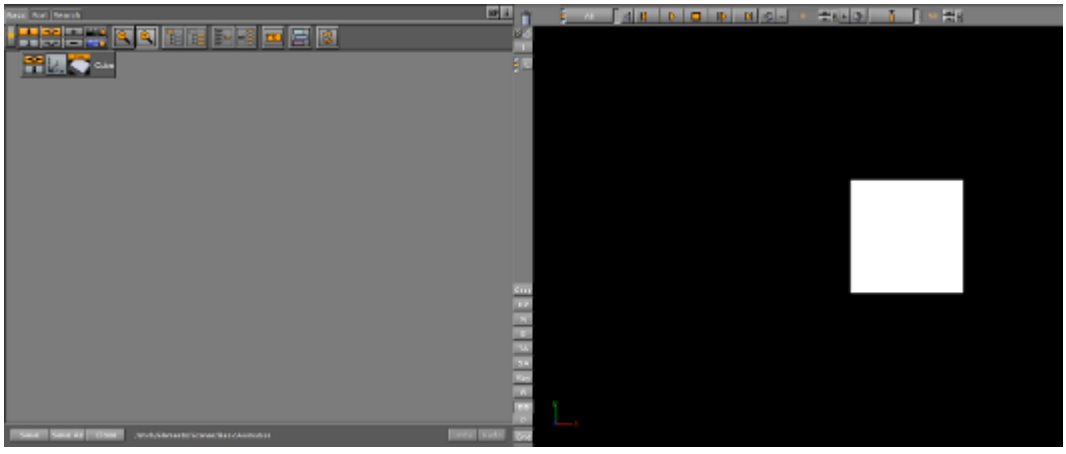
11.7.2 To Create the Basic Animation Scene

1. Create a new scene, and save it as Basic Animation.
2. Open the Basic Animation Scene.
3. From the Main menu, click Built Ins.
4. Click the **GP** (Geometry plug-in) tab to show all available Geometry plug-ins.

5. In the Folders panel, click the **Default** folder.



6. Drag the **Cube** icon from the plug-in list into the Scene Tree.
7. Click **Save**.
8. The Scene Tree and Scene Editor should then look like this:



11.7.3 To Create a Basic Animation

1. Open the Basic Animation Scene.
2. Set the time-line, in the Scene Editor (1), to 0 (fields).

3. Create the first Key Frame:

   a. Click on the Cube in the Scene Editor.
   b. Click the **Set Key Frame** button (2) or press <Enter>. This sets the first Key Frame.
4. Create the second Key Frame:
   a. In the Scene Editor, drag the Cube to the right, so that it shifts to a different position.
   b. Click the **Set Key Frame** button or press <Enter>. This sets the second Key Frame.

> ⚠ **Tip:** Use the Transformation editor to move the Cube to precise position.

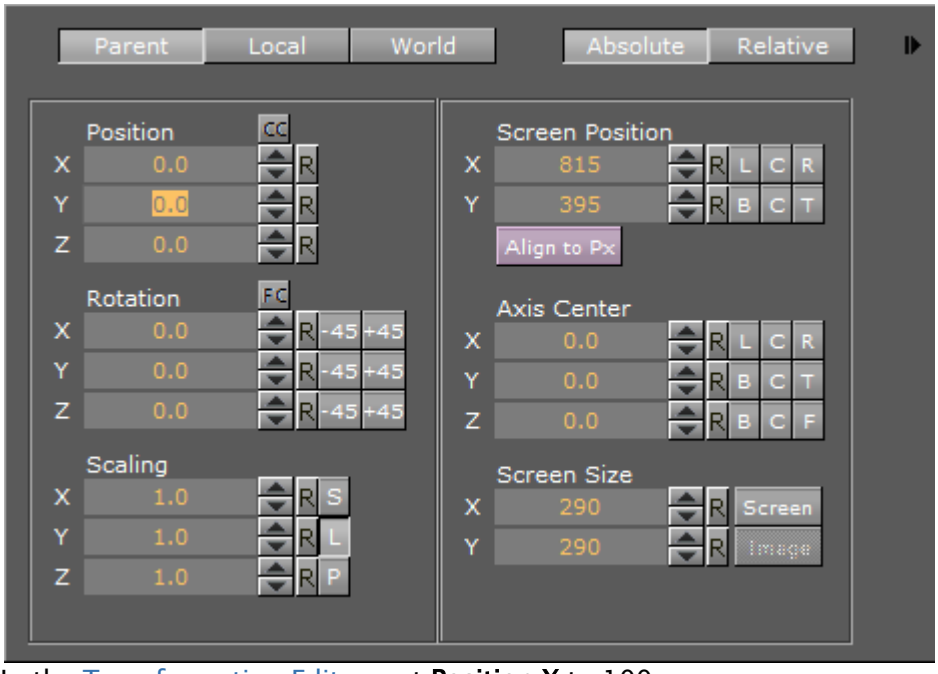
> ⚠ **Note:** On the Cube Container, in the scene tree, an Anim icon is automatically added, which shows that one or more properties on this Container has been animated.

5. Repeat from step 1 to create more movement, if required.
6. Click the **Start Animation** button (3) to see the created animation.

11.7.4 To Create Key Frames with the Transformation Editor

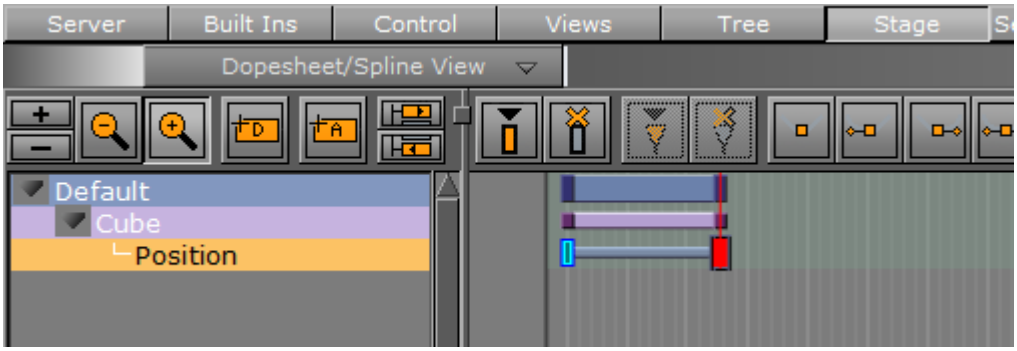
Use the Transformation editor to move the Cube to precise locations then set a Key Frame.

1. After the first Key Frame is set, click the  (Transformation) icon on the Cube container in the scene tree.



2. In the Transformation Editor, set **Position X** to 100.
3. Click the **Set Key Frame** button or press <Enter>. This sets the second Key Frame.

4. Move the Cube as required with the **Position** parameters to a new position. In the new position click the **Set Key Frame** button, or press <Enter>, to set a Key Frame.
   A representation of the animation can be seen in the Stage Tree Area.



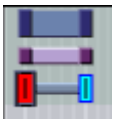
11.8 Create An Advanced Animation

Once you have mastered Create a Basic Animation, where the position of a cube is moved, you can move on to more advanced tasks. This section provides some additional procedures, so that a cube can continue its journey rightward, rotate while moving, and gain a nicer appearance by adding color to it.

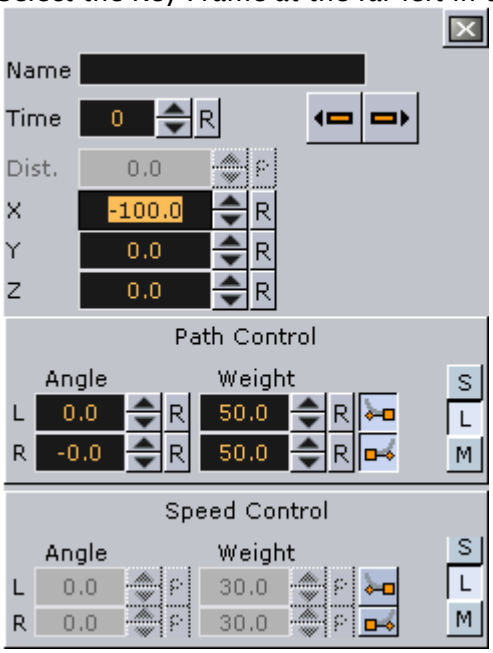
This section contains the following procedures:

- To Edit the First Key Frame
- To Edit the Second Key Frame
- To Create an Additional Key Frame
- To Add a Stop Point
- To Create Key Frames for Rotation
- To Modify the Rotation Settings
- To Run the Animations Independently
- To Add and Animate a Material

11.8.1 To Edit the First Key Frame

1. Enable the Key Frame Editors  .

2. Select the Key Frame at the far left in the Dopesheet Editor.



3. In the Key Frame Editors, set the **Position** to X = **-100**.

11.8.2 To Edit the Second Key Frame

1. Select the Key Frame at the right in the Dopesheet Editor.
2. In the Key Frame Editors, set the **Position** to X = **0**.

> ⚠ **Tip:** Set the position either by typing in the input box, using the up/down buttons, or clicking the R (reset/default setting) button.

When playing the animation now, the cube will still slide rightward, but from a point at the left part of the Scene Editor to the center.

3. Click the **Play** button in the Time-line Editor.

11.8.3 To Create an Additional Key Frame

1. In the Time-line Editor, set the time-line field to **100**.
2. In the Scene Editor, drag the cube to the right, so that it shifts to a different position.
3. Set an additional Key Frame.
4. Click the **Play** button in the Time-line Editor.
   When playing the animation now, the cube will slide rightward from a point at the left part of the Scene Editor, through the center, and all the way to the right part.

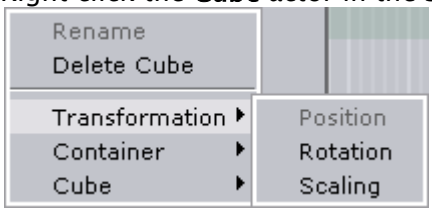
11.8.4 To Add a Stop Point

1. Select the required Director.
2. In the Time-line Editor, set the time-line field to **50**.
3. Create a stop event.

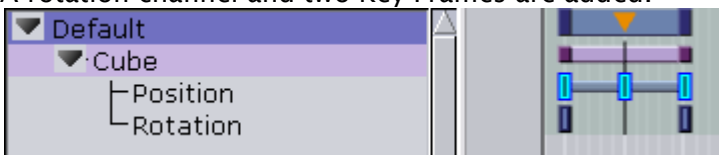
4. Click the **Play** button in the Time-line Editor.
   When playing the animation now, the cube will slide rightward until it reaches the center of the Scene Editor, and then come to a halt.

11.8.5 To Create Key Frames for Rotation

1. Right-click the **Cube** actor in the Stage Tree.



2. In the menu that shows, click *Transformation -> Rotation*.
   A rotation channel and two Key Frames are added.



11.8.6 To Modify the Rotation Settings

1. Select the Key Frame at the right in the Dopesheet Editor.
2. In the Key Frame Editors, set the **Rotation** to X = **360**.
3. Click the **Play** button in the Time-line Editor.
   When playing the animation now, the cube will slide rightward until it reaches the center of the Scene Editor, while rotating all the while before coming to a halt.

11.8.7 To Run the Animations Independently

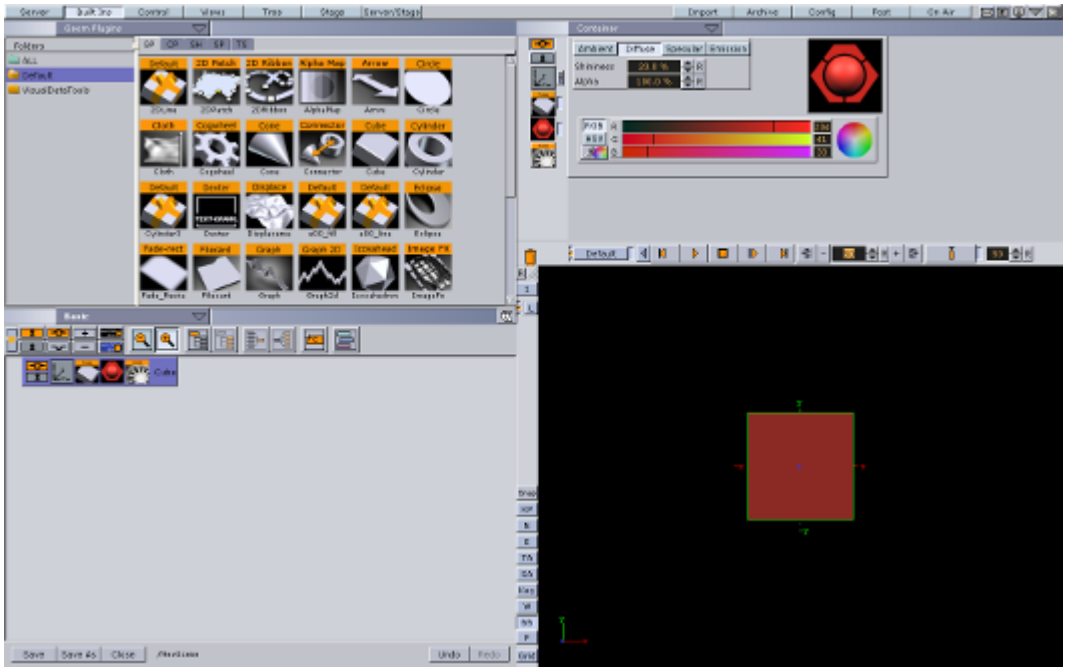
1. Create a Director at the root level of the Stage Tree.
2. Rename the new Director to **Rotation**.
3. Drag the rotation channel to the new Director.
   The Stage Tree and should now look like this:

   When playing the animations now, the output depends on which Director is selected. If the **Default** Director is selected, only the position animation will be played. If the **Rotation** Director is selected, only the rotation animation will be played. Selecting both/all Directors will play both the position and rotation animations.

11.8.8 To Add and Animate a Material

1. Create a Material and define its color.

2. Drag the Material to the Container in the Scene Tree.
3. In the Scene Tree, click the Material in the Container.
   The material editor is shown.



4. In the Time-line Editor, set the time when you want the Key Frame to be created.
5. In the Time-line Editor, click the **Set Key** button.
6. Change the material's color.
7. To create the next Key Frame, in the Time-line Editor, click the **Set Key** button.
8. Click the **Play** button in the Time-line Editor.
   The cube now has a color that changes as part of the animation.

11.9 Advanced Animation Functions

This section details the creation of a Scene with the more advanced animation features in Viz Artist. As the basis, take the Scene created in Create a Basic Animation and improved on in Create an Advanced Animation.

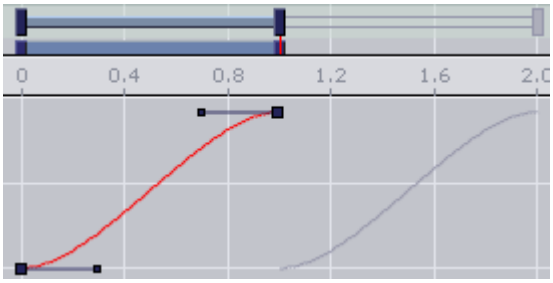
This section contains the following topics:

- Animating Channels
- Path Control
- Spline Control
- Animate Audio
- Animate Cameras
- Animate Light
- Merging Animations
- Multi Select Animation

11.9.1 Animating Channels

Repeat Animation Channels (Loop)

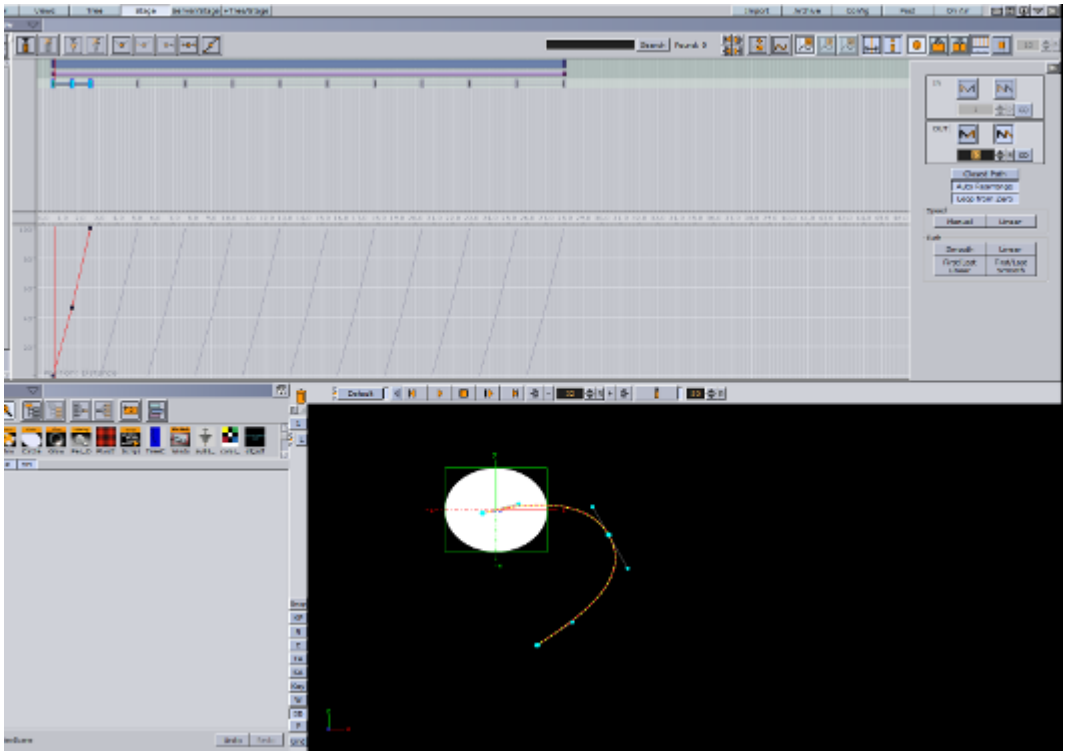
The loop operation makes it possible to repeat the animation of a channel either at the beginning or end of the channel time-line.



To Repeat the Animation of a Channel

1. Select a channel.
2. In the Channel Editor, enable the **Loop In/Out** buttons.

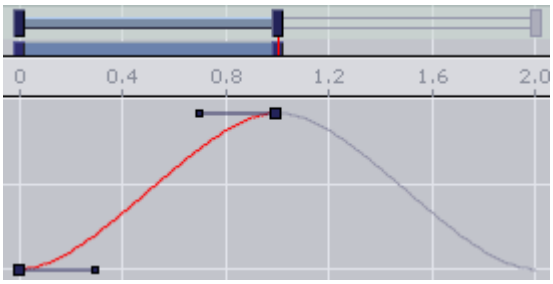
> ⚠ **Note:** Loop at the beginning is only available if the belonging actor has a defined offset value.



3. Define how many times the loop should run before coming to a halt. Either enter a number in the field or click the **Infinite** button.
   When enabling loop, the looped animation shows as gray in the in the Dopesheet Editor

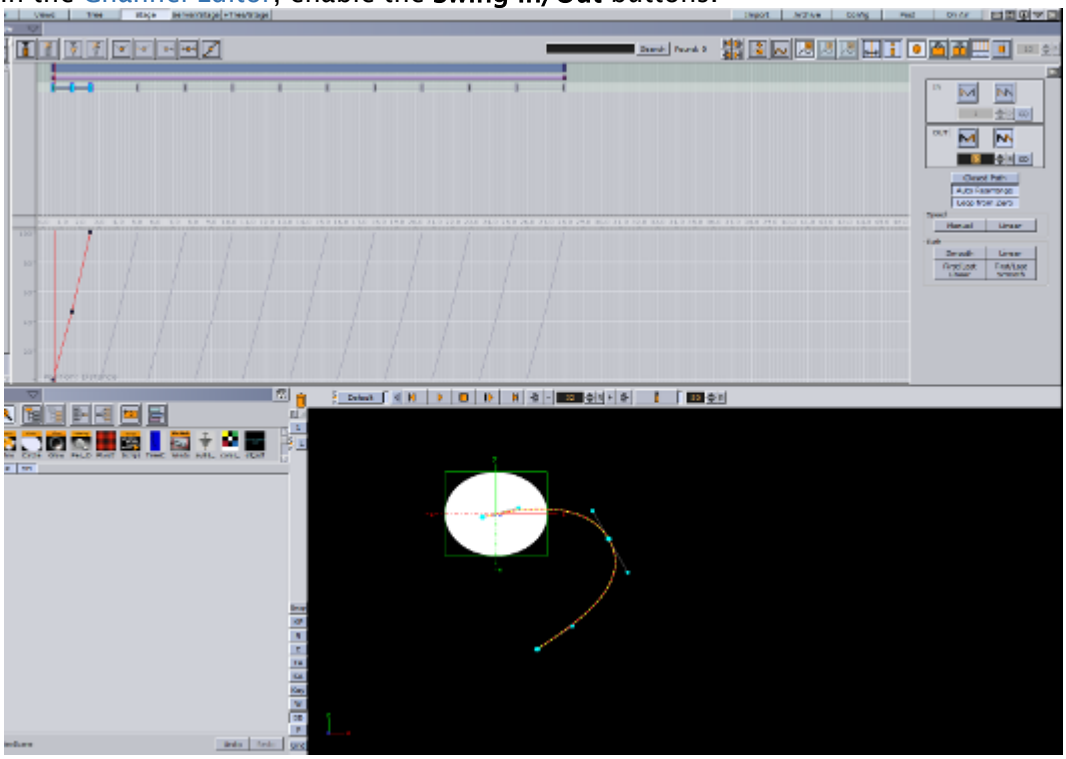
Repeat Animation Channels in Reverse (Swing)

The swing operation makes it possible to repeat the animation of a channel in reverse play either at the beginning or end of the channel time-line.



To Create a Swinging Animation

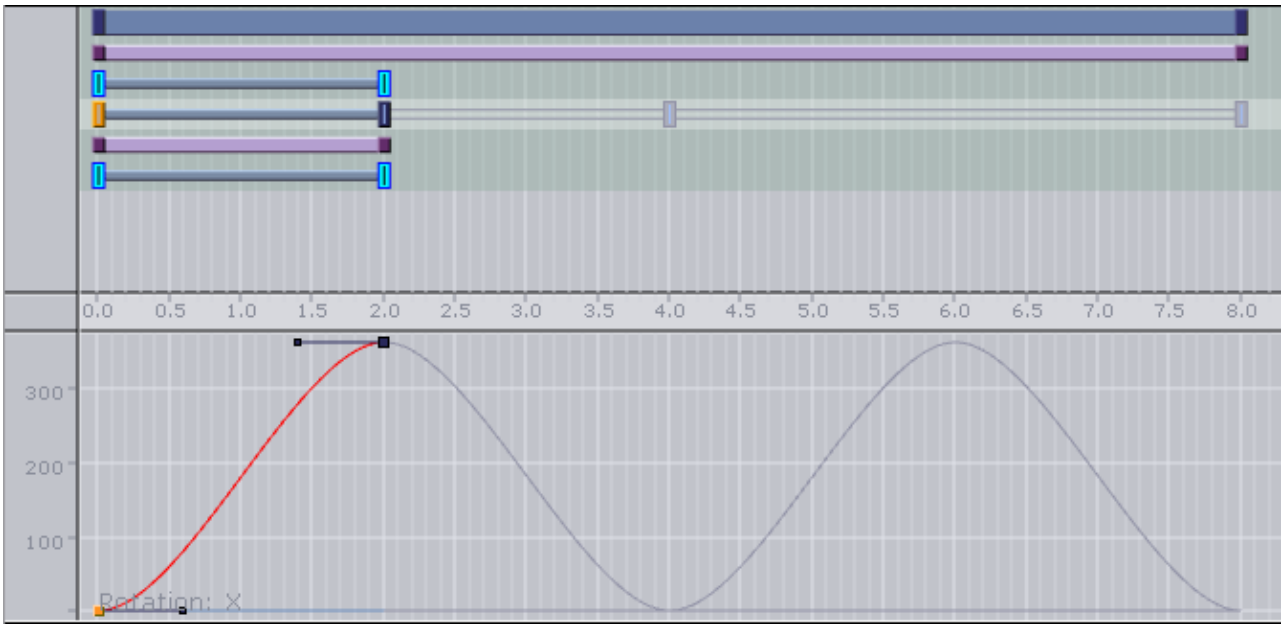
1. Select a channel
2. In the Channel Editor, enable the **Swing In/Out** buttons.



When enabling swing, the animation shows as gray in the in the Dopesheet Editor.

Combine Swing and Loop

It is possible to combine the swing and loop operations on a channel. The animation will then first be added in reverse play (swing) and then started once again from the beginning to the end (loop). This will save you a great amount of time if in need of a recurring animation, for example a bouncing ball. Animations with swings and loops show as gray in the in the Dopesheet Editor.



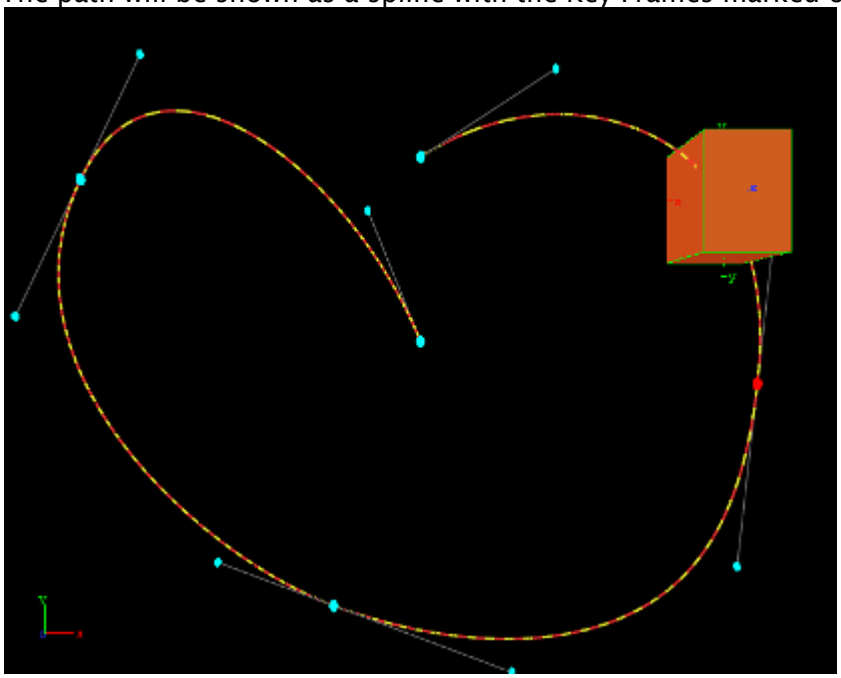
11.9.2 Path Control

The position animation follows a path you define by the Key Frames. A spline will be interpolated from the Key Frames. You can view and edit this spline in the Scene Editor.

To Edit the Spline in the Scene Editor

1. Enable the bounding box:
   - Click the **BB** button to the left of the Scene Editor, or
   - With the mouse pointer over the Scene Editor, press <B>.
2. In the scene tree, select the animated container.

3. Right-click in the Scene Editor and select **Animation**.
   The path will be shown as a spline with the Key Frames marked on it.



4. Drag the Key Frames around to modify the animation trajectory.
   The red and yellow markings on the spline provide information about the speed the object is moving along its trajectory. One segment represents a field, the longer the segments are, the faster the object will travel.

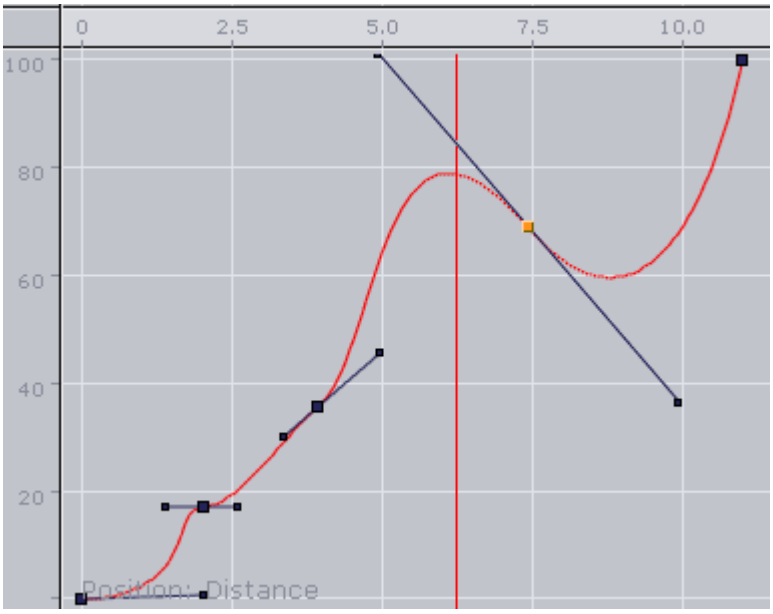
As every Key Frame has handles (seen in blue) you can use them to modify the Key Frame's trajectory. To do so select a handle and drag it around.

By default the handles are set to be locked and proportional. If you want to switch to a mirrored mode, press <Shift> while you drag the handle around. Pressing `<Ctrl>` will allow you to edit the single handle only.

An other option to set the handles is the object editor. Select the Key Frame you want to change the handles in and set either **Single**, **Locked** or **Mirrored** and change the values in **Weight L** and **R**.

11.9.3 Spline Control

In the spline editor the distance spline is shown, which sets the current position of the container along the path. The y-axis in the spline editor is scaled to use the percentage of the trajectory, while the x-axis still uses the time. In the spline editor you can only change the position of a Key Frame along the time-line, but not its percentage of the total trajectory.



By default the handles at every Key Frame on the spline are set to be linear.

To Edit the Spline

- Drag the handles around to change their positions.
  By default the handles are set to move proportionally. So as you move the handle of a Key Frame the other one will move to.
- Press <Ctrl> while you drag a handle to modify only the selected one.
  This “single handle mode” will be enabled for the Key Frame as long as you do not reset it by a click the Key Frame icons in the Stage Editor Bar.

> ⚠ **Example:** If you, for example, modify the spline in such a way that the value decreases, the container will move backwards.

To Move Key Frames in the Spline

- Select the required Key Frame and drag it.
  Multiple Key Frames can be selected to move at once.

To Insert Key Frames from the Spline

1. Moving the cursor on the spline exactly where you would like to add the Key Frame.
2. Click on the required location to set the Key Frame.

To Delete a Key Frame from the Spline

1. Select the required Key Frame
2. Press <Delete> or click Delete Key Frame button in the Stage Editor Bar.
   Multiple Key Frames can be selected to be deleted at once.

11.9.4 Animate Audio

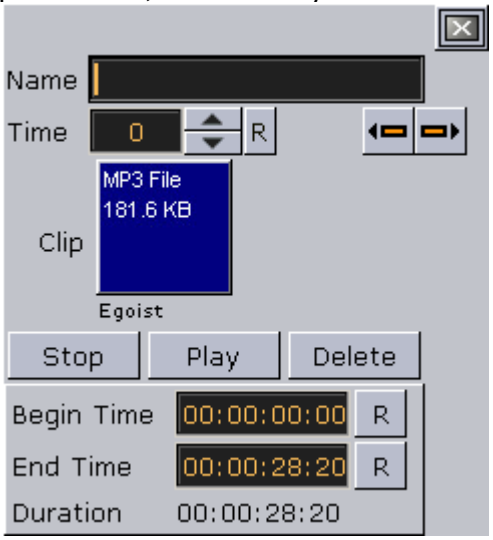
An audio animation plays an audio clip in an animation.

To Enable Audio in an Animation

1. Apply the audio container function to a container in the Scene.
2. Add an actor for this container to the Stage Tree, if it is not present yet.

> ⚠ **Tip:** The easy way to do is to drag the container from the scene tree onto the Stage Tree.

3. Open the shortcut menu and select **Audio** followed and then **Clip**.
4. Select a Key Frame in this channel to open the object editor.
5. Switch to the **Server/Stage** view and drag an audio clip from the Server Panel onto the **Clip** drop zone in the object editor.
6. In the object editor for the Key Frames of the AudioClip channel, configure the following parameters, as necessary.



- **Name:** Sets a name for the Key Frame. This name is for use only for external commands or scripts.
- **Time**: Shows the time value in fields the Key Frame is placed. Editing this value will change to position of the Key Frame.
- **Clip**: Drag an audio clip from the Server Panel onto this drop zone.
- **Stop:** Stops playback.
- **Play**: Plays the clip.
- **Delete:** removes the clip.
- **Begin Time**: Sets the begin time within the clip. Everything before this time will not be played in the animation. Another way to achieve this is to drag the Key Frame at the beginning of the clip.

- **End Time**: Sets the end time within the clip. Everything after this time will not be played in the animation. Another way to achieve this is to drag the Key Frame at the end of the clip.
- **Duration**: Shows the duration to be played.

11.9.5 Animate Cameras

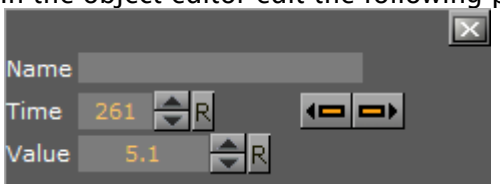
You can animate cameras by adding an actor for a camera to the Stage Tree is by setting Key Frames in the Scene Editor.

To Animate a Camera

1. Set the Time-line Marker to when the Key Frame should be added to start with the animation of the camera.
2. Click the **Add Key Frame** button in the Stage Editor Bar.
3. Change one of the selected camera properties and set the next Key Frame. This will add an actor in the Stage Tree for the camera whose properties you have changed.
   Another way to create camera animations is to enable position and/or direction tracking in the camera editor to animate the camera through a container transformation animation.

To Animate the Current Camera

1. Set a Key Frame in the current Camera channel.
2. In the object editor edit the following parameters for the Key Frame as necessary.



- **Name** Set a name for the Key Frame. This name is for use only for external commands or scripts.
- **Time** Shows the time value in fields the Key Frame is placed. Editing this value will change to position of the Key Frame. If you click arrow buttons in the Stage Editor Bar the previous or next Key Frame will be selected.
- **Value** Sets the camera you want to switch to.
  The animation of the current Camera allows you to switch the camera through which to view at a given time.

To Create a Zoom Animation

If you click Spline button Stage Editor Bar or press <Shift+Q>, the whole spline which is currently shown will be zoomed to.

If you have selected an actor or a Director, the zoom will be adjusted to fit this entry.

When zooming the whole spline, both the time-axis for animation editor and the spline editor will be zoomed, as well as the value-axis in the spline editor.

Another way to zoom the time-axis is by pressing **X** to zoom in and **Y** or **Z** to zoom out. As an alternative you can use the mouse wheel.

If you press <Shift+Y> you will set the zoom along the time-axis to zoom the whole spline while <Shift+X> will set the zoom along the time-axis to show the whole animation.

In the spline editor you can zoom the value axis too. This is done by pressing *A* to zoom in or *S* to zoom out. If you press *<Shift+A>* you will zoom the whole spline along the value axis.

You can zoom also if you press one of the "zoom keys" and move the cursor around with the middle button pressed.

If you have changed the zoom level, you can switch back to standard zoom at any time by a click designated button in the Stage Editor Bar or if you press <Shift+W>.

As you maybe zoomed the view in such away you are no longer able to edit the spline comfortably, you can pan by moving around the cursor while the middle button is pressed.

Zooming along the time axis will zoom both the animation editor and the spline editor.

11.9.6 Animate Light

The only way to add an actor for a light to the Stage Tree is by setting Key Frames in the Scene Editor. To do so set the time-line value to the time you want to start with the animation of the light and set the first Key Frame. Now change one of the selected lights properties and set the next Key Frame. This will add an actor for the light which property you have changed to the Stage Tree.

An easier way the animate a light is to enable position and/or direction tracking in the light editor. To learn more about the light editor, see Light Editor .

11.9.7 Merging Animations

In Viz Artist, it is possible to merge container groups into one single compound object. If an animation has been defined for at least one of the containers, on merging a new Director will be created, given the name of the merged object in the scene tree. There will be a Sub-Director by the name of the original Director the animations were created in. The animations created for the single containers will be kept, but cannot be changed anymore unless you split the object into the containers it has been created from.

The animations will be kept also if you save the newly created object in the database.

Please be aware that you cannot add Sub-Directors to the Directors created automatically by merging.

11.9.8 Multi Select Animation

If two or more objects are selected multiple animations can be created in one step. This adds independent animations channels for each container within the selection.

1. Select the objects to be animated.
2. Create the animation, as required (see Create a Basic Animation).
   An Actor is created for each selected animated object. Once the animation has been created each Actor can be edited separately.

11.10 Create An Over The Shoulder Scene

An Over the Shoulder (OTS) graphic is a graphic which is placed over the shoulder of, for example, the news anchor, and can show on the left or right side of the screen. An OTS graphic can be used to highlight a story with a large image and a headline, and sometimes a sub-headline.

This section contains information on the following procedures, which when completed in sequence create an OTS Scene:

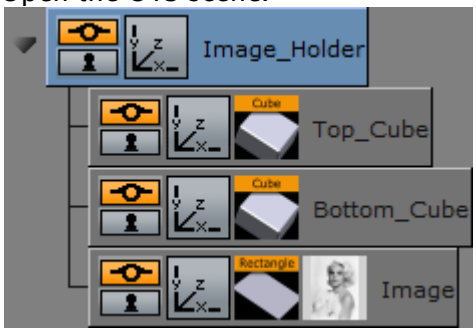
- To Create a Scene
- To Add an Image and Basic Geometries
- To Add Materials
- To Scale Geometries and Place them in Z Space
- To Add Text
- To Add Gradients
- To Add Background items
- To Add Flares
- To Animate
- To Animate Sub Containers
- To Animate Single Containers
- To Animate Single Objects
- To Animate Out
- To Fade out the Graphic with the Alpha plug-in
- To Copy the Directors, Flip them Around, then Reverse the Animation
- To Create a Separate Out Animation
- To Add Stop Points
- To Add Controls

11.10.1 To Create a Scene

- Create a new scene and save it as OTS.

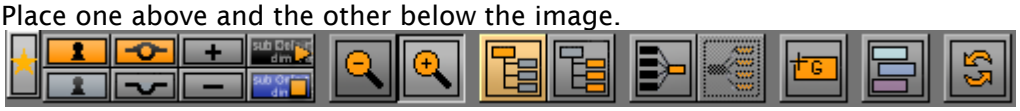
11.10.2 To Add an Image and Basic Geometries

1. Open the OTS scene.



2. Add a Rectangle geometry to the Scene Tree.
3. Add an image (for example, a headshot) to the Rectangle container.

4. Rename the Rectangle container to Image.
5. Add a Cube geometry to the scene. Place it above the Image container.
6. Open the Cube plug-in editor and do the following:
    a. Set **Center Y** to Bottom.
    b. Adjust the **Size X** value to fit the width of the image.
    c. Adjust the **Size Y** value to e.g. 10.0.
    d. Adjust the **Size Z** value to e.g. 1.0.
7. Copy the Cube container. Place it below the original Cube container.
    - Press <Ctrl+C> then <Ctrl+V> to place a copy below the original container.
    - Press <Ctrl+Left mouse button> to drag and drop a copy to a new location.
8. In the transformation editor of the Cube container set **Center Y** to Top.
9. Rename the Cube containers to Top_Cube and Bottom_Cube.
10. Open the transformation editor of each Cube container and move the position of the Cubes. Place one above and the other below the image.

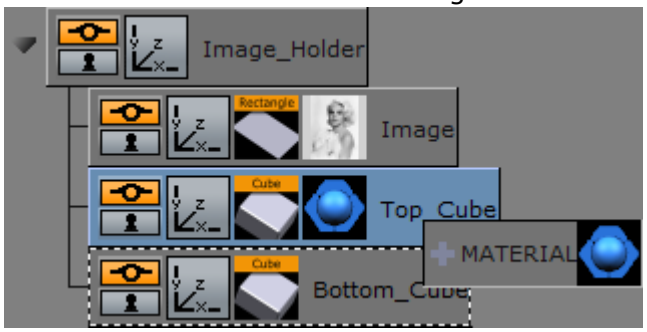


11. Group all containers under a new root container:
    - Select all containers and click the **Group** button, or
    - Select all containers and press <Ctrl+G> (when Caps Lock is disabled).
12. Rename the new group container to Image_Holder.
13. Save the scene.

> ⚠ **Note:** Geometries can also be scaled with the container's transformation editor. But this will have an effect on the its proportions when, for example, beveled edges are added.

11.10.3 To Add Materials

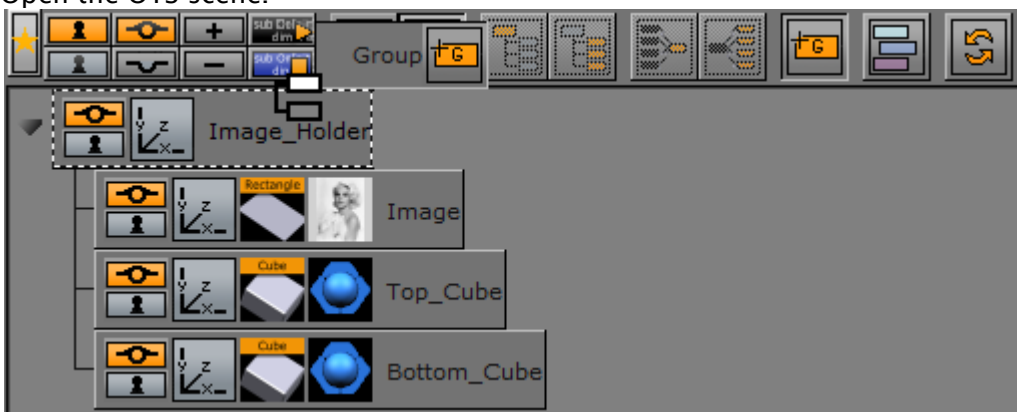
1. Open the OTS scene.
2. Expand the Image_Holder container.
3. Right-click the Top_Cube container and select **Add Material** from the context-menu.
4. Open the Material Editor and define a color (e.g. blue or red).
5. Copy the Material onto the Bottom_Cube container:
    - Drag the Material from one container to the other, or
    - Right-click the Material and from the context menu select Copy 'MATERIAL' and then Paste 'MATERIAL' onto the receiving container.



6. Save the scene.

11.10.4 To Scale Geometries and Place them in Z Space

1. Open the OTS scene.



2. Add a new group container. Place it above the Image_Holder container:
   - Select the Image_Holder container and press <Ctrl+Insert>, or
   - Drag the **New Group** button to the Image_Holder container.
3. Rename the New Group container to Body.
4. Add a Wall geometry to the scene. Place it as a sub-container of the Body container.
5. Rename the new container to Wall.
6. Open the Wall editor (not the transformation editor) and do the following:
   a. Set the **Center** option to Center.
   b. Adjust the **Height** value to 540 (for example).
   c. Adjust the **Width** value to 75 (for example).

7. Group the Body and Image_Holder containers under a new container:
   - Select the Body and Image_Holder containers and click the **New Group** button, or
   - Select the Body and Image_Holder containers and press <Ctrl+G> (when Caps Lock is disabled).
8. Rename the new group container to Main.
9. Copy the Material from one of the Cube containers to the Wall container.
10. Open the Body container transformation editor.
11. Adjust the **Position Z** value to -30.0 (This will avoid potential conflicts with the geometries in the Image_Holder container).
12. Save the scene.

> ⚠ **Note:** If the position and/or scaling of the Wall geometry need to be adjusted, make sure to adjust those values on the Body container, as this will keep future additions to the container aligned.

11.10.5 To Add Text

1. Open the OTS scene.
2. Add a New Group container. Place it below the Image_Holder container.
3. Rename the New Group container to Text_Holder.

4. Add a Font to the scene. Place it as a sub-container of the Text_Holder container.
5. Copy the Font container. Place it below the first Font container.
6. Rename the Font containers to headline and sub-headline.
7. Open the Text Editor of both containers and do the following:
    a. Enter some text (e.g. headline, sub-headline).
    b. Set the **Orientation** option to Center.
8. Open the transformation editor of the headline and sub-headline containers and do the following:
    - Scale and position the containers to fit each other.
9. Add a Rectangle geometry to the scene. Place it above the Font containers and rename it to Text_Rectangle.
10. Open the transformation editor of the Text_Rectangle container and do the following:
    a. Adjust the **Position Y** value. Center the rectangle between the text objects.
    b. Enable single (S) Scaling.
    c. Adjust the **Scaling X** value to equal the width of the image.
    d. Adjust the **Scaling Y** value to 0.1 (for example).
11. Copy the Material from one of the Cube containers to the Text_Rectangle container.
12. Open the transformation editor of the Text_Holder container and do the following:
    a. Adjust the **Position Y** value to equal the Image_Holder objects.
    b. Adjust the **Scaling** (Locked) if needed.
13. Save the scene.

> ⚠ **Note:** If the position and/or scaling of the font containers need to be adjusted, make sure to adjust those values on the Text_Holder container, as that will keep future additions to the container aligned.

11.10.6 To Add Gradients

There are several ways to add gradients or ramps. The example below details a very simple way, which uses a gradient image with alpha.

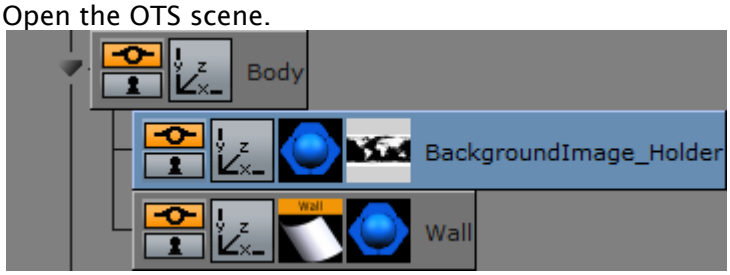
1. Open the OTS scene.
2. Open the Text_Holder container.
3. Add a gradient image to the Text_Rectangle container.
4. Open the transformation editor of the Text_Rectangle container.
5. Adjust the **Scaling** values to make the rectangle equal the intended width of the text.
6. Save the scene.

> ⚠ **Note:** If the PixelFX pxGradient plug-in is used there are more parameters that can be animated. Gradients from other design tools do not need to be imported.

11.10.7 To Add Background items

This procedure details how to add a background image.

1. Open the OTS scene.



2. Add a New Group container. Place it as a sub-container of the Body container.
3. Rename the New Group container to BackgroundImage_Holder.
4. Add an alpha image to the BackgroundImage_Holder container (for example, a map of the world).
5. Copy the Material from the Wall container to the BackgroundImage_Holder container.
6. Open the transformation editor of the Backgroundimage_Holder container and do the following:
   a. Adjust the **Scaling** values as required.
   b. Adjust the **Position Z** value (for example, 2.0) so that it shows in front of the Wall object.
7. Save the scene.

> ⚠ **Note:** If the position and/or scaling of the background items as a group need to be adjusted, make sure to adjust those values on the Body container, as that will keep future additions to the container aligned.

11.10.8 To Add Flares

This procedure details how to make the designs look more dynamic by the addition of items such as flares. A flare is basically an alpha image.

1. Open the OTS scene.

2. Add a New Group container. Place it below the Body container.
3. Rename the New Group container to Flares.
4. Add a New Group container. Place it as a sub-container of the Flares container.
5. Rename the New Group container to Flare_Holder.
6. Add a flare image (with alpha) to the Flare_Holder container.
7. Copy the Material from the Wall container to the Flare_Holder container.
8. Open the transformation editor of the Flare_Holder container.
9. Adjust the **Scale**, **Rotation** and **Position** of the flare so that it makes the intended effect on the object it is applied to.
10. Add the Expert plug-in to the Flare_Holder container.
11. Open the Expert plug-in editor.
12. Set **Render Mode** to **Add**.
    The source color is now added to the target color.

13. Save the scene.

> ⚠ **Note:** If more than one flare is required repeat the process, or copy the Flare_Holder container and make the necessary changes (e.g. position and scaling).

11.10.9 To Animate

Before animations are added, move the OTS graphic to its final on-screen position (normally to the top left or right) where it should be seen by the viewer.

1. Open the OTS scene.
2. Open the transformation editor of the Main container.
3. Adjust the **Scaling**, **Rotation**, and **Position** values.
   Make sure any conflicts are resolved now. Notice that when the Scene is rotated, objects may "drift" apart as they are placed at different depths (Z space).
4. Save the scene.
   This procedure focuses mainly on the animation of a root container, mainly to start with the simplest factor; how do I want my graphics to animate in to view?

1. Open the OTS scene.
2. Set Key Frames to 60/50 (for example) fields.
3. Click the **Set Key Frame** button.
4. Open the transformation editor of the Main container and do the following:
   - Adjust the **Position X** value to 0.0.
5. Set Key Frames to 0 fields.
6. Click the **Set Key Frame** button.

> ⚠ **Note:** Notice that an Animation object has been added to the Main container.

7. Click the **Start animation** button on the Time-line Editor.
   The graphic now moves from Position X at 0 fields to its final on-screen position at 60 fields.
8. Open the Stage Editor.
   The Main container and the animated Position property are located under the Default Director.
9. Right-click the Main container and from the context menu select *Transformation -> Scaling*.
   This will add a time-line for the **Scaling** property of the Main container.
10. Click the first Key Frame on the Scaling animation's time-line on the Dopesheet Editor.
    This will open the Scaling animation Stage Object Editor.
11. Set the **Scaling** values to 0.0.
12. Click the **Start animation** button on the Time-line Editor.
    The graphic will now animate from Position X at 0 fields with Scaling at 0.0 to its final size on-screen.
13. Save the scene.

> ⚠ **Note:** If the Viz Configuration is set to 50 hertz the default animation time-line will be 50 fields. If set to 60 hertz, it will be 60 fields. It is always important to configure the required output format before work is started on the designs.

11.10.10 To Animate Sub Containers

This procedure focuses on the animation of a sub-containers. In this case the image and text holder containers.

1. Open the OTS scene.
2. Open the Stage Editor.
3. Drag the Image_Holder container onto the Default Director.
4. Right-click the Image_Holder container and select *Transformation -> Scaling* from the context menu.
   This will add a time-line for the Image_Holder container's Scaling property.
5. Click the Key Frame for the Scaling animation. In the Key Frame Editors set it to 30 (for example) fields, to delay the introduction of the image.
6. Copy the animation object from the Image_Holder container to the Text_Holder container.
7. Save the scene.

> ⚠ **Note:** The Image_Holder and Text_Holder containers are set to grow from the Center. To grow the containers from other angles, open the transformation editor, for each container, and adjust the **Axis Center** properties. If the OTS is placed to the right, try setting the Text_Holder container's Axis Center X to R (Right), and conversely.

> ⚠ **Note:** Always scrub the time-line in the time-line editor to check for potential conflicts (caused by rotation of objects).

11.10.11 To Animate Single Containers

This procedure focuses on the animation of a single container, and how to add additional Key Frames to the animation. In this case the background image is animated and an additional Key Frame is added to grow the size of the background image before it shrinks back to its final size.

1. Open the OTS scene.
2. Select the BackgroundImage_Holder container.
3. Set the number of fields to 30 in the Time-line Editor.
4. Open the Transformation Editor of the BackgroundImage_Holder container.
5. Set **Scaling** to 0.0.
6. Click the **Set Key Frame** button.
7. Set **Scaling** to the original value.
8. Set the number of fields to 60 in the Time-line Editor.
9. Click the **Set Key Frame** button.
10. Open the Stage Editor.
11. Select the **Scaling animation** time-line for the BackgroundImage_Holder.
12. Move the Time-line Marker to a position closer to the end of the animation.
13. Click the **Add a Key Frame** button.
14. Click the new Key Frame and set the **Scaling** value to be larger than the final size.
15. Save the scene.

> ⚠ **Note:** Always scrub the time-line in the time-line editor to check for potential conflicts (caused by rotation of objects).

11.10.12 To Animate Single Objects

This procedure focuses on the animation of single objects. Animating single objects increases the dynamics of the scene, especially when the scene has reached its final position on-screen and all other animations have stopped.

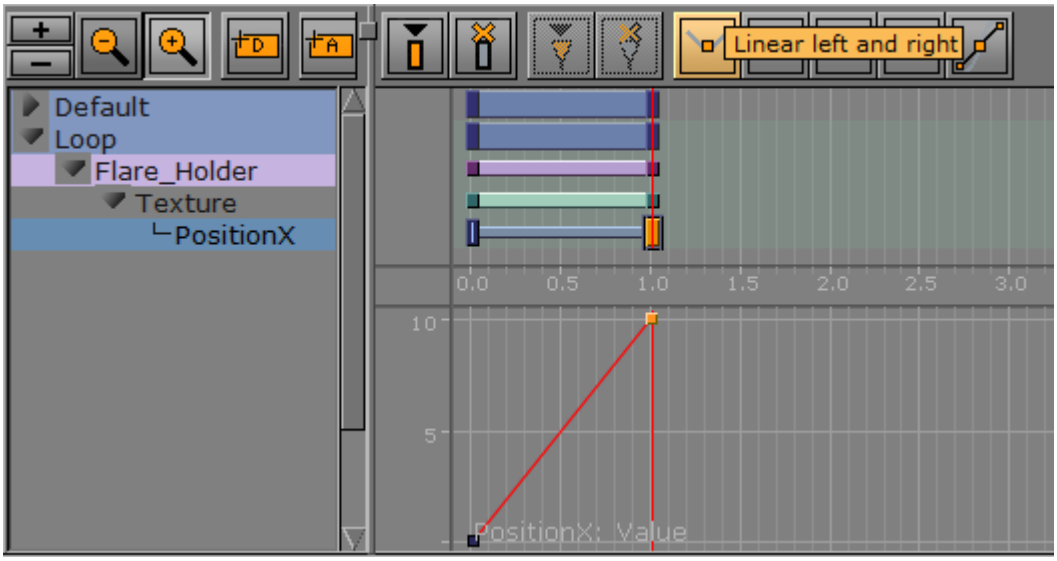
This procedure describes how to make a flare move, continuously, along another object.

1. Open the OTS scene.
2. Open the Stage Editor.

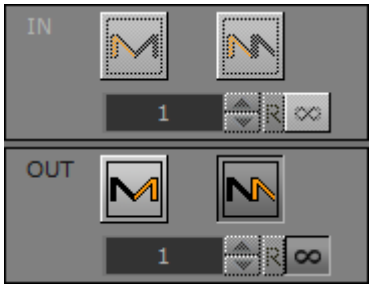
3. Create a new Director and name it Loop:
   - Click the **+D** button, or
   - Right-click the Stage Tree and select **Add Director**.
4. Select the Loop Director.
5. Set the Time-line Editor to 0 fields.

6. Open the Texture Editor of the Flare image:
   - Set **Position X** to 0 and click the **Set Key Frame** button.
   - Set **Position X** to 10 and click the **Set Key Frame** button.

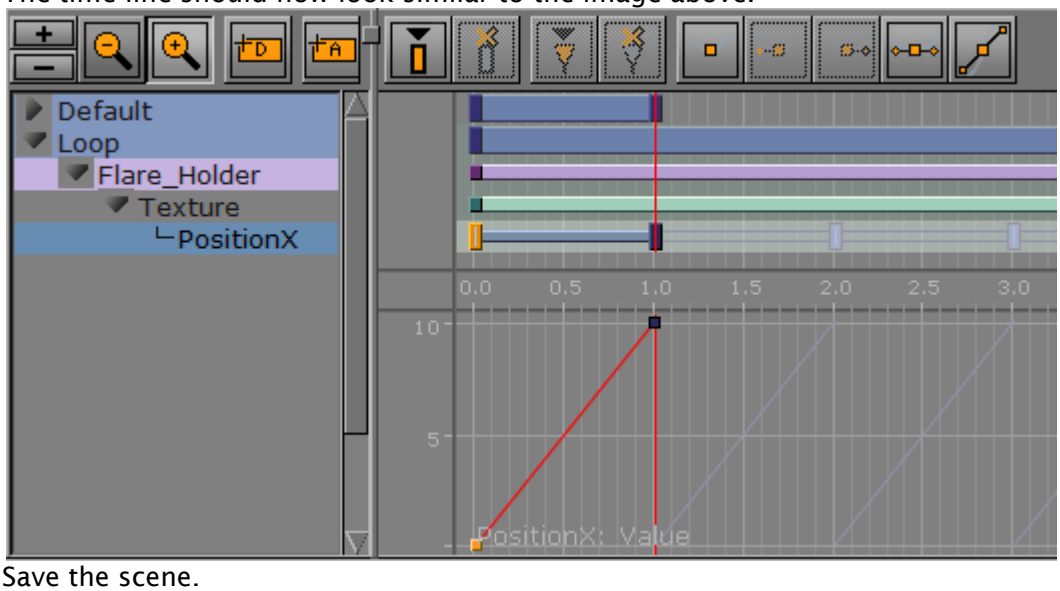
> ⚠ **Note:** When **Position X** is set to 10 it makes sure that there are no gaps between the repeating textures.



7. Select both of the **Position X** Key Frames and click the **Linear left and right** button.



8. Click the **Position X** channel to open its Channel Editor. Click the **Loop out** and **Infinite** buttons.
   The time-line should now look similar to the image above.



9. Save the scene.

> ⚠ **Note:** Now that there is an animated Flare it is always good to revisit the **Position**, **Rotation**, and **Scaling** properties of the Flare_Holder container.

11.10.13 To Animate Out

Animations to the scene so far has brought the graphic on-screen for the viewer to see. However, the scene should also be able to animate out instead of simply removing it (hard cut).

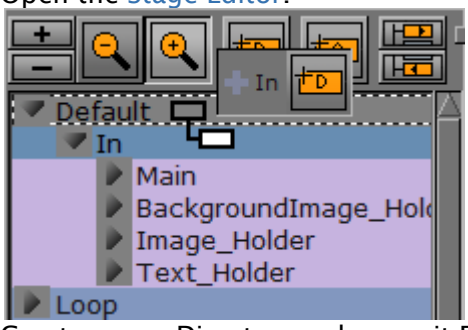
This can be achieved in three ways:

> ⚠ **Note:** With one or a combination of the following three methods an out animation can be created. The scene will now work in terms of animation, however, it will not work well if someone needs to control, for example, when or how the out animation should start. For this we need to add controls (using control plug-ins) that will allow an operator to control the scene on air.

11.10.14 To Fade out the Graphic with the Alpha plug-in

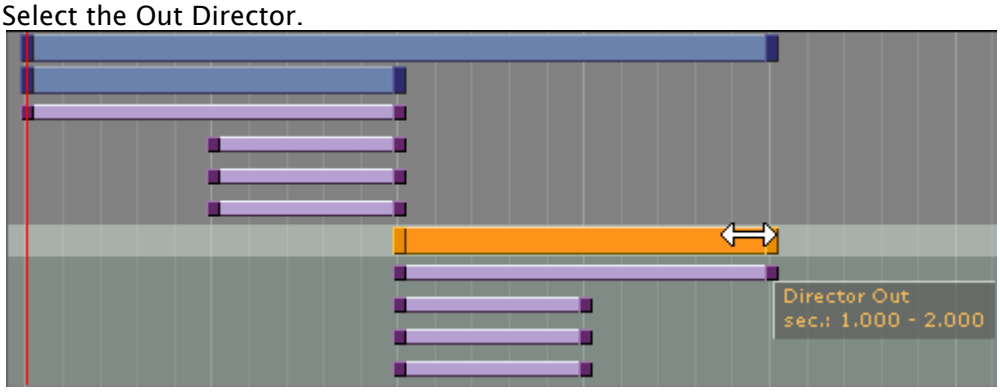
1. Open the OTS scene.
2. Add the Alpha plug-in to the Main container.
3. Set the time-line to the end of the in animation (at 60 fields for example).
4. Open the Alpha plug-in editor.
5. Set the **Alpha** value to 100%.
6. Click the **Set Key Frame** button.
7. Set the **Alpha** value to 0%
8. Click the **Set Key Frame** button.
9. Save the scene.
   When played the animation should now animate in and fade out to Alpha at 0%.

11.10.15 To Copy the Directors, Flip them Around, then Reverse the Animation

1. Open the OTS scene.
2. Open the Stage Editor.
   

3. Create a new Director, and name it Default.
4. Place the old Default Director under the new Default Director
5. Rename it to In.
6. Copy the In Director. Place it as a Sub-Director of the Default Director.
7. Rename the copied Director to Out.

8. Select the Out Director.



9. Hold <Ctrl>, select the Out Director's first Key Frame, and drag the Key Frame past the last Key Frame to reverse the animation.
10. Save the scene.
    When played the animation should now animate in and out, where the out animation is simply a reverse of the in animation.

11.10.16 To Create a Separate Out Animation

Manually create an 'out' animation from scratch, by the extension of the existing animations to animate out of view.

Also, repeat the previous procedure and adjust the different animations as required.

11.10.17 To Add Stop Points

When the scene design is finished Controls can be added that will allow the operator to control the animation while on air. To do this add one or multiple stop points to the Default Director (always the Default Director).

1. Open the OTS scene.
2. Open the Stage Editor.
3. Select the Default Director.
4. Place the Time-line Marker where the Stop point is to be added (for example, where the In animation stops, and the Out animation starts).
5. Select the Default Director again.
6. Click the **Add a Stop/Tag** button.
   This will add a Stop/Tag to the Default Director.
7. Name the first Stop point.
8. Save the scene.

> ⚠ **Note:** Stop point names can be user defined. However, when the scene is preview with a remote (external) Viz Engine, either from a newsroom system, or when connected through a control application to a Media Sequencer, the pilot1 stop/tag is the default preview point on the Default Director.

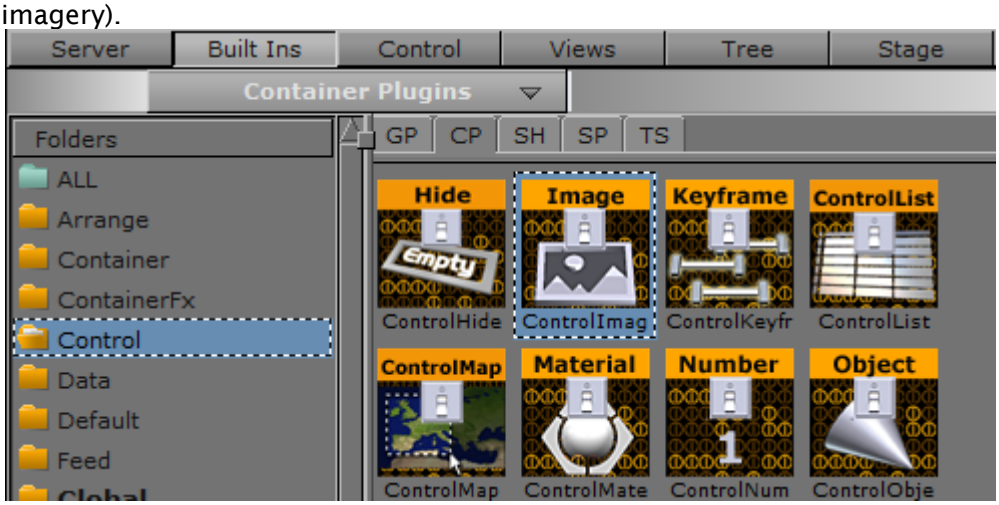
> **Note:** The placed pilot1 stop/tag point on the Default Director's time-line, at a given field, enables Vizrt control applications to create a thumbnail and show it in, for instance, the playlist.

Now the scene is ready for the operator to control the animation. However, the operator still lacks one key item; they need to control the content of the graphics (e.g. text and imagery).

11.10.18 To Add Controls

Stop/tag points are added to the Director to allow the operator to control the animation. However the operator, when required, should be able to control the content of the Scene. For this purpose Control plug-ins are used.

1. Open the OTS scene.
2. Expand the Scene Tree, or those containers that require any form of control (e.g. text and imagery).



3. Click the Built Ins button on the Main Menu Left.
4. Click the Container plug-ins (CP) tab and select the Control section.
5. Add the Control Text plug-in to the headline and sub-headline containers.
6. Add the Control Image plug-in to the Image container.
7. Open the headline container Control Text plug-in editor.
8. Set **Field Identifier** to 01.
9. Repeat the process for the sub-headline and Image containers and set **Field Identifier** to 02 and 03, respectively. Update the **Description** fields as required, by default it will use the container name.
10. Save the scene.

> **Tip:** Number the Field Identifiers to help order them in the control application interface.

> **Note:** To check that an operator can use the scene test it with a control application (for example, Viz Pilot with the Template Wizard, Viz Trio, Viz Multichannel, or Viz Weather).

> ⚠ **Note:** All plug-ins are installed locally. plug-ins used in the design must be available on the production system for the scene to work.

11.11 Create A Stand-Alone Scene

This section details how to create a basic lower third as a stand-alone Scene. This scene can then be imported to a Viz Trio to create a template for pages which are used for playback.

> ⚠ **IMPORTANT!** When creating these transition effects also refer to the User Manuals for the required Control Applications: Viz Trio, Viz Pilot, Viz Weather, etc.

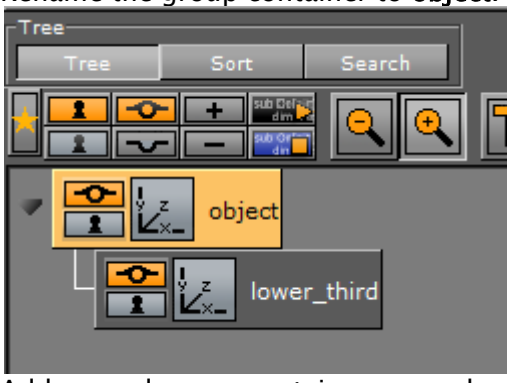
> ⚠ **Note:** For ease of use, the Scene example has been split up in several procedures. For the example to work, it is recommended to follow each procedure in the order given.

This section contains information on the following topics:

- To Create a Scene
- To Add a Background
- To Add Text
- To Create an In and Out Animation
- To Add Stop Tags
- To Add Key Functions to the Container
- To Add Expose Properties
- To Edit Multiple items with a Single Value

11.11.1 To Create a Scene

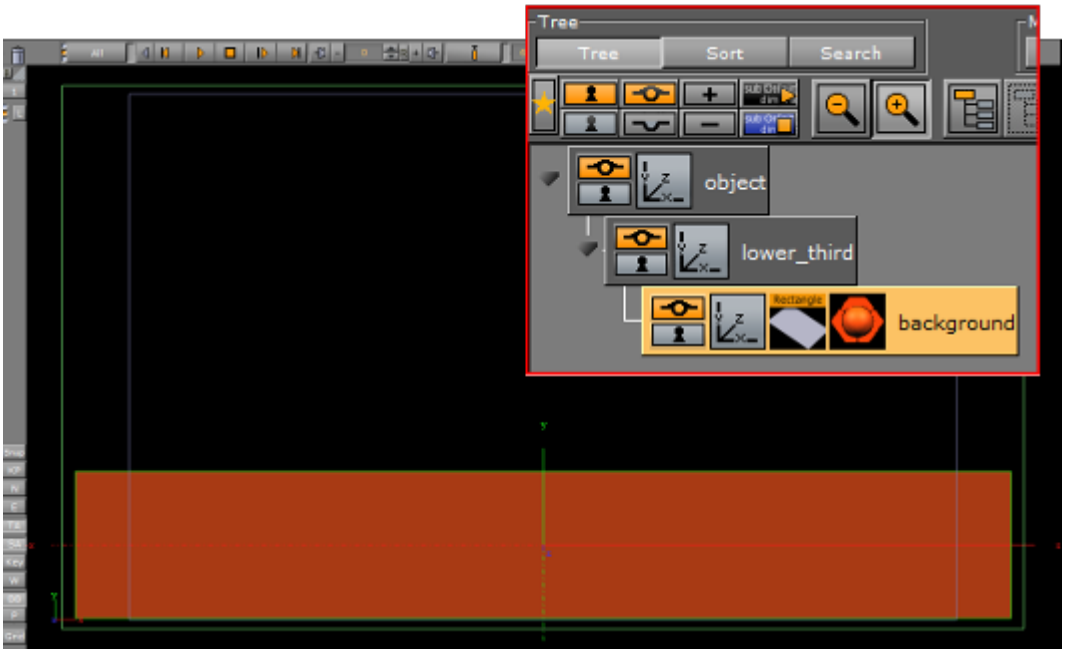
1. Add a group container to the Scene Tree.
2. Rename the group container to **object**.



3. Add second group container as a sub-container to the **object** container.
4. Name the second group container **lower_third**.
5. Save the new scene as **stand_alone_scene** (in this example).

> ⚠ **Tip:** For ease of use set the Scene Editor Title (TA) and Safe (SA) areas to ON.

11.11.2 To Add a Background



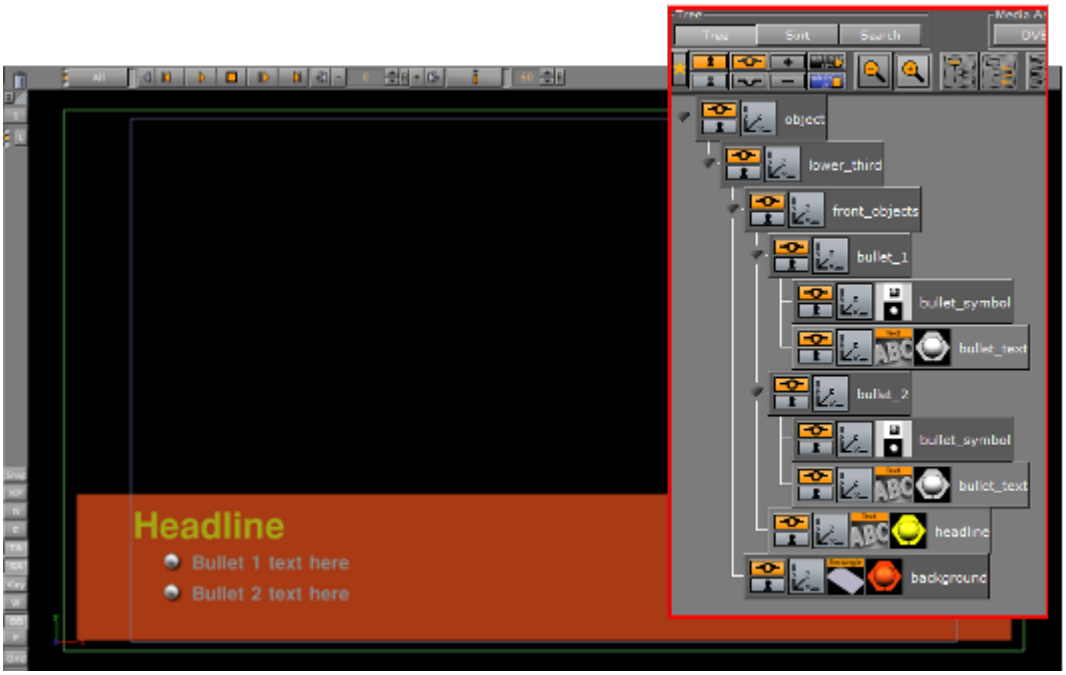
1. Add a Rectangle geometry as a sub-container of the **lower_third** container,
2. Rename it to **background**.
3. Position the rectangle object so that it covers the lower third of the screen:
   - Rectangle editor: **Width** `800` and **Height** `125`.
   - Transformation editor: **Position Y** `-160` and **Position Z** `-100`.

   > ⚠ **Note:** Position Z: Since this item is a background item it should be a little bit behind the other objects on the Z-axis.

4. Add a material object to give the Rectangle a color.

11.11.3 To Add Text

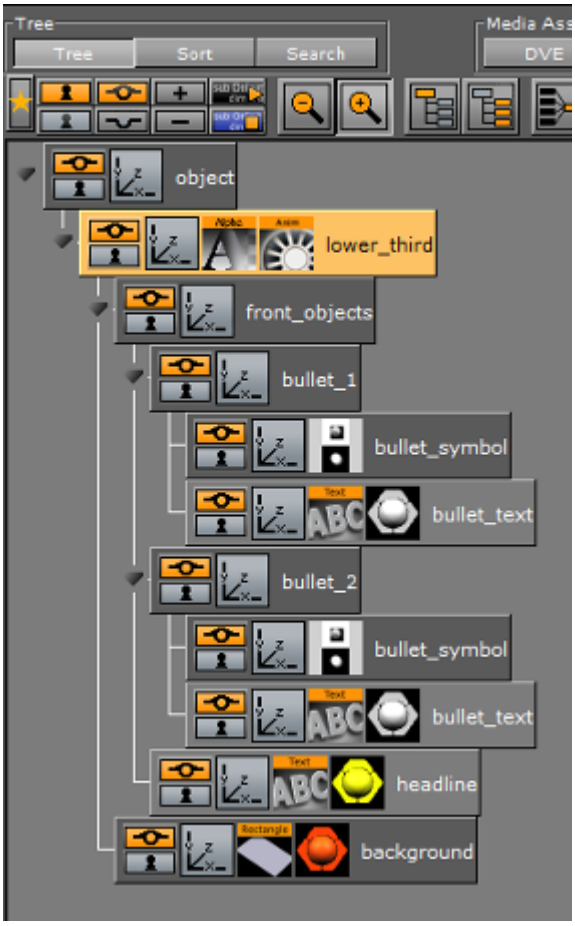
Add a headline item and two text lines, lead by a bullet symbol.



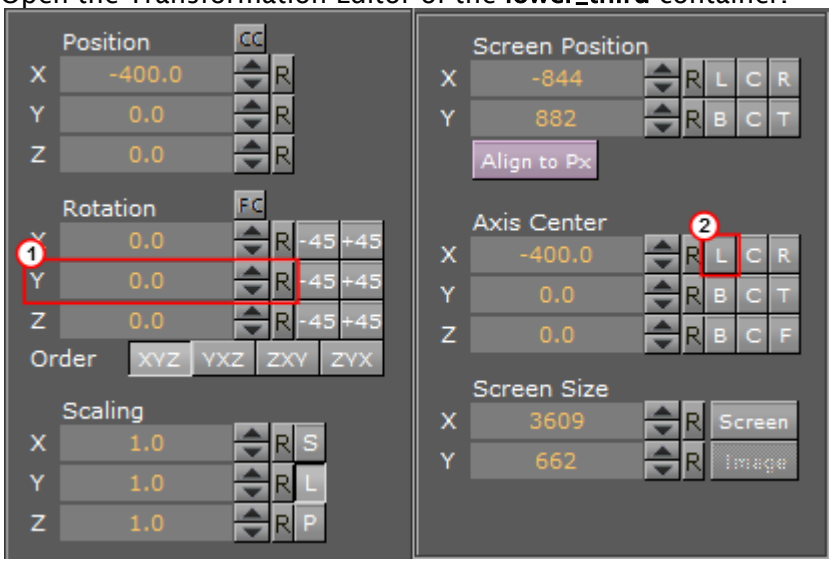
1. Add a container as a sub-container to the **lower_third** container.
2. Name the container **front_objects**.
3. Add a Fontstyle as a sub-container of the **front_objects** container.
4. Name the Fontstyle container **headline**.
5. Open the Fontstyle text editor and:
   - Enter the word **Headline** in the **text input** field
   - Set the Horizontal orientation to **Left**.
6. Add a material to the **headline** container.
7. Open the transformation editor of the **headline** container.
8. **Scale** and **Position** the Fontstyle so that it is placed in the upper left corner of the lower third background.
9. Add a group container as a sub-container of the **front_objects** group.
10. Name the container **bullet_1**.
11. As sub-containers of **bullet_1**, add:
    - A Fontstyle: Name container **bullet_text**
    - Any Image or Object that can function as a bullet symbol: Name the container **bullet_symbol**
12. In **bullet_text** text editor:
    - Enter the text **Bullet 1 text here**
    - Set the horizontal orientation to **Left**.
13. Add a material to the **bullet_text** container.
14. **Scale** and **Position** the **bullet_text** and **bullet_symbol** containers.
15. Position the **bullet_1** container under the **Headline** text.
16. Make a copy of the **bullet_1** container as a sub-container of the **front_objects** group.
17. Rename the copy to **bullet_2**.
18. Adjust the **Position Y** of **bullet_2**.
19. In **bullet_text** text editor enter the text **Bullet 2 text here**

11.11.4 To Create an In and Out Animation

To make the whole lower third rotate in from the left side of the screen and then fade out, the **lower_third** container must be animated.

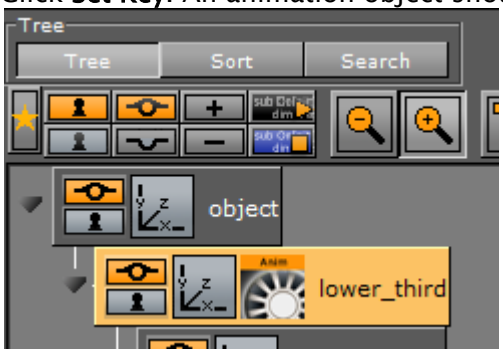


1. Open the Transformation Editor of the **lower_third** container.



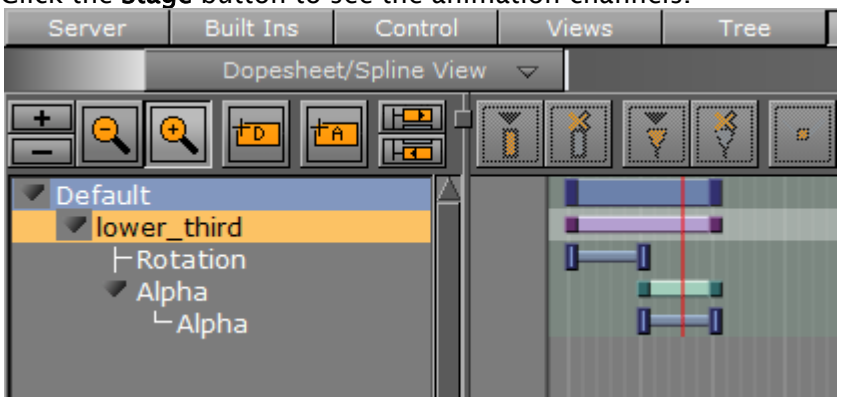
2. Click the **Axis Center X** set **left** button  (2).
3. Change the **Rotation Y** value so the container moves out to the left side and is hidden (1).
4. Click **Set Key** (3).

5. Click the **Rotation Y** reset button  (1) to set the **Rotation** value to zero.
6. Click **Set Key**. An animation object should now be visible on the **object** container.



7. Play the animation in the Scene Editor.
8. Add the Alpha plug-in to the **lower_third** container.
   - *Built Ins -> Container plug-ins -> Global*

9. Open the Alpha editor and:
   a. Set the Alpha value to 100%.
   b. Press the **Set Key** button.
   c. Set the alpha value to 0.0%.
   d. Press the **Set Key** button.
   This has created an alpha animation in addition to the rotation animation.
10. Click the **Stage** button to see the animation channels.



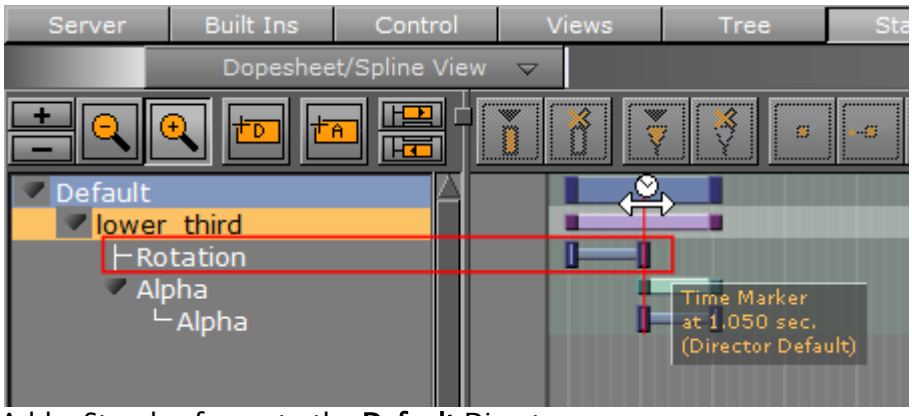
> ⚠ **Note:** It is possible to move the Key Frames to achieve a different timing. Either drag with the mouse or click on them and alter the time settings in the Key Frame Editors.

11.11.5 To Add Stop Tags

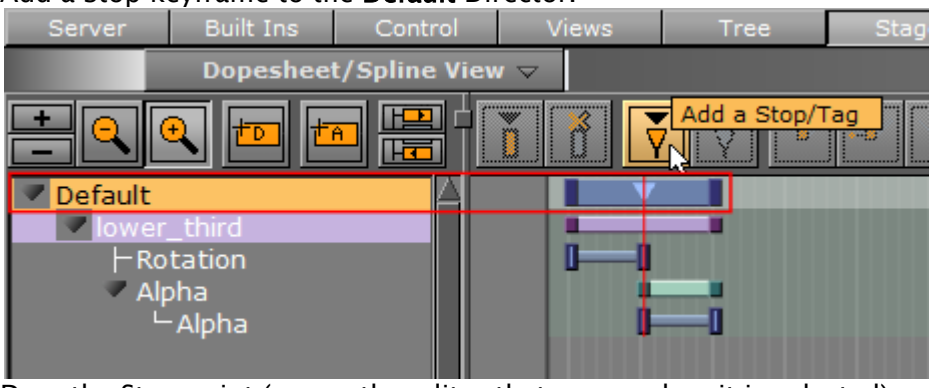
When the Scene is played out, the objects should rotate in from the left and then fade out. To stop the animation, after the rotate in, a stop point must be added before the animation can continue and fade out.

1. Move the time-line (the thin red vertical line) to the second key frame of the Rotation channel

   > **Note:** An exact spot is not needed as it can be adjusted afterwards.



2. Add a Stop keyframe to the **Default** Director.

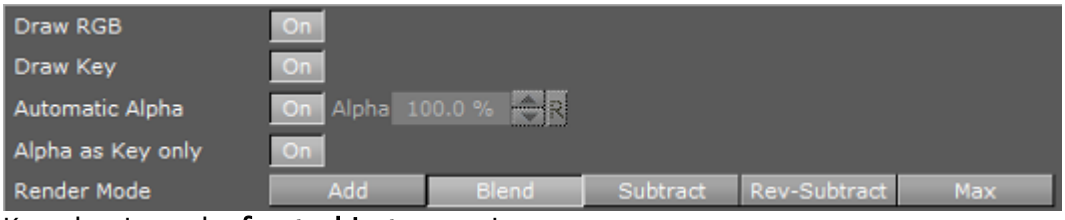


3. Drag the Stop point (or use the editor that opens when it is selected) so that it has the same time settings as the second keyframe on the Rotation channel.
4. Play the animation to check that the animation stops when the rotate in has finished.
5. Click the **Continue** button to continue with the Alpha fade.

11.11.6 To Add Key Functions to the Container

1. Add the Key plug-in to the **background** container.
   - *Built Ins -> Container plug-ins -> Global*

2. Click the Key plug-in icon to open its editor:
   - Set **Alpha as Key only** to **On**. This is to avoid a dirty key if the background has some level of transparency.
   - Set **Render Mode** to **Blend**.



3. Add a Key plug-in to the **front_objects** container.
4. Click the Key plug-in icon to open its editor:
   - Set **Alpha as Key only** to **Off**.
   - Set **Render Mode** to **Add**.
     This is to avoid the foreground objects to cut a hole in the Key signal of the background object.

11.11.7 To Add Expose Properties

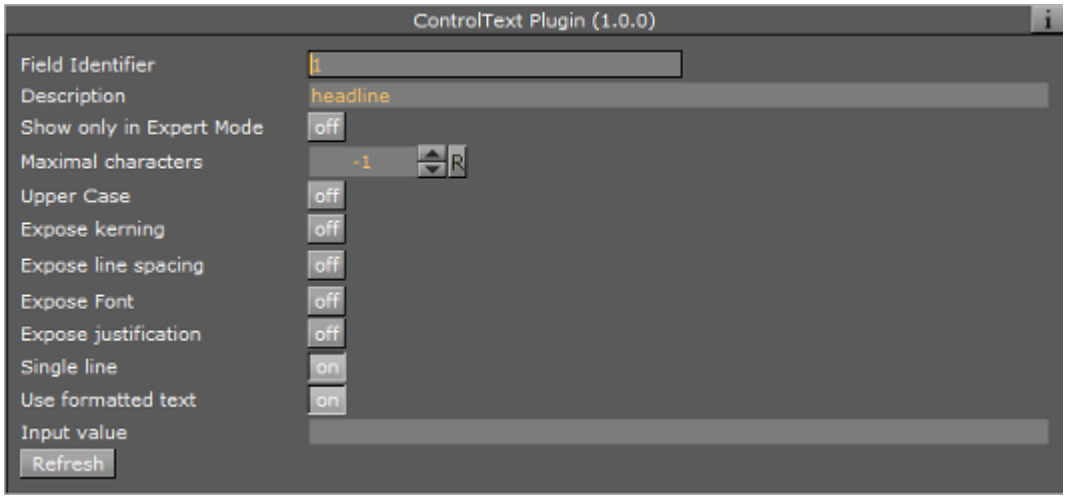
To make a Scene ready for import into Viz Trio, Control plug-ins must be added. The Control plug-ins enable the designer to expose scene properties to a Viz Trio user.

1. Add the **Control Text** plug-in to the **headline** and both **bullet_text** containers.
   - *Built Ins -> Container plug-ins -> Control*

> ⚠ **Note:** The **Control Object** plug-in is automatically added to the top most container in the Scene Tree, which should always be named **object.**

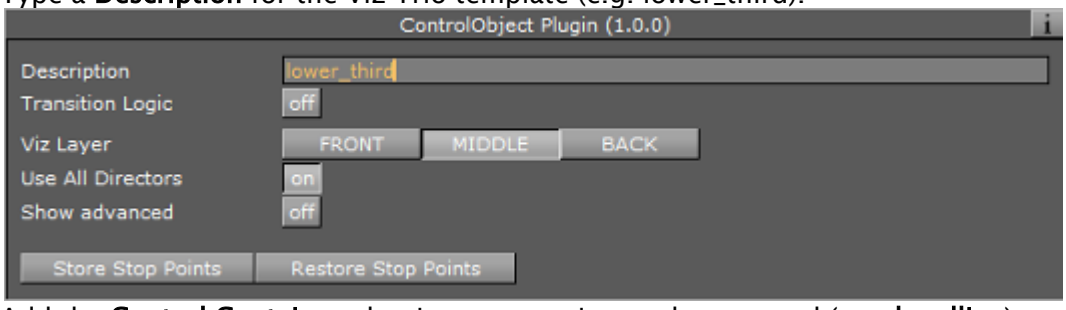
2. Click on a **Control Text** plug-in icon to open its editor:
   a. Set the **Field identifier**. The Field identifier must be a numeric value and the value will be used to give the Viz Trio page a tab order.
   b. Add a **Description**.
   c. Repeat for the other two **Control Text** plug-ins.

> ⚠ **Note:** In this scene the **headline** container will typically get identifier 1, and the two **bullet_text** containers will get identifier 2 and 3.



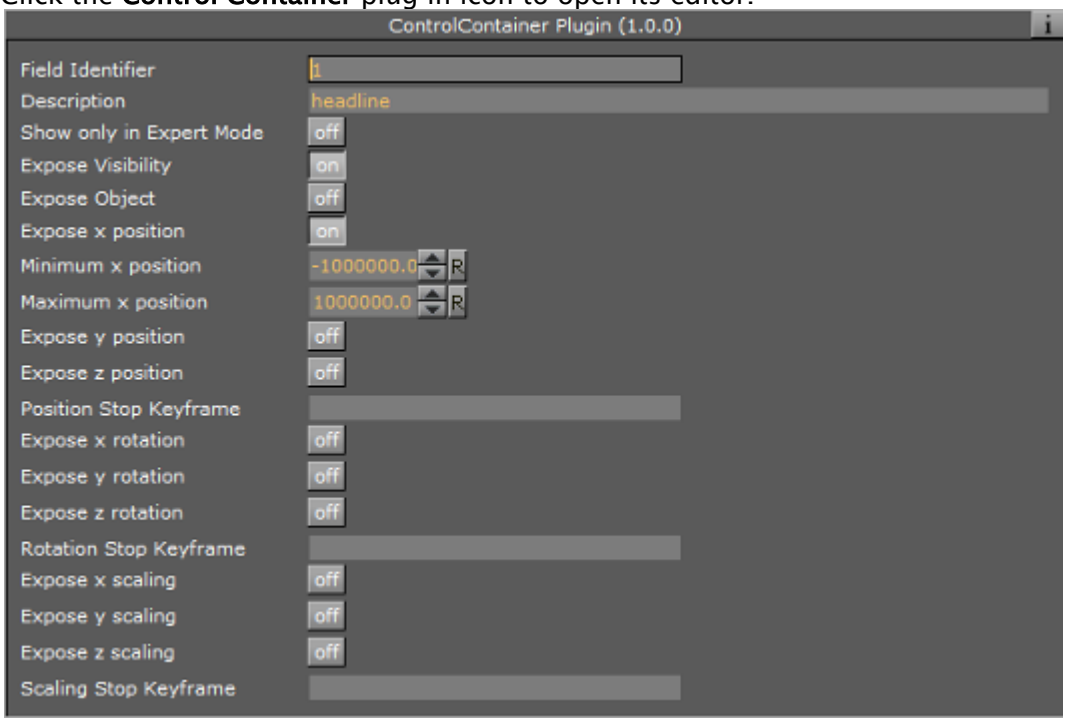
> **Note:** More properties can be exposed to the Viz Trio operator. If the **Position**, **Rotation**, and **Scale** properties are exposed the operator can hide or show an object and add material.

3. Click on the **Control Object** icon to open its editor.
4. Type a **Description** for the Viz Trio template (e.g. lower_third).



5. Add the **Control Container** plug-in to a container to be exposed (e.g. **headline**).

6. Click the **Control Container** plug-in icon to open its editor.



7. Type a **Field Identifier**.

> ⚠ **Note:** If there is another control plug-in on that container, like a **Control Text** or **Control Image**, use the same Field identifier.

- X/Y/Z properties for a keyframe in an animation must be specified by a Stop Key Frame name.

11.11.8 To Edit Multiple items with a Single Value

- Use the same **Field Identifier** for multiple control plug-ins to make two or more fields as a single item that will receive the same value.
  This can, for example, be used with a bar that scales and a text label that shows its value. The scale value and the text can then be set in a single operation, which makes sure that there is no mismatch between the bar size and the written text value.

11.12 Create Transition Effects

This section details how to design custom transition effects for use with Viz Trio. The scene transition mechanism uses the **GFX Channel** Media Asset.

> ⚠ **IMPORTANT!** When creating these transition effects also refer to the User Manuals for the required Control Applications: Viz Trio, Viz Pilot, Viz Weather, etc.

⚠ **Tip:** The Dynamic textures may also be available as part of an already existing transition effect scenes in the **dynamic** folder (at root level).

The Dynamic textures must be named **layer1** and **layer2**. These are not ordinary images as they have the possibility to load into them an actual scene.

- **layer1**: Represents the first scene.
- **layer2**: Represents the scene to transit to.

It is possible to make any kind of animation, but there are a few design conventions to follow:

- The scene should start with a full screen view of the dynamic texture layer1, and end with a full screen display of the dynamic texture layer2. Use the screen size section in the transformation editor to set the size of the texture container to fit the screen exactly
- A transition scene is a scene that is created to move from one scene to the next in a dynamic way (not by cut)

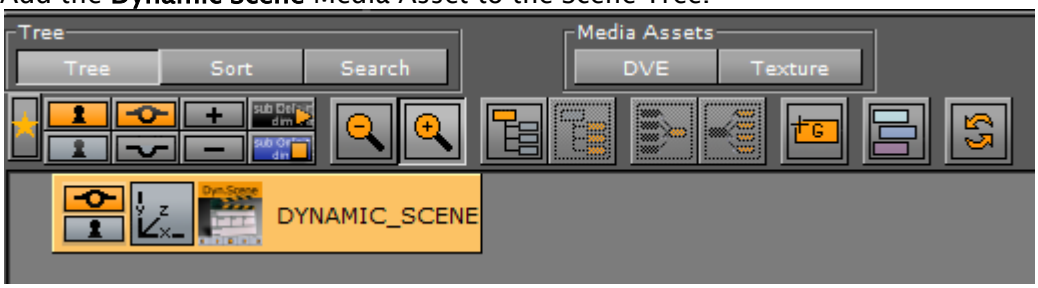
⚠ **Note:** All scenes must be placed in a **transitions** folder in the scene database to make it visible in the control application.

This section contains the following procedures:

- To Create Dynamic Textures
- To Create a Transition Scene

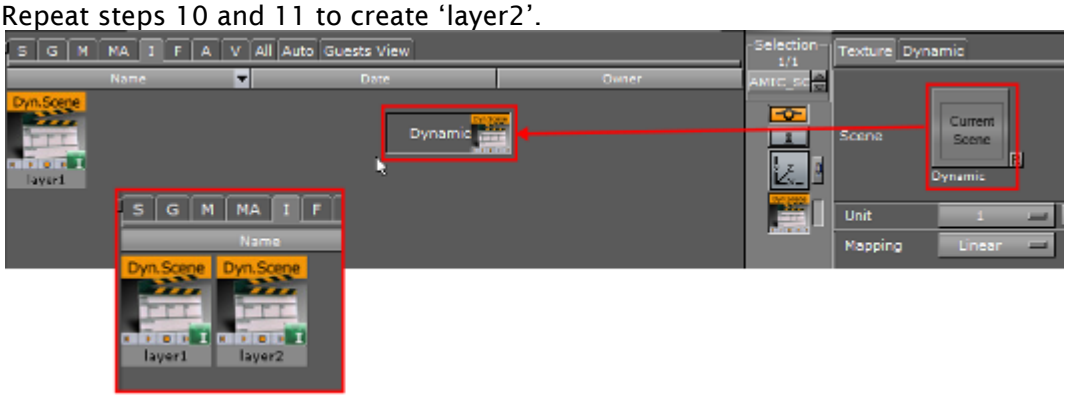
11.12.1 To Create Dynamic Textures

1. Add the **Dynamic Scene** Media Asset to the Scene Tree.



2. Open the Dynamic Scene properties **Dynamic** tab.
3. Configure the **Dynamic** tab properties as required.
4. Open the Transformation editor.
5. Set **Screen Size** to **Screen**.
6. Click **Server**.
7. Create a folder at the root level and name it **dynamic.**
8. Select the **I** (Image) tab.
9. Open the Dynamic Scene properties **Texture** tab.
10. Drag the **Current Scene** placeholder icon into the Server Panel.
11. Save as 'layer1'.

12. Repeat steps 10 and 11 to create ‘layer2’.

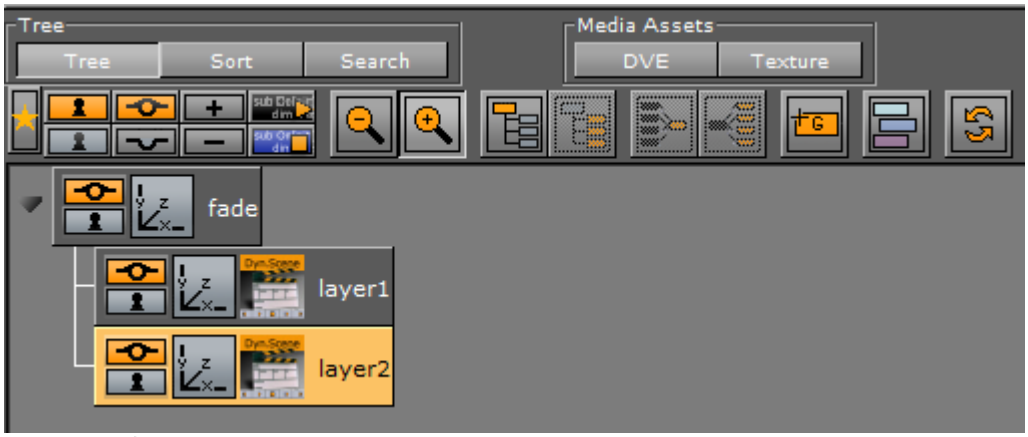


13. Delete the **DYNAMIC_SCENE** container.

11.12.2 To Create a Transition Scene

1. Add a group container to the Scene Tree.
2. Name the container **fade**.
3. Add the dynamic textures ‘layer1’ and ‘layer2’ as sub-containers to the **fade** container.

> ⚠ **Note:** There might not be images in the preview window, but only the bounding box.

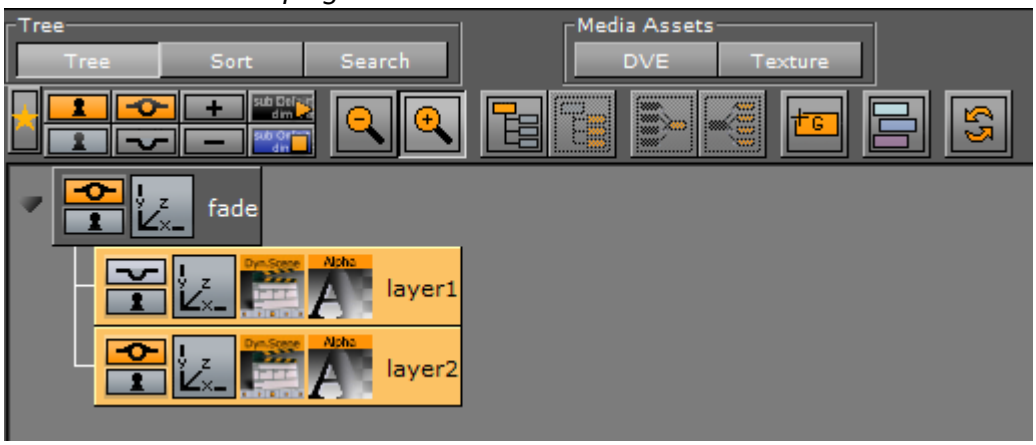


4. Open the **fade** container Transformation editor.
5. Set **Screen Size** to **Screen** (resize the container to fit the whole screen).

> ⚠ **Note:** The scaling of the main group (**fade**) to screen size will make sure that the dynamic textures (**layer1** and **layer2**) will be the correct full frame size.

6. Add the **Alpha** plug-in to the **layer1** and **layer2** containers.

- *Built Ins -> Container plug-ins -> Global*



7. Create a fade **out** animation for the **layer1** container:
   a. Click the **Alpha** plug-in icon.
   b. Set the **Alpha** to 100%.
   c. Click **Set key**.
   d. Set the **Alpha** value to 0%
   e. Click the **Set key**.

8. Create a fade **in** animation for the **layer2** container:
   a. Click the **Alpha** plug-in icon.
   b. Set the **Alpha** to 0%.
   c. Click **Set key**.
   d. Set the **Alpha** value to 100%
   e. Click the **Set key**.

> ⚠ **Tip:** In the **Stage**, place the time-line somewhere in the middle before the scene is saved as it usually gives the best representation of the transition effect.

9. Click **Server.**
10. Create a folder at the root level and name it **transitions.**
11. Save the scene to the transitions folder.

> ⚠ **Tip:** Transition scenes are commonly stored in a **transitions** scene folder at the root of the database directory structure.

12 Import And Archive

This section details how to import files and archives, and how to archive (export) graphical resources such as one or several Scenes including fonts, images and so on to a file (archive) on disk.

Scenes can be exported as video or image files. This process is called Post Rendering.

Also supported is the import of Scenes from third party file formats.

This section contains information on the following topics:

- Import of Files and Archives
- Archive of Graphical Resources
- Deploy items
- Post Render Scenes
- Placeholder names used for file-name expansion

12.1 Import Of Files And Archives

Viz Artist can Import Files (Fonts, Images, Geometries, Scenes, Audio clips) and Import Archives (archives have the filename extension `.via`).

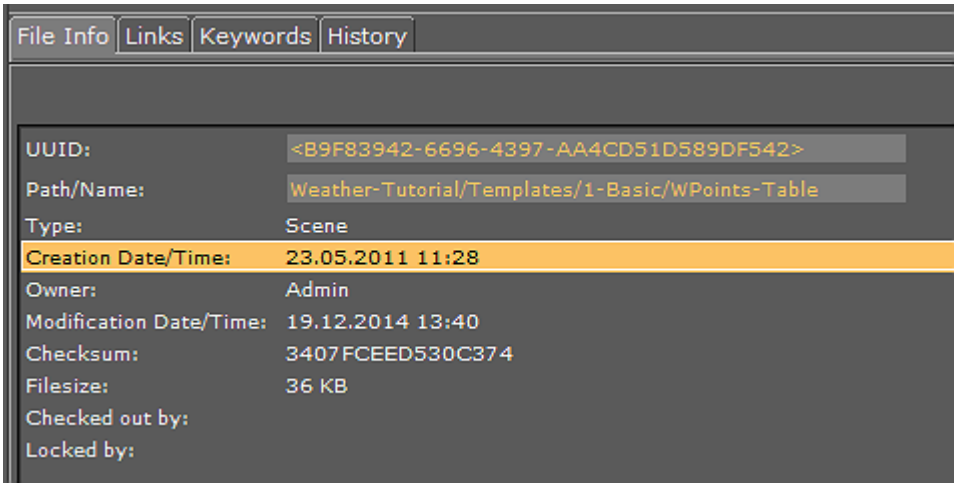
Archives are, if possible, self contained and complete. This means that if a user archives one or more Scenes, then **all** referenced items in the scene, including links, fonts, images and so on, will be archived. A user can choose to archive only individual items such as some fonts or some images.

All items used in Viz Artist are stored in the Graphic Hub (GH). Viz Artist connects to the GH through a network connection, even if the GH is local.

The Graphic Hub is a **database**, and every file in the Graphic Hub is uniquely identified by its **UUID** (a unique identifier number).

Files in the Graphic Hub can have links. This means that possibly several Paths/Names can link (or point) to the same file in the GH database.

Right-click on any item in the Server Panel and select **Show Properties**, to view its UUID, its Path/Name and Link information.



⚠ **Tip:** Vizrt strongly recommends to have a clear and well communicated plan for organizing the paths and naming conventions for all resources in the Graphic Hub. This will make life easier for users and import/export operations will be easier to do without the creation of clutter.

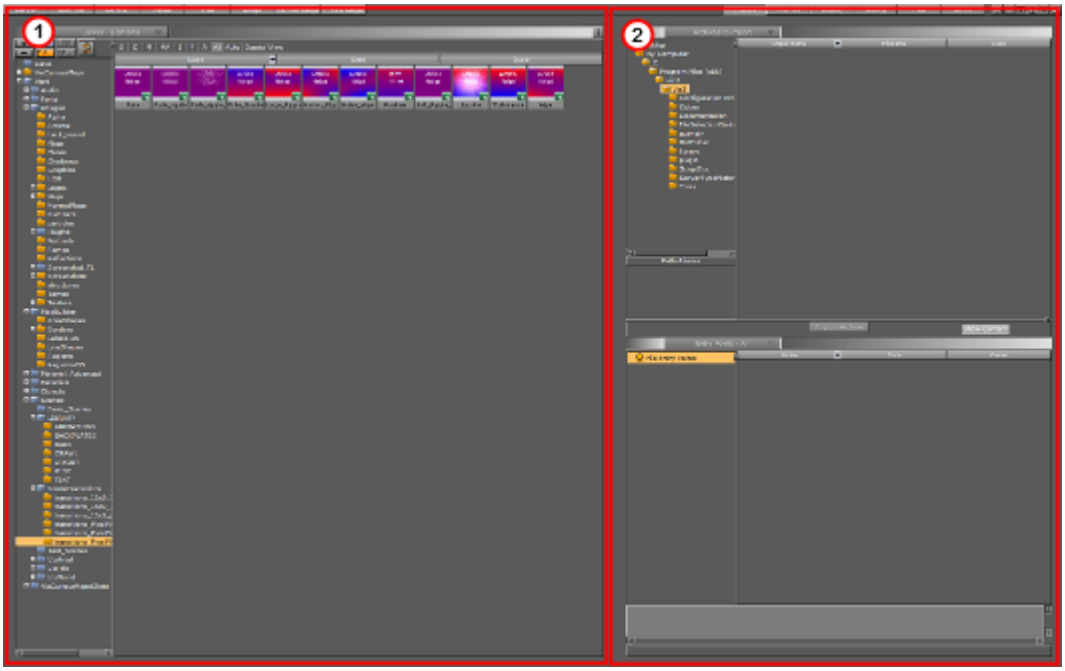
⚠ **Note:** If Viz Artist is running in Viz Engine mode, the import is limited to only import archives.

This section contains information on the following topics:

- Import Window
- Import Menu
- Supported Files for Import
  - Fonts
  - Image Files
  - Video Files
  - 3D Formats
  - Scene Formats
  - Audio
  - Geometries
  - Substances
  - Archives

12.1.1 Import Window

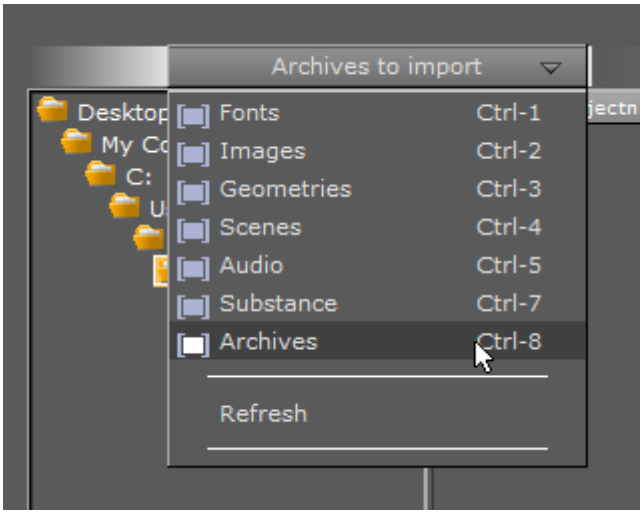
To Open the Import Window, click Import on the main menu or press <F9>.



The Import Window consists of two panels:

- **Server Panel:** Gives access to all folders and projects in the Graphic Hub (1).
- **Import Panel:** Gives access to the source drives, where assets can be imported from (2).

12.1.2 Import Menu



The Import menu is positioned at the top of the Import Window. When an item type is selected from the menu, only the assets of that particular type will be listed in the file list.

12.1.3 Supported Files for Import

The following file formats are supported for import:

- Fonts

- Image Files
- Video Files
- 3D Formats
- Scene Formats
- Audio
- Geometries
- Substances
- Archives

Fonts

The most common file formats for fonts are supported. Please observe however, that path rendering of fonts is not supported for fonts imported with **Font Import** set to `Windows (Advanced)` in **Font/Text Options** in Viz Configuration.

Type	Comments
Adobe Type1 fonts	.pfb files. Postscript Type 1 fonts.
Adobe Type2 fonts	Postscript Type 2 fonts.
True type fonts	Recommended, most common format.

Image Files

Viz Artist supports a wide range of industry standard image file formats.

Format	Comments
.bmp	Bitmap
.bw	Black and white raster image (SGI). Legacy support.
.exr	OpenEXR HDR image format (ILM)
.gif	Graphics Interchange Format.
.hdr	High Dynamic Range Image File.
.jp2	JPEG 2000 compressed images
.jpeg, .jpg	JPEG standard image format

Format	Comments
.pict	QuickDraw native graphics format
.pix	Alias Systems Corporation bitmap file format
.png	Portable Network Graphics. Lossless raster graphics.
.psd	Layers defined in a Photoshop file will be imported as separate images into the database.
.tga	Truevision TARGA
.tiff, .tif	Tagged Image File Format
.xpm	X11 pixmap

Video Files

Format	Comments
MPEG2 IFrameHD with alpha	Can only be imported with the Adobe After Effects importer (see Adobe After Effects)

3D Formats

Format	Comments
3ds Max (.3ds / .fbx)	Autodesk Max 3ds file format. Fbx is a 3d file format supported by many 3d vendors.
Autocad (.dxf)	Drawing Interchange Format.
Maya (.fbx)	FBX for Maya
Softimage (.fbx)	FBX for Softimage
VRML 2.0	Virtual Reality Modeling Language
VRML including animation	Virtual Reality Modeling Language

Format	Comments
Wavefront (.obj)	Geometry definition file format

Scene Formats

Scenes created in certain other 3D applications can be imported to Viz Artist. However, any geometry that is to be imported must be a polygon geometry. Pay special attention to this when importing geometries from CINEMA 4D, as these must have been saved as a **.c4d** file with the **Save Polygons for Melange** option enabled (Located under *Edit -> Preferences -> Files* in CINEMA 4D), to make sure that all items saved will be stored with their corresponding polygon data.

Format	Comments
After Effects (.aep)	See Adobe After Effects. **Note**: A license is required for the use of Adobe After Effects and the import of Adobe After Effects Scenes, issued by Vizrt.
CINEMA 4D (.c4d)	See CINEMA 4D. **Note**: A license is required for the import of CINEMA 4D Scene, issued by Vizrt.
Autodesk (.fbx)	FBX is a 3D asset exchange format compatible with many 3D tools (see FBX Files)

Audio

There are two ways to capture audio in the Viz Engine: through Matrox or a DirectSound compatible device.

Video clips can have interleaved audio in it. Maximum 16 channels, minimum 2 (stereo) as mono is not supported. The format is limited to 24 bit and 48 kHz.

The following formats are supported for Clips:

Format	Comment
WAVE	Up to 96 kHz, 24 bit and 16 Channels. The recommended format.
MP3	All Formats (Stereo only).
Ogg Vorbis	All Formats, up to 16 Channels.

Geometries

All geometries imported to Viz Artist must be polygon geometries.

Format
.obj
.3ds

Substances

Format
.sbsar

Archives

Archives can contain all items that Viz Artist/Engine supports (see Import Archives).

12.1.4 Import Archives

An Archive (filename extension '.via') is a single file that holds various types of items and remembers the project or folder they were stored. Archives are typically used for backup and for transferring Scenes from one Graphic Hub (GH) to another GH.

⚠ **IMPORTANT!** Archives created with Viz Artist 2.x must not be imported to Viz Artist 3.x in this way. They need to be imported using the Graphic Hub Manager (see Graphic Hub User Guide).

This section contains information on the following topics:

- View Archive Content
- Sorting Archive Content
- To Import an Archive to the Root of a Graphic Hub
- To Import an Archive to a Specific Location in a Graphic Hub
- To Import Part of an Archive
- To Search for Import Files

View Archive Content

Use the drop-down menu in the preview panel to see the different aspects of an item's composition.

- **Entry Point View**: Shows all Entry Points. An entry point is created for each item or project or folder added to an archive by drag and drop.

- **Folder View**: Shows all the geometries that are contained in the Archive (includes referenced items as well).
  

Sorting Archive Content

Use the column header bars to sort the archive content. Click on each column header bar to sort the contents ascending or descending. The small arrow shows which way the contents are sorted.

To Import an Archive to the Root of a Graphic Hub

⚠ **IMPORTANT!** Using this method of import, the archive will be imported to the 'root' (/) of the GH while maintaining the path and names of the items in the archive.

1. Click **Import**.
2. Select **Archives to import**:
   - Click on the **Import** Menu, and click on **Archives**, or press <Ctrl+7>
   - 
3. In the Import tree, search for the Archive to import.
4. Select the required Archive.
5. Click on **Import Archive** (1).
   

   ⚠ **Tip:** Press **Show Content** (2) button to show the Archive content in the panel below. The Show Content button can be deactivated for faster performance (some Archives can take a long time to show).

6. If there are files which have the same UUID as an item in Viz Artist, the question **Do you want to replace?** is asked:
   

   Select from
   -  : Replace the item in all detailed folders
   -  : Replace all items with the archive files, in all folders
   -  : Do not replace the item. The import procedure continues with the next file
   -  : Do not replace any items. The import continues and does not import files found as items in Viz Artist
   -  : Stop the import of all files

To Import an Archive to a Specific Location in a Graphic Hub

⚠ **IMPORTANT!** Use this import procedure to control which Project or Directory the archive should import to. The Path and Names of items in the archive will be preserved and will use the location in the GH, selected by drag and drop, as a “start point”.

1. Click **Import**.
2. Select **Archives to import**:
   - Click on the **Import** Menu, and click on **Archives**, or press <Ctrl+7>
   - 
3. In the Import directory tree, search for the Archive to import (see Browse for Files to Import).
   
4. Drag the archive or archives selected for import to either:
   - the target Folder (or Project) in the Server folder tree (1), or
   - the Server item Panel (2).

   
5. The archive starts to import.

   ⚠ **Tip:** Press the Show Content button (3) to show the Archive’s content. The Show Content button can be deactivated for faster performance as some big archives can take a long time to display.

   
6. If there are files which have the same UUID as an item in Viz Artist, the question **Do you want to replace?** is asked.
   
   Select from
   -  : Replace the item in all detailed folders
   -  : Replace all items with the archive files, in all folders
   -  : Do not replace the item. The import procedure continues with the next file
   -  : Do not replace any items. The import continues and does not import files found as items in Viz Artist
   -  : Stop the import of all files

To Import Part of an Archive

1. Click **Import**.
2. Select **Archives to Import**:
   - Click on the **Import** Menu, and click on **Archives to Import**, or press <Ctrl+7>
   - 
3. In the Import directory tree, search for the Archive to import.
4. Select the required Archive in the right side panel.
5. From the top menu, select **Entry Points View**.
   
6. Select an import option:

- To import all entry point files from the archive, drag the **File Entry Points** file folder and drop it in the left panel.
- To import a folder and its contents from the archive, drag the required folder from the folder entry points and drop it in the left panel.
- To import one or more files, drag and drop them into the left panel.

> ⚠ **IMPORTANT!** The archive is imported into the folder selected in the Graphic Hub tree. All reference files are included to guarantee full functionality. This method of import will "flatten" the imported items. This means that the path or directory structure of the items in the imported archive will not be kept. Use this method of import for importing parts of, or items of an archive.

> ⚠ **Note:** To import projects or folders, the same procedure applies, with the exception that you select **Folder View** from the pull-down menu, and drag and drop the required folder.

To Search for Import Files

Search for files before or after the import for an Archive. Use this feature to see if similar or the same files are present in the system and to find the file's location after import.

1. <Right-click> any of the items in the Archive.
2. Select **Search for this file in current GH**.

12.1.5 Import Files

Use File Import to import these items:

- Fonts
- Images
- Geometries
- Scenes
- Audio clips

Path Alias

Use a path alias to help streamline the import process. Path Aliases, defined in Viz Configuration, create shortcuts to frequently used file paths for import operations.

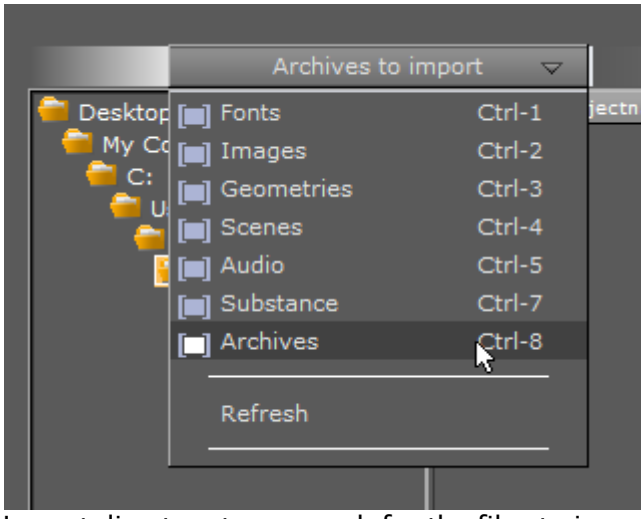
Browse for Files to Import

Use the import file list, on the left side of the Import Window, to browse the directories on a disk:

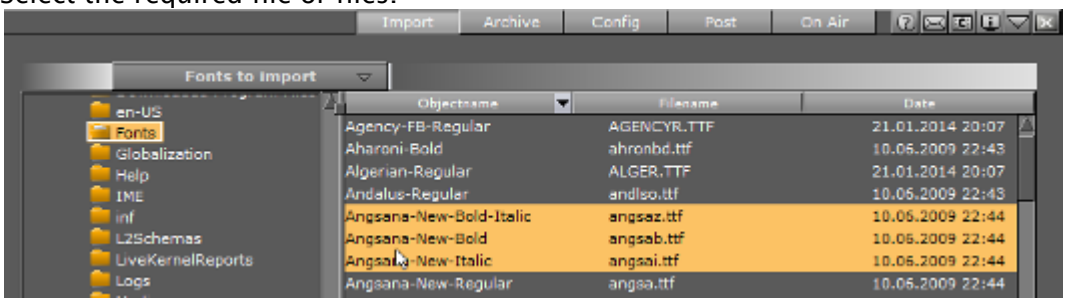
- Double-click a directory to show its sub directories
- Click a directory to show the contents in the right window
- Click a path alias to quickly jump to this directory

- Hold down the <Ctrl> key and click all required files to select multiple files.

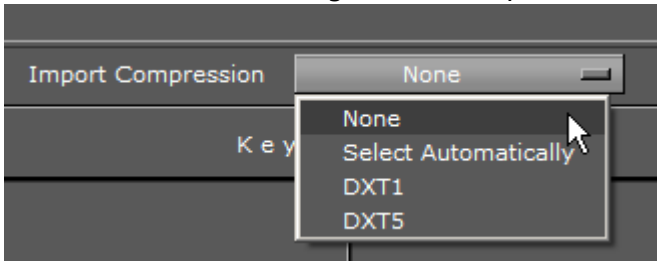
To Import Selected Files

1. Select the target folder in the Server panel.
2. Click **Import**.
3. Select the type of file to import:
   - Click the **Import** Menu and select the type of file to import, or press the keyboard shortcut.
   - 

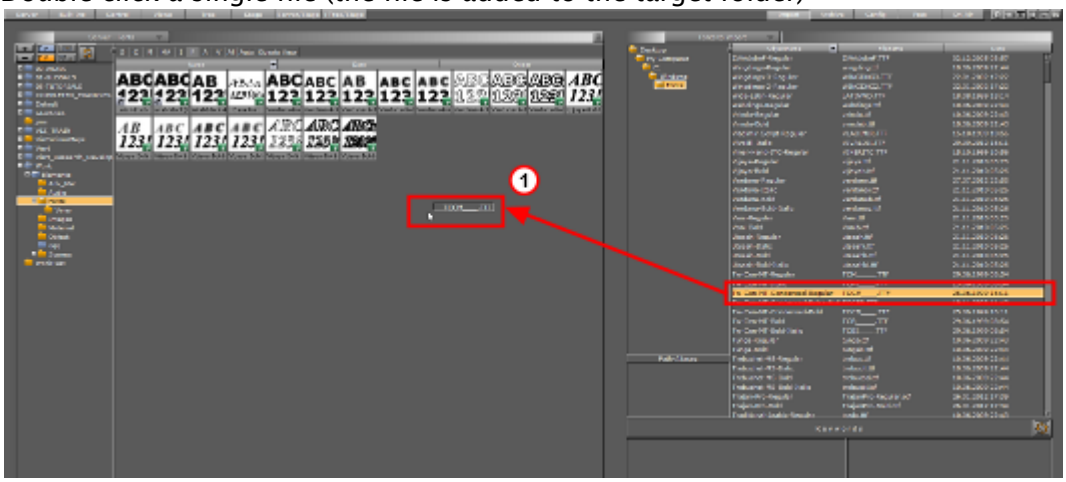
4. In the Import directory tree, search for the files to import.
5. Select the required file or files.



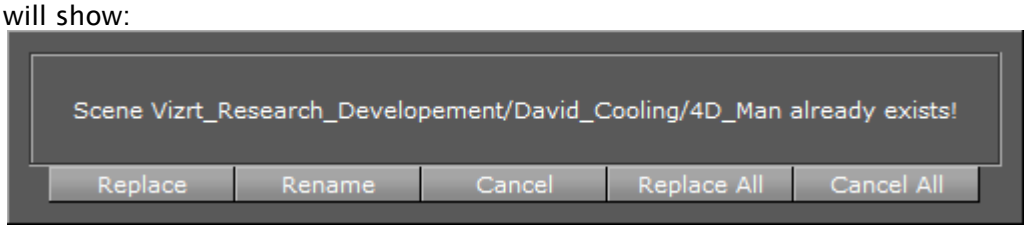
6. **Images only:** Select the import compression for the image or images. Click on **Import Compression**. Options are:
   - **None**
   - **Select automatically**: If the source image has an alpha channel, DXT5 will be used. If the source image does not have an alpha channel, DXT1 will be used.
   - **DXT1**: Works best on images without an alpha channel.
   - **DXT5**: Works best on images with an alpha channel.



7. Import the selected file or files:
   - Drag the file or files to the target folder, in the Server Panel (1), or
   - Double-click a single file (the file is added to the target folder)



8. If a reference to a file, which is to be imported, already exists in the Graphic Hub, this dialog will show:

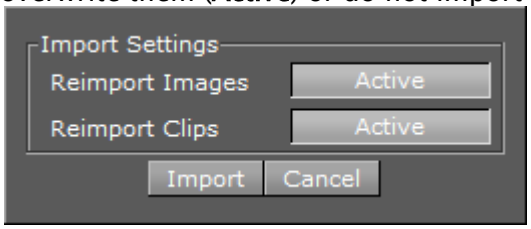


Select from:

- Replace : Overwrite the file in Viz Artist
- Rename : Rename the file and import (new file). Opens a new dialog, type and new name and press **OK**
- Cancel : Stop the file import of the file shown in dialog window. The import procedure continues with the next file
- Replace All : If there are multiple files, overwrite all files
- Cancel All : Stop the import of all files

9.

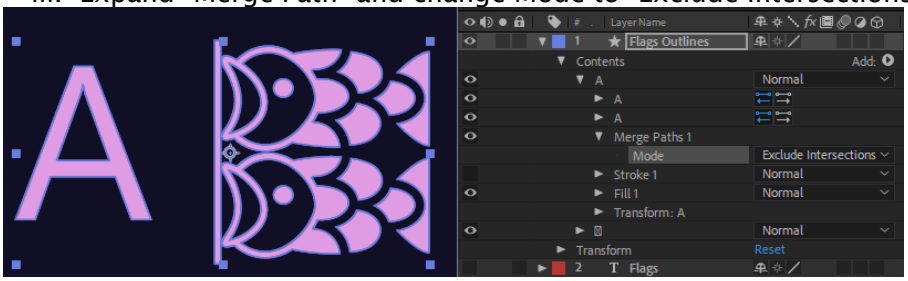
10. **Adobe After Effects projects only**: If image or clip files already exist in Viz Artist, import and overwrite them (**Active**) or do not import files which already exist (**Inactive**).



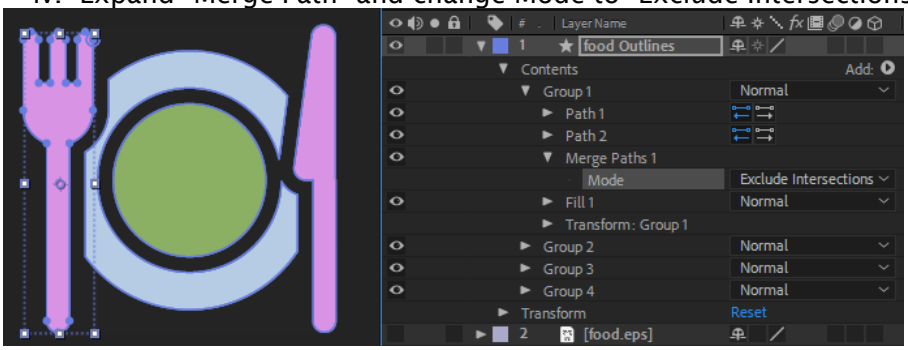
> ⚠ **Tip:** If there are no changes in the image or clips files, select **Inactive**. The import will be much faster, because no output from the Adobe After Effects Renderer is required.

Shape layer in After Effects can be imported to the scene as 2 types of objects, video and geometry plugins, depends on its complexities

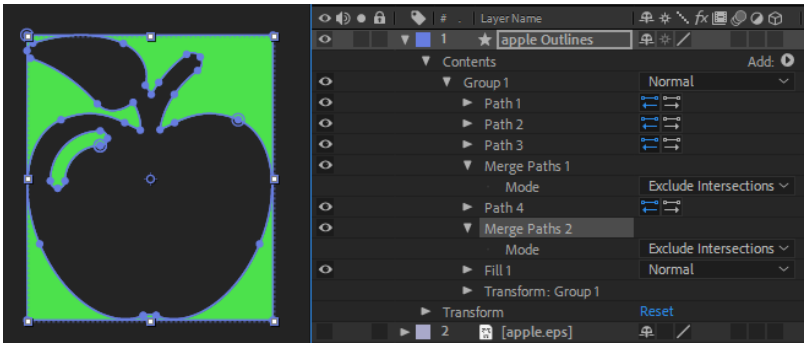
a. Rectangle created by Rectangle or Rounded Rectangle tools will be imported as Noggi object
b. Ellipse created by Ellipse tool will be imported as Eclipse object
c. Star and Polygon will be imported as Star object if
    i. Inner and Outer roundness is 0 and
    ii. Star point is integer number
d. Shape created by Pen tool will be imported into Polygon object
e. Converted shape from text, each character of it will contains multiple paths. Shape of each character merge all paths in a special mode (be decided by After Effects among Add, Subtract, Intersect and Exclude Intersections depends on the overlapping and direction of each path) that we not support it even most of character works with our simple merging method, "Exclude Intersections", manually change Merge Paths mode of all characters to "Exclude Intersections" can make this layer be import into Polygon object instead of video.
    i. Use Horizontal or Vertical Type Tool to create a text layer
    ii. Right click on the text layer and select "Create Shapes from Text"
    iii. Expand "Merge Path" and change Mode to "Exclude Intersections"



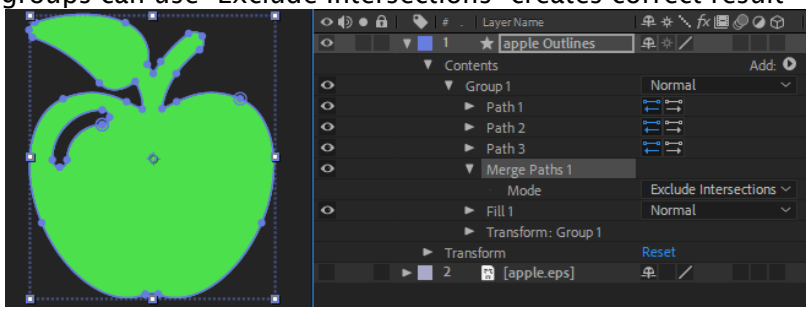
f. Convert shape from imported external vector graphic file such as .eps. This is similar to the converted text but it might have many groups of paths
To import .eps file to After Effects and make it be able to imported as as Polygon object in Viz Artist
    i. Press CTRL+I and select a vector files
    ii. Right click on the vector layer of automatically created composition and select "Create Shapes from Vector Layer"
    iii. Expand Contents and every groups inside, each group is the same as character of converted text
    iv. Expand "Merge Path" and change Mode to "Exclude Intersections"



Some converted vector shape have multiple group of merged paths and not work with "Exclude Intersections" merging and mostly one group with single path is used as boundary.



In this case that group (Merge Paths and its above Path) can be deleted and other groups can use "Exclude Intersections" creates correct result



⚠ We support layer animation but not object animation inside, the layer will be imported as video if any properties of them have a key-frame.

12.2 Archive Of Graphical Resources

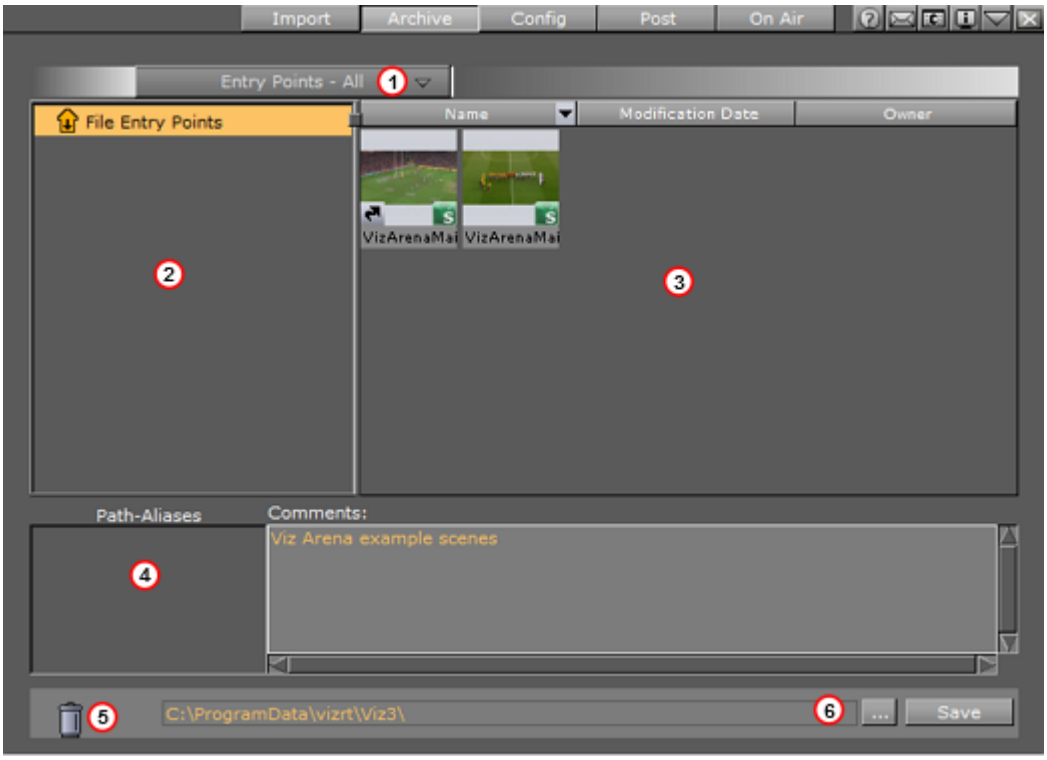
The archive function exports graphical resources, such as one or several Scenes including fonts, images and so on to a file (archive) on disk. The archive will by convention have a '.via' filename extension.

The archive can be use for the safekeeping and/or for transferring of data to another Graphics Hub.

This section contains the following topics and procedures:

- Archive Properties
- Entry-Points Menu
- To Create an Archive

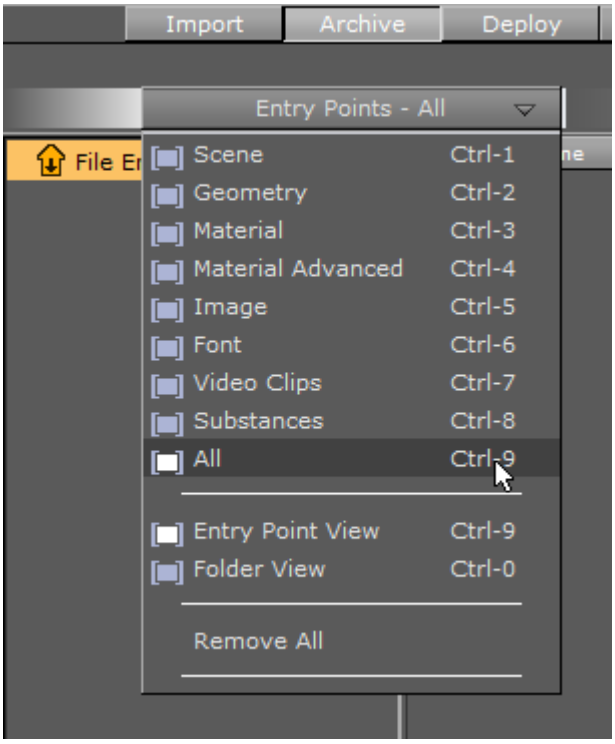
12.2.1 Archive Properties



- **Entry-Points** (1): A pull-down menu is located just below the main menu and allows the selection of data types.
- **File Entry Points panel** (2): The target area for the folders to archive.
- **Data Panel** (3): The target area for resources to archive. Drag items such as Scenes, Fonts, Images etc. to this area to include them in the archive.
- **Use Path-Aliases** (4): Select aliases that represent actual directories or paths. Path-Aliases must be defined in advanced (see the **Configuring Viz** section of the Viz Engine Administrator Guide). Aliases are typically short and descriptive names used as a convenient shortcut to directories which could have long or cumbersome names.
  Use the **Comments** area to add descriptive comments about the archive.
- **Trash-can** (5): Remove items not needed to be archived.
- Browse and select a file or specify an archive name (**6**) before pressing Save. The extension .via will be attached to the file name if not specified.
  Entry points represent the main items in an archive. Other items in the archive are referenced and attached automatically to the archive.

12.2.2 Entry-Points Menu

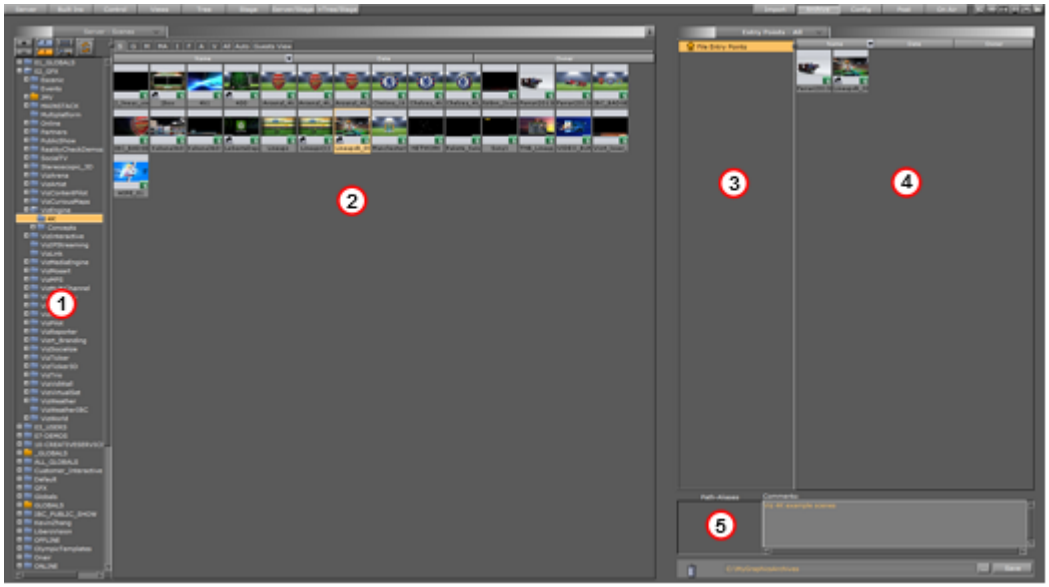
At the top of the archive view there is a pull-down menu enabling you to switch the way the content of your archive is displayed:



- **Scene** to **Video Clips**: Only shows the items of the selected file type.
- **All**: Shows items of all file types.
- **Entry Point View**: Shows only entry points.
- **Folder View**: Shows the folder hierarchy.
- **Remove All**: Removes all items from the archive.

12.2.3 To Create an Archive

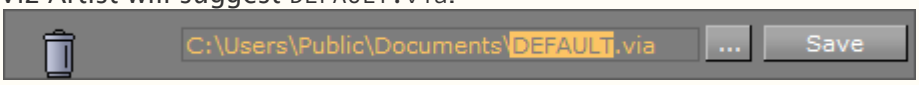
1. In the main menu, click **Archive**.
   The Archive view is shown.



2. From the Server side (1), drag folders into the Archive view's left panel (3) and files into the right panel (4).
3. If required, enter a comment in the comments area (5).

> ⚠ **Note:** Comments can be a maximum of 255 characters.

4. Either manually or by clicking the ellipsis button (...), enter a file path for the archive.
5. After the file path, enter a file name for the archive.
6. Click the **Save** button.

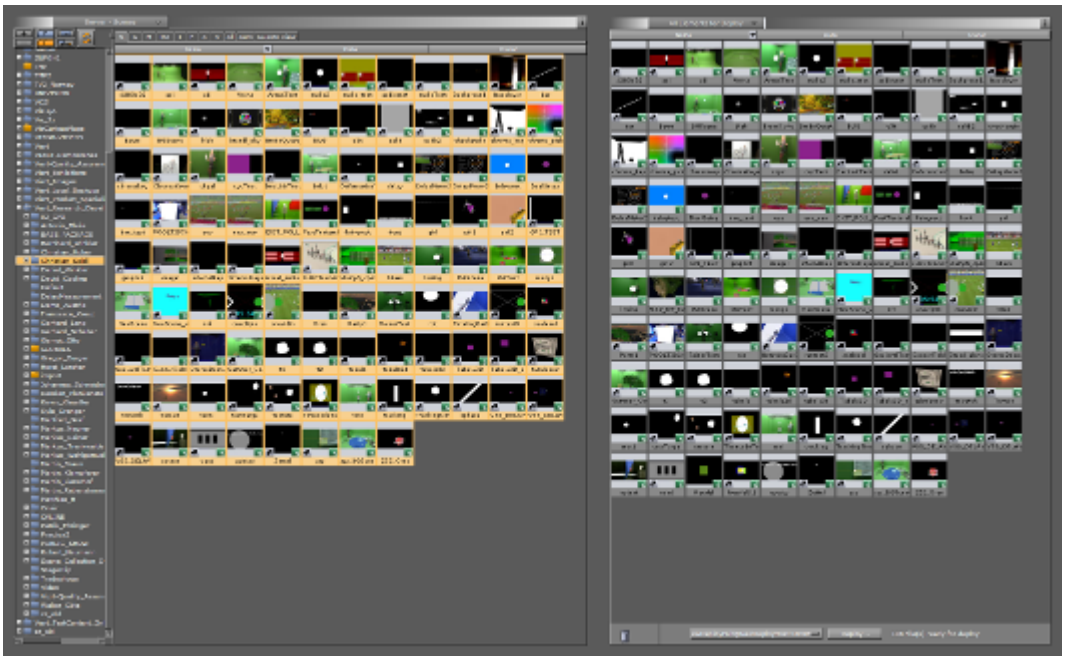
> ⚠ **Note:** If navigating to a preferred location before adding the file name for the archive, Viz Artist will suggest `DEFAULT.via`.
>
> 
>

> ⚠ **IMPORTANT!** When archiving a set of transition logic scenes in Viz Artist, the archive will not include the Geometries that are generated by Viz Template Wizard when creating a template for the control applications. It is therefore recommended to archive scenes intended for Viz Pilot by using Viz Pilot or Viz Template Wizard. If Viz Artist is used, the Geometries can be added manually.

12.3 Deploy Items

The Deploy panel makes it possible to copy items, with all their necessary data, and deploy them from one server to another. For example, if a Scene is deployed, all items referenced by the Scene (Images, Fonts, etc.), will also be deployed.

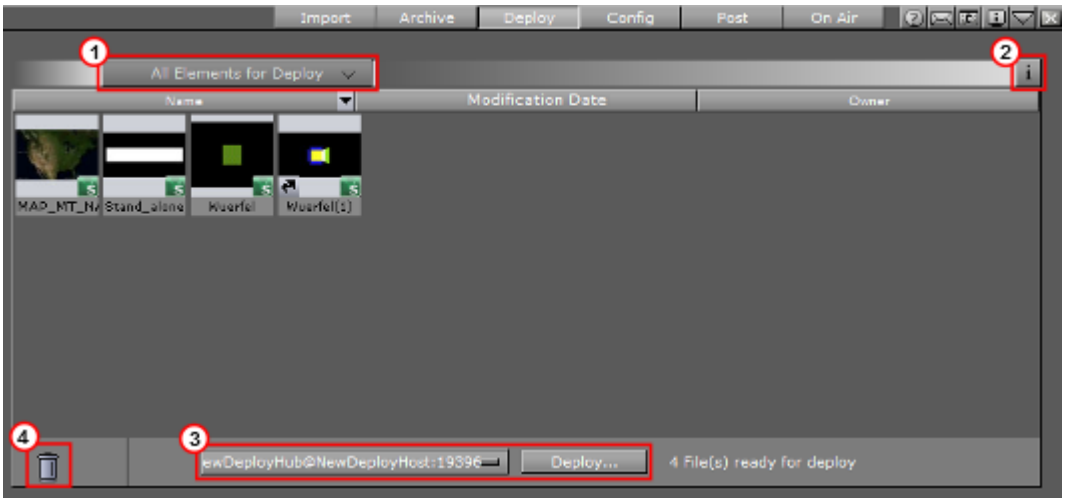
The **Deploy** button is only available when at least one deploy server is configured in Viz Configuration. If you configure multiple servers, you can deploy items from one server to another. See the **Database** page of the **Configuring Viz** section of the Viz Engine Administrator Guide for more information.



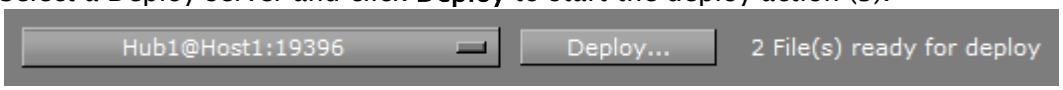
This section contains the following topics:

- Deploy Panel Overview
- To Deploy Selected items
- To Deploy Single items

12.3.1 Deploy Panel Overview



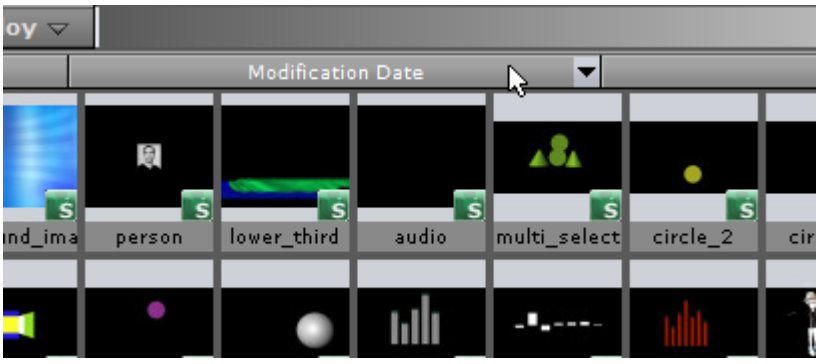
- Open an item filter menu (1).
- Click the information button  (2) to open the **Deploy Information** panel. This panel displays information about the deploy action. You can choose keep messages or not when you close the Deploy Information panel . The icon will turn to orange when information is added.
- Select a Deploy server and click **Deploy** to start the deploy action (3).



- Drag and drop unwanted items from the Deploy panel to the trash bin (4).

Order Items to be Deployed

Use the column header bars to order the items to be deployed. Click on each column header bar to order its contents ascending or descending. The small arrow shows which way the contents are ordered.



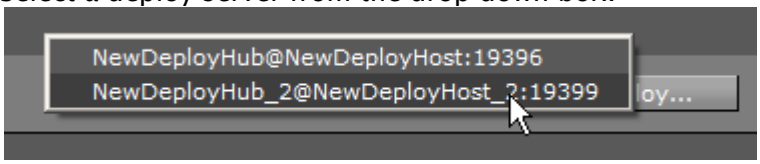
Item Context Menu in the Deploy Panel

1. Right-click an item to open its context menu.

2. Select **Search for this file in current GH**.
   The item will show in the Server Panel.

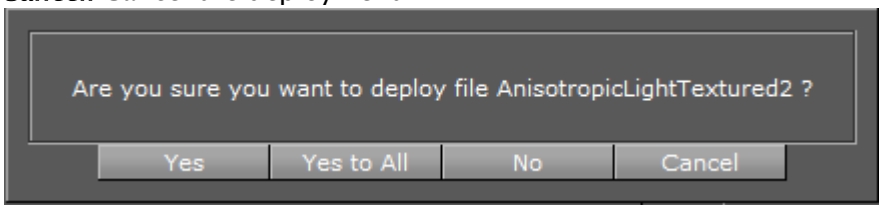
12.3.2 To Deploy Selected items

1. Click on **Deploy**.
2. Drag and drop all the required items from the Server Panel to the Deploy Area.
3. Select a deploy server from the drop-down box.



4. Click **Deploy....**

5. Select from:
   - **Yes:** Deploy each item one at a time.
   - **Yes to All:** Deploy all items together.
   - **No:** Cancel each item one at a time.
   - **Cancel:** Cancel the deployment.

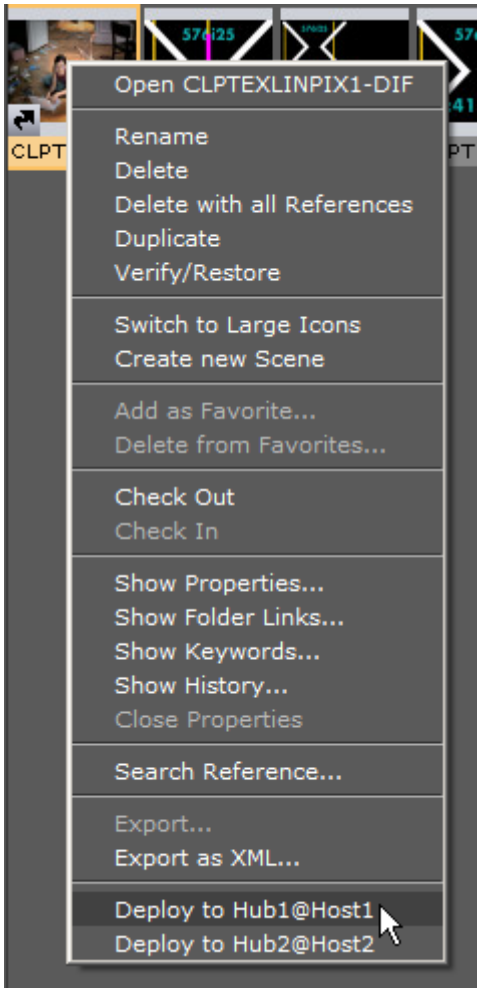


6. After deployment or cancellation select to clear the deploy list or not.

> ⚠ **Note:** If the list is not cleared all the items will stay in the Deploy list until required again. The list will automatically clear when Viz Artist is restarted.

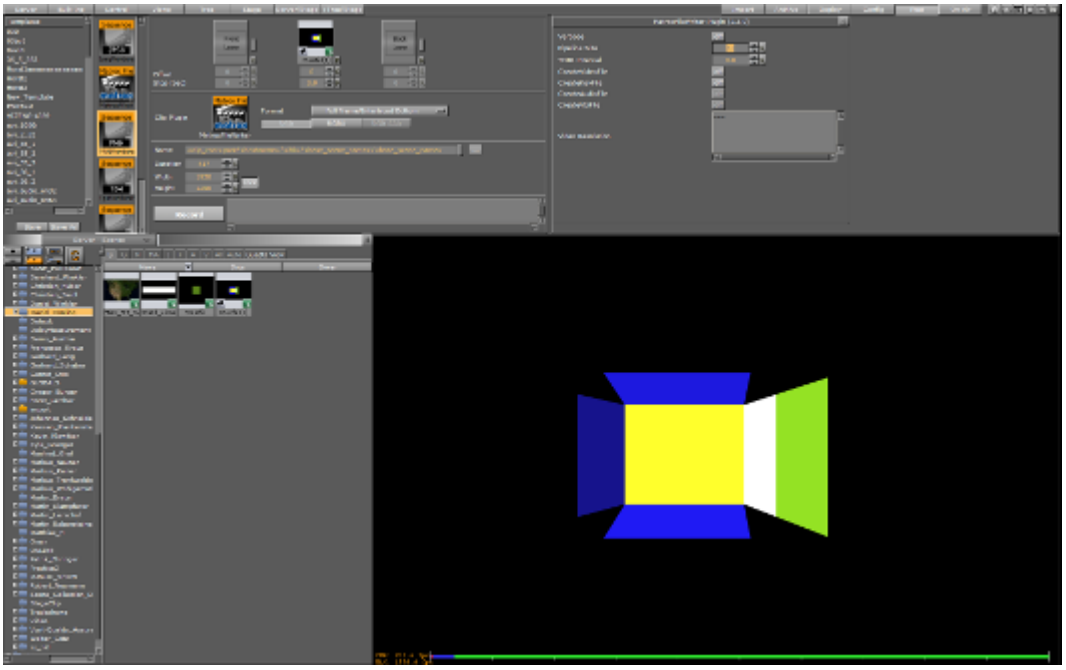
12.3.3 To Deploy Single items

1. Right-click on the selected item in the Server Panel.
2. In the context menu, select **Deploy to <selected server>**.



12.4 Post Render Scenes

Scenes can be exported as video or image files. This process is called post rendering.

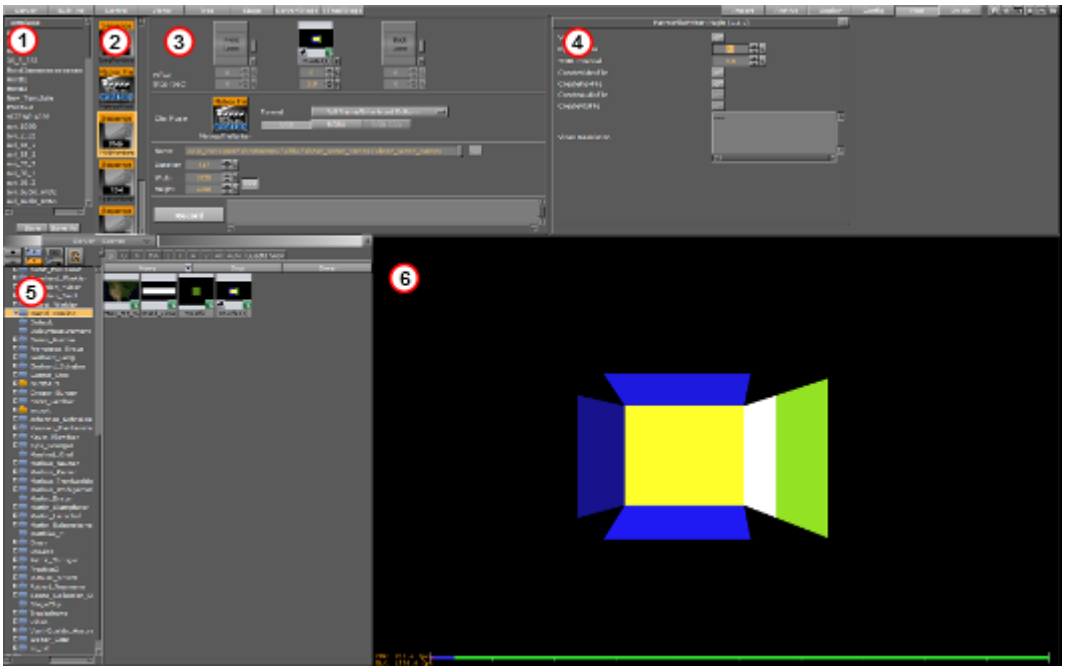


Note: Post rendering is not possible with clips or live video feeds. However, these items can be post rendered through the SoftClip plug-in, if it is animated through the stage.

This section contains information on the following topics:

- Post Render Screen
- To Post Render a Scene
- Post Render Templates
- Post Rendering plug-ins

12.4.1 Post Render Screen



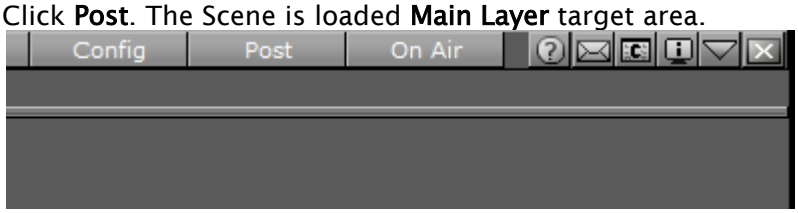
The post render panel consists of:

- Template (1) (see Post Render Templates)
- Installed plug-ins are listed (2) (see Post Rendering plug-ins)
- Post rendering setup (3):
  - **Main Layer**: Shows the Scene to be rendered. When Post is clicked, the Scene currently open will be set as the Main Layer
  - **Front Layer**: Sets a Scene running in front of the Main Layer
  - **Back Layer**: Sets a Scene running behind the Main Layer
  - **Offset**: Sets an offset for the Main, Front and Back Layer, so they will start at different times
  - **Stop**: Defines the number of seconds to pause during the animation stop points of a Scene. If there are stop points in the Scene, these are normally lost in the post rendering process, unless an interval is assigned here
- Plug-in property editor (4)
- Server Panel (5)
- Output (6)

12.4.2 To Post Render a Scene

1. Open the Scene to be post rendered.

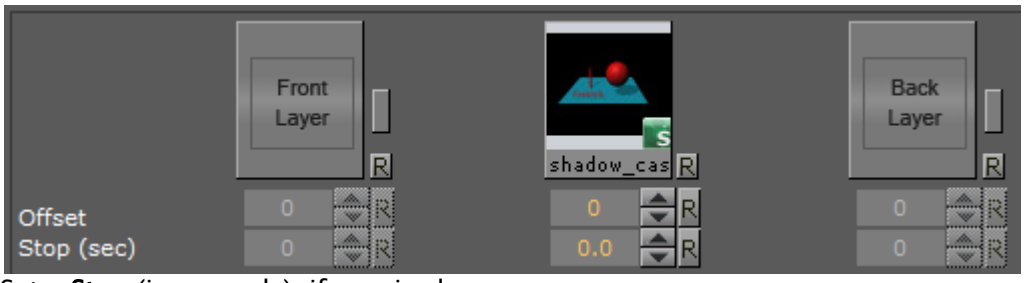
2. Click **Post**. The Scene is loaded **Main Layer** target area.



> ⚠ **Note:** When Post is clicked, the Scene currently open on will be set as the Main Layer

> ⚠ **Tip:** It is also possible to click on **Post**, then select a Scene and drag it to the Main layer.

3. Set an **Offset**, if required. How the offset affects the Scene can be seen in the Post Rendering area.



4. Set a **Stop** (in seconds), if required.
   This stops the post rendering after the assigned interval.
5. Drag a selected post render plug-in, to **Clip plug-in** target area. Select from:
   - JPEG
   - PNG
   - MOV
   - TGA
   - TIFF
   - VideoRenderer
   - MatroxFileWriter. The Matrox File Writer plug-in requires a Matrox card. See Post Rendering plug-ins section for additional Matrox File Writer settings.

6. Select a frame **Format**. Viz Artist/Engine always renders 50 images per second for PAL or 60 for NTSC. Dependant on the post render format, not all of them or not the full frames are stored. Options are:
   - **Full Frame**: Stores 50 or 60 full height frames for each second of video. The number depends on the video settings, PAL/50 or NTSC/60 frames, and the size also changes with the video format (720x576 for PAL digital and 720x486 for NTSC digital). Normally TV is 25 or 30 interleaved frames, but in this case the post rendering produces twice the amount of pixels needed. An editing system (or wherever you put the images you render) can throw away 50% of the pixels and still produce a “normal” video stream.
   - **Fields Top/Fields Bottom**: Stores 50 fields per second of video. It is the same amount of data as interlaced top/bottom, but instead of shuffling the data the post rendering process writes different files for each field.

- **Interl. Top, Interl. Bottom**: Stores 25 interlaced frames for each second of video. Each frame consists of 2 fields (half size of PAL or NTSC for example 720x288). These two fields are combined line by line. Interlaced top means that the first line of the first field is the first line in the interlaced frame. In interlaced bottom the first line of the second field is the first line in the interlaced frame. This determines field dominance.
- **Full Frame Skip**: Stores every other full frame, resulting in 25 (or 30 for NTSC) frames per second.

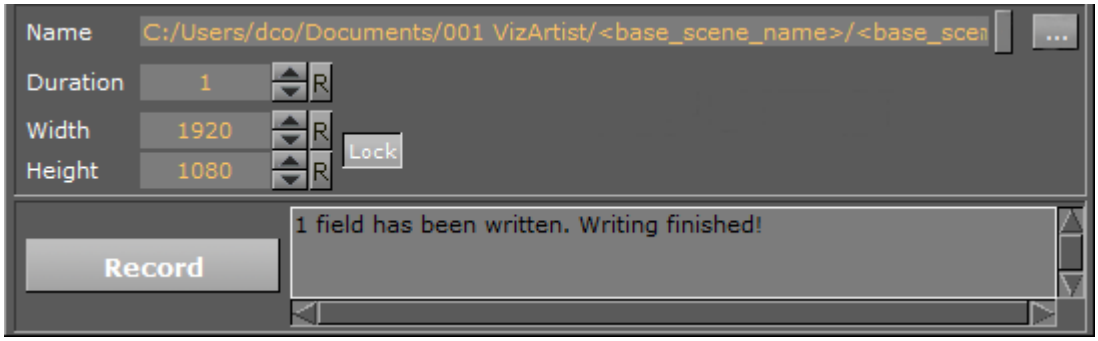
7. Select the color settings. Select from:
   - **RGB**: Creates RGB images. Here a pixel is represented by 3 bytes containing a red, green and blue sample.
   - **RGBA**: Creates a red, green and blue sample, along with an additional color sample, all in a single image. When shown on top of other images, RGBA pixels are blended with the background pixel according to the value of this alpha component.
   - **RGB AAA**: Creates an RGB and an Alpha image for each image rendered.

   > ⚠ **Note:** These settings are already fixed for the Matrox File Writer (see Post Rendering plug-ins).

8. In **Name**, select a destination file path and name.

   > ⚠ **Tip:** When a Scene is added to the Main Layer, the path and filename are automatically assigned. To keep a previously selected path, click the **Keep Path** button (small rectangle to the right of the Name field).

   > ⚠ **Tip:** With the cursor over **Name** a tool tip shows the path name.



9. Set the parameters for:
   - **Duration**: Sets the number of frames to be rendered.
   - **Width**: Sets a scaling value for the width.
   - **Height**: Sets a scaling value for the height.

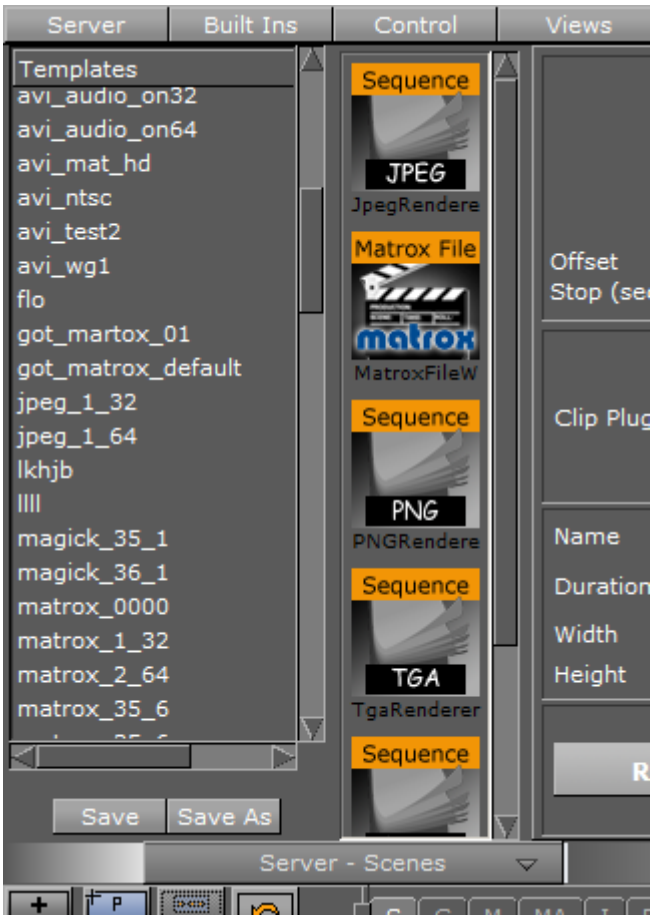
   > ⚠ **Tip:** To scale the height and width proportionally click Lock. When disabled the lock height and width scale can be set separately. The default setting is locked.

10. Click Record .

    At any time, while rendering, click Stop Recording to stop the rendering. When the rendering is stopped the file(s) already written will be kept.

12.4.3 Post Render Templates

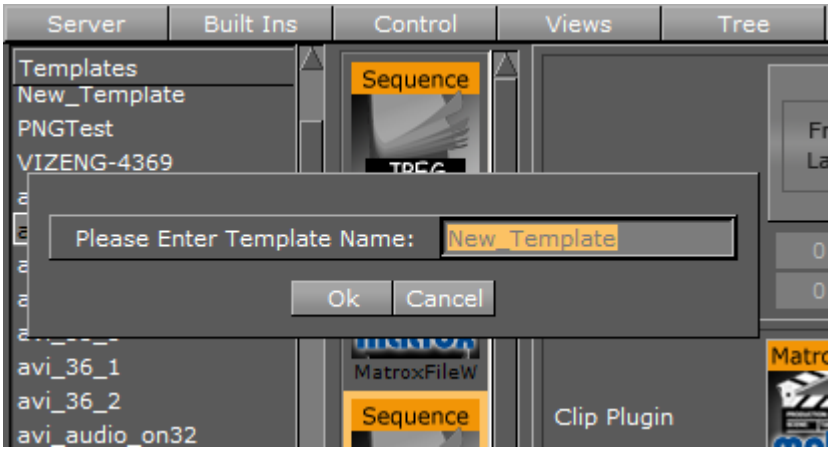
If Scenes are post rendered often, with the same settings, create a Post Render Template. To use a saved template double-click it in the Template panel.



⚠ **Note:** Post rendering templates are saved in the Graphic Hub *and not* on the local machine.

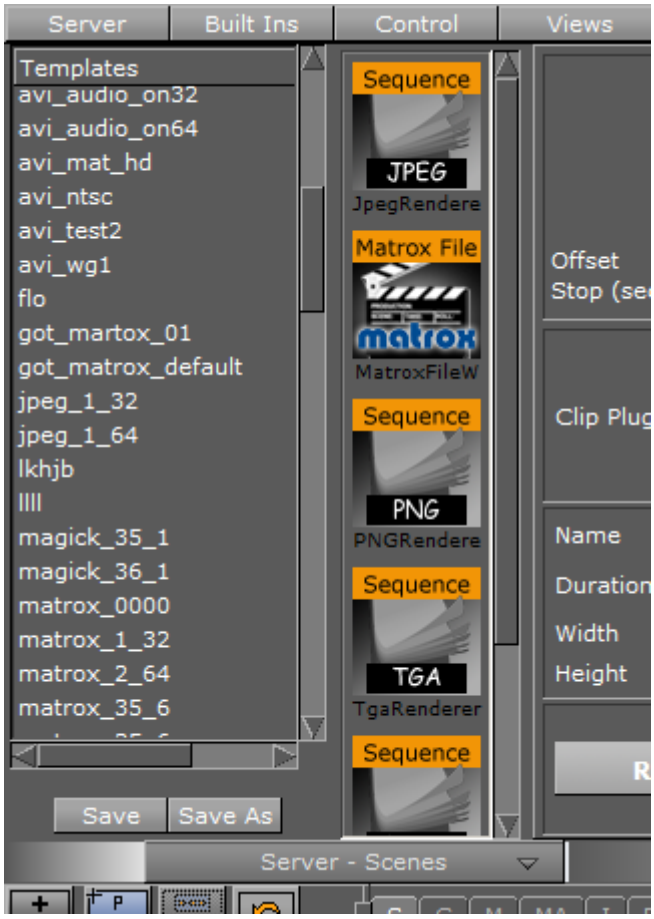
To Save a Post Render Template

1. Configure all the settings in the procedure To Post Render a Scene.
2. In the Templates panel, click **Save**.
3. Enter a name for the template.
4. Click **OK**.
   Saved templates show in the Templates panel.



To Change a Post Render Template

1. Double-click the post rendering template in the Templates panel.
2. Change the required settings.
3. Click **Save**.

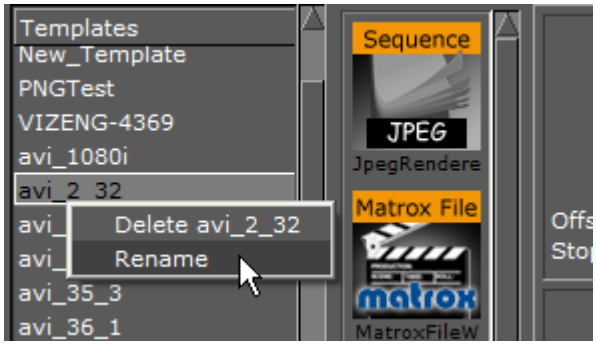


To Load a Post Render Template

- Double-click the post rendering template in the Templates panel.

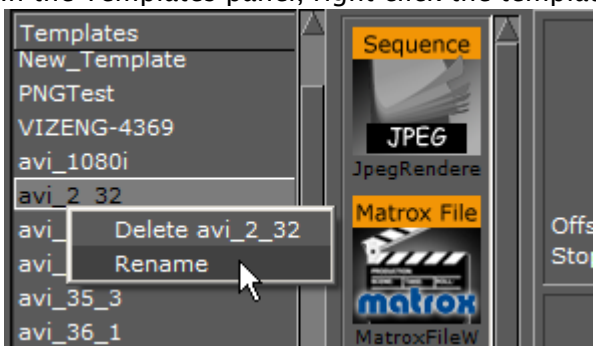
To Rename a Post Render Template

1. In the Templates panel, right-click the template and select **Rename**.
2. Enter a new name and click **OK**.



To Delete a Post Render Template

- In the Templates panel, right-click the template and select **Delete <name>**.



12.4.4 Post Rendering plug-ins

Post rendering plug-ins define the format of the post rendered files. Viz Artist offers a set of image and video rendering plug-ins:

> ⚠ **Note:** Additional plug-ins can be installed at any time.

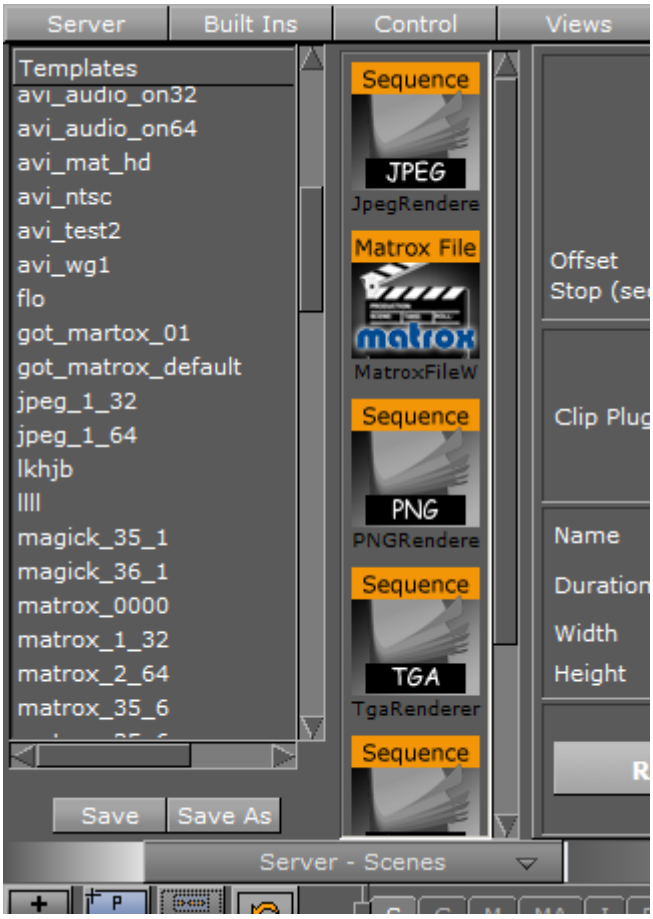


Image Render plug-ins

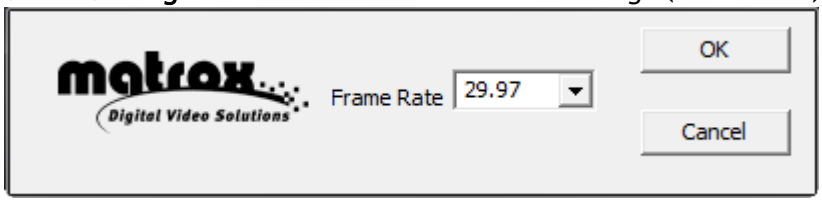
Image rendering plug-ins, such as JPEG, PNG, TGA and TIFF, will all create a single image of every frame in a Scene. The images are named as set in the **Name** field in the post render setup, with incremental numbers.

Some of the plug-ins offer a Quality setting that allows to setup the quality level of the jpeg images to be created.

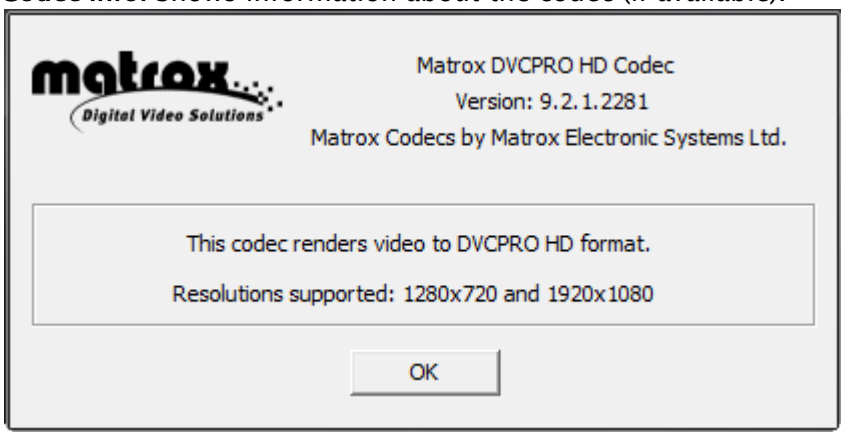
Video Clip Render plug-ins

The video rendering plug-ins render a Scene to a Video clip. If the Video Renderer or QuickTime Renderer plug-in are used, a menu showing all installed video codecs will show:

- **Available Codecs:** Lists all available codecs.
- **Codec Settings:** Shows the available codec settings (if available).



- **Codec Info:** Shows information about the codec (if available).



- **FPS:** Sets the frame rate per second.
- **Flicker Filter:** Sets the Flicker filter rate.

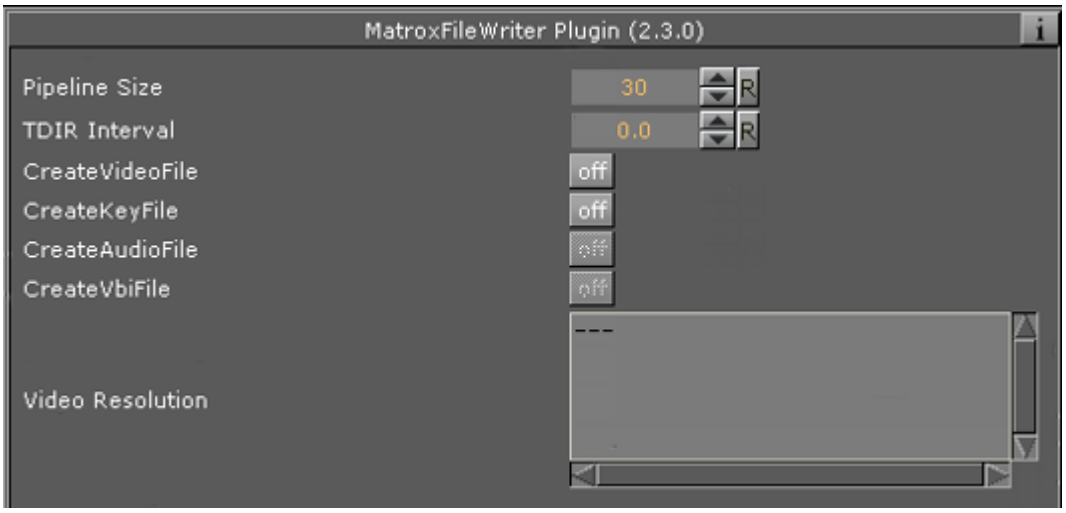
Select the codec to be used and specify the parameters for Frame rate and Flicker Filter. If the video codec allows, specify codec parameters.

The **Matrox File Writer** plug-in is also a video rendering plug-in that is available to users with a Matrox board installed, and will render the Scene into a Matrox encoded clip. It uses the Matrox video board to encode and write clips to file in real-time.

> ⚠ **Note:** The **Video Output** DVI out setting must be set to **Inactive** (see the Configuring Viz section of the Viz Engine Administrator Guide).

When a Scene is to be Post Rendered (To Post Render a Scene),

> ⚠ **Note:** These additional settings for the **Matrox File Writer** plug-in:



- **Pipeline Size**: Defines the number of internal buffers in frames for the matrox framework to operate. for SD+720p+1080i should be around one second (~25-30). It is recommended that you leave the default values.
- **TDIR Interval:** Sets the interval time, in seconds, to update the video file while it is recorded. The value must be 1 or more (less than one will not activate the TDIR feature).

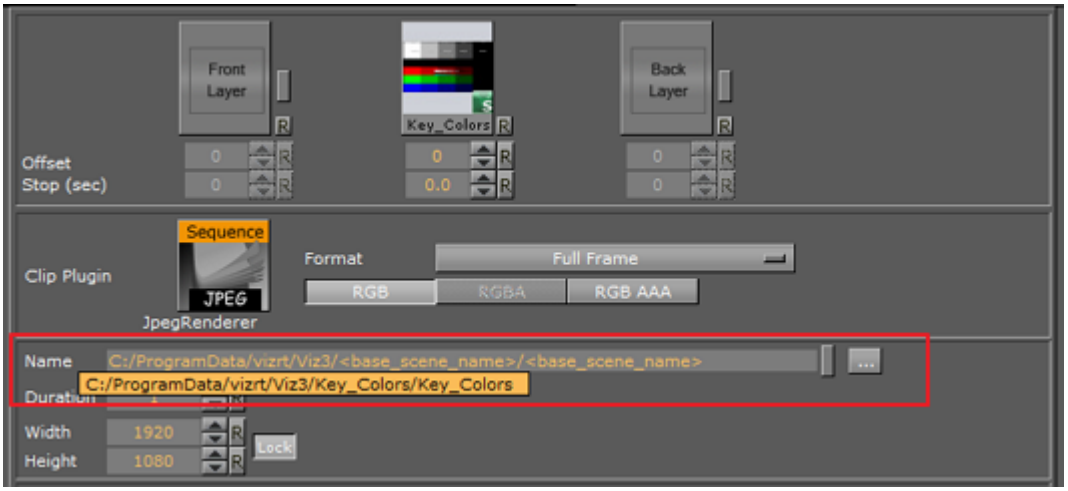
  > ⚠ **Note:** TDIR is only supported on '.avi', '.p2.mxf', 'xdcam.mxf', 'xdcam_422.mxf', '.mov', and '.wav' containers. The MATROX_MOV .mov containers write a separate file (<basename>.Ref<.extension>) that is updated to read the .mov file. This reference file needs to be used while the file is growing. The QT_MOV containers update the .mov file directly.

- **CreateVideoFile**: Creates a fill clip when set to **On**.
- **CreateKeyFile**: Creates a key file when set to **On**.
- **CreateAudioFile**: Not in use.
- **CreateVbiFile:** Not in use.
- **CreateProxyFile**: Specifies a custom resolution copy of the clip when set to **On**. These settings behave in a similar manner to the video file options and show when this is set to On.

- **Video Resolution**: Shows clip resolution information.

12.5 Placeholder Names Used For File-Name Expansion

Placeholder names (variables) are available when final (i.e. fully expanded) path and file-names are created or searched for. You can hover the mouse over the receiving variable or name to see a tooltip for what the expanded filename will be. The illustrative screen-shot below show how this looks in the post-render window.



The available placeholders for post-render mode are:

Variable	Definition
<absolute_scene_name>	Expands to the complete path of the loaded scene
<base_scene_name>	Expands to the scene name
<clip_root>	Expands to the clip_root
<hostname>	Expands to the host name
<if0>	Expands to the IPV4 network address of the first network interface
<if1>	Expands to the IPV4 network address of the second network interface
<if2>	Expands to the IPV4 network address of the third network interface

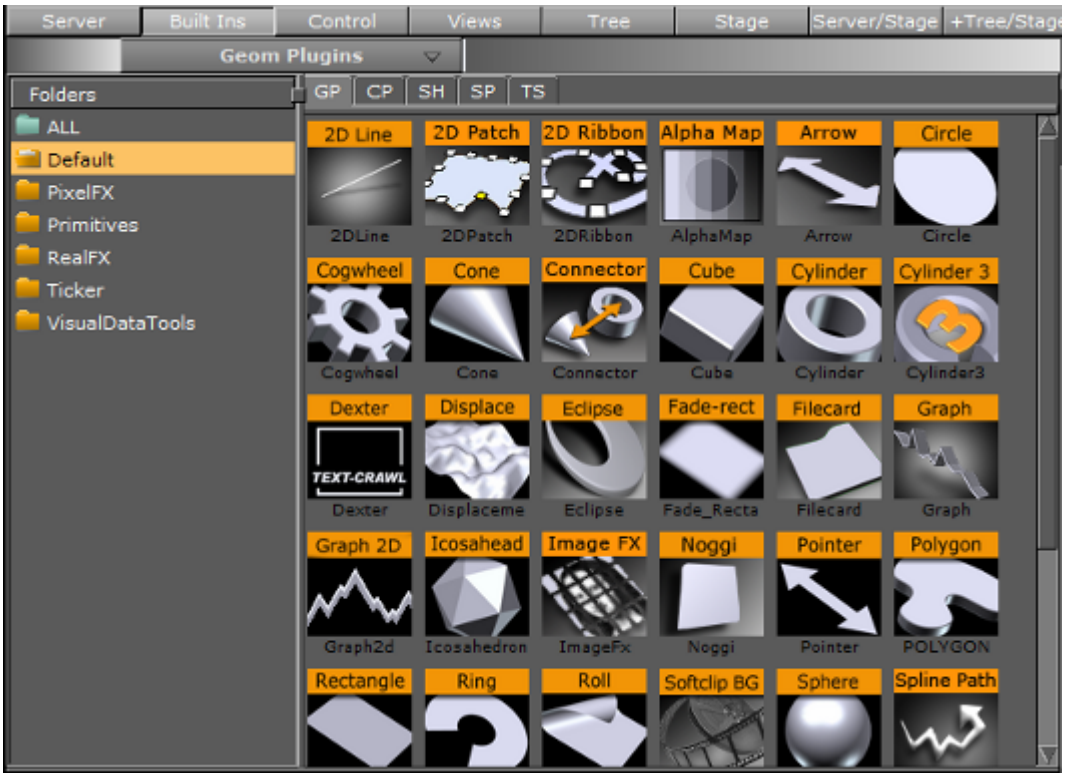
Example usage

```
<clip_root>post/<hostname>/<if0>/<base_scene_name>
```

The available placeholders for clip-out channel are:

Variable	Definition
<hostname>	Expands to the host name
<if0>	expands to the IPV4 network address of the first network interface
<if1>	Expands to the IPV4 network address of the second network interface
<if2>	Expands to the IPV4 network address of the third network interface
<absolute_scene_name>	expands to the complete path of the loaded scene
<absolute_scene2_name>	expands to the complete path of the loaded scene2

13 Geometry Plug-Ins



The default path for the Geometry plug-ins is: `<viz install folder>\plug-in\<plug-in name.vip>`

13.1 Default

The following Geometry plug-ins are located in the Default folder:

- 2D Line
- 2D Patch
- 2D Ribbon
- Alpha Map
- Arrow
- Circle
- Cog Wheel
- Cone
- Connector
- Cube
- Cycloid
- Cylinder
- Cylinder3
- Dexter
- DisplacementMap
- Eclipse

- Fade Rectangle
- Filecard
- Graph
- Graph2D
- Icosahedron
- Image FX
- Noggi
- Pointer
- Polygon
- Rectangle
- Ring
- Roll
- SoftClip Draw Pixels
- Sphere
- Spline Path
- Spline Strip
- Spring
- Star
- Torus
- Trio Scroll
- Wall
- Wave

13.1.1 2D Line

The 2D Line plug-in draws a simple 2D line through given 2D coordinates, respectively point values.

⚠ **Note:** This plug-in is located in: *Built Ins -> Geom plug-ins -> Default*

This section contains the following topics:

- 2D Line Properties
- To Create 2D Lines

2D Line Properties

- **use LOD:** Enables level of detail. Range: On - Off. Default: ON
- **Enable Outline:** Enables drawing of an outline. Range: On - Off. Default: ON
- **Width:** Defines the width of the drawn line. Range: 0.0 - 500.0. Default: 5.0
- **Point values:** Lists x,y pairs separated by ':' character. x and y values are separated by blanks. Example: 0 0: 100 100: 200 70: 300 240: 400 280: 500 240: 600 400.

- **New Line:** Press before values are entered (i.e. Point values), or else the line will not be rendered visible.
- **Clear All:** Deletes all line-segments.

To Create 2D Lines

1. Enter the **Point Values**. For example: 0 0 : 200 45 entered in the Point Values field will create a line that starts at the point X1(0), Y1(0) and ends at point X2(200), Y2(45).
2. Click the **New Line** button to create the line.

13.1.2 2D Patch

The 2D Patch is a two dimensional planar curved grid of polygons. The grid is defined by control points that are located on the perimeter of the grid. The parameters of each one of the control points is the location X and Y, and the assigned texture coordinates U and V. The calculation of the internal grid points is done with a mix of a one-dimensional cubic spline and a two dimensional bilinear interpolation.

It is possible to specify the interpolation direction to be just in one direction, or in both. Depending on the shape that should be created, it is important to choose the right direction (for example it is not too difficult to create an annulus with this plug-in, after choosing the right interpolation direction along the radius).

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Geom plug-ins -> Default*

This section contains the following topics:

- 2D Patch Properties
- 2D Patch Workflow
- 2D Patch Tips
- 2D Patch Known Limitations

2D Patch Properties

- **use LOD:** Enables/disables dynamic level of detail.
- **X and Y Number of Intervals:** Sets the number of intervals between the control points in the X and Y direction respectively. Parameter name: NX/NY.
- Interpolation Direction: Defines the direction of interpolation from the control points on the perimeter to the internal grid points. Parameter name: interDirect.
- **X Size, Y Size, Reset XY (button) and Reset UV (button):** Gives initial values for the control points. The values of X and Y-Size will take affect just after hitting the Reset XY button. The Reset UV button is setting default values to the texture coordinates. Parameter name: SX/SY, resetXY/resetUV.

- **X Resolution and Y Resolution:** Defines the number of subdivisions to polygons between the control points in the X- and Y-direction respectively.Parameter name: MX/MY.
- **Interactive mode:** A toggle to enable or disable the interactive mode of direction manipulation with the mouse. Parameter name: interactive.
    - **Editing Range %:** Sets the amount of “effect” the movement of one point (move with <Shift> pressed) should have on the neighboring points. Parameter name: EditingRange.
    - **Editing Mode:** When moving a point, it is possible, by pressing <Shift>, to move the other points simultaneously. How the points move depends on the editing mode that is chosen. Parameter name: EditingMode.
    - **Constant (button):** The other points will move their positions exactly equal to the one you move.
    - **Linear (button):** The other points move in a linear way, based on the distance from the point that is being moved.
    - **Exponential (button):** The other points move in an exponential way, based on the distance from the point it is being moved.
- **Draw Control Points:** A toggle to draw or not to draw the control points. Enables the Control Point Size parameter. Parameter name: drawControlPoints.
    - **Control Point Size:** Sets the control point size.
- **Show Control Points Values**: Enables the manipulation of the control points in the rendering window. Parameter name: controlPointSize.
- **New edit mode:** When enabled, and you press down <Ctrl>, you can see the control points and also you have the opportunity to move the control points. The “Interactive mode” and the handle “script/plug-ins event mode” must be enabled.
- **X, Y, U and V:** Sets position and texture coordinates for each of the control points. These values will typically be changed in the rendering window only with the mouse.

> ⚠ **Note:** There are 12 control points that each have their own X,Y,U and V parameters.

2D Patch Workflow

1. Set the right number of control points, according to your estimation of how complex the patch is you are going to create. This number can be changed at any time. The plug-in will redistribute the new points along the already defined patch.
2. Estimate the final length and width of the patch.
3. Move the patch to the right place in your scene. If it is going to be on some horizontal plane, it might be easier first to edit it in the default orientation, and afterwards rotate it to the right place. There is no problem to continue working on it after the rotation has been made, but because of the perspective, it might be less straightforward.
4. Turn on the Draw Control Points toggle and the Interactive toggle. While working with the cursor, you do not have to be very close to the control point that you would like to manipulate. The plug-in will find by itself the nearest control point to where is your cursor is. The selected control point will get yellow color, as compared to all the rest that are white, to indicate that it has been selected for editing.

> ⚠ **Note:** You will be able to do the direct manipulation just if the object is the selected object in the Viz Artist container tree. Remember also, that regardless of the position of your cursor, if the patch is the selected object, and the interactive toggle is on, always one control point will become selected and will be changed while working with the cursor. Hitting the space key on the keyboard cancels the current selection. The cursor then returns to normal functionality until the patch object is selected again.

5. With the left button, according to the selection roles above, you can change the X and Y location of each of the control points.
6. Once the shape of the patch is more or less defined, it is a advisable to set the resolution parameter. Performance wise you should set it as low as possible, but it should be high enough to guarantee a smooth enough look.
7. Choose an image and drag it on the patch in the usual way. After doing it open the image texture editor and set the Mapping property to Vertex, and the Wrap property to Clamp.
8. The next step is to map more accurately the image on the patch. Set first the Texture Length according to your estimation, then choose the Texture Direction, and then play with the Texture Head Location to see the texture flowing along the patch. This is a good point to review again all the parameters you set before.
9. The last step is to define an animation if necessary. Typically with the patch the animation will be just on the Texture Head Location. To learn more about animation see Create Animations .

> ⚠ **Caution:** While defining an animation on the parameters of a 2D Patch, you must make sure that the right window is open in Viz Artist. Without it, the changes that you are doing with the cursor will not take effect with regard to the animation.

2D Patch Tips

- It is sometimes useful to work with 2D Patch while in wireframe mode. If you have a texture with alpha, to see it correctly, you need to turn the image off with the small enable/disable button near the image icon.
- Remember that if the 2D Patch is not selected, the interactive mode is not active. As result of this, the first click with the mouse on unselected object will be always with the normal Viz Artist functionality of the cursor, of moving, rotating and so on the object.
- If you are done with the interactive session of defining a 2D Patch, turn the **Interactive** toggle **off**.
- While working on a 2D Patch in the interactive mode, it is not possible to select another object with the cursor in the Viz Artist Scene Editor. The way to select another object is first to hit *SPACE*, and thereafter select the other object.
- Doing animation on the 2D Patch is quite expensive performance wise. Be aware to it, and try to use as much as possible a small number of control points and polygons.
- For the texture used for the 2D Patch, make sure that along the flow direction, you have on both texture edges a clean line with alpha equal to 0. Other wise you will get a wake of the image edges in front or behind it.

- Do not change the number of control point along an animation. It is possible, if necessary, to change the resolution number.
- While defining an animation, make sure to remove from it, if necessary, the toggle values of Show Control Points, Interactive Mode and so on.

2D Patch Known Limitations

- When you use 2D Patch in a scene, by choosing an object in the Viz Artist rendering window, the Viz Artist user interface is not updated automatically.
- While copying a 2D Patch, to make it work in the interactive mode, it is necessary to save the scene first.
- If the texture coordinates on the control points are getting outside the [0,1] range artifacts may show. Be aware to keep it within the [0,1] range.

13.1.3 2D Ribbon

The 2D Ribbon is a curved strip. It is ideal to make customized curves to symbolize, for example, frontiers on a map or similar. The way the object is curved is easily edited in the 2D Ribbon property editor. It is possible to enable inter activity so that counterpoints of the ribbon can be dragged with the cursor.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Geom plug-ins -> Default*

The 2D Ribbon can be used in Ribbon (1), Filled Area (2) or Double Outline (3) modes:

This section contains the following topics and procedures:

- 2D Ribbon Properties
- To Create a 2D Ribbon
- 2D Ribbon Tips
- 2D Ribbon Known Limitations

2D Ribbon Properties

The properties of the 2D Ribbon plug-in are different for each selected mode:

-
  - Ribbon Mode Properties
  - Filled Area Mode Properties
  - Double Outline Mode Properties

Ribbon Mode Properties

- **Use LOD:** Enables or disables the dynamic Level of Detail (LOD)
- **Geo**: TBA
- **Working Mode:** Available modes are Ribbon, Filled Area and Double Outline. Default is the Ribbon mode.
- **Number of Control Points:** Sets the number of control points along the ribbon (when **Slave** mode is set to **off**)
- **Resolution:** Defines the number of subdivisions to polygons between the control points.
- **Closed Curve:** Creates a closed a curve like a circle. The forward end touches the backward end.
- **Constant Width:**
  - Set to **on**: Any width modification is along all of the 2D Ribbon.
  - Set to **off**: Any width modification is around the nearest Control Point.

    ⚠ **Note:Preserve Constant Width While Possible** is not available when this is selected.

- **Add Caps:** Shows the Begin and End cap parameters when set to **on**. Removes the Begin and End caps when set to **off**, even if they are set to **on** in the two parameters:
  - **Begin Caps**: Applies a cap at the start of the 2D Ribbon when set to **on**. Also sets the resolution of the start cap.
  - **End Caps**: Applies a cap at the end of the 2D Ribbon when set to **on**. Also sets the resolution of the end cap.
- **Preserve Constant Width While Possible**: Makes the 2D Ribbon try and maintain its set width when modified (available when **Constant Width** is set to **off**) when set to **on**.
- **Interactive Mode:** Enables or disables the interactive mode of direction manipulation with the cursor:
  - **Editing Range %:** Sets the amount of 'effect' the movement of one point (move with <Shift>) should have on the neighboring points
  - **Editing Mode:** When moving a point, press <Shift> to move the other points simultaneously. How the points move depends on the editing mode that is chosen, **Constant**, **Linear** or **Exponential**
  - **Constant**: The other points will move their positions exactly equal to the one you move
  - **Linear**: The other points move in a linear way, based on the distance from the point that is being moved
  - **Exponential:** The other points move in an exponential way, based on the distance from the point it is being moved
- **Reset Options:** Sets the 2D Ribbon to initial values for the Control Points and ribbon size. Changed values of Ribbon **Length** and Ribbon **Width** take affect after ☐ has been clicked
  - ☐ **Reset**: Resets all created curves but does not delete the existing control points.
- **Texture Options:** Enables 2D Ribbon Texture options when set to **on**:
  - **Texture Direction:** Shows the texture in different directions. For example, if a pointer right-shows used as texture, the pointer shows to the right side if the **Horizontal** option is activated. Otherwise to the left side if the **-Horizontal** option is

set. Same for the **Vertical** and **-Vertical** option. The rendered pointer shows to the top or to the bottom side.
    - **Texture Head Location:** Moves the texture either to top or bottom direction
    - **Texture Length:** Stretches the texture
    - **Use Texture Factor:** Enables the texture factor parameter when set to **on**.
- **Rotate Around X Axis**: Rotates the created 2D Ribbon around the X axis when set to **on**.
- **Show Spline:** Shows the spline of the 2D Ribbon when set to **on**.
- **Use Progressive Visualization:** When set to **on** the activated parameters will affect the rendering:
    - **Progress Mode:** Slices the curve in the x-direction by length or by the control point index. Select from two options, **By Length** and **By Control Points**.
    - **Path Position:** Limits the workspace so there are less control points to prepare.
- **Render Bumps:** The curves are top-barbed which is useful for weather broadcasting.
    - **Warm**: Gives a rounded edge to the bumps.
    - **Cold**: Gives a sharpe edge to the bumps.
    - **Occluded**: Gives alternate Warm and Cold bumps.
- **Slave Mode**: Enables one ribbon to act as the master and the other as the slave. The slave ribbon uses the **Number of Control Points** and **Resolution** set for the master.
- **Show Control Points Values:** Enables the manipulation of the Control Points in the render window. X, Y, and W values set the X/Y position and the width of the curve at the different Control Points. These values are typically be changed when edited directly in the Scene Editor.

Filled Area Mode Properties

For details see Ribbon Mode Properties.

Double Outline Mode Properties

For details see Ribbon Mode Properties.

To Create a 2D Ribbon

1. Add the 2D Ribbon plug-in to the Scene Tree.
2. Set **Interactive Mode** to **on**.
3. Set the **Number of Control Points**.

> ⚠ **Note:** This number can be changed at any time. New points are distributed along the already defined ribbon (2).

4. Open **Reset Options**:
    a. Enter (estimate) the **Ribbon Length** and **Ribbon Width** (enter a new figure and press <Enter>).
    b. Click  to activate the settings.
5. Toggle the **Constant Width**, as required.

6. Position the 2D Ribbon in the Scene.

> ⚠ **Note:** If the 2D Ribbon is to be on some horizontal plane, it might be easier first to edit it in the default orientation, and afterwards rotate it to the right place. There is no problem to continue work on it after the rotation has been made, but because of the perspective, it might be less straightforward.

7. Click the (Events) button in the Scene Editor.
8. Modify the 2D Ribbon shape, either:
   - Use the mouse cursor to modify the Control Points. Click on or near the Control Point. The selected control point will show yellow (1)
   - **Left button**: Click and drag the nearest Control Point
   - **Right button**: Modify the 2D Ribbon width

   > ⚠ **Note:** If the Container, which contains the 2D Ribbon plug-in, is selected, and **Interactive Mode** is set to **on**, one Control Point is always active.

   or
   - Set **Show Control Points Values** to **on**. Modify the Control Points as required.

9. Set the **Resolution**. For best performance set the resolution as low as possible, but high enough to guarantee a smooth enough look in the corners of the 2D Ribbon, and also a smooth enough mapping of the image/texture on the 2D Ribbon.
10. Choose an image and drag it on the 2D Ribbon. Open the Image texture editor and set:
    - **Mapping** to **Vertex**
    - **Wrap** to **Clamp**
11. Map the image more accurately on the 2D Ribbon. Set **Texture Options** to **on**:
    a. Set the **Texture Length** as required.
    b. Set the **Texture Direction** as required.
    c. Set the **Texture Head Location** to see the texture flowing along the 2D Ribbon.
12. This is a good point to review again all set parameters.
13. Define an animation, if required. Typically with the 2D Ribbon the animation will be just on the Texture Head Location.

> ⚠ **IMPORTANT!** While defining an animation on the parameters of a 2D Ribbon, make sure that the right window is open in Viz Artist. Without it the changes that are made with the cursor will not take effect, with regard to the animation.

2D Ribbon Tips

- It is sometimes useful to work with 2D Ribbon with use LOD set to on.
- If you have a texture with alpha, to see it correctly set the image to off with the small enable/disable button near the image icon.
- When finished with the creation of a 2D Ribbon, make sure that **Interactive Mode** is set to off

- When the 2D Ribbon **Interactive mode** is set to on, and the Scene Editor is active, it is not possible to select another object with the cursor in the Scene Editor. To select another object make sure that **Interactive mode** is set to off and the Scene Editor is inactive
- Animation on the 2D Ribbon is quite heavy on performance. Try to keep the number of Control Points and Polygons as low as possible.
- When a texture is used with a 2D Ribbon, make sure that, along the flow direction, on the texture edges there is a clean line with alpha equal to 0. If not there will be a wake of the image edges in front or behind it.
- Do not change the number of Control Point along an animation. It is possible, if necessary, to change the resolution number.
- While defining an animation, make sure to remove from it, if necessary, the toggle values of Show Control Points, Interactive Mode and so on.

2D Ribbon Known Limitations

- When you have in the scene a 2D Ribbon, by choosing an object in the Viz Artist Scene Editor, the Viz Artist user interface is not updated automatically.
- While copying a 2D Ribbon, to make it work in the interactive mode, it is necessary to save the scene first.
- If the texture coordinates on a Control Point are close to or outside the [0,1] range, artifacts may show. Keep it within the [0,1] range.

See Also

- 2D Follow
- Create Animations

13.1.4 Alpha Map

The Alpha Map plug-in enables the creation of an alpha map from a grayscale image by translating the intensity of the grayscale image to an opacity (alpha) value. This is useful for entities which do not directly support an image as their alpha channel.

The plug-in does not accept an arbitrary image even if it only contains grayscale values. The image format must be luminance, for example, a single channel containing intensity values.

If a RGB image in dropped onto the plug-in, an error message is generated:

```
ERROR in plug-in AlphaMap: Image <8AFD1E04-B938-41D1-A743CF251028403B> not a valid luminance or alpha image
```

⚠ **Note:** This plug-in is located in: *Built Ins -> Geom plug-ins -> Default*

This section contains the following topics:

- Alpha Map Properties
- To Add an Alpha Map

Alpha Map Properties

- **Detail:** Sets the detail level of the map, the lower the value the higher the level of detail.
- **Image:** Drag the image to create the map to the drop zone. The image must be a valid alpha image.
- **X-size:** Sets the size of the map along the X-axis.
- **Y-size:** Sets the size of the map along the Y-axis.
- **Texture X Offset %:** Sets the offset of the texture in the X-axis.
- **Texture Y Offset %:** Sets the offset of the texture in the Y-axis.
- **Texture X Width %:** Stretches or compresses the texture in the X-axis.
- **Texture Y Width %:** Stretches or compresses the texture in the Y-axis.
- **Patch mode:**
  - **Triangle Strip:** Uses the same size of triangle polygon on the whole surface.
  - **Optimized Quads:** Uses bigger triangle polygons, where the surface of the displacement map is flat (reduce the total number of polygons and improve performance).
- **Inverse**: Mirror all coordinates.
- **Texture Coordinates:** Draws the texture over again when set to **Repeat** or stretches the texture to fit when the texture is to small to fit onto the Alpha Map rectangle when **Clamp** is selected.
- **Texel Precision:** Uses **Subtexel** for better texture image quality:
  - **Texel**: No sub-pixel or sub-texel corrections are made.
  - **Subtexel**: There is a limited number of pixels available on the screen, if a line does not run through a real pixel, it must be moved to the nearest one, this introduces a positional error. If **Subtexel** is selected Viz Artist breaks up pixels into smaller sub pixels in memory, so that the line can be drawn to the nearest sub pixel.
- **Smooth:** Creates a smooth look of the map, without the reduction of the polygon details of the map. In some cases the number of polygons are required to obtain the correct lightning.
- **Texture Coordinates:** Decides if texture coordinates are to be created, and to which level. Select from **None**, **Full** or **Part (partial)**. This is required if a texture is applied with a vertex mapping. Texture mapping consumes resources, if not needed select **None**.
- **Color:** Sets the color of the alpha map rectangle.

To Add an Alpha Map

1. Drag the Alpha Map plug-in to the Scene Tree.
2. Add an alpha (must be luminance only) image to the Alpha Map editor **Image** drop zone.
3. Drag a video source icon onto the alpha map's container.

13.1.5 Arrow

The Arrow built in geometry allows you to create and customize an arrow. The arrow is a 2D object. By clicking on the arrow icon in a container, the arrow editor opens. The default arrow has

two heads. At the tip of each arrowhead the reference points are placed. The left point is 1 and the right is 2.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Arrow Properties
- To Create an Arrow

Arrow Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Point 1X/1Y/1Z/2X/2Y/2Z:** Sets the point offsets.
- **Width:** Sets the width of the arrow shaft.
- **Style 1:** Select if the arrow end at point 1 should have a head or not.
- **Style 2:** Select if the arrow end at point 2 should have a head or not.
- **Arrow Width:** Sets the width of the arrow heads.
- **Arrow Length:** Sets the length of the arrow head.
- **Percent:** Scales the arrow in percent of the size defined by X-, Y- and Z-values.
- **Mode:** Defines if the arrow lies in an XY, XZ or YZ plane.

To Create an Arrow

1. Create a group and add the Arrow plug-in to it.
2. Add a material and/or an image to the same container as the arrow plug-in to add color and/or texture to it.

13.1.6 Circle

With this plug-in you can create a circle with different corner levels.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Circle Properties
- To Create a Circle

Circle Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Radius**: Sets the radius of the circle.
- **Corners**: Sets the number of corners the circle is to be constructed of. The circle is set together by tiled triangles and the higher the number of corners is set, the more triangles are used to construct the circle. The higher the value is set the more rendering performance will be used.
- **Mode:** Changes the space perspective.
  - XY (front - view)
  - XZ (bottom - view)
  - YZ (left - view)

To Create a Circle

1. Create a group and add the Circle plug-in to it.
2. Add a material and/or an image to the same container as the circle plug-in to add color and/or texture to it.
3. Set the circle radius.
4. Set the number of corners.

13.1.7 Cog Wheel

Use this plug-in to create a cog wheel. It has a range of properties to adjust the look of the cog wheel.

To avoid decreasing the system performance, set the tessellation to a lower value since it does not make much difference in quality.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Cog Wheel Properties
- To Create a Cogwheel

Cog Wheel Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Tesselation:** Sets the level of detail.

- **Corners:** Sets the number of corners the cogwheel is to be constructed of. The cog wheel gets one tooth for each corner, so the number of teeth changes correspondingly.
- **Height:** Sets the height or width of the cog wheel.
- **Tooth height:** Sets the height of the cog wheel teeth.
- **Diameter:** Sets the diameter of the cog wheel.
- **Hole:** Creates and sets the size of a hole in the cog wheel.
- **Bevel:** Sets the degree of bevel at the cog wheel.
- **Inner Bevel:** Enables or disables bevel in the cog wheel hole.
- **Show Top:** Enables or disables visualization of the top.
- **Show Bottom:** Enables or disables visualization of bottom.
- **Center:** Allows you to select where the geometrical center should be placed on the cog wheel, either **Center**, **Bottom** or **Top**.

To Create a Cogwheel

1. Add the **Cog Wheel** plug-in to a container.
2. Add a **material** to the same container.
3. Set **Hole** parameter to 50.0.
4. Set the **transformation** parameters for the container to:
    - Set **rotation** X: 90.0, Y:45.0 and Z: -45.0.

13.1.8 Cone

This plug-in creates a cone geometry.

Set the tesselation to a reasonable value since it does not make a great difference in quality.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Cone Properties
- To Create a Cone

Cone Properties

- **Use LOD:** Enables or disables dynamic level of detail.
- **Tesselation:** Sets the degree of detail.
- **Height:** Sets the height of the cone.
- **Diameter:** Sets the diameter of the cone.
- **Corners:** Allows you to decide the number of corners the cone is to be constructed of.
- **Show Bottom:** Enables or disables visualization of the cone bottom.

- **Center:** Allows you to select where the geometrical center should be placed on the cone, either Center, Bottom or Top.
- **Smooth:** Enables a smoothing of the cone edges.
- **Rounded Tip:** Enables the user to adjust the roundness of the cone's tip.
- **Rounded Tip Height:** Sets the parameter for the roundness of the tip. The angle of the cone's side does not change, the rounded tip is created by 'taking away' from the pointed tip. Therefore the cone will not have the full height.

To Create a Cone

1. Add the **Cone** plug-in to a container.
2. Add a **material** to the same container.

13.1.9 Connector

This plug-in is useful to connect two objects with a line in a required color and width.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Connector Properties
- To Connect Two Objects

Connector Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Container 1 and 2:** Placeholder for container 1 and 2. Drag the container onto the drop zone. Click **R** to remove the container.
- **Connection Point 1 and 2 By X, Y and Z Axis:** Sets the X, Y and Z-value of container 1 and 2 where the line should start. Can be set to Min, Center, Max, CenterBB or Origin.
- **Line Width:** Sets the width of the connecting line.
- **Drawing Mode**: Available options are Line and Mesh.
- **Line Color:** Defines the color of the line.

To Connect Two Objects

1. Create a new group and add the **Connector** plug-in to it.
2. Create two new group containers and add a geometry object (e.g. Cylinder and Cone) to each group.

3. Open each container's transformation editor and move the objects a part.
4. Add a material and/or an image to the geometry objects.
5. Open the connector plug-in editor and drag and drop the two containers to the Container 1 and Container 2 placeholders, respectively.
6. Adjust the color line.
7. Open the Connector plug-in editor and play with the settings.
8. Add more objects to Container 2 and animate it.

See Also

- Scene Editor Scene Editor Buttons
- BoundingBox plug-in

13.1.10 Cube

Use this plug-in to create cubes with particular widths, heights, depths and other attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Cube Properties
- To Create a Cube

Cube Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Size X**: Sets the size of the cube in the X-axis. Parameter name: size_X.
- **Size Y:** Sets the size of the cube in the Y-axis. Parameter name: size_Y.
- **Size Z**: Sets the size of the cube in the Z-axis. Parameter name: size_Z.
- **Center Y:** Sets the position of the cube center along the Y-axis. You can choose between Center, Bottom or Top. Parameter name: center_Y.
- **Tesselation:** Sets the degree of detail. Parameter name: tesselation.
- **Adaptive Tesselation**: Adjusts tesselation on each axis separately, depending on the extension of the cube on that axis. The reference value is an extension of 100. For a face of size 200 the tesselation-parameter will be doubled, for a face of size 50 the tesselation is halved. The automatically adapted tesselation still never exceeds its maximum value of 100. Parameter name: adaptiveTesselation.
- **Bevel:** Adds a bevel of the given size to corners and edges of the cube. The size of the bevel reduces the size of the cube's axis-aligned faces which: in case 'Adaptive tesselation' is enabled: will cause them to become less tesselated accordingly. Tesselation of the bevel itself is affected by 'adaptive tesselation' too. Parameter name: bevel.

- **Rounded Bevel:** If enabled the beveled edges and corners will show 'rounded'. How much the round bevel is tesselated will be affected by 'adaptive tesselation'. Parameter name: roundedBevel.
- **Show Top, Bottom, Front, Back, Left, Right:** Shows or hides the cube's face and adjacent corners and edges. Parameter names: Top, Bottom, Front, Back, Left and Right.
- **Show Size 0:** Enables/disables display of the cube at axis value zero (0). If one of the axis values are set to 0 the cube will remain visible if the setting is enabled (On). Disabling (Off) it will make the cube become invisible as long as one of the axis values are 0. A situation where this parameter is useful is when creating bar chart animations. In such animations it would then be possible to hide the bar at value 0 by disabling this setting.

To Create a Cube

1. Create a new group container and add the Cube plug-in to it.
2. Add a material and/or an image to the same container.
3. Open the cube editor and set Rotation Y and Z values to 45.0.

13.1.11 Cycloid

Use this plug-in to create a Cycloid; The path traced by a point on a wheel as the wheel rolls, without slipping, along a flat surface.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Cycloid Properties
- To Create a Cycloid

Cycloid Properties

- **Shape**
  - **EpiCycloid**: A plane curve produced by tracing the path of a chosen point of a circle which rolls without slipping around a fixed circle
  - **Hypo Cycloid:** A special plane curve generated by the trace of a fixed point on a small circle that rolls within a larger circle.
- **Corners:** Sets the number of corners the Cycloid is to be made up of.
- **Hole:** Creates a hole in the Cycloid.
- **Radius:** Sets the radius of the hole.
- **Factor Outer:** Sets the outer factor of the Cycloid.
- **Factor Inner:** Sets the inner factor of the Cycloid (when **Hole** is set to On).

To Create a Cycloid

1. Create a new container.
2. Add the Cycloid plug-in to it (1 (default Hypo Cycloid)).
3. Open the Cycloid editor.
4. Modify the Cycloid parameters, for example (2).
    - **Corners:** 8
    - **Hole:** On
    - **Radius:** 20
    - **Factor Outer:** 10
    - **Factor Inner:** 10
    - **Slice**: 5
5. If required, add a material and/or a texture.

13.1.12 Cylinder

Use this plug-in to create cylinders with different heights, widths and depths and other attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Cylinder Properties
- To Create a Cylinder

Cylinder Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Height:** Sets the height of the cylinder.
- **Diameter:** Sets the diameter of the cylinder
- **Rotation:** Sets the rotation around X-axis.
- **Angle**: Sets the opening angle from 0-360˚. As the value decreases below 360˚ an increasing angle will open in the cylinder.
- **Corners:** Sets the number of corners the cylinder is to be made up from.
- **Hole:** Creates a hole in the cylinder, making it into a tube.
- **Center:** Sets the position for the center. You can choose between Center, Bottom or Top.
- **Bevel:** Sets the size of bevel at the cylinder.
- **Inner Bevel:** Enables/disables bevel in the cylinder hole if bevel is set at the bevel parameter.
- **Show Top:** Turns off/on visualization of cylinder top.
- **Show Bottom**: Turns off/on visualization of cylinder bottom.

- **Show size 0:** Enables/disables display of the cylinder at height value 0. If you set height to zero, the cylinder is visible by default. If you disable this option, the cylinder becomes invisible. If you make a pie chart visualization you will have use for this option.

To Create a Cylinder

1. Create a new group container.
2. Add the Cylinder or Cylinder3 plug-in.
3. Open the Cylinder editor.
4. Set Hole to 50.0.
5. Add a material and/or an image to the same Container.
6. Open the cylinder editor.
7. Set Rotation Y to 75.0 and Z to 90.0.

13.1.13 Cylinder3

Use this plug-in to create cylinders. Compared to the Cylinder plug-in, Cylinder3 provides the designer with more advanced options and settings.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Cylinder3 Properties
- To Create a Cylinder

Cylinder3 Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Height:** Sets the height of the cylinder.
- **Diameter:** Sets the diameter of the cylinder
- **Rotation:** Sets the rotation around X-axis.
- **Angle:** Sets the opening angle from 0-360˚. As the value decreases below 360˚ an increasing angle will open in the cylinder.
- **Opening Mode:** Describes in which direction the cylinder will be opened when adjusting the Angle parameter:
  - **CW:** Opens the cylinder in a clockwise direction.
  - **CCW:** Opens the cylinder in a counter-clockwise direction.
  - **Center:** Opens the cylinder in clockwise and counter-clockwise directions simultaneously.
- **Corners:** Adjusts the resolution of the cylinder by setting the number of corners the cylinder is to be made up from.

- **Hole:** Creates a hole in the cylinder, creating a tube.
- **Center:** Sets the center position of the cylinder to **Center**, **Bottom** or **Top**.
- **Bevel:** Adjusts the bevel size of the cylinder.
- **Inner Bevel:** Enables or disables bevel in the cylinder hole if a value is given for the bevel parameter, even if **Hole** is set to `0.0`. When creating a cylinder with an outer bevel and flat top and bottom surfaces, this setting must be set to `Off`.
- **Show Top:** Turns visualization of the cylinder top surface on or off.
- **Show Bottom:** Turns visualization of cylinder bottom surface on or off.
- **Show size 0:** Enables or disables display of the cylinder when the value of **Height** is set to `0`. By default, the cylinder is visible when Height is set to `0`. By disabling this option, the cylinder becomes invisible. This is useful when designing pie chart visualizations.
- **Show inner bevel:** If this toggle is off, the inner bevel, respectively the inner cylinder is not rendered.
- **Show outer bevel:** If this toggle is off, the outer bevel, respectively the outer cylinder is not rendered.
- **Show caps:** Applicable only to cylinders with an opening angle. This setting determines whether a surface, or "cap", should be added to the angle's sides. This setting only affects the geometry object if the angle is less than 360°. In combination with the **Back Face** setting of the Expert container plug-in, this setting allows for the design of cylinders with a cut out section exposing the cylinder's inside.
- **Static texture:** If the texture mapping of the texture is set to **Vertex** mode, this setting defines whether or not the texture will bend around the object.
- **Texture anchor:** Changes the texture orientation in **X**-direction to left or right. Requires the angle to be less than 360°.

To Create a Cylinder

1. Create a new group container.
2. Add the Cylinder3 plug-in and open the Cylinder plug-in editor.
3. Add a material or an image to the Container.
4. Open the Cylinder plug-in editor and set the desired parameters.
5. Use the transformation editor to set the container's position and rotation.

13.1.14 Dexter

Dexter is a macro-based text-ticker tool that generates static horizontal text crawl and vertical text rolls. The Dexter editor is organized in five sections, Control, Design, Text, Trigger and Stopper. When adding the plug-in, two sub-containers are also added; Animation (visible) and Templates (hidden). In addition, it has a set of scripting possibilities.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Common Parameters
- Control Parameters
- Design Parameters
- Text Parameters
- Trigger Parameters
- Stopper Parameters
- Dexter Parameters
- Script Programming Parameters
- Script Programming Tokens
- Script Programming Syntax
- Script Programming Templates
- Script Programming Newline Logic
- Script Programming Triggers

Common Parameters

These buttons are found top right of the Dexter editor and are not active in control mode:

- **Control**: See Control Parameters.
- **Design**: See Design Parameters.
- **Text**: See Text Parameters.
- **Trigger**: See Trigger Parameters.
- **Stopper**: See Stopper Parameters.
- **Info:** Shows the delimited borders defined by the mark settings.
- **Update:** Applies the changes on the already existing containers.
- **Build:** Rebuild all objects.
- **Clear:** Clear the text objects.

Control Parameters

Use the Control section to start, stop, continue and reset the scroll/roll. It also contains buttons to select the velocity mode and speed of the ticker.

- **Start** starts the animation.
- **Stop** stops the animation.
- **Reset** resets the animation.
- **Cont** continues the animation.
- **Loop** loops the scroller/crawl.
  The velocity control allows you to define how the speed of the crawl/roll is to be calculated.
- **Duration (all groups same)** makes all group use the same amount of time to crawl/scroll from start to end. The duration will be measured at the group containing the longest text string.

- **Duration (for each group)** makes all groups use exactly the same time. As a consequence, longer text strings will scroll faster than shorter ones.
- **Speed** makes all groups scroll at the same speed, as a consequence long message will take longer time to scroll than shorter messages. The speed unit is pixels per frame divided by 10.

Design Parameters

In the design section of the Dexter editor set a range of parameters to how the text will show:

- **Screen width:** Sets the width of the screen used for placing the text objects and culling. Possible values range from 20 to 4000.
- **Screen height**: Sets the height of the screen used for placing the text objects and culling.
- **Loop kerning:** Defines the spacing between two loops. Possible values range from 0 to 1000.
- **Group kerning**: Sets the vertical distance between group items.
- **Crawl mode**: Creates a normal horizontal scrolling text.
- **Roll mode**: Creates a vertical scroll.
- **Default Font:** Shows the default font. To load a font, drag it from the Server Panel and onto the font drop zone. To unload a font, simply drag it to the trashcan. This must be set to get any text unless you specify a font in the macro and all text are executed with macros.
- **Autosplit text**: If you have a long text string written in one line, without the auto-split feature, this is built as a big text object. This is not good for culling and performance. If you activate auto-split, long text objects will be split up into smaller ones which is better for performance.
- **Corr. val.:** Every time the auto-split feature splits a text object, the correction value is added to the gap between the two new objects. Although it reads the length of a space character from the font to make a proper placing of the objects there is a slight difference in length between a split and a non-split text. Therefore, the correction value is inserted to correct text that does not look good because of incorrect placing when the objects are split.
- **Min. object length:** Sets the minimum object length for an auto-split object.
- **Col1**: Allows you to set the default color 1. To set it, either drag a material from the Server Panel onto the col1 icon, or use the color editor below.
- **Col2**: Allows you to set the default color 2. To set it, either drag a material from the Server Panel onto the col1 icon, or use the color editor below.
- **Col3**: Allows you to set the default color 3. To set it, either drag a material from the Server Panel onto the col1 icon, or use the color editor below.

Text Parameters

The Text Editor contains the text area where you enter the script commands. It also contains a list of all tokens and parameters. By clicking on a token or parameter, you automatically insert it into the script text at cursor position.

- **Space=Text:** Forces the parser to interpret spaces at the beginning of a line/after a token has ended as legal text (in its default state those spaces are abandoned).

- **LoadFile**: Allows you to browse for a script file. The default extension for the file is `.edx`. The script file must contain a macro script with standard syntax. When the file loads, the contents are written onto the text field.

Trigger Parameters

- **Act.:** If this toggle is on, the marker will be evaluated.
- **Mark:** The name of the marker for orientation.
- **Position:** Here you can set the marker position where an event should occur.
- **Command:** You can type in here required Viz commands.
  The concept of marks and triggers allows you to send commands to Viz Artist/Engine when the animation reaches a certain position. To make it work you need to define two items:

First of all, mark the trigger positions in the text crawl/roll. This is done using the token `<MARK>`.

Animation starts<MARK A>here.

There are six marks, named A, B, C, D, E and F, that can be put wherever desired in the text.
If **Info** is enabled, a representation of the mark will be visible on-screen.

In addition to the six user-definable marks, there are two marks that are created automatically, named BEGIN and END. They are located at the beginning and end of the crawl/roll. Please keep in mind that the END mark is located before all the containers created for the loop feature and that marks are not looped.

After setting marks and building the crawl/roll, trigger actions must be specified. All eight marks are listed in the trigger section of the editor. The **on**/**off** switch left to the mark name is used to set that mark active/inactive. The position value right to the mark name sets the position where the triggering should occur (value `0` is at the end of Dexter's bounding box. When animation is running and the mark in the text reaches this position, the specified command is triggered. If you switch on **Info** you will see a line representing the trigger position. Simply switch it on and watch the marks and lines interacting to see how this works. Right to the trigger position you will find a text field where you can enter the command which should be sent to Viz Artist when a mark triggers. You may use every command available via external control, for example you may change values in Viz Artist, start animations, execute plug-ins, and so on.

First you may issue two or more commands when a mark reaches its trigger position. Simply split them using a ; character (semi-colon). Also you may use the DEXTER location to send pseudo-commands to Dexter.

Currently two such pseudo-commands are implemented: The first one is DEXTER*STOP, which stops Dexter's animation, the second one is DEXTER*PAUSE X, where Dexter will stop the animation for X seconds (of course this is a floating point value, so DEXTER*PAUSE 0.5 is also possible) and continue with the same speed as before.

⚠ **Note:** These two pseudo-commands are not supported by Viz Artist command interface; they are only available using Dexter's trigger feature. Dexter supports the THIS_SCENE pseudo location as well; you may want to use this to refer to objects/animations in the local scene.

Stopper Parameters

- **Enable:** If this toggle is on, the markers will be evaluated.
- **Use Mark:** If this toggle is switched to `on` the given positions will be used.
- **Position:** Sets the position of the `Ease-Out` and `Ease-In` range.
- **Ease Out:** The text object is leaving this position during the movement.
- **Continue:** This parameter has two options. `Wait for Cont` and `Wait Nr Fields`. In first case the text object waits until an `continue` Viz command is being sent. In second case the text object waits for the number of fields. Consider that two fields comply one frame.
- **Ease In**: The text object slows down and waits for a continue signal or until the number of fields are reached.
- **Global:** Stops global or local to a certain group.
  The Stopper is an extension to the triggering ability of Dexter. It allows you to create ease in/out animations at defined points along the crawl/roll.First you need to create a set of marks using the <MARK> token to define the stop points in your text. Then open the Stopper page in the Dexter editor and check the **Enable** button.

Select which marks to use. If you have defined your stop points using the <MARK C> token you need to select C here. If you enable **Info** you will see a representation of the animation curve in the editor. Using the position, ease in and ease out slider you can move and alter that animation like you need. (Please keep in mind that the ease out part of the animation is executed before the ease in part: although usually it is the other way round. But here we do not want to start an animation smoothly and then stop it, we want to stop it first and then restart it again).

Using the continue switch you can select what Dexter should do after the animation has stopped: Either wait a certain number of fields or wait until the Continue button in the Control section is pressed. If you have more than one group in crawl mode you may want to check the **Global** as well. If it is checked, all groups will be halted as soon as a marker in any group reaches the stopper position. If it is unchecked only the group that contains this marker is stopped.

Dexter Parameters

Name	Type	Function
text	string	This is the script Dexter uses for building the text.
screenw	float	Screen width (used for placing the text objects and culling) 20/4000.
screenh	float	Screen height (used for placing the text objects and culling) 20/4000.
time	float	Animation length (in seconds) or animation speed 1/500.

Name	Type	Function
file	string	File name (default extension .*).
loopmode	bool	Loop on or off.
loopkern	float	Defines the spacing between two loops (global) 0/1000.
groupkern	float	Defines the spacing between two groups (global) 0/1000.
showarea	bool	View the culling area on or off.
doautosplit	bool	If on, it splits text objects into pieces.
autosplitcorr	float	Correction value for autosplit feature -1e6/+1e6.
autosplitmin	int	Minimum number of characters for split text objects 0/100000.
time-line	float	Value for preview control (works if animation is stopped) 0/1000.
deffont	string	Default font (e.g. Peak/AvantGarde/AvantGarde-Book).
col1	unsigned long	Default color value.
col2	unsigned long	Color value #2.
col3	unsigned long	Color value #3.
durmode	int	Selects animation mode (duration/speed) 0/2.
roll	bool	Switches between vertical (0) and horizontal (1) animation.
trigonbegin	bool	Turns on/off trigger BEGIN (see 2.6).
trigona	bool	Turns on/off trigger A.
trigonb	bool	Turns on/off trigger B.
trigonc	bool	Turns on/off trigger C.

Name	Type	Function
trigond	bool	Turns on/off trigger D.
trigone	bool	Turns on/off trigger E.
trigonf	bool	Turns on/off trigger F.
trigonend	bool	Turns on/off trigger END.
trigposbegin	float	Position of trigger BEGIN -100000/100000.
trigposa	float	Position of trigger A -100000/100000.
trigposb	float	Position of trigger B -100000/100000.
trigposc	float	Position of trigger C -100000/100000.
trigposd	float	Position of trigger D -100000/100000.
trigpose	float	Position of trigger E -100000/100000.
trigposf	float	Position of trigger F -100000/100000.
trigposend	float	Position of trigger END -100000/100000.
trigcombegin	string	Command for trigger BEGIN.
trigcoma	string	Command for trigger A.
trigcomb	string	Command for trigger B.
trigcomc	string	Command for trigger C.
trigcomd	string	Command for trigger D.
trigcome	string	Command for trigger E.
trigcomf	string	Command for trigger F.
trigcomend	string	Command for trigger END.

Name	Type	Function
status	bool	While animation is running, this parameter is set to 1 (read-only).

Boolean values can either be 1 (on) or 0 (off). Color values are 4 byte unsigned integers, byte 0 = red, 1 = green, 2 = blue, 3 = alpha.

Script Programming Parameters

Name	Default value	Function
AnchorX, AnchorY, AnchorZ	0	X, Y and Z position of a new text object.
AnimDirection	Default	Animation direction, set to anything but “Default” to reverse animation
AutoSplitCorr	0	Correction value for autosplit feature
AutoSplitMin	20	Minimum number of characters for split text objects
Col1, Col2, Col3	255/255/255 /255	Holds the color value from the gui/external control (parameter Col1)
Command	-	Sends a command to Viz Artist/Engine.
Detail	Auto	Sets detail/fontstyle of text objects (see above).
DoAutoSplit	0	Enables/disables auto-split feature.
Enlighted	0	Set to 1 if you want enlighted text.
FontStyle	-	Sets the font used to create text objects (e.g. Vizrt/ AvantGarde/AvantGarde-Book).
GroupKerning	-	Sets horizontal spacing between groups (like Jump).
Jump	0	Sets horizontal spacing between groups.
LastContainer	-	Holds the container number of the last created container.
LastX, LastY	0	If a new text object is created, the last X and Yposition will be stored here.

Name	Default value	Function
MarkOffsX, MarkOffsY	0	X and Y offset for marks (see Script Programming Tokens).
Material	255/255/255/255	The material used for creating text objects.
ObjectKerning	5	Sets spacing between two text objects.
RollAlign	C	Align groups in roll mode. values are R/r (right), L/l (left) and C/c (center).
RollBorder	0	Sets left/right border for groups (if aligned left or right in roll mode).
Scaling	20	Sets scaling in percent (20 means a scaling of 0.2).
ShadowDirection	320	Direction of shadow.
ShadowDistance	10	Distance of shadow
ShadowMaterial	204/204/204/204	Material of shadow.
ShadowSharpen	0	Controls **Sharpen** parameter for the shadow of the selected font.
ShadowZOffset	-1	Z offset of shadow.
Sharpen	0	Control **Sharpen** parameter of selected font.
SoftShadowLevel	1	Level of soft shadow (1-4).
Step	15	Sets the amount AnchorX is increased on a newline (see newline logic).
Template	-	Sets template to be used for creating text objects (see Script Programming: Templates).
TextKerning	0	Adjusts text kerning (in text object).
UseShadow	0	Set to 1 to enable shadow.

Name	Default value	Function
UseSoftShadow	0	Set to 1 to enable soft shadow.
WordSpacing	0	Adjusts word spacing (in text object).
XScaling, YScaling, ZScaling	20	Scaling value for X, Y and Z axes.

Color values are described in the form r/g/b/a with decimal numbers (base 10). Col1 is also used as the default color value (Material is initialized with this value). Material can also hold an existing material in the material pool, e.g. “Vizrt/Artdeco/artdeco.0” (see **Col1, Col2, Col3** in Script Programming Parameters).

Detail can have one of the following values: 1, 2, 3, 4, 5, 6 or T. 1 through 6 are detail levels, T means texture font. Every other value (like the default “Auto” will be interpreted as detail level auto) (see **Detail** in Script Programming Parameters).

Script Programming Tokens

Name	Syntax	Function
ADD	<ADD Parameter Value>	Adds Value to Parameter (see example below).
CLEAR	<CLEAR Parameter>	Clears Parameter (sets it to "").
CONTAINER	<CONTAINER Path>	Inserts a container into the text (see example below).
ENDMAC	<ENDMAC>	Ends macro definition (nested macros are not allowed).
EXEC	<EXEC Name>	Executes the macro Name.
GROUP	<GROUP>	Begins a new group of containers.
IMAGE	<IMAGE name>	Inserts an image in the text.
MACRO	<MACRO Name>	Defines the macro Name.
MARK	<MARK Type>	Sets a mark (see Script Programming Triggers).
MAX	<MAX Parameter Value>	If Value is higher than Parameter then Parameter = Value.
MIN	<MIN Parameter Value>	Works like MAX but takes the smaller value.

Name	Syntax	Function
NEWLINE	<NEWLINE>	See Script Programming Newline Logic.
NOTEXT	<NOTEXT>	See Script Programming Newline Logic.
SET	<SET Parameter Value>	Sets Parameter to Value.

The ADD token adds either a string or a float to one. If either Parameter or Value are strings, the two strings can be combined to one. If both are floats, an arithmetic addition will be done.

For example:

```
<SET Test Vi><ADD Test zrt>
```

results in “Vizrt”, but:

```
<SET Test 5><ADD Test 6.3>
```

results in Test holding the value 11.3.

The CONTAINER token is used to define a container path for inserting a container into the text. A container path may either look like “1/3/2/3/1” (the same system as used in the GUI), “$Containername” or “#Containernumber”.

```
<CONTAINER 1/3/2/4> <CONTAINER $$Sphere> (or: <CONTAINER “$Sphere”>) <CONTAINER #307>
```

The MARK token sets a mark that can be A, B, C, D, E or F.

The GROUP token begins a new group of containers. Instead of writing <GROUP> you may leave one line empty as this has the same effect. Use groups for making lines in roll mode.

Script Programming Syntax

The syntax consists of tokens and text. <tokens> are surrounded by <>, e.g. <ENDMAC>. Spaces are used to separate tokens and arguments. Text is everything that is not bracketed. Tokens can be everywhere in the text, e.g. Viz is <SET Material 234/34/58/114>great is valid. You do not need to write every token in a new line. Nested tokens are not allowed. A macro cannot execute itself.

In addition to the tokens, there is the escape character **$**:

- **$Parameter** inserts the value of Parameter in the text
- **$_** inserts a space character in the text (for use in tokens)
- **$<** inserts a < character
- **$>** inserts a > character
- **$$** inserts a $ character
  The $ character can be placed in tokens and in the text, e.g. $<Dexter$> says:

```
<SET Text Hello><ADD Text $_World>$Text
```

Inside tokens you may use double quote characters to insert text.

```
<SET Text “This is a text”>
```

Any $ characters and spaces within two double quote characters are ignored from parsing and executing. Two double quote characters are interpreted as one:

```
 <SET Name ""Harry"">
```

Script Programming Templates

You can define templates for the creation of text objects. Under the Dexter container you will find a container named “Templates”. Under this container you can create a set of template containers. These can hold animations, plug-ins, key, alpha, etc. Give every template container a unique name. To access it in the script, you need to set the Template parameter to the name of the template you want to use:

```
<SET Template AlphaKey1>
```

After you set the Template parameter, Dexter will use this template container as basis for every text object. To switch this behavior off set Template to its default value (which is ""):

```
<CLEAR Template>
```

⚠ **Caution:** Certain parameters (like material) are overwritten by Dexter after the container has been created from the template.

Script Programming Newline Logic

The token <NEWLINE> (see Script Programming Tokens) describes a new text object, which is separated from the previous one. Every newline moves AnchorX (see **AnchorX, AnchorY, AnchorZ** in Script Programming Parameters) a bit further, depending on the value of **Step**. This token will be inserted internally to prevent you from typing <NEWLINE> over and over again.

The following text

```
One Two
```

is converted to

```
One <NEWLINE> Two
```

No newlines:

- From the beginning of the script until the first occurrence of text
- After a line that doesn't contain text (if the line is empty a new group will be created)
- After the tokens MACRO, ENDMAC and NOTEXT, and the last newline before those tokens will be deleted
  For example:

Lines	Explanation
No new lines because no text occurred.	
<SET Camera Bobby>	No text, no new line.
Return of the	Here is the first new line.
Killer Tomatoes II	No new line because of token MACRO in the next line. From here on there are no new lines until the end of the MACRO.
<MACRO SetDir>	
<SET Director Jimmy>	
<ENDMAC>	
<EXEC SetDir>	No new line because there is no text in the line.
Director:	New line.
$Director	New line.
<SET FontStyle Arial>	No new line because there is no text in the line.
Camera:	New line.
$Camera	
<NOTEXT>	The <NOTEXT> token will not add new lines before the next text item occurs (see Script Programming: Tokens).
x	New line.

Lines	Explanation
No new lines because no text occurred.	
y	New line.
<SET Material $Col2>	No new line because there is no text in the line.
z	New line.
1<NEWLINE>2	New line for 1 and 2.

The macro SetDir in the example above can be used as follows without creating a newline:

```
Hello<EXEC SetDir>World
```

The same rules applies to automatically inserted **GROUP** tokens (see Script Programming Tokens).

Script Programming Triggers

The concept of marks and triggers allows you to send commands to Viz Artist/Engine when the animation reaches a certain position. Therefore you need to define two items:

- First is setting a mark in the text. You have six different marks to place wherever you think they are useful, together with the automatically created BEGIN and END mark this gives you a total of eight marks.
- The second items are the triggers, where you have as well eight different triggers. While marks are set with a token in the text, triggers can be defined in the trigger section of the Dexter editor or using external control. You can define a position and a command for every trigger. When the corresponding mark reaches the trigger position, the command is issued to Viz Artist/Engine.

  > ⚠ **Note:** Only marks and triggers of the same type work together! If for example mark B reaches the position of trigger D, nothing will happen.

  Marks are set using the MARK token, for example: Animation starts exactly{{<MARK D>}} here!

User defined marks are numbered A through F and are color coded in the trigger editor and as graphical symbols when show area is turned on. In addition to the user defined marks there are the automatically created marks BEGIN and END, which are inserted at the beginning and ending of the crawl/roll. The commands being issued do not have to be numbered, this is optional. If you do not give them a number, -1 is added before the command is sent to Viz Artist/Engine. you can as well define more than one command, separate them by a semicolon character.

You may invoke two internal commands using the trigger function with the `DEXTER*` location: stop and pause.

```
DEXTER*STOP DEXTER*PAUSE X
```

⚠ **Note:** X is the time in seconds.

13.1.15 DisplacementMap

This plug-in enables you to create topographical (height) maps by means of a greyscale image. The intensity value of the gray scale image is translated to a height value.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- DisplacementMap Properties
- To Create a Displacement Map

DisplacementMap Properties

- **Detail:** Sets the detail level of the map, the lower the value the higher the level of detail.
- **Image:** Drag the image you want to use creating the map, onto the placeholder. The image must be a valid luminance or alpha image.
- **X-size:** Sets the size of the map along the X-axis.
- **Y-size:** Sets the size of the map along the Y-axes.
- **Height:** Sets the height of the map.
- **Texture X Offset %:** Sets the offset of the texture in the X-axis.
- **Texture Y Offset %:** Sets the offset of the texture in the Y-axis.
- **Texture X Width %**: Stretches or compresses the texture in the X-axis.
- **Texture Y Width %:** Stretches or compresses the texture in the Y-axis.
- **Patch Mode**
  - **Triangle Strip**: Uses the same size of triangle polygon on the whole surface.
  - **Optimized Quads:** Uses bigger triangle polygons where the surface of the displacement map is flat, thereby reducing the total number of polygons and improving performance.
- **Inverse:** Mirrors all topographic coordinates through the zero level. This will make a mountain top to a crater.
- **Texture Coordinates:** Repeat/Clamp sets if the texture is the be repeated or clamped if it is too small to fit onto the displacement maps rectangle. Clamp stretches the texture to make it fit, Repeat starts drawing the texture over again when it reaches the end.
- **Texel Precision**
  - **Texel:** No subpixel/subtexel correction is made.

    - **Subtexel**: There is a limited number of pixels available on the screen, if a line does not run through a real pixel, it must be moved to the nearest one, this introduces a positional error. If subtexel is selected Viz Artist will break up pixels into smaller sub pixels in memory so that the line can be drawn to the nearest sub pixel.
- **Smooth:** Use this parameter to smooth the look of the map, without reducing the polygon details of the map. In some cases you need the number of polygons you have to obtain the correct lightning.
- **Texture Coordinates:** Decides if texture coordinates are to be created, and to which level, either **None**, **Full** or **Partially**. You need this enabled to some level, if a texture is applied using a vertex mapping. Texture mapping is resource consuming, so if you do not need it, keep it off.

> ⚠ **Note:** Please be aware not to increase to number of polygons in detail not too much, as this is a common mistake.

To Create a Displacement Map

1. Create a new group and add the displacement plug-in to it.
2. Add material to the group container.
3. Open the displacement plug-in editor, and drag and drop a gray-scaled image onto the image placeholder.
4. Open the group container's transformation editor and adjust the Rotation values.

> ⚠ **Note:** An RGB or similar image does not work, and you will get an information in the log field. The image must be a valid luminance or alpha image.

13.1.16 Eclipse

This plug-in enables you to create an eclipse shape.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Eclipse Properties
- To Create an Eclipse

Eclipse Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Corners**: Changes the number of triangles the object has. If you want a good quality increase the value. A value of 40 is a reasonably value.
- **Inner Radius 1 and 2:** Changes the inner radius in X-Position (1) and the inner radius in Y-Position (2).
- **Outer Radius 1 and 2**: Changes the outer radius in X-Position (1) and the outer radius in Y-Position (2).
- **Hole Offset X, Y and Z**: Moves the inner hole to X, Y, Z-position. With this parameter you can shape great objects like a volcano.
- **Rotation:** Rotates the eclipse. The result is visible if you change the angle less than 360°.
- **Angle:** Change this to create a view like a cake respectively a divided circle.
- **Create Uniform Normals:** Adapts the varying normals to the majority of normals in same direction.

To Create an Eclipse

1. Create a new group and add the eclipse plug-in to it.
2. Add material to the group container.
3. Open the eclipse plug-in editor, and adjust the following values:
    - Set Inner Radius 1 and 2 to 40.0
    - Set Hole Offset X to 20.0
4. Open the group container's transformation editor and adjust the Rotation values.

13.1.17 Fade Rectangle

The Fade Rectangle's four sides can be set to fade in a way specified by the user. Both the degree of fading an the area influenced by the fading can be customized.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Fade Rectangle Properties
- To Create a Faded Rectangle

Fade Rectangle Properties

- **Use LOD**: Enables/disables dynamic level of detail.
- **Tesselation:** Sets the degree of detail.
- **Width:** Sets the width of the fade rectangle.
- **Height**: Sets the height of the fade rectangle.

- **Lock Blend**: Allows you to decide whether to adjust the blending looked for both the X- and Y-axis or single (each axis separately).
- **Blend Range**: Allows you to set the blend range if looked mode is selected.
- **Blend Range X and Y:** Sets the blend range of the X and/or Y axis if single blend is selected.
- **Blend Factor:** Sets the alpha ramp of the fade rectangle. To see the effect set the alpha value to 0 and adjust the blend factor.
- **Color:** Shows the color of the rectangle. Set the color either in the editor below or drag a material from the Server Panel onto the color icon.

To Create a Faded Rectangle

1. Create a new group and add the fade rectangle plug-in onto it.
2. Add an image to this group.
3. Open the fade rectangle editor and change the required color through the color parameter and adjust the width and the height.

13.1.18 Filecard

Use this plug-in to create file cards with different widths and heights and other attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Filecard Properties
- To Create a Filecard

Filecard Properties

- **use LOD:** Enables/disables dynamic level of detail.
- **Width**: Sets the width of the file card.
- **Height:** Sets the height of the file card.
- **Tab Position X and Y:** Sets the tab position in x-direction and y-direction.
- **Tab Angle:** Reduces/increases the size of the curve.
- **Bevel:** Reduces/increases edge softness. Increase this value to get a softer edge.
- **Tesselation:** Changes the number of triangles of the rendered file card to increase the visual quality. Please consider that the number of triangles will affect the performance of the system.

To Create a Filecard

1. Create a new group and add the file card plug-in to it.
2. Add a material and/or a texture to the group.
3. Open the file card editor and set the following parameters:
    - Set Width to 700.0
    - Set Tab Position X to 70.0, and Tab Position Y to 60.0
    - Set Tab Angle to 45.0
    - Set Bevel to 25.0

13.1.19 Graph

The Graph object allows you to create a 2D or 3D graph with up to 50 values.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Graph Properties
- To Create a Graph

Graph Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Width**: Sets the width of the graph.
- **Height:** Sets the height of the graph.
- **Start**: Sets the starting point of the graph.
- **End:** Sets the ending point of the graph.
- **Line Width:** Defines the line width.
- **1st/last segment interpolation:** Switches the interpolation of the first and last segment on or off.
- **3D:** Enables/disables 3D visualization of the graph.
- **Extrusion depth**: Sets the graph's extrusion depth.
- **Add backside:** Enables/disables visualization of the graph backside.
- **Use vertex color:** Enables a base color for the graph. Set the values in the color editor or drag a material onto the small square below the parameter.
- **Line color:** Enables the use of a vertex color.
- **X0/Y0 to X49/Y49:** Enables up to 50 different value points of the graph by giving coordinates on the X- and Y-axis.

To Create a Graph

1. Create a new group and add the graph plug-in to it.
2. Add a material and/or a texture to the group, or open the graph editor and enable Use Vertexcolor and set the Line Color.
3. Open the graph editor and set the Start and End points.
4. Enable 3D and set Extrusion Depth to 30.0.
5. Add points for the X- and Y-directions (i.e. X0/Y0, X1/Y1 and so on).

13.1.20 Graph2D

Use this plug-in to create 2D and 3D graphs, lines and bars from imported files.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Graph2D Properties
- To Create a Channel File for a Graph2D
- To Create a 2D Graph

Graph2D Properties

- **use LOD:** Renders higher or lower detail for the object. This setting depends on the camera/object distance, and renders with marginal quality casualties if this toggle is switched on. LOD stands for Level of Detail.
- **Channel:** Affects the Channel File to search for. The file holds the vertex and other data to define the whole graph.
- **Use:** Loads the file through a Channel File or remote access.
- **Max (%):** Sets the rage for rendering. For example, if the range is set to 50% the graph will be cut 50% in width.
- **3D:** Enables 3D rendering of the graph, and the **Show Back** option.
  - **Show Back:** Enables the object to show the back in case the object will be rotated.
- **Type:** Following types are available: Graph, Line and Bars. Take a change if you want another visual depiction. **Graph** enables the Reduction and Minimum Points settings. **Line** enables the Constant Width, Line Width (%), Reduction (%) and Minimum Points settings. Bars enables the Bars Width (%) and Bar Animation settings.
  - **Reduction (%):** Reduces points by the Douglas-Peucker algorithm. The Douglas-Peucker algorithm is an algorithm for reducing the number of points in a curve that is approximated by a series of points.
  - **Minimum Points:** Changes the number of points for rendering the graph.

- **Constant Width:** Sets a constant width.
- **Line Width (%):** Changes the width of the line.
- **Reduction (%):** Reduces points by the Douglas-Peucker algorithm. The Douglas-Peucker algorithm is an algorithm for reducing the number of points in a curve that is approximated by a series of points.
- **Minimum Points:** Changes the number of points for rendering the graph.
- **Bar Width (%):** Changes the width of all bars.
- **Bar Animation:** Renders each bar separately when **One by One** is selected. Renders all bars at the same time when **All together** is selected.

- **Use Colors:** Sets the color for positive and negative values. This setting is only available for graphs and bars (not line).
- **Show Frame:** Shows the frame around the graph. Useful for orientation.
- **Aspect Ratio:** Stretches the graph evenly in x-direction.
- **Refresh Data File:** Reloads the Channel File, which holds the data value for the x- and y-points.

To Create a Channel File for a Graph2D

1. Create a text file (for example, **CHANNEL2**), with no file extension.
2. Add this data: 5 0 0 10 20 20 0 30 20 40 0.
   - The first number (5) holds the number of points available in the file.
   - The other definitions are the X and Y points -> (0/0), (10/20), (20/0) etc.
3. Create this directory: *<viz data folder>\plug-in\graph2D.* and place the channel file in it.

> ⚠ **IMPORTANT!** The channel file must always be named **CHANNEL<Channel number>**.

To Create a 2D Graph

1. Create a new Container.
2. Add the **Graph2D** plug-in.
3. Add a Material and/or a Texture.
4. Set the **Channel** parameter to the Channel file (for example: **2** for **CHANNEL2**).
5. Click **Refresh data file** if the Channel file contents have been modified.

13.1.21 Icosahedron

An Icosahedron is a Platonic solid composed of twenty faces that span twelve vertices, each face of which is an equilateral triangle. An icosahedron can be considered a rough approximation for a sphere.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Icosahedron Properties
- To Create an Icosahedron

Icosahedron Properties

- **Use LOD**: Enables/disables dynamic level of detail.
- **Smooth:** Enables smoothing of edges.
- **Depth**: Sets the number of planes at the icosahedron. The more planes you set the more it looks like a sphere.
- **Tesselation**: Allows you to choose level of detail setting between **Low**, **Middle** and **High**.

To Create an Icosahedron

1. Create a new group and add the Icosahedron plug-in to it.
2. Add a material and/or a texture to the group.
3. Open the Icosahedron editor and disable the Smooth option, set Depth to 3 and Tesselation to Middle.

13.1.22 Image FX

Image FX is a tool that enables you to create a wide variety of transitions between one or more images. The plug-in uses many different effect models that can be customized through parameters. Some properties use advanced mathematical formulas to create the effects, and not all of them can be explained meaningfully in normal words. Because of this, not all properties can be explained completely here, you will have to “play” with some properties to see the effects they create in a given situation (what some properties are set to will have impact on the effect that other properties give).

The images are split into a number of user-defined sections and every section is controlled independently based on progress maps. Any number of images of any Viz Artist supported format can be controlled using ImageFx. All images will be resized (if needed) to the first image size. The image transition order can be modified on the fly and images may be added to the sequence using external commands.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section also contains information on the following topics:

- Image FX Properties
- To Add, Load and Remove Images

Image FX Properties

At the top of the editor you will find a set of buttons to switch the properties.

Image

- **Add and Remove:** Images dropped onto the Add/Remove drop zone will be added/removed as sub-containers of the plug-in container in the scene tree.
- **Load from Scenetree:** Loads images that are added directly as sub-containers to the ImageFX plug-in container.
- **Clear all:** Removes all sub-containers to the ImageFX plug-in container.

Geometry

The image is split into a number of stand-alone sections that can be modified independently.

- **Geometry Type:** Selects which kind of geometry the images will be split up into.
  - **Tiles:** GL quads, the number of quads are X tessellation times Y tessellation.
  - **Mesh:** (vertices) GL quad strip, the number of vertices are (X tessellation +1) times (Y tessellation +1). Modification can apply to every vertex in the mesh.
  - **Triangles:** GL triangles, the number of triangles are X tessellation times Y tessellation times 2. Modification can apply to one triangle (3 vertices).
  - **Random Triangles:** GL triangles, triangles are generated using random points. Number of triangles is a minimum of X tessellation times the Y tessellation. Modification can apply to one triangle (3 vertices).
  - **Random Seed**: Is relevant if you have chosen random triangles. It specifies a seed for the random number generator. Even though Viz Artist use random numbers, the layout of a specific random seed will always look the same. Press the wireframe button and click the new random seed button and you will see the effect.
- **Tesselation:** Sets the degree of detail (enable wireframe to view the effect).
- **Spacing**: Sets a spacing between the geometrical parts in the image.
- **Border**: Draws a border with the width defined here between single sections if spacing has a value.
- **Crop**: Enables/disables cropping of the border.
- **Alpha**: Sets an alpha value for the border.
- **Color**: Sets a color for the border.

Trigger

In the trigger section, you can select between different progress maps that defines how every section of the image is to be triggered by the global progress. Each section of the image receives its base value, the global progress is a value that changes as you animate by using one of the

options in the control menu. Since every section gets a different base value, it will be triggered differently by the global progress.

⚠ **Example:** In a domino effect which starts at the bottom left the section at the bottom left will receive a value of 100 and the section at the top right will receive a value of 1. As the progress moves section with high values will be influences, fast then sections with low values. If we continue with the above example with a fade effect, the bottom left section will fade out when progress is 20 but the top right section will only fade out when progress is 100.

The options for progress maps are:

- **Clock**: Sets the values for the sections that are computed based on a clock hand movement. It has the following properties:
  - **Center**: Sets the center for the clock hand movement.
  - **Start Angle:** Sets the starting angle for the clock hand.
  - **Whirl**: Applies a whirl effect to the movement.
  - **Direction:** Sets the direction for the clock hand movement.
- **Domino**: Sets the values for the sections that are computed based on a domino movement. It has the following options:
  - **Order**: Sets the order in which the tiles will be moved. Choose between **Lines** (Line by line movement), **Columns** (Column by column movement, **Center** (From center and outwards movement), **Bottom Left** (Bottom left tiles first and then diagonal movement towards top right corner) or **Top Left** (Top left tiles first and then diagonal movement towards bottom right).
  - **Direction**: Sets the starting point of the movement.
- **Noise**: Sets the values for the sections that are computed based on user-defined noise map created using noise algorithms (perlin noise).
  - The settings create a noise texture and the image transformations are based on this image.
  - When you edit the noise properties it is a good idea to enable the *show* button. This will let you see the image that are created and you will see changes you do instantly.
  - **Wave**, **Cloud**, **Marble** and **Noise** are predefined noise parameter settings. It is normally a good idea to start with one of these and thereafter edit until you get the required effect. The four images below show how the four noise parameters look like when the show option is enabled:

    
  - **Turbulence 1** to **3**: Sets the noise level. The settings must be *played* with to see what kind of effect you get as they influence each other. So what effect one gives depends on the state of the other.
  - **Frequency:** Defines something that can be described as the “wave length” of the noise function. A large number gives a small wave length which will create a lot of changes to the image, while a small number will create a smoother image.
  - **Exponent:** Controls the ratio between the white areas and the black areas without changing the shape significantly. Exponent zero give a full white picture, and as the value will grow, the black area will become larger and larger.

    - **Scale X/Y**: Are quite similar to the usual texture scale, but they are working with reverse logic. In the usual texture scaling, increasing the scale is like zooming into the image, but in Image FX it works like zooming out. Also the units are different, but beside this, they are quite similar.
- **Random**: Computes the values for the sections randomly.
- **Image**: Computes the values for the sections based on pixel values of an image you drag onto the drop zone. What the function does is to create links between the animated image tiles and the pixels of the image and then use the luminance level of each pixel to set the base value for the tiles. By default 100% luminance sets high base value and 0% sets a low base value. This option allows you to generate any progress map you want by using other tools.
- **Static:** Gives all sections the same value. The complete image will move out and the other will move in.
- **Random Seed:** Specifies a seed for the random number generator. Is relevant if you have chosen random order. Even though Viz Artist uses random numbers, the animation for a specific random seed will always look the same.

Effect

The trigger section defined the order in which the tiles would be influenced by the global process. The effect section allows you define what kind of effect/transition that will be used.

- **Target Image**
    - **Static:** The target image is not effected at all the by the effect. A typical example would be a reveal effect where target image is “hidden” behind main image and a transition effect on the main image will reveal the target image.
    - **Dynamic:** The target image is controlled by the effect. A typical example would be main image flying out of the frame and at the same time target image flying in. If a dynamic target image is used, its effect can be either being the invert of main image or can be defined separately.
- **Interpolate**
    - **Linear**: A linear interpolation will be used for the transformations.
    - **Smooth**: A smooth interpolation will be used for the transformations.
- **Position > Move**: You can specify a position for the source image, for the target image and a random position.
    - For the source image, which is the image that is currently shown, you specify a source position. This is the position that the image tiles will move to when the animation runs.
    - If no target position is specified, the target image will animate from the source position to the initial position of the source image.
    - If you specify a target value the target image will move from that position and onto the initial position of the source image.
    - Switch on **X**, **Y** and **Z** to alter the values.
    - **Absolute:** All sections will move to the same end position.
    - **Relative**: The sections will maintain their original relationship.

- **Position > Explode**: Final positions for the sections are calculated based on a user-controlled explode algorithm. The tiles of the image are treated as particles in an explosion and they are moved in one to three axis depending on what you specify. All tiles are thrown out of a emitter that can be defined to have any opening angle, so the particles can be sent within a narrow angle, like a canon fire, or in all direction like a explosion. The parameters are:
  - **Duration**: Sets the duration for the movement of the tiles/particles.
  - **Opening Angle:** Sets the opening angle for the emitter. Remember to choose more than one axis to get any clear effect.
  - **Angle Rotate X**: Rotates the emitter hole on the X-axis.
  - **Angle Rotate Y**: Rotates the emitter hole on the Y-axis.
  - **Force** sets the degree of force that throws the particles out in space.
  - **Force Spread %:** Randomizes the initial impels of the tiles. Spread 0 % means all items have the same impulse, 100% means a high degree of randomness.
  - **Gravity**: Sets a gravity force for the environment.
  - **Use Axis**: Selects which axes or axis to use.
- **Rotation:** Rotates the sections.
- **Scaling**: Scales the sections.
- **Alpha**: Changes the alpha value of the sections from **Alpha Start** to **Alpha End**.
  To all options (except alpha) a certain degree of user-controlled randomness can be added.
- **Random Seed**: Specifies a seed for the random number generator. Is relevant if you have a random property. Even though Viz Artist uses random numbers, the animation for a specific random seed will always look the same.

  The final effect of transformation can be a combination of effects where the timing between the different effects can be tuned using the mini-stage. Move the start and end Key Frames to time the effects.

⚠ **Example:** A position explode effect can be combined with a rotation effect so the sections will rotate as they explode. An alpha effect can be added to fade out the sections at the end.

The pivot properties have relevance when you use rotation on the tiles. The settings decide where the rotation center should be on the tiles. To see the effect, create a low tesselation transition with tiles and a 180 degree X or Y rotation. Change the pivot properties for Y pivot if you have an X rotation or the X pivot if you have an Y rotation and see the effect.

Control

The global progress value that is used to animate the sections can be controlled in different ways:

- **Auto:** When using auto mode ImageFx will start the effect as soon as the “Take” button is invoked. Useful for external control to run a sequence of images.

- **Animation:** The global progress value is animated and the effect follows the stage progress. In this mode you can combine the effect animation with other animations.
- **Global Animation:** When using a sequence of images the complete progress will animate all images.

To Add, Load and Remove Images

1. Add the images you want to create transitions between by dragging them from the Server Panel and dropping them onto the **Add** drop zone.
    - The added images will be visible in the scene tree as sub-containers of the ImageFX plug-in container.
2. To add images directly as sub-containers to the plug-in container you must click the **Load from Scenetree** button to make the plug-in recognize the added images.
3. Remove images by dragging them from the Server Panel and onto the **Remove** drop zone, or simply remove them directly from the scene tree.
4. To remove all, click the **Clear all** button.

13.1.23 Noggi

Use this plug-in to create geometry objects of beveled rectangles with some attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Noggi Properties
- To Create a Noggi

Noggi Properties

- **Width:** Changes the width in X-direction.
- **Height:** Changes the height in Y-direction.
- **Bevel (top/right, top/left, bottom/left and bottom/right):** Changes the bevel on each corner.
- **Stretch:** The result is an object like a parallelogram. Stretch is in percent (= tan(alpha)). If Geometry Type is Outline.
- **Horizontal Alignment:** Changes the object alignment to left, center or right. This change will affect the y-rotation.
- **Vertical Alignment:** Changes the object alignment to bottom, middle or top. This change will affect the x-rotation.
- **Geometry Type:** Renders the object as area- or with outline look.
- **Inner Outline:** Changes the vertex sequence in outline mode. Inner Outline is important for face orientation when extruding the geometry.
- **Outline Width:** Changes the thickness of the rendered outline.

- **Edge Points:** Change the number of triangles to render a beveled corner. Increasing the number increases the bevel resolution.
- **Crop:** Enables/disables cropping of the object. Cropping and vertex texture are only available if Geometry Type is Area.
    - **Crop Face (Left, Right, Bottom and Top):** Crops the object on each given side.

To Create a Noggi

1. Create a new group and add the Noggi plug-in to it.
2. Add a material and/or a texture to the group.
3. Open the transformation editor and set Position X to -150.0.
4. Open the Noggi editor and set the following parameters:
    - Set Width and Height to 600.0 and 100.0.
    - Set Bevel top/right and bottom/left to 0.0.
    - Set Bevel top/left and bottom/right to 20.0.

13.1.24 Pointer

The pointer is similar to the Arrow. The main difference is that the pointer has a joint link, so it looks more like a *bend* arrow. The pointer has two reference points which controls the pointer's position. The first point is in the joint link and the second point is at the arrow's head.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Pointer Properties
- To Create a Pointer

Pointer Properties

- **Use LOD**: Enables/disables dynamic level of detail.
- **Point 1 X**: Sets the position of point 1 on the X-axis.
- **Point 1 Y**: Sets the position of point 1 on the Y-axis.
- **Point 2 X**: Sets the position of point 2 on the X-axis.
- **Point 2 Y**: Sets the position of point 2 on the Y-axis.
- **Width**: Sets the width of the pointer shaft.
- **Cap Style Begin:** Allows you to select the style on the pointers beginning. The options are **Flat**, **Circle** or **Arrow**.
- **Cap Style End**: Allows you to select the style on the pointers end. The options are **Flat**, **Circle** or **Arrow**.

- **Arrow Length**: Sets the length of the arrow head.
- **Arrow Width**: Sets the width of the arrow head.
- **Circle Radius:** Sets the radius of the circle, if cap style is selected.
- **Circle Segments:** Sets the resolution of the outer side of the pointer angle. Switch to wireframe to see the effect as you change the value.
- **Joint Style**: Sets the style of the pointer joint, either **Round** or **Miter**.

To Create a Pointer

1. Create a new Container.
2. Add the Pointer plug-in to it.
3. Add a material and/or a texture to the group.
4. Open the Pointer editor and set the following parameters:
    - Set Point 1 X and Y to 50.0.
    - Set Point 2 X and Y to 75.0.
    - Set Width to 20.0.
    - Set Cap Style Begin and End to Circle and Arrow, respectively.
    - Set Arrow Length, Width and Radius to 50.0.

13.1.25 Polygon

The Polygon Geometry plug-in enables the creation of user defined polygon shapes. The polygons are drawn and edited directly in the Scene Editor. Created Polygons can also be rotated to create a 3D object (see Revolve: Spline Rotation in 3D).

To create a Polygon, add the Polygon plug-in to a container and start drawing by clicking in the Scene Editor to create polygon nodes. A click, hold and drag converts the node into a spline handle for drawing curves. If an existing node is clicked on it removes the node, and if a line or spline is clicked on, between two nodes, it inserts a new node at this location. There are several Polygon plug-in Editor Shortcuts. If more than one polygon is created, in the same container, and if they overlap each other, they will act as masks.

Instead of drawing polygons manually, they can be imported as Adobe Illustrator files (*.ai) as well.

> ⚠ **Note:** Only Adobe Illustrator AI 88 format files are supported.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Polygon Properties
- Polygon Scene Editor Context Menu

- To Select an Option from the Context Menu

Polygon Properties

To toggle between the Polygon and Container properties:

- Click the Polygon or Transformation icon in the:
  - Scene Tree Container, or
  - The Properties panel
- In the Scene Editor:
  - Outside the polygon bounding box: Right click and select Properties -> Geometry.
  - Inside the polygon bounding box: Right click and select Edit container.

- **Precision**: Sets the precision of the spline calculation.
- **Import**: Imports Adobe illustrator files. Opens a window to select a file and import. Only Adobe Illustrator AI 88 format files are supported.
- **Revolve**: See Revolve: Spline Rotation in 3D.
- **Storage**: Creates a Spline with coordinates. Press <Enter> to apply.

Polygon Scene Editor Context Menu

Right-click, and hold, in the polygon green bounding box, in the Scene Editor, to open the polygon context menu. To select a menu option see To Select an Option from the Context Menu.

> ⚠ **Note:** The right-click must be inside the created polygon shape box. If a click is outside the bounding box then the standard Scene Editor menu will be shown.

- **New spline:** Creates a new polygon in the same container.
- **Close/Open:** Opens or closes a polygon.
- **Cut**: Cuts the selected polygon or spline nodes.
- **Copy**: Copies the selected polygon or spline nodes.
- **Paste**: Pastes the selected polygon or spline nodes.
- **Mirror X/Mirror Y:** Mirrors the created shape in the X or Y axis.
- **Flip x/Flip y:** Flips the created shape in the X or Y axis.
- **Edit container:** Opens the Container editor and exit polygon edit mode. To edit the polygon again, click on the polygon icon in the properties editor or the Scene Tree.

To Select an Option from the Context Menu

1. Right-click and hold.
2. Move the cursor to the required option.
3. Release to select.

Create a Polygon

This section contains the following procedures:

- To Create a Simple Spline
- To Draw a New Spline
- To Move a Spline
- To Move More Than One Spline
- To Add or Delete a Spline Point
- To Move Spline Points
- To Create or Delete a Spline Point Handle
- To Resize a Spline Point Left or Right Handle
- To Resize a Spline Point Handle
- To Mirror Spline Sections
- To Flip a Spline
- Scene Editor Grid
- To Create a Spline in the Scene Editor Grid Snap
- To Move a Spline in the Grid with Snap
- Revolve: Spline Rotation in 3D
- Polygon Revolve Properties
- To Rotate a Spline

To Create a Simple Spline

1. Create a Container.
2. Add the Polygon plug-in.
3. Click on the Polygon icon (in the Container or the Properties editor).
4. The polygon splines are created by each click in the Scene Editor:
   - Click in the Scene Editor to create a spline point
   - Click and drag in the Scene Editor to create a spline node, complete with handles (continue the drag to increase the length of the handles)

   
5. While the polygon plug-in is active each click in the Scene Editor will create more polygon points, and create the polygon shape.

   ⚠ **Tip:** At any time press and release <Delete> to remove the last created point. Press and hold <Delete> to remove more than one spline, or remove the whole polygon.

   Once the spline has been created it can be modified, as required, with various shortcuts (see Polygon plug-in Editor Shortcuts). List below are some common examples:

To Draw a New Spline

1. Right-click in an existing spline.
2. Click **New Spline**.
3. Add intersections as detailed in To Create a Simple Spline.

To Move a Spline

1. Select the spline to be moved:
    - Press <S> and click on a spline.
2. Press <Alt> and drag.

> ⚠ **Tip:** If there is only one spline in the Container, press <Alt> and drag.

To Move More Than One Spline

1. Select the splines to be moved:
    - Press <S> and drag around the splines.
2. Press <Alt+S> and drag.

To Add or Delete a Spline Point

- Click anywhere on a spline to add another spline point.
- Click on a spline point to delete it

To Move Spline Points

1. Press and hold <Ctrl>.
2. Click on a spline point and drag.
   When <Ctrl> is pressed move any spline points as required.

To Create or Delete a Spline Point Handle

1. Press <Shift>:
    - **Create**: Click on a spline point and drag.
    - **Delete**: Click on a spline point with a handle.

To Resize a Spline Point Left or Right Handle

1. Press <Shift>.
2. Click on a spline point handle and drag. Only the selected handle will move.

To Resize a Spline Point Handle

1. Press <Ctrl>.
2. Click on a spline point handle and drag. The left and right handles will move as one handle.

> ⚠ **Note:** The handle clicked on can be lengthened as required. The other handle will stay the same length

> ⚠ **Note:** If either the left or right handle has been moved individually, this will set the two handles together again

To Mirror Spline Sections

1. Right-click on a spline, or select particular nodes of it with <S>.
2. Select either:
   - **Mirror X**: Mirrors whole spline along the X axis.
   - **Mirror y:** Mirrors whole spline along Y axis.

To Flip a Spline

1. Right-click on a spline.
2. Select either:
   - **Flip x:** Flips whole spline over X-axis.
   - **Flip y**: Flips whole spline over Y-axis.

Scene Editor Grid

The Scene Editor Grid Tool-bar can be used as a great tool to help draw polygons. Together with the grid snap option polygon points can positioned, simply and accurately, in the 3D space.

Make sure that the polygon is aligned with the grid for the grid snap to work.

To Create a Spline in the Scene Editor Grid Snap

1. In the Scene Editor click (Grid).
2. Click (Plane Type).
3. Click (Free Grid).
4. Click (Snap to Grid).
5. Create a spline. Each spline point will snap to the nearest cross hairs on the grid.

To Move a Spline in the Grid with Snap

1. In the Scene Editor click (Grid).
2. Click (Plane Type).
3. Click (Free Grid).
4. Click on (Move Object to Plane).
5. Press <Alt>.
6. Click on the Spline and drag.

> ⚠ **Note:** If there is more than one Spline to be moved see To Move More Than One Spline.

The Spline will snap to the next set of cross hairs, and so on, as it is dragged.

Revolve: Spline Rotation in 3D

The Spline rotation creates a 3D mesh by rotating the created spline.

Use the Polygon Revolve Properties panel to modify the Spline as required.

Polygon Revolve Properties

The Polygon Revolve properties are available after the first spline point has been added to the Scene Editor. Click on **Rotate** to activate.

- **Precision**: Sets the precision of the spline calculation.
- **Import**: Imports Adobe illustrator files. Opens a window to select a file and import. Only Adobe Illustrator AI 88 format files are supported.
- **Rotate**: Creates the revolve mesh and activate all its settings.
  - **Rotate in degrees box**: Sets the degree of rotation.
- **Tessellation**: Sets the degree of detail for the mesh.
- **Shading Angle**: Defines how the shading normal are aligned on the revolve object surface.
- **Culling**: Sets which side of the spline to view:
  - **Back**: View the back side of the spline.
  - **Front**: View the front side of the spline.

    > ⚠ **Note:** To view both sides at the same time, add the Expert plug-in to the Container and active **Back Face**.

- **Revolve Around**: Sets the axis to rotate the spline around.
  - **Local Axis**: Calculates the rotation around the Container axis when the Rotation angle is changed.
  - **First to Last**: Rotates around the first and last spline point (**Revolve Axis** is grayed out) when the Rotation angle is changed.
- **Revolve Axis**: Sets which axis to rotate around:
  - **Y**: Rotates around the Y axis.
  - **X**: Rotates around the X axis.
- **Direction**: Sets which direction to view the rotation:
  - **Right-hand**: View from the right hand side.
  - **Left-hand**: View from the left hand side.
- **Show Hull**: Shows the rotated shape (1 and 2).
- **Show Caps**: Shows the original polygon shape at the beginning and end of the rotation (1 and 3).

To Rotate a Spline

1. Create a spline.
2. Click on **Rotate**.

3. Modify the created 3D object with the Polygon Revolve Properties.

See Also

- Polygon plug-in Editor Shortcuts
- Revolve: Spline Rotation in 3D

13.1.26 Rectangle

Use this plug-in to create a simply 2D rectangle with some attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Rectangle Properties
- To Create a Rectangle

Rectangle Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Tessellation:** Sets the degree of detail.
- **Bevel**: Enables and sets the degree of bevel at the rectangle edges. Bevel may have an effect on the height of the rectangle. For example if Height is set to 100.0 and Bevel is set to 51.0, the effective height is 102.0.
- **Width**: Sets the width of the rectangle.
- **Height**: Sets the height of the rectangle.
- **Corners:** Sets the number of corners the rectangle is build up from, it become visible, when you set a bevel on the rectangle.
- **Use Vertexcolors:** Enables you to set the four color parameters below. If this option is set, the rectangle does not respond to container color anymore.
- **Upper Left**: Sets the color and alpha of the upper left section of the rectangle. Press the colored button to enable color editing on that section and change the color in the color editor below. You can also drag a material from the Server Panel and onto the color icon.
- **Lower left**: Sets the color and alpha of the lower left section of the rectangle. Press the colored button to enable color editing on that section and change the color in the color editor below. You can also drag a material from the Server Panel and onto the color icon.
- **Upper Right**: Sets the color and alpha of the upper right section of the rectangle. Press the colored button to enable color editing on that section and change the color in the color editor below. You can also drag a material from the Server Panel and onto the color icon.
- **Lower Right**: Sets the color and alpha of the lower right section of the rectangle. Press the colored button to enable color editing on that section and change the color in the color

editor below. You can also drag a material from the Server Panel and onto the color icon.

- **Mode:** Sets the orientation of the plane to XY, XZ or YZ.

To Create a Rectangle

1. Create a new group and add the Rectangle plug-in to it.
2. Add a material and/or a texture to the group, and/or open the Rectangle plug-in editor and enable and set the vertex colors.
3. Open the transformation editor and set Position Y to -150.0.
4. Open the Rectangle editor and set the following parameters:
    - Set Bevel to 15.0.
    - Set Width to 600.0.
    - Set Height to 30.0.

13.1.27 Ring

Use this plug-in to simply create an open or closed ring with some attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Ring Properties
- To Create a Ring

Ring Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Corners**: Sets the number of corners on the ring.
- **Inner Radius:** Sets the size of the inner radius.
- **Outer Radius**: Sets the size of the outer radius.
- **Rotation**: Rotates the ring like a “turning wheel”. This is typically used in combination with an open angle, to place the angle at the required point.
- **Angle**: Defines an open angle on the ring.
- **Mode:** Changes the object view coordinate. Available options are XY, XZ and YZ.

To Create a Ring

1. Create a new group and add the Ring plug-in to it.

2. Add a material and/or a texture to the group.
3. Open the Ring plug-in editor and **animate** the Angle.
    - Set Angle to 0.0 (60/50 fps).
    - Set Angle to 360.0 (120/100 fps).

13.1.28 Roll

The roll allows you to create a rectangle that rolls up in different ways. This is typically used for different kinds of unveiling. To make the roll look realistic, the expert plug-in must be added to the container to enable visualization of backface and two-sided lightening.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Roll Properties
- To Create a Roll

Roll Properties

- **Use LOD**: Enables/disables dynamic level of detail.
- **Size X**: Sets the size of the rectangle on the X-axis.
- **Size Y**: Sets the size of the rectangle on the Y-axis.
- **Tesselation:** Sets the level of detail.
- **Angle**: Sets the angle of the rolling up.
- **Diameter:** Sets the diameter of the roll.
- **Helix factor:** Defines how tight the roll is to be rolled up. With a high helix factor, there will be much space between the layers of the roll.
- **Drawing:** Selects how the roll rectangle is to be drawn up. If you switch to wireframe mode and toggle between the two settings you will see the difference:
    - **Segments:** Draws the rectangle up by using many long lines stretching from edge to edge. This mode does not look as good as **Mesh** but it demands less performance of the render engine.
    - **Mesh:** Draws the rectangle using many small equally sized triangles. The number and size depends on the level of tessellation selected. This drawing mode can be heavy to render if the tessellation is set high. **Mesh** creates a better lightning of the roll rectangle.
- **Style:** Sets the style of the roll:
    - **Cigar roll:** Makes the rectangle roll up in a normal manner.
    - **Wipe:** Rolls the carpet up without creating a roll.
- **Outline:** Enables an outline view for the roll. You must in addition add the expert plug-in and enable the outline setting there.

- **Time:** Sets the time for the roll sequence. Animate this value to create an animation of the roll object. To learn more about animation see Create Animations.

To Create a Roll

1. Create a new group and add the Roll plug-in to it.
2. Add the Expert plug-in to the group.
3. Open the Expert editor and enable (On) the Back Face property.
4. Add a material and/or a texture to the group.
5. Open the Roll editor and set the following parameters:
    - Set Size X to 150.0.
    - Set Size Y to 20.0.
    - Set Angle to 10.0.
6. Adjust the Time parameter to see the roll effect.

13.1.29 SoftClip Draw Pixels

IMPORTANT! *Deprecated plug-in*. Since Viz version 3.8.2, this plug-in is no longer supported, and has ceased to be bundled with the Viz Artist installer.

Plug-in for playing video clips by drawing pixels.

Choose a video file to play. Playback can be achieved by either animating over the frame numbers or by turning on live playback. While in live playback mode, the frames will be shown synchronized to Viz rendering, or according to the Frames Per Second (FPS) setting, depending on the play mode that is chosen.

On machines with multiple CPUs turning on the decode thread will help balancing the CPU load over the CPUs. If you use live play, the Preload frame count can help reducing spike-loads and avoid dropped frames.

Note: This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- SoftClip Draw Pixels Properties

SoftClip Draw Pixels Properties

- **use LOD:** Enables/disables dynamic level of detail.
- **Clip file**: Loads the AVI clip file.
- **Live Play:** The clip is being automatically played.

- **Play mode:** Plays the frames synchronized to Viz rendering or according to the given Frames per second.
- **Frame per second:** Renders the number of frames per second.
- **Loop video:** Plays the video in an endless loop.
- **Use decode thread:** Useful if the machine owns more than one CPU core. It increases the rendering performance if this parameter is set.
- **Preload frame count:** Increases the performance through loading frames in RAM before they are currently needed. Recommended setting is 1.
- **Frame skipping:** Skip frames if the machine does not have enough performance to render all frames in the given time range instead of slowing down the video playback.
- **Force Opaque (fill alpha channel):** Ignores the transparency.
- **Preferred CPU (0 = Any):** Use this option if you want choose manually a core for rendering and processing. Choose 0 for automatically selection.
- **Priority:** Allows you to set the priority if the entire achievement is needed because of clip rendering lapse. Setting priority to Low should not reduce the overall render performance.
- **X, Y and Z Position:** Changes the X, Y and Z coordinates for clip translation.

13.1.30 Sphere

Use this plug-in to create a simple sphere with some attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Sphere Properties

Sphere Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Tesselation**: Sets the degree of detail for the sphere.
- **Rotation**: Allows you to rotate the sphere around its Y-axis.
- **Open Angle:** Allows you to create an open angle in the sphere.
- **Texture**: Allows you to select between three different positions of the texture mapping coordinates when a vertex texture is mapped onto the sphere. To visualize the effect of the different settings, add a plain texture onto the sphere, like a chessboard image and set the texture mapping to vertex in the texture editor. Create an opening in the sphere and change the sphere texture parameters. Increase and decrease the opening angle with the different settings enabled and the effect should be visible.
    - **Absolute:** Compresses and expands the texture mapping coordinates as the opening of the sphere changes.

- **Relative**: Cuts/adds texture coordinates as the opening changes. The cut/add is done on the one side of the opening angle. The coordinates are anchored at the other side.
- **Centered**: Does the same as the relative mapping, but cuts/adds coordinates from both sides. The texture is anchored in a centered position on the sphere. The center of the sphere is 180° from the point where the opening angle starts.

13.1.31 Spline Path

Use this plug-in to create complex paths with an arrow look.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Spline Path Properties
- To Create a Spline Path

Spline Path Properties

- **use LOD:** Enables/disables dynamic level of detail.
- **Node:** Defines the node index. For example node 3 or node 25.
- **X, Y and Z**: Sets the 3D point to define a node.
- **Length:** Stretches the entire path in X-direction.
- **Width:** Stretches the entire path in Y-direction.
- **Threshold**: Increase this value if you do not need a high path accuracy but more performance.
- **Add, Move, Remove and Clear:** Add adds a new node with an specific index. Move moves an existing node to another X, Y or Z-point. Remove removes an existing node. Clear clears all nodes.
- **Mode:** The following options are available:
  - **Tube:** Renders the entire object.
  - **Band XY:** Renders only the front side.
  - **Band XZ:** Renders only the top side.
  - **Band YZ:** Renders only the left side.
- **Arrow:** Creates an array look.
- **Arrow Length:** Sets the arrow length (only when **Arrow** is set to On)
- **Arrow Width:** Sets the arrow width (only when **Arrow** is set to On)

To Create a Spline Path

1. Create a new group container.

2. Add the Spline Path plug-in.
3. Add the Expert plug-in.
4. Open the Expert editor
5. Set **Back Face** to On.
6. Add a Material and/or a Texture to the group.
7. Open the Spline Path editor.
8. Set **Node** to 0 / **X** to 50.0 / **Y** to 50.0
9. Click Add.
10. Set **Node** to 1 / **X** to 150.0 / **Y** to 25.0
11. Click Add.
12. Set **Node** to 2 / **X** to 250.0 / **Y** to 10.0
13. Click Add.
14. Set **Node** to 3, **X** to 350.0, **Y** to 45.0
15. Click Add.

13.1.32 Spline Strip

The Spline Strip plug-in is used for creating 3D interactive loft extrusions which are able to be animated in their length and surface. You can use modifiable lines, circles or rectangles for the extrusion. Arrows and terrain alignment are also supported.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Spline Strip Properties
- To Create a Simple Spline Strip

Spline Strip Properties

- **Path Node Index:** Defines where Path Nodes get added, moved or removed. The first node within a path owns the index 0. If you add nodes at the end, then the Path Node Index is increased automatically to provide a simple path generation.
- **X, Y and Z:** Sets the Path Node's X, Y and Z-position.
- **TC:** Sets the Path Node's TC-position.
- **Add Node:** Adds a path node at the current Path Node Index with position X, Y, Z and TC.
- **Move Node:** Moves the current path node to your required X, Y, Z and TC position.
- **Remove Node:** Removes a path node at the current Path Node Index.
- **Interactive Mode:** If you have finished the process of setting path nodes then you are able to edit it simpler by mouse. Just drag and drop the shown nodes to a new position. Left Mouse Button: move node in XY. Middle Mouse Button: move node in XZ.
- **Interactive Orientation:** World, Local, View. Move nodes interactive with world-, local-, or viewpoint- coordinates.
- **Node Size:** Sets the interactive node's size.

- **Storage:** Is used for storing your defined path nodes in a #-separated string. It is possible to edit nodes directly in the Storage Text field.
- **Clear Storage:** Clears the spline strip object and its storage.
- **Use Time Code:** Activates Time Code functionality.
- **TC Current:** Contains the current Time Code of an external Time Code source.
- **TC Start:** Sets the starting point of your Time Code animation.
- **TC Reference:** If you are using more than one spline strip then they probably won't start all at the same Time Code position. But you can change that by a simple time shift with the Help of TC Reference.
- **Path Steps:** Defines SplineStrip tessellation.
- **Use Threshold:** You can optimize your spline strip with that option. Path Steps are added at necessary positions only. So the number of Path Steps increases with a decreasing Threshold.
- **Threshold**: Sets the Threshold for Path Steps generation.
- **Path Start:** Sets the starting point of the path geometry.
- **Path Length:** It defines the length of our spline strip object. Feel free to animate that value!
- **Path Alignment:** Fits the spline strip to Terrain plug-ins in Viz Artist/Engine. (e.g. use Terrain for drawing paths over a mountain). Available options are; Free and Topography.
- **Surface Offset:** Shifts Path on Y axis.
- **Use Surface Normal:** If activated then the surface offset is computed via terrain normals.
- **Show Path:** Shows the Path which you have defined at the beginning. This option is quite helpful when you are editing Path Nodes.
- **Shape Type**: If you activate the Organic Mode, your spline strip will start to wobble in dependency of the adjusted Organic Amplitude, Organic Speed, and Organic Period's Length. Available options are; Static and Organic.
- **Shape**: Defines the shape used for the loft extrusion. Available options are; Line, Circle and Rectangle.
- **Shape Width**: Sets the shape's width.
- **Shape Height:** Sets the shape's height.
- **Shape Tesselation:** Defines how many steps are used to draw the required shape.
- **Shape Rotation:** You are able to move the spline strip shape up to 360 degrees.
- **Shape Orientation:** When set to Free, shapes are rotated in dependency of the adjusted Shape Rotation value. When set to XZ, YZ or YX, shapes are auto rotated to obtain a constant SplineStrip width in the required aspect. When set to Terrain, shapes are rotated to fit the terrain beneath them.
- **Translate Vertices To Terrain:** Moves every single vertex to fit the terrain. This mode only makes sense if you are using Line extrusion.
- **Flip Normals:** Flips spline strip normals.
- **Backface Culling:** Switch off drawing spline strip's backside.
- **Organic Amplitude:** Adjusts the animated diameter reduction from 0% to 100%.
- **Organic Speed:** Changes speed of organic mode animation. Also negative values are allowed.
- **Organic Period's Length:** Sets the diameter reduction function's period length.
- **Termination**: Closes our spline strip object at the Beginning, End or at Both sides. Available options are; None, Beginning, End and Both.

- **Arrow:** Puts an Arrow to the Beginning, End or Both sides. Available options are; None, Beginning, End, Both
- **Object Width:** Changes arrow width.
- **Object Height:** Changes arrow height.
- **Object Length:** Changes arrow length.
- **Object Tesselation:** Changes arrow tessellation.
- **Generate Texture Coordinates:** Activate this function if you are using vertex maps on your spline strip.
- **Stretch Texture to Path Length:** If activated then the map is stretched to the actual path length.
- **Draw Mode:** Switch between fill- and wire-frame mode. Available options are; Fill and Wire.
- **Buffer Geometry:** The whole spline strip data is buffered to memory. If you use spline strips with a huge number of vertices then Buffer Geometry is able to improve performance. Activate this mode only if you do not use any kind of animation in your current spline strip.
- **Color:** Defines path's and interactive-mode-node's color. Color is also used if Wire is activated and no maps are applied onto your spline strip object.
- **Child Distance:** Defines distance between path end and child container.
- **Rotate Child:** Activates child container rotation.

To Create a Simple Spline Strip

1. Create a new Container.
2. Add the Spline Strip plug-in.
3. Open the Spline Strip editor.
4. Use all default values.
5. Click **Add Node**.
6. Change **X** to `100.0`.
7. Click **Add Node**.
8. Change **X** to `200.0`, **Y** to `50.0`.
9. Click **Add Node**.
10. Set **Shape Width** and **Shape Height** to `50.0`.
11. Set **Shape** to **Circle**.
12. Set **Shape Tesselation** to `20`.
13. Set **Path Lenght** to `150`.

14. Optionally, click **Interactive Mode**.
15. Change **Node Size** to for example `15.0`.
16. Activate script and plug-in events in the Scene Editor by clicking the **E** button.
17. Drag and drop the small Cubes as required.

13.1.33 Spring

The spring object creates a spiral spring. Be aware this object can easily be very heavy to render if a high level of tessellation and corners are selected.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Spring Properties
- To Create a Spring

Spring Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Tesselation**: Sets the degree of detail for the spring
- **Corners**: Sets the number of corners the spring should have.
- **Gradient:** Sets the size of the spring gradient.
- **Diameter torus:** Sets the diameter of the springs.
- **Diameter cross section:** Sets the diameter of the spring cord.
- **Show Bottom**: Shows/hides bottom of the spring.
- **Show Top**: Shows/hides Top.
- **Center**: Sets the location of the spring center, either **Center**, **Bottom** or **Top**.

To Create a Spring

1. Create a new group and add the Spring plug-in to it.
2. Open the Spring editor and set the following parameters:
    - Set Turns to 6.0.
    - Set Gradient to 20.0.
    - Set Diameter torus to 200.0.
    - Set Diameter cross section to 100.0.
    - Set Center to Center.
3. Add a material and/or a texture to it.

13.1.34 Star

Use this plug-in to create a Star with different attributes.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topic and procedure:

- Star Properties
- To Create a Star

Star Properties

- **Corners/Tips**: Sets the number of corners on the inner circle of the star (equals the number of tips on the outer circle of the Star). Default: 5, Min: 3, Max: 300.
- **Star Radius**: Sets the radius of the outer circle (where the tips of the star are located). Default: 50, Min: 1, Max: 200.
- **Star Inner Radius Factor**: Determines the factor of the outer radius to the inner radius of the star. A value of 1.0 means that the inner radius of the star is identical to the **Star Radius** value. Default: 2, Min: 0.25, Max: 50.
- **Hole**: Places a hole in the Star geometry. Default: 0 (Off).
- **Hole Radius**: Sets the radius of the outer circle where the tips of the hole are located. Default: 25, Min: 1, Max: 200.
- **Hole Inner Radius Factor**: Determines the ratio of the outer radius to the inner radius of the hole. A value of 1.0 means that the inner radius of the hole is identical to the Hole Radius. Default: 2, Min: 1, Max: 50.

To Create a Star

1. Create a new container.
2. Add the Star plug-in to it (1 (default Star)).
3. Open the Star editor.
4. Modify the Star parameters, for example (2).
    - Corners: 10
    - Star Radius: 60
    - Star Inner Radius Factor: 6.0
    - Hole: On
    - Hole radius: 50
    - Hole Inner Radius Factor: 9.0
5. If required, add a material and/or a texture.

13.1.35 Torus

Use this plug-in to create a torus with different attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Torus Properties
- To Create a Torus

Torus Properties

- **Use LOD**: Enables/disables dynamic level of detail.
- **Tesselation:** Sets the degree of detail for the torus.
- **Corners**: Sets the number of corners on the torus.
- **Radius**: Sets the radius of the torus from its center in the middle of the hole.
- **Radius cross section:** Sets the radius of the torus cross section or “tube”.

To Create a Torus

1. Create a new group and add the Torus plug-in to it.
2. Open the Torus editor and set the following parameters:
    - Set Radius to 50.0.
    - Set Radius cross section 20.0.
3. Add a material and/or a texture to it.

13.1.36 Trio Scroll

The Trio Scroll plug-in is essentially a Geometry plug-in for positioning and scrolling a fixed set of items in a specified direction, often used for credit lists or crawls.

Scrolling templates can be built in Viz Trio using the Create New Scroll feature. The scroll scene that is automatically created by Viz Trio can be edited in Viz Artist, but it is also possible to build them manually.

Some of the supported features are:

- Scrolls can be created by receiving item data from Viz Trio’s XML format.
- Fine-tuned control of spacing between individual items.
- Easepoints (ease in and out) on particular items.
  Scrolls are usually built in Viz Trio, but it is possible to add items in Viz Artist by adding merged objects under the `base_elements` group and clicking the **Initialize** button.

Adding the Trio Scroll plug-in to the scene tree will automatically add the Control Object plug-in (if it is not present).

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Trio Scroll Properties

Trio Scroll Properties

- **Field Identifier:** When making templates for Viz Trio, this should be a numeric value, Range: 1-n. For Viz Pilot and Viz Ticker, this can also be a descriptive text string. This value is used by the control clients to identify the editable item in the scene. In Viz Trio, the numbers from the control plug-ins are used to create the tab-order between the editable items.
- **Description:** A description of the tab field or editable object. This will be used as a description for the items when used in the control clients.
- **Show only in Expert Mode:** If Viz Trio is connected, changes are possible if this toggle is set to Off.
- **Mode:** Selects the direction the scroller should move across the screen.
- **Scroll area width**: Selects the area in width the scroller should use.
- **Scroll area height:** Selects the area in height the scroller should use.
- **Scroll speed:** Sets the scroller speed.
- **Wanted total scroll time:** Sets the scroll time.
- **Actual total scroll time:** Sets the scroll time.
- **Global spacing**: Sets spacing between the pages.
- **Loop:** Enables the Loop property set On or Off (default) looping. When set to Off looping will not stop the playlist from being played till its end.
- **Show bounds:** Enables and shows the scroll's bounding-box on the preview and program renderer.
- **Auto-start scroll:** Enables the Auto-scroll on take property.
- **Start position:** Sets the start position for the scroll. This setting will affect the Actual total scroll time.
- **Alignment:** Sets the alignment of the scroll. The position is relative to the position of the scene design. Available options are: None, Left/Bottom, Center and Right/Top.
- **End-of-scroll director:** Triggers a Continue on a specified end director. The director is triggered when the scroll leaves the scroll area. It is only supported in normal mode (real-time), not in post rendering or NLE mode.
- **Show Advanced:** All parameters under the Advanced section are for debugging, by Vizrt developers or very advanced users only. For more information contact Vizrt support.
  The nameless text parameter with the value "<entry></entry>" shows the contents of the scroll items in the internal VDOM format. It contains the scroll items that can dragged into a scroll page in the Trio client with drag and drop
  - **Current position**: Scrubs the scroller position.
  - **Command**: Executes various scroller commands. After setting its value the click on **Execute Command**. If a command has a result it is reflected in **Command result**. The following commands are used from the Trio client:

    ⚠ **Example:** `SET_FOCUS [<item name> [<tab-field name>]]` – This is for example used if you double click in Trio on a scroll item to highlight it in the preview.

> ⚠ **Example:** `MOVE_ELEMENT <from item name> <to item name>` - This is used when changing the order of scroll items by drag and drop.

> ⚠ **Example:** `REMOVE_ELEMENT <item name>` - This is used to remove a scroll item.

> ⚠ **Example:** `UPDATE_TOTAL_TIME` – This commands the plug-in to recalculate its total time, typically after adding a new scroll item.

> ⚠ **Example:** `SET_TEMP_EASEPOINT_VALUE` – Gets used when editing the **Easepoint** values.

- **Command result**: See **Command**.
- **Reload prototypes**: Updates the scroll items due to changes in their templates (scenes) - use a context menu action in Viz Trio to do this (see the Viz Trio User Guide).
- **Start/Stop/Continue**: Controls the scroll (used in the local preview and the live controls for playback).
- **Execute command**: See **Command**.

See Also

- **Create New Scroll** in the Viz Trio User Guide.
- Control Object plug-in
- Trio Scroll item plug-in

13.1.37 Wall

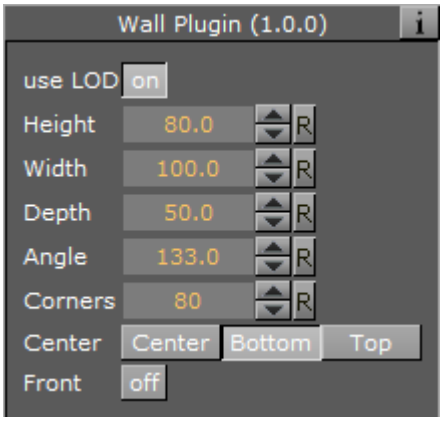
The wall object is a curved wall. The size and curving of the wall can be customized.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

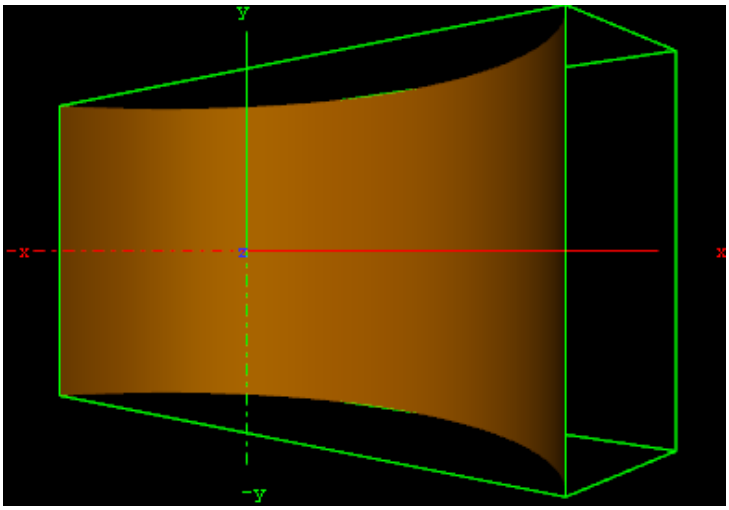
- Wall Properties
- To Create a Wall

Wall Properties



- **Use LOD**: Enables/disables dynamic level of detail.
- **Height**: Sets the height of the wall.
- **Width**: Sets the width of the wall.
- **Depth**: Sets the depth of the wall curving. The higher this value is set, the more curved the wall will show.
- **Angle**: Sets the angle of the walls curving. Use this parameter together with Depth: To achieve the wanted curving of the wall.
- **Corners**: Sets the number of internal corners the wall should be built up from.
- **Center**: Sets the center axis of the object. Choose between Center, Bottom or Top.
- **Front**: Lets you switch between front and back.

To Create a Wall



1. Create a new group and add the Wall plug-in to it.
2. Open the Wall editor and set the following parameters:
    - Set Height to 200.0.
    - Set Width to 400.0.

- Set Depth to 400.0.
- Set Center to Center.

3. Add a material and/or a texture to it.
4. Open the Transformation Editor and set Rotation Y to -45.0.

13.1.38 Wave

The wave allows you to create a wave sequence on a flat surface. The properties of the sequence are visualized in a graphical display in the wave editor.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Default

This section contains the following topics:

- Wave Properties
- To Create a Wave

Wave Properties

- **Size X** and **Y**: Sets the size of the rectangle where the waves show.
- **Geom Type**: Allows you to select between **Circular** and **Linear** waves.
- **Stretch**: Allows you to customize the form of the circular waves. By altering the stretch **X** and **Y** values, you can for instance obtain an oval or a compressed form of the waves.
- **Time**: Is the time-line for a wave sequence. You see a thin red vertical line moving in the graphical display as you alter the time value. You can also alter the value by dragging the red line.
- **Tess**.: Sets the tessellation / level of detail.
- **Angle**: Sets the angle of the waves when a linear mode is chosen.
- **Ref.Pnt X** and **Y**: Sets the position of the waves starting point.
- **W. Length**: Sets the length of the waves.
- **Damp. mode**: Enables damping of the waves. The amplitude of the wave is reduced by the damp length calculated from the center of the wave.
- **Damp. Length:** Sets the length the distance that is used to damp the waves in the **Damp. mode** function.
- **Offset**: Sets an offset value of the **Ref.Pnt.**
- **Ampl**: Amplifies the waves, resulting in both bigger and higher waves.
- **Inverse OFF/ON:** Inverts the wave curves.
- **Envelope Generator:** Draws the wave manually when enabled. When disabled the wave is built by a sine wave function.

- **Position X** and **Y**: Allows you to edit the position values of the wave control points. Select a point by clicking on it. Its color changes to red to show that it is selected. Now you can drag it around.
- **Lock**: Allows you to decide whether the points controlling the crest of the waves should be edited in a locked manner, with the points controlling the troughs of the waves or conversely, if they are to be edited separately.
- **Spline Curve:** The control points of the spline curve, that shows the wave sequence, can be edited by using the cursor. Click on a point and drag it to the required position.

To Create a Wave

1. Create a new group and add the Wave plug-in to it.
2. Open the Transformation Editor and set Rotation X to -75.0.
3. Add a material and/or a texture to it.
4. Open the Wave editor and set the following parameters:
    - Set Size X and Y to 350.0.
    - Set W.-Length to 8.0.
5. Click the Lock button to unlock the spline curve editor.
6. Play with the spline curve handles, Time, Position X and Y values to see how the wave behaves with different settings.

13.2 PixelFX Plug-Ins

The Lens Flare plug-ins, which are part of the PixelFX plug-ins set, simulate the effect of streaks and spots of light, caused in real cameras when light enters and bounces inside a camera lens. This plug-in set contains geometries that simulate different shapes, which can be used for lens flare or simple flare effects.

The Flare draws itself as a combination of a geometry shader and a pixel shader. When a Flare Container is created a non-editable pixel shader will be added to the created Flare container.

In addition, the Expert Container plug-in is automatically added and configured in such a way as to cause the shape used to always be seen in front of all other objects, blending with the rest of the scene to resemble the effect of light bouncing off the lens.

Note: Lens flare plug-ins reside in their own containers. Applying these plug-ins to a container with geometry will overwrite the geometry and disable any shaders in child containers.

The baseline plug-in within this set is the pxLensMulti plug-in, which is a container plug-in, whose purpose is to manage other geometry plug-ins. The one exception in this set is the pxLensEnergyBolt, which does not simulate lens flare, but rather the effect of electricity or lightning and it is not managed by pxLensMulti.

The following Geom plug-ins are located in the PixelFX folder:

- Lens Flare Geometries
- pxLensEnergyBolt
- pxLensRays

See Also

- PixelFXLenseFlare

13.2.1 Lens Flare Geometries

The following lens flare plug-ins are simply different shapes for the composition of final lens flare effects. Each has its own parameters to define its appearance.

- pxLensGlowBall
- pxLensGlowSpikes
- pxLensPolyitem

- pxLensRandomPoly
- pxLensSpark
- pxLensStripes

Each plug-in can take on very different appearance depending on how the parameters are adjusted. The best advice for working with these plug-ins is to simply play around with the parameter configurations. The following example shows varying manifestations of the pxLensRandomPoly.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> PixelFX

See Also

- PixelFXLenseFlare

13.2.2 pxLensEnergyBolt

pxLensEnergyBolt produces the effect of electricity-like lightning between two points in space. These points may be defined manually by entering positions, or by tracking the positions of other containers. The parameters of Radius, Core, End Caps, Distortion and N define the effect's look and Speed defines change rate.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> PixelFX

See Also

- PixelFXLenseFlare

13.2.3 pxLensRays

pxLensRays can be used to produce the effect of rays. The rays can be static or move randomly based on the Speed parameter. The shape is defined with the following parameters:

- **Angle**
- **Width**
- **Size**
- **Inner radius**

The number or density of the rays is defined by the Rays parameter. Rays can receive their color from a predefined or user defined gradient ramp, or from an image, by dragging the image into the Image box.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> PixelFX

See Also

- PixelFXLenseFlare

13.3 Dynamics

The following Geom plug-ins are located in the Dynamics folder:

- Cloth
- Cloth Flag
- Flag

13.3.1 Cloth

Cloth provides a simulation of an elastic vertex system. The vertex system can be configured with many different parameters. Most of the parameters depend on each other, and some must be set in conjunction with others. The stress of the system will be mentioned below, and the rule is that the greater the stress is the greater the chance is that the system will run amok (unstable). With low stress factors the system is completely stable.

Adding the Cloth plug-in to a container will also add the Expert plug-in.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Dynamics

This section contains the following topics:

- Cloth Properties
- To Create a Cloth Effect

Cloth Properties

- **Reinitialize:** Restarts rendering of the object.
- **Elastic:** Changes the elastic properties of the object. If set it to `0.1`, the object behavior will resemble that of latex, and at `1.0` more like cloth.
- **DampingFct:** Specifies how much speed each vertex should loose. The lower this factor is the less speed the object will lose.
- **Pause:** Pauses all calculations and freezes the system's animation.
- **Tessellation direction U and V:** Sets the tessellation in the two directions on the objects' surface. The calculation time needed will increase with more vertices.

- **Speed:** Determines how many calculations are needed per frame. The less calculation done the slower the object will move. If the number of calculations are increased, the object moves faster.
- **Wind X, Y and Z:** Gives the speed and direction of the wind affecting the object.
- **TurbulenceSpeed:** Gives the speed and direction of the wind turbulence affecting the object.
- **TurbulenceFactor:** Multiplies the wind speed.
- **Gravity X, Y and Z:** Defines gravity affecting all vertices of the object.
- **Scaling u (width):** Changes the width of the rectangle which is including the animated cloth without increasing the number of triangles.
- **Scaling v (height):** Changes the height of the animated rectangle which is including the animated cloth without increasing the number of triangles.
- **Draw normals:** Shows the normals for each triangle.
- **Draw force:** Shows the appealing wind in color.

To Create a Cloth Effect

1. Add the **Cloth** plug-in to a Container.
2. Add a **material** to the same Container.
3. Set **WindX** to 2.0.
4. Click the **Reinitialize** button to see the result.

See Also

- Expert

13.3.2 Cloth Flag

The Cloth Flag plug-in provides a simulation of an elastic vertex object. The vertex object can be configured with many different parameters. Most of the parameters depend on each other, and some must be set in conjunction with others. Adding the Cloth Flag plug-in to a container will also add the Expert plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Dynamics

This section contains the following topics:

- Cloth Flag General Properties
- Cloth Flag Geometry Settings
- Cloth Flag Move Settings
- Cloth Flag Advanced Settings

Cloth Flag General Properties

- **Wind Rotation, Elevation and Force:** These three factors combined gives the speed and direction of the wind affecting the flag. The allowed values are:
  - **Wind Rotation:** From `-720.0` to `720.0`
  - **Wind Elevation:** From `-90.0` to `90.0`
  - **Wind Force:** From `0.0001` to `100.0`
- **Gravity Direction**: Vectors that define in which direction gravity is affecting the vertices of the flag:
  - **U:** Horizontal.
  - **V:** Vertical.
  - **None:** Gravity is not applied.
- **Pause:** Setting **Pause** to **On** pauses all calculations performed and the animation in both the scene editor and *On Air* output.
- **Reinitialize:** Restarts rendering of the object.

Cloth Flag Geometry Settings

- **Tessellation direction U and V:** Sets the tessellation in the two directions on the systems surface. Calculation time increases with more vertices.
- **Scaling U (width):** Changes the width of the rectangle which is including the animated cloth without increasing the number of triangles.
- **Scaling V (height):** Changes the height of the animated rectangle which is including the animated cloth without increasing the number of triangles.
- **Reinitialize:** Restarts rendering of the object.

Cloth Flag Move Settings

- **Translate X, Y and Z:** Offsets the position output using the position values set for the container.
- **Rotate X, Y and Z:** Offsets the rotation output using the rotation values set for the container.
- **Scale:** Offsets the scale output using the scale value set for the container.
- **Reinitialize:** Restarts rendering of the object.

Cloth Flag Advanced Settings

- **Turbulence Speed:** Determines how fast the direction of speed will oscillate.
- **Turbulence Factor:** Multiplies wind speed.
- **Elastic:** Changes the elastic properties of the system. If set to `0.1`, the rendered output will have properties resembling latex, and at `1.0` more like cloth.
- **Damping:** Specifies how much speed each vertex should lose.
- **Simulation Quality:** Reduces or increases the number of calculations performed for the plug-in, for performance reasons.

- **Wireframe:** Shows the wireframe for the plug-in.
- **Draw normals:** Shows the normals for each triangle.
- **Draw force:** Shows the appealing wind in color.
- **Reinitialize:** Restarts rendering of the object.

See Also

- Expert

13.3.3 Flag

The Flag plug-in provides a simpler simulation of an elastic vertex object than the Cloth Flag plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Dynamics

This section contains the following topics:

- Flag General Properties

Flag General Properties

- **Width:** Defines the width of the object. Accepted values are from `0.0` to `200`
- **Height:** Defines the height of the object. Accepted values are from `0.0` to `200`
- **Amount:** Sets the intensity of the effect. Accepted values are from `0.0` to `200`
- **Wavelength:** Defines the length between the animated waves. Accepted values are from `0.0` to `200`
- **Wave Speed:** Defines the speed of the animated waves. Accepted values are from `-100` to `100`
- **Turbulence Speed:** Defines the speed of the turbulence in the animation. Accepted values are from `-100` to `100`
- **Y Stretch:** Defines the anisotropic value of the Y-axis. Accepted values are from `0.2` to `100`
- **Tessellation X:** Defines the amount of tessellation on the X-axis. Accepted values are from `1` to `100`
- **Tessellation Y:** Defines the amount of tessellation on the Y-axis. Accepted values are from `1` to `100`
- **Plane:** Selects the dimensional plane for the rendered object.
- **Fixed Edge:** Selects which edge should be fixed, or none. The fixed edge will act as a boundary for the animation.
- **Direction:** Automates the animation direction angle if set to **`Auto`** (default).
- **Dir Angle:** If **Direction** is set to `Custom`, provides the designer with the option of defining the animation direction angle. Accepted values are from `-720` to `720`

13.4 Primitives

The following Geom plug-ins are located in the Primitives folder:

- N Quad

13.4.1 N Quad

Use this plug-in to create a rectangle with different attributes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> Primitives

This section contains the following topics:

- N Quad Properties
- To Create an N Quad

N Quad Properties

- **Use LOD:** Enables/disables dynamic level of detail.
- **Tesselation U**: Sets the degree of detail.
- **Tesselation V**: Sets the degree of detail.
- **Orientation**: Switches the orientation from **TopBottom** to **LeftRight**.
- **Top/Left Width**: Sets the width.
- **Bottom/Right Width:** Sets the width.
- **Height/Width**: Sets the height.
- **Depth**: Sets the depth.
- **Center X:** Switches the center in X between **Center**, **Top** and **Bottom**.
- **Center Y**: Switches the center in Y between **Center**, **Left** and **Right**.
- **Shearing**: Sets the shearing value, so the bottom is shifted.
- **Image Mapping:** May be set to **Stretch** or **Tile**.
- **Crop Top/Left**: Crops the top.
- **Crop Bottom/Right:** Crops the bottom.
- **Border Geometry**: Enables/disables the top and bottom border width parameters. When enabled, only the border outline is visible.
- **Top Border Width**: Sets the width of the top border.
- **Bottom Border Width:** Sets the width of the bottom border.

To Create an N Quad

1. Create a new group and add the N Quad plug-in to it.
2. Add a material and/or a texture to the group.
3. Open the transformation editor and set Rotation X and Z to -45.0.
4. Open the N Quad editor and set the following parameters:
    - Set Orientation to Left-Right.
    - Set Top/Left Width to 200.0.
    - Set Bottom/Right Width to 0.0.
    - Set Height/Width to 300.0.
    - Set Depth to 100.0.
    - Set Center X and Y to Center and Bottom, respectively.

13.5 RealFX Plug-Ins

The RealFX plug-in set enables you to create particle effects in Viz Artist.

Particle systems are a computer graphics technique to simulate certain physics-based effects, which are otherwise very hard to reproduce with conventional rendering techniques. Examples of such effects which are commonly replicated using particle systems include fire, explosions, smoke, weather effects, sparks, falling leaves, dust, meteor tails, or abstract visual effects like glowing trails, magic spells, etc.

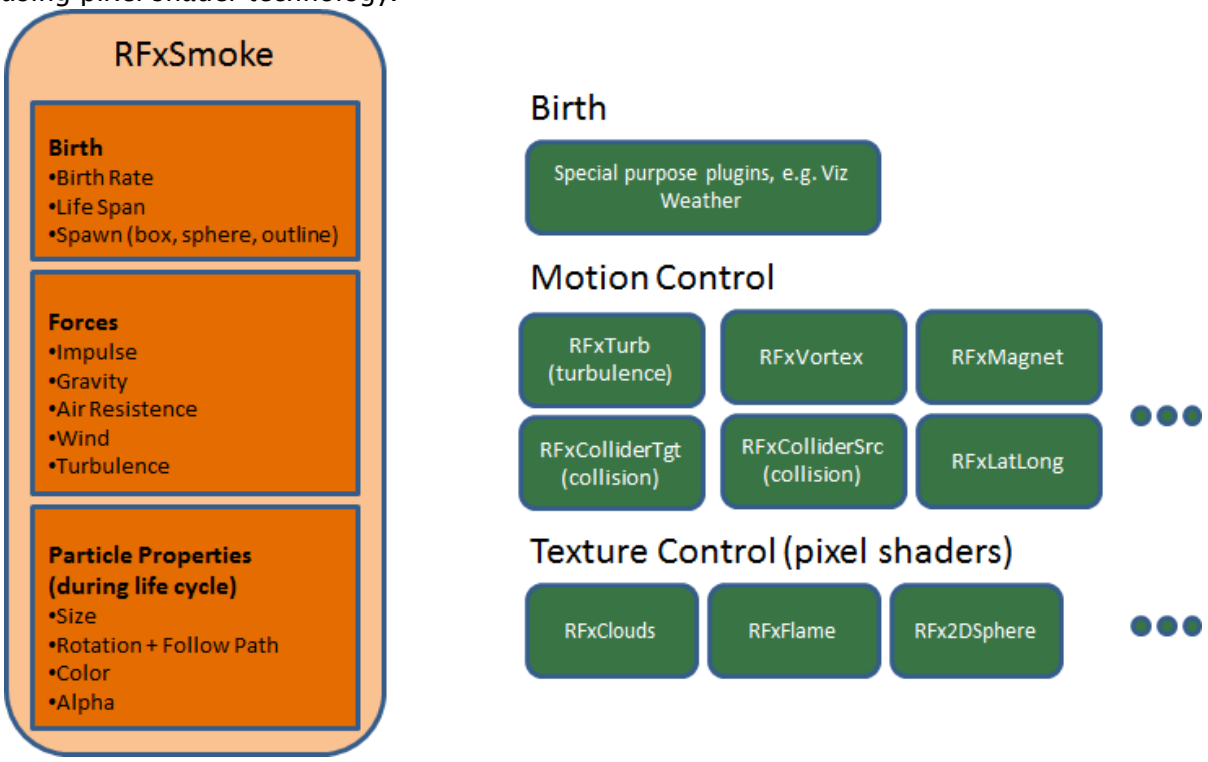
The particle effects in Viz Artist/Engine run in real-time, meaning that there are a few inherent constraints that must be taken into account when considering best practices for employing this plug-in set. For example, there is a trade-off between the number of particles and performance optimization; more generally there needs to be a considered balance between performance and visual quality.

RFxSmoke is the baseline plug-in within the RealFX plug-in set. The remaining plug-ins in this set are applied on top of RFxSmoke in any given container. RFxSmoke includes built-in functionality and the ability to host the additional functionality contained in the other plug-ins in this set. Part of the built-in functionality, e.g. turbulence, is kept for compatibility with previous version of Viz Artist.

There are three categories of additional plug-ins:

- **Birth plug-ins:** Refers to where the particles are spawned.
- **Motion control plug-ins:** Governs the position, direction, velocity, size and color of each particle.

- **Texture control plug-ins:** Affects the texture mapping and the “look” of each particle by using pixel shader technology.



The following Geom plug-ins are located in the RealFX folder:

- RFxSmoke

See Also

- RealFX in Shader plug-ins
- pxColorWorks in Container plug-ins

13.5.1 RFxSmoke

RFxSmoke is the baseline plug-in within the RealFX plug-in set, which creates realistic visualization of smoke, fire, explosions and much more (the level of creative skill is the only limitation).

The basic function of the object is that it emits particles from a point and moves them in some direction until they die or fade away. With the use of different texture mappings and settings, many effects can be achieved.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> RealFX

This section contains the following topics:

- RFxSmoke Properties
- Spawn

- Impulse
- Forces
- Size
- Rotation
- Color
- Alpha

RFxSmoke Properties

- **Number of particles:** Sets the number of particles to be active at the same time. The number of **required** particles is calculated by the **Birth Rate** multiplied by the **Life Span**. If this is lower or equal the number of particles there will be a constant flow of particles, otherwise the plug-in will collect particles until the minimum number of particles has been reached and then spawn them.
- **Birth Rate**: Sets the birth rate of particles per second.
- **Life Span**: Sets the life span of the objects in seconds. In the field to the right of Life Span, set a percentage over which the life span of the particles will vary. If **Life Span** is set to three and the variation to 50%, the life span of particles will vary randomly between 1,5 and 4,5 seconds. This creates more realism, since particles in real smoke or fire does not have equal life span.
- **Preroll (fields)**: Sets the amount of fields to calculate before RFxSmoke goes on air.
- **Reference**: Sets the particles reference to either **Container** or **World**. If **Container** is selected, after emitted the particles will relate their position to the containers position. If the emitter object is moved, the particles will alter their position correspondingly. If **World** is chosen, the particles will maintain their position and path even if the emitter is moved, they have their own world coordinates.
- **Billboard:** If **Particle** is selected, the particles will maintain a frontal position against the camera when the camera moves.
- **Sort:** Enables/disables sorting.
- **Array**: Enables/disables the use of array.
- **Reset Particles:** Removes all already emitted particles.
- **Show Forces**: Show help lines that show the forced settings of some of the parameters.
- **Freeze Motion:** Halts the emitting process.

Spawn

Spawn parameters are related to the position of the smoke emitter. In the Spawn panel define the shape and size of the area in which the emitter should emit the particles.

- **Type:** Defines the shape or type of spawn area. Click **Cube**, **Sphere**, **Container** and **Image** to set if the spawn area should be shaped as a **Cube** or a **Sphere**, or if a container or image should be used.

- **Size:** Sets the size, selected from the **X**, **Y** and **Z** values. If **Sphere** is selected and only the **X** value is set, the emitter will create particles while it moves randomly along a line. If a **Y** value is also set, the emitter will move within a circle. If a **Z** value is set, the emitter will move within a sphere. If **Cube** is selected the same applies, but the stages would then be a line, a rectangle and a Cube.
- **Spawn Range:** Sets the range, within the **Sphere** or **Cube**, the emitter should use to distribute the particles. Default is 100%, where it uses the whole spawn area. If the value is reduced, a section, which starts from the middle of the spawn area, where no particles are being emitted, will increase correspondingly. If the value is set to 1%, the emitter only emits particles at the outer edge or surface of the cube or sphere.

> ⚠ **Tip:** Container is especially valuable when animating a Scene, as the smoke will be emitted from the outline of an animated Container. To set the Container, which the smoke should emit from, drag the Container from the Scene Tree onto the drop zone.

> ⚠ **Note:** A Container must be 2D, such as a rectangle, circle, square, etc.

Impulse

In the Impulse panel set the parameters for how particles are emitted.

- **Speed**: Sets the speed the particles have when they are emitted. In the field to the right of **Speed**, set a percentage over which the particle speed will vary.
- **Rot X:** Sets the X axis angle of the opening in the emitter, the emitter hole.
- **Rot Z**: Sets the Z axis angle of the opening in the emitter, the emitter hole.
- **Angle X:** Sets the X axis opening angle of the emitter hole. It can be done locked with the Z axis, or independently. Click **Lock X/Y** to The button to the right of the value field to lock or unlock the axes.
- **Angle Z**: Sets the Z axis opening angle of the emitter hole.
- **Twist**: Creates a twist of the opening angels of the emitter hole. It can also be described as a rotation of the emitter around the Y axis.
- **Align to Center:** Disables all the above impulse parameters and aligns the particles above the emitter.

Forces

In the Forces panel are the parameters to create environmental effects, like wind, gravity and air resistance.

- **Gravity:** Sets the degree of gravity, which affects the path of the particles. If set high, the smoke will go downwards. With a negative gravity the particles will rise faster.

- **Air Resistance:** Sets the degree of air resistance to creates a force that prevents the particles from rising.
- **Turbulence X**, **Y** and **Z:** Defines a simulation of turbulence on the particles as they rise. The turbulence effect is achieved through a jittering of the particles. The **X**, **Y** and **Z** values set the axis of the jittering movement.
- **Wind:**
  - **Force:** Sets a level cross wind. A positive value blows from the left to towards the right, a negative the other way around. For example: If a horizontal wind force is not required, use the two next parameters to adjust the angle of the wind.
  - **Rot X/Z:** Sets the rotation of the wind force on the X and Z axis. Modify these values to change the direction of the wind defined in the **Wind Force**.

Size

In the Size panel are the parameters to the size of the particles at the moment of their birth and death. Between the time of birth and death, the size is a product of a linear interpolation.

- **Birth X/Y:** Sets the size of each particle at the time it is being emitted.
- **Death X/Y:** Sets the size of each of the particle at the time it fades away and dies.
- **Lock X/Y:** Locks or unlocks the axes.

Rotation

In the Rotation panel, you can define a static rotated position or a spin of the particles. Both the static rotated position and the spin can either be set as an offset equal on all particles or as a range where the particles will be rotated or span randomly. To see the effect, set the number of particles to a very low value and alter the rotation parameters.

- **Initial:**
  - **Offset:** Sets the degree of rotation the particles should have. This rotated position is static and will remain throughout the life of each particle.
  - **Range**: Sets a degree range of rotation the particles should have. This rotated position is static and will remain throughout the life of each particle. If for example the value is set to 30, the static rotated position of the particles will vary randomly between 30˚ and -30˚.
- **Spin:**
  - **Offset:** Sets the degree of spin each particle should have throughout its life time. The higher the value, the higher the number of spins. A positive value creates an counterclockwise rotation, a negative creates a clockwise.
  - **Range**: Sets a spin degree range for the particles. They will spin randomly between the parameter value and the same value mirrored through zero.
- **Follow Path**: If the RFxSmoke object is animated, enable this option to make the particles align their position to the animation path in the same way as is the case with the follow path option in the transformation editor.

Color

In the Color panel set the color of the particles throughout their life cycle. The particles can be set to change their color through stages from their birth to their death by the addition of more Key Frames (single click the color stage).

The left side of the color stage is at birth time and the right is at death time. By default there are two color Key Frames at each end, that can be clicked on, to set their color. The color range between two Key Frames will be an interpolated transformation. More Key Frames can be added to create a more complex color transformation between particles birth and death.

To delete color Key Frames, select them and press <Delete> or <Backspace>, on the keyboard.

Alpha

In the **Alpha** panel set the alpha values of the particles throughout their life cycle. The alpha editor contains a curve with points that can be moved, in both X and Y directions, to change the form of the curve. The X axis is the life time of the particles and the Y axis is the alpha value.

To select a point, or create a new one, left-click on the graph and drag the point to the required X and Y position. To delete a point, select it and press <Delete> on the keyboard.

See Also

- Create Animations

13.6 Ticker

The following Geometry plug-ins are located in the Ticker folder:

- Scroller

13.6.1 Scroller

The Scroller plug-in is a geometry plug-in that generates a dynamic line of scrolling items, mostly used for creating scrolling tickers.

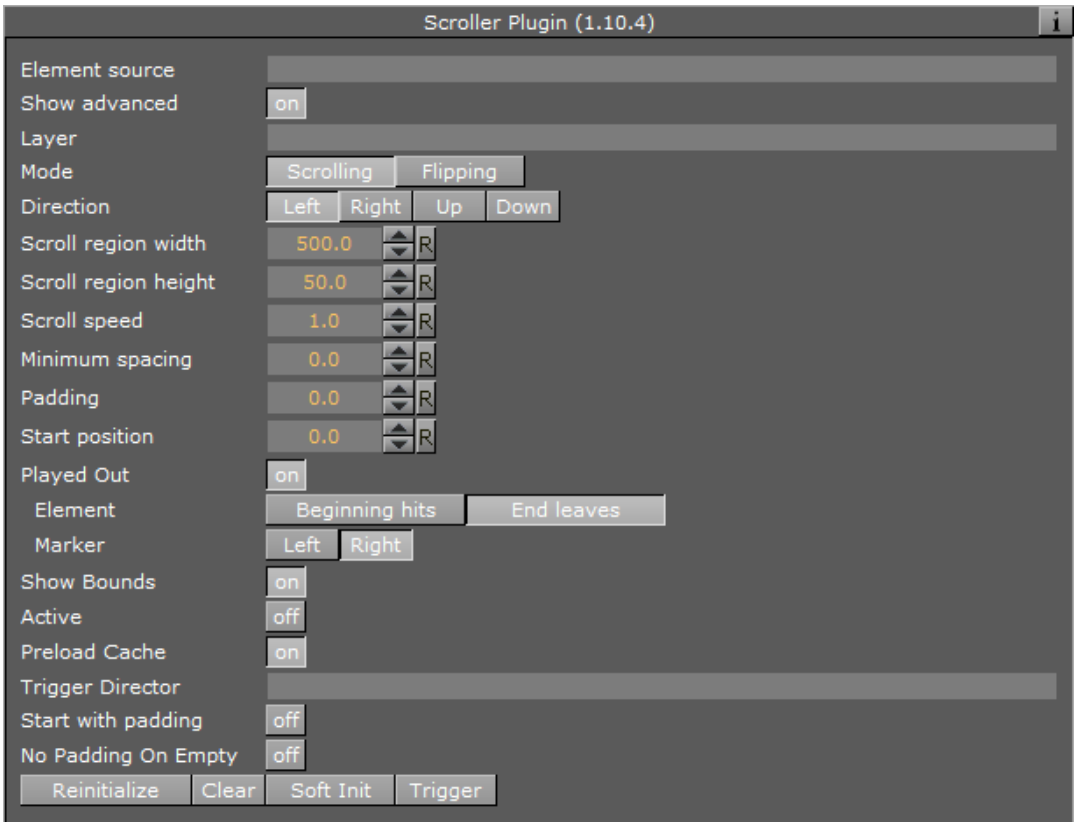
When used with Viz Ticker the scene is designed as a transition logic scene where each data item is an instance of a scene template designed as part of the background scene. This is unlike traditional transition logic scenes where each data item would be based on a specific foreground scene controlled by the background scene. Based on the scene template that resides with the background scene the Scroller is able to construct and configure the scene templates on demand for Viz Ticker that manages and reuses them.

To design a scrolling carousel, the Scroller and one or more scene templates must be added to the background scene. The scene templates serve as prototypes for the containers that the Scroller plug-in generates. Alignment of the generated containers is done by editing the alignment of the scene templates.

⚠ **Tip:** When saving a Scroller scene, remember to hide the scene template containers. This will prevent the scene template from being shown on-screen.

⚠ **Note:** This plug-in is located in: *Built Ins -> Geom plug-ins -> Ticker*

Scroller Properties



- **Element Source:** Element descriptions are added here. Descriptions must be entered in the element source syntax, or in the name of a Shared Memory segment that uses the Shared Memory protocol. When designing and testing a scene, the element source syntax is used to make the system scroll instances of the design containers.
- **Show Advanced**: Shows additional parameters. Defaults to `Off`.
- **Layer**: Defines the layer name for the scroller. The layer name must be identical to the name of the director that holds the in and out stop points for the carousel.
- **Mode:** Defines if elements should be shown as scrolling or flipping elements for scrolling and flipping carousels, respectively.
- **Direction:** Defines the direction of the scroller.
- **Scroll Region Width**: Defines the width of the scroller. This option is only relevant for left-to-right/right-to-left scrollers.

- **Scroll Region Height:** Defines the height of the scroller. This option is only relevant for up-down/down-up scrollers.
- **Scroll Speed**: Defines the speed of the scroller. The x position of the scrolling items will be decremented with this value for each frame that passes.
- **Minimum spacing:** The minimum spacing between scrolling elements.
- **Padding**: Defines an extra padding value that will be added to the width of the bounding box of an item when calculating the spacing between the items. Negative padding can be used to make items overlap.
- **Start Position:** Set the initial position as a percentage value in the range `0-100`. A value of 100 means the scroller starts fully populated with content. The default value is `0`.
- **Played Out**: Tracks movement of scrolling element and notify ticker service on specified position.
    - **Element**: Position of the scrolling element to be checked with the marker. When set to `Beginning hits`, notify when beginning of element hits the marker. When set to `End leaves`, notify when end of element leaves the marker.
    - **Marker**: Position of the marker related to side of Scroll region.
- **Show Bounds**: If enabled, the area in which the scroller tries to maintain the illusion of a continuously scrolling flow of items is visible on-screen.
- **Active:** Scrolling or flipping is available when enabled.
- **Preload Cache**: Use this on configurations with no ringbuffer, to move all spike-load related to item creation to the initialization phase. Designs named `DESIGN_designname_Xnnn` will have `nnn` instances created in advance, where `nnn` is an integer.
- **Trigger Director**: Director to be triggered when the **Trigger** button is pressed.
- **Start with padding**: Sets padding for the first scrolling item, for example when **Start Position** is set to `100 %` or the scroller container is hidden at the beginning.
- **No Padding On Empty**: Does not create any extra spacing between elements for elements with zero bounding box.
- **Reinitialize:** Reinitializes the plug-in by deleting all cached objects and reconstructing the initial node structure. This parameter can be used to update the scroller after changing for example the item source.
- **Clear**: Clears all current ticker items in the rendering window.
- **Trigger**: Triggers the director named in the Trigger Director field provided that there are items in queue to be scrolled in. If no director is specified, the scroller is activated.

> ⚠ **Tip:** The Viz Ticker User Guide contains information on design conventions when using the Scroller plug-in.

See Also

- Viz Ticker User Guide
- Control plug-ins
- Toggle plug-in
- Transition Logic

13.7 Visual Data Tools

13.7.1 Overview

Visual Data Tools (VDT) show any kind of statistic data. All the VDT plug-ins use the Shared Memory transport channel.

Shared Memory is a map which has user defined variables, indexed by a string, also known as a key-value pair. There are three types of Shared Memory Maps.

For more information about Shared Memory see Shared Memory (SHM).

The VDT plug-in suite consists of:

- **Geometry plug-ins**: Used to draw the chart (detailed in this section)
- Container plug-ins: Used to label, data scale, etc., the chart
  To learn more about the use of Visual Data Tools, download a tutorial on www.vizrt.com under the Training section.

The following Geometry plug-ins are located in the Visual Data Tools folder:

- Advanced Bar Chart Creation
- Area Chart
- Bar Chart
- Line Chart
- Pie Chart
- Scatter Chart
- Stock Chart

See Also

- Container plug-ins
- Geometry plug-ins
- Scripting
- Advanced Bar Chart Creation
- Area Stack
- Bar Stack
- Data Fit
- Data Import
- Data Label
- Data Storage
- Line Stack
- Tutorial on www.vizrt.com under Training

13.7.2 Advanced Bar Chart Creation

This section outlines some of the more advanced features of the **Bar Chart** geometry plug-in. The example provided is aimed at designers familiar with the Viz Artist user interface and prior knowledge of advanced scene design techniques, such as the use of control plug-ins and scripting.

This brief tutorial consists of the following sections:

- Establishing the Basic Bar Chart
- Adding a Script to Control Data
- Calculating Scale Manually, for External Applications, or for Scripts
- Adding Labels to the Bars
- Animating the Bars
- Animating the Labels
- Adding Color

Additionally, feel free to download a more extensive Tutorial from www.vizrt.com under Training, which also covers the other VDT plug-ins.

Establishing the Basic Bar Chart

In this first section, the basic bar chart will be established and set up to be controlled by an External Control Application, such as Viz Trio or Viz Pilot. In an environment where the data will be modified directly through the plug-in, there will be no need for the shared memory or script as outlined below.

1. Create a new scene and add an empty container. This container will be used as a parent container. Rename the container to `object`.
2. Add the **Bar Chart** plug-in to a sub-container in the scene tree and name it `BarChart`, then open the plug-in properties.
3. In the **Data** field, add a series of comma-separated values. Here, the values `10, 70, 20, 30` are used.

   > ⚠ **Note:** By using the **Data Delim.** field, the data delimiter can be changed from comma to any other delimiter, for example the pipe character `|`, to suit the needs of the production environment the scene will be used in.

4. Currently, the **Shared Mem.** parameter is set to `Inactive`. This means that the data used by the plug-in is being read from the **Data** field, as set in step 2. However, in most cases some sort of shared memory would be used in order to share the same data with other plug-ins, in particular the **Data Label** plug-in that will be introduced in a later step.

   In this example, the **Scene** shared memory will be used. This Shared Memory setting does not interfere with any other scene(s) or machine, only the local scene. By clicking the `Scene` button, the field **Key Data** becomes available, where the name of the shared memory used for the bar data must be specified. An external control application, such as Viz Pilot or Viz Trio, can now be used to set the values that are to be represented by the bar chart. Enter `barData` as the shared memory name.

5. Add an empty container to the scene tree, and then the **Control Parameter** and **Script** container plug-ins to the newly created container. Rename the container to `ValuesControl` and open the Script plug-in properties.

Adding a Script to Control Data

1. Next, a script allowing the data to be controlled with the **Control Parameter** plug-in needs to be added:

```
Dim oldKey As String = ""
Sub OnInitParameters()
RegisterParameterString("barsValues", "Bar Values", "1,2,3,4,5,6", 40, 1000, "")
RegisterParameterString("barsSMName", "Shared Mem Name", "barData", 40, 1000, "")
End Sub

Sub OnParameterChanged(parameterName As String)
If parameterName = "barsSMName" Then
If oldKey <> "" Then Scene.Map.DeleteKey(oldKey)
Scene.Map.CreateKey(GetParameterString("barsSMName"))
oldKey = GetParameterString("barsSMName")
Else
Scene.Map[GetParameterString("barsSMName")] = GetParameterString("barsValues")
End If
End Sub

Sub OnInit()
Scene.Map.CreateKey(GetParameterString("barsSMName"))
Scene.Map[GetParameterString("barsSMName")] = GetParameterString("barsValues")
End Sub
```

> ⚠ **Tip:** As alternative to a string with a data delimiter, the data source could also be a *Double-Array*. This can be achieved by selecting `Array` as **Transfer Mode** in the Data Bar plug-in properties. In that case, instead of assigning a string, an `Array[Double]` data type would have been assigned in the example script above. This can be convenient if the data is provided by an external source, for instance a database or as the result of parsing a text file.

After entering the above script into the script editor, click the **Compile & Run** button. The scene tree should now look like this:

2. Open the **Control Parameter** plug-in properties. Change the **Field Identifier** to `1001`, set the **Data Type** to `Text`, and enter `SCRIPT*INSTANCE*barsValues` in the **Parameter** field. By opening the Control Objects tab in the Control area, the *Bar Values* can now be changed directly. Any changes here should be reflected immediately on the rendered output in the **Preview** window when pressing `Enter`.

3. Apply a material on the BarChart container. The shape of the rendered bars can be controlled by adjusting the **Corners**, **Tesselation Length**, and **Bevel** parameters in the Bar Chart plug-in properties. For a classic 2-dimensional bar, a **Corners** value of `2` will suffice. For a smooth cylinder shape, the corners value can be increased up to a maximum of `1000`.

Activate *Wireframe* rendering mode in the preview window by clicking the `W` button in the scene editor to get a better understanding of these parameters. This is how it may look with Corners set to `25`, a **Tesselation Length** of `20`, with **Bevel**, **Bevel Top** and **Bevel Bottom** set to `On`, **Bevel size (%)** to `15` and **Bevel Steps** to `10`:

4. Drag a **Rectangle** container plug-in to the scene tree to create a new container, and add the **Expert** container plug-in. In the Expert plug-in properties, select `Outline` as **Drawmode** to create a bounding box.
   Often, it is useful to fit the entire data into a given screen bounding box. This can be achieved positioning the newly added rectangle on the screen, representing the area where the graph should be drawn. Here, the rectangle has width of `500` and a height of `300`. It is positioned at `0.0` in Z-space, as is also the container holding the Bar Chart itself.

   Open the rectangle container's Transformation Editor and check the effective width and height of the boundary rectangle:

   
5. To match the width of the graph to the bounding box, open the **Bar Chart** plug-in properties and change the **Chart Width** to `Total` and the **Bar Width** to that of the bounding rectangle (`500.0`). Then set **Horizontal Alignment** to `Center`.

   > ⚠ **Tip:** The **Max. Width** parameter defines how thick a bar can become. If the number of input values are reduced, the bars width will increase within the defined **Bar Width** area. If there are only two bars left, they will look really big. The Max. Width parameter is useful to limit the impact of this behavior.

6. Mapping the graph to fit the whole height of the bounding area is done by enabling some of the properties inherent of the Bar Chart plug-in. In the property panel, set **Data Fit** and **Data Auto Scale** to `On`, then set **Data Fit Size** to `Current`. Then either type in, or click and drag, the **Data Start** and **Data Stop** values as required. The sum of these two should equal the height of the bounding box.

   Open the **Control Object** tab in the **Object area**, and enter the values used for the example data scale calculation above, `-14, 70, -30, 12, 15, 20, 5`, as values for the **Bar Values** object. The preview is automatically updated. The result should look like this, with the Bars perfectly matching the bounding area:

Please go to straight to Adding Labels to the Bars to continue with the tutorial. The following section is intended as reference for users wanting to perform this scaling manually, or by an external application or a script.

Calculating Scale Manually, for External Applications, or for Scripts

The minimum and maximum values of the data needs to be known to be able to calculate the required scale value. For example, if using the values `-14, 70, -30, 12, 15, 20, 5` as input data, the minimum value is `-30` and the maximum value is `70`. If negative values are not used, set the minimum value to minimum to `0.0` (more mathematically speaking; `min=minimum(0.0, minimum(inputdata))`). For the example values above, this results in a range of **100** (from min to max). These 100 units must be matched to the *bounding box height*, which is **300**.
Dividing `300` by `100` returns a scale value of `3.0`.

Set **Data Fit** to `On` and enter this value in the **Data Scale** field (1):

Open the **Control Object** tab in the **Object area**, and enter the values used for the example data scale calculation above, `-14, 70, -30, 12, 15, 20, 5`, as values for the **Bar Values** object. The preview is automatically updated. Notice how the graph aligns to the bounding box size, however, it is not centered in the middle.

To center the graph within the bounding box, a *data offset* translation value must be calculated. The formula for calculating this value is:

where *dataScale* is the value calculated in the previous step, and *height* is the bounding height (here: `300`). Using the values in this example, the formula would be

Set **Data Origin Offset** to `On` and enter the resulting value, `-60`, in the **Data Offset** field. The result should look like in step 11, with the bars perfectly matching the bounding area.

Adding Labels to the Bars

In order to have labels representing the bar value above each bar, the **Data Label** container plug-in is used.

1. Add the **Data Label** plug-in to the **BarChart** container and open the plug-in properties. Select `Scene` as **Shared Mem.** and enter `barData` in the **Key Data** field, then set the **Copy Container** parameter to `On`.

2. Create a new container as a child of the **BarChart** container, and add a font and a material to the new container. Rename the container to `label`. Set the size and text parameters as desired. Here, the font container's **Scaling** is `.25`, and the font orientations are set to **Orientation center** and **Vertical orientation first line**.

3. Open the **Bar Chart** plug-in properties and set **Pos. Container** to `On`. Combined with the **Copy Container** parameter set in step 13, the parameter Pos Container will let Viz copy the **label** container for each label value and position the sub-containers of the BarChart container in relation to the respective bar representing the value. Container offsets can also be defined if desired.

   > ⚠ **Tip:** When creating 3D bars where the **Corner** value is set to more than `2`, the label needs to be offset in Z-space. A way to circumvent this is to add the **Expert** container plug-in, and setting Z-Buffer Ignore to `On`.

4. Open the **Control Area** and change one of the bar values in the **Control Object** tab, in order to trigger a refresh. This should create the new **label** containers and populate them with data. If modifications are required on the label prototype (the first child container), delete all other siblings and alter one or more of the data input values, in order to trigger a new refresh and thus a re-creation of the sibling containers.

Animating the Bars

Basic animation of the bars is typically controlled by the **Bar Chart**'s **Stop%** parameter, with values from `0.0` to `100.0`. For a basic animation, set **Stop%** to `0` and add a key-frame at frame `0`, then set the **Stop%** value to `100` and add another key-frame.

There are various other parameters that further influence the animation:

- By activating the **Const. Speed** setting, animation of each bar will have the same duration.
- The **Relative Length** setting gives each bar its own 100 percent length, meaning 7 bars equal 700 percent.
- When **Progress Mode** is set to `Bars`, each bar subsequently grows to its value; and when set to `ChartX` or `Y`, the whole chart grows horizontally or vertically, respectively, with the incoming bars already at their final height.

> ⚠ **Note:** When **Progress Mode** is set to `Chart Y`, columns representing negative values will not be visible.

Animating the Labels

The labels can be animated in a similar fashion, as the **Data Label** plug-in have the same **Stop%**, **Relative Length** and **Const. Speed** parameters. Additionally, the values can be animated while the bars grow by setting **Animate Values** to `On`. By adding an **Alpha** plug-in to the label containers, and activating the setting **Use Alpha**, the labels will fade in as the bars appear.

The parameter **Use Scale** is used when an interpolation on the scale value of each label between `0.0` and `1.0` is required. As a result of the fact that the scaling interpolates to the fixed value of `1.0`, combined with the fact that the font label in most cases is somehow scaled, fonts often appear too big. This problem can be overcome by adding a parent container to the font. The **Data Label** plug-in will put the text into the first child container containing a *Text Geometry* plug-in, so the label content is preserved.

Adding Color

Colors can be assigned to each individual bar, by setting the **Map** parameter of the Bar Chart plug-in to **Color**. RGB and alpha values can be set by adjusting the **Color ID**. However, Viz Trio and Viz Pilot will not be able to control the colors, there is no control plug-in capable to do so. This can be resolved by using a script and to expose the script parameters to the control application. The following script example allows to set a clear color for all bars, and it has a button that calculates the min/max bar and assigns different colors to those bars:

```
Sub calcMinMax()
Dim chart As Container = GetParameterContainer("bars")
If chart = null Then Exit Sub
Dim valueString As String = Scene.map[GetParameterString("barsSMName")]
Dim values as Array[String]
Dim minValue As Double = 1000000
Dim maxValue As Double = -100000
```

```
Dim minIndex As Integer = -1
Dim maxIndex As Integer = -1
Dim i As Integer = 0
valueString.split(",", values)

For Each str In values
If str.length <> 0 Then
If CDbl(str) < minValue Then
minValue = CDbl(str)
minIndex = i
End If

If CDbl(str) > maxValue Then
maxValue = CDbl(str)
maxIndex = i
End If
End If
i = i + 1
Next
Dim col As Color
Dim clearColor As Color = GetParameterColor("clearColor")

For i = 0 To values.Size
If i = maxIndex Then
col.Red = 0.317
col.Green = 0.71
col.Blue = 0.0
ElseIf i = minIndex Then
col.Red = 1.0
col.Green = 0.14
col.Blue = 0.0
Else
col = clearColor
End If

chart.Geometry.SetParameterInt("ColorID", i)
chart.Geometry.SetParameterColor("Color", col)
Next
End Sub

Sub resetColors()

Dim chart As Container = GetParameterContainer("bars")

If chart = null Then Exit Sub

Dim valueString As String = Scene.map[GetParameterString("barsSMName")]
Dim values as Array[String]

Dim i As Integer = 0
valueString.split(",", values)

Dim col As Color = GetParameterColor("clearColor")
```

```
For i = 0 To values.Size
chart.Geometry.SetParameterInt("ColorID", i)
chart.Geometry.SetParameterColor("Color", col)
Next
End Sub

Sub OnExecAction(buttonId As Integer)
If buttonId = 0 Then calcMinMax()
If buttonId = 1 Then resetColors()
End Sub

Sub OnInitParameters()
RegisterParameterContainer("bars", "Bar Chart")
RegisterParameterString("barsSMName", "Shared Mem Name", "bars", 40, 1000, "")
RegisterPushButton("minmax", "Calc Min/Max", 0)

Dim clearColor As Color

clearColor.Red = 1
clearColor.Green = 1
clearColor.Blue = 1

RegisterParameterColor("clearColor", "Clear Color", clearColor)
RegisterPushButton("reset", "Reset Colors", 1)
End Sub

Sub OnInit()
resetColors()
End Sub
```

The **ControlParameter** is set to **Text**, and the parameter to control is `SCRIPT*INSTANCE*clearColor`.

See Also

- Control Chart
- Data Fit
- Data Import
- Data Label
- Data Storage
- Presenter
- Bar Stack

13.7.3 Area Chart

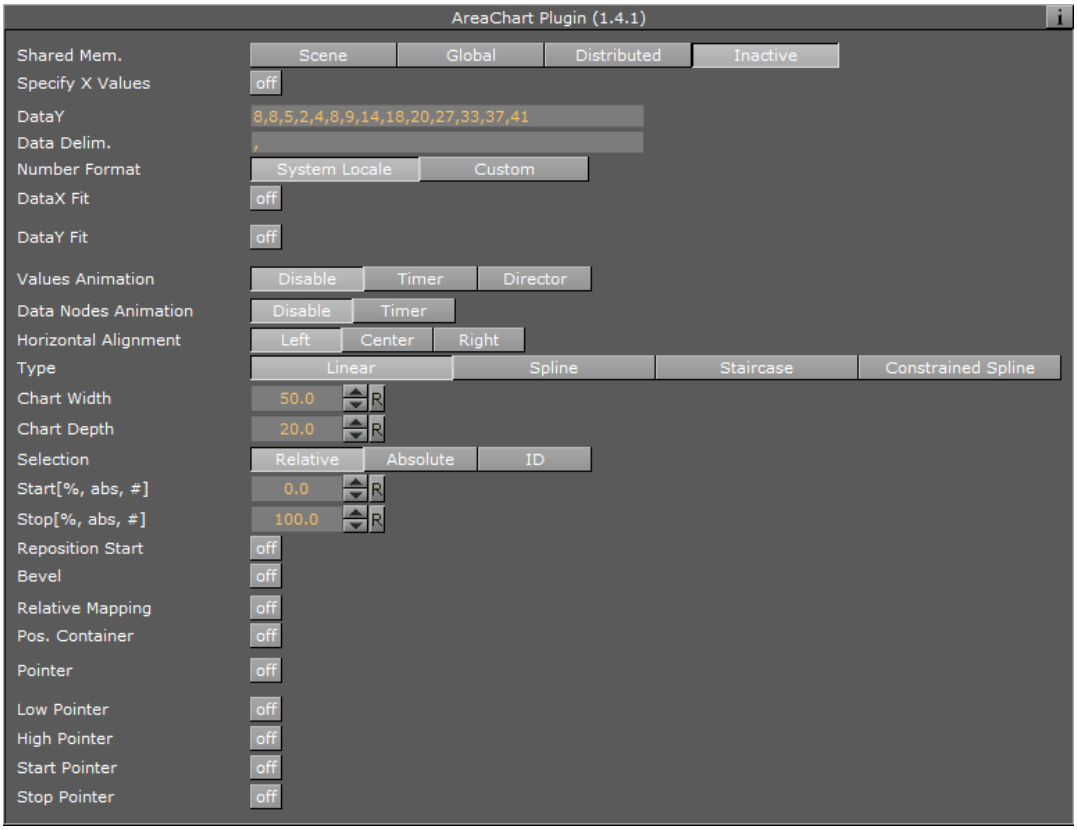
Th Area Chart plug-in draws an area chart, filled with data out of a Shared Memory Map. You can use delimited strings or arrays for data transfer via Scene, Global or Distributed Shared Memory.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> VisualDataTools

This section contains the following topics:

- Area Chart Properties
- To Create a Scene with Area Chart
- To Create a Scene with Area Chart and Data Storage
- To Create a Area Chart with Data Animation
    - Chart Animation using Data Import
    - Chart Animation using Control Chart

Area Chart Properties



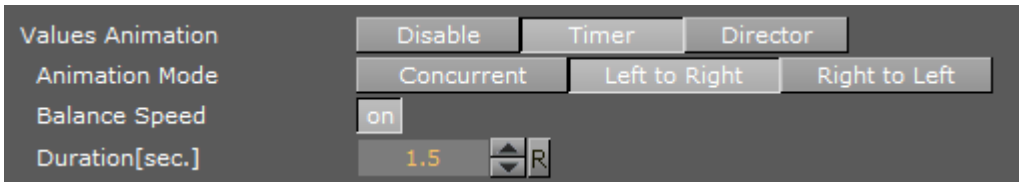
- **Shared Mem.:** Changes between **Scene**, **Global** and **Distributed** Shared Memory. Use **Inactive** memory to not forward any values via Shared Memory.
- **Specify X Values:** Enables DataX input.
    - **Key DataX:** Defines Shared Memory key name for X values.
    - **DataX**: Defines parameter for X values if Shared Memory is set to **Inactive.**
- **Key DataY:** Defines Shared Memory key name for Y values.
- **DataY**: Defines parameter for Y values if Shared Memory is set to Inactive.
- **Transfer Mode:** Sets string or array based data transfer (not available when set shared memory is set to **Inactive**).
- **Data Delim.:** Defines the value separator sign(s).
- **Number Format**: Defines if Viz should get the decimal point separator from the `System Locale` or `Custom` settings. If Custom is enabled:

- **Decimal Symbol**: Defines which symbol is used as decimal point when **Custom Number Format** is set.

> ⚠ **Note:** If the Area Stack plug-in is used, the following Area Chart properties need to be defined in the Area Stack plug-in.

- **DataX, Y Fit:** Enables data normalization.
  - **DataX, Y Scale:** Scales input by the selected factor.
  - **DataX, Y Offset:** Adds an offset to the incoming data.
  - **DataX, Y Auto Scale:** Enables automatic data normalization.
  - **DataX, Y Fit Size**: **Total:** Scales the whole chart to the defined borders. **Current** scales the current chart segment to the set borders.
  - **DataX, Y Detect Limits:** Detects minimum and maximum of all values and scales them to adjusted Start and Stop. This option is used to upscale the interesting part of the chart - especially if there are only little changes between the data values.
  - **DataX, Y Threshold**: Adds a definable offset to the detected limit.
  - **DataX, Y Start:** Lower Auto Scale edge.
  - **DataX, Y Stop:** Upper Auto Scale edge.

- **Values Animation**: Sets the vertical animation type to be created when data values change, but the number of nodes is still the same.



  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.** Also select an **Animation Mode.**
  - **Director**: Creates data transition animation with the Stage Director, that controls the Animation Progress parameter from 0.0% (old data) to 100.0% (new data). Also select an **Animation Mode.**
- **Animation Mode**:
  - **Concurrent**: Changes all nodes concurrently.
  - **Left to Right**: Transitions from left to right.
  - **Right to Left**: Transitions from right to left.
- **Balance Speed**: Relates node values to the position of transformed data when set to **On** and Animation Mode is set to **Left to Right** or **Right to Left**.

> ⚠ **Example: Balance Speed**: If the values of four nodes change from 0,0,0,0 to 1,9,90,900 in **Left to Right** mode at 50% of the transformation. **Balance Speed Off**: Only the first two nodes will be transformed to new values. **Balance Speed On**: The first three nodes are already transformed to new values, and the last node also transformed to 44% of the new values, because the data change

from 0 (0+0+0+0) to 1000 (1+9+90+900), at 50% of transformation (in this case the values is changed for 500 of 1000), the first 3 nodes are already finished and the last node is changed to 400.

- **Duration sec.**: Sets the duration of the animation in seconds for **Timer** mode.

- **Data Nodes Animation:** Sets the horizontal animation type to be created when number of data nodes change is detected.

  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**
- **Duration sec.:** Sets the duration of the animation in seconds for **Timer** mode.

⚠ When number of nodes is changes, area chart looks for added and removed nodes from their values, adds new nodes to the chart from zero width and resizes all nodes to their new size, specified in duration.

- **Horizontal Alignment**: Sets horizontal orientation to left, center or right.
- **Type:** Creates different graph looks. Available options are Normal, Spline, Staircase and Constrained Spline.
  - **Normal:** Enables direct connections between entered values.
  - **Spline:** Interpolates extra values to chamfer the graph.
  - **Staircase:** Creates a staircase look.
  - **Constrained Spline**: Same as Spline mode but with a different algorithm to prevent overshooting.
- **Stair Width**: Defines width for a single stair or for all stairs if Specify X Values is activated.
- **Chart Width**: Adjusts the chart width.
- **Chart Depth:** Adjusts extrusion.
  - **Tesselation Width:** Sets tessellation in Spline mode.
- **Selection:** Specifies Start and Stop type.
  - **Relative:** 0%: 100%
  - **Absolute:** Depends on the specified data.
  - **ID:** Value ID starting from 0
- **Start%, abs,:** Graph starting point.
- **Stop%, abs,:** Graph stopping point.
- **Reposition Start:** Translates the whole chart always to the same starting point independent of varying Start and Stop positions.
- **Bevel:** Activates Bevel mode.
  - **Bevel Bottom:** Chamfers the chart's bottom
  - **Bevel Size:** Adjusts bevel's size
  - **Bevel Steps:** Sets roundness via the number of segmentation steps

> ⚠ **Note:** A chamfer is a beveled edge connecting two surfaces.

- **RelativeMapping:** Stretches texture to fit graph width.
- **Pos. Container:** Translates every child container to a bar's top.
  - **Container Offset:** Adds a certain offset to each container.
  - **Container Offset Z:** Adds a Z Axis offset to each container.
  - **Center Container:** Centers each translated container.
- **Pointers Shared Mem.**: Selects the map where the pointer values should be distributed to.
- **Key Data Pointers**: Defines Shared Memory Keyname.
- **Pointer**: Activates the Pointer.
  - **Container**: Defines Container which should be used as pointer.
  - **Selection**: Selects the navigation type.
  - **Position%, abs,:** Sets pointer position in dependency of the selected navigation type.
  - **Offset**: Defines pointer offset from the chart.
  - **Normal Offset**: Rotates offset to the direction of the current surface normal.
  - **Rotate**: Rotates the pointer to the direction of the current surface normal.
  - **Offset Z**: Adds an additional Z offset to the pointer container.
  - **Center Container**: Centers Container - useful for varying pointer sizes.
- **Pointers:** Activates each pointer as required:
  - **Low Pointer / High Pointer / Start Pointer / Stop Pointer**
  - **Container:** Selects a container for the pointer.

To Create a Scene with Area Chart

1. Create a new Container.
2. Add the Area Chart plug-in.
3. Open the Transformation Editor.
   Set these parameters:
   - Position X: -100.0.
   - Position Y: -100.0.
   - Rotation Y: -25.0.
2. Add a Material and/or a Texture.
3. Open the Area Chart editor.
4. Enter these parameters:
   - Set DataY values: 10,20,30,20,50,60,80,45,20.
   - Set **Type** to Spline.
   - Set **Chart Width** to 200.0.
   - Set **Chart Depth** to 30.0.

To Create a Scene with Area Chart and Data Storage

1. Create a new Container.
2. Add the Area Chart plug-in.

3. Add the Data Storage plug-in to Area Chart Container.
   Select the Area Chart plug-in.
   a. Set **SharedMem.** to Scene.
   b. Rename the **Key DataY** to `MyDataY`.
   Open the **Data Storage** editor.
   a. Enter `MyDataY` in the **Key Data1** field.
   b. Set the **SharedMem.** to Scene.
   c. In **Data1** enter: 10,40,50,20,80,90,60,50.

To Create a Area Chart with Data Animation

Chart Animation using Data Import

The following steps are to prepare data sets for animation in Excel file.

1. Start Microsoft Excel.
2. Enter **ExcelData1** into cell A1.
3. Add some sample values in the cells below (A2-A8: 80, 35, 45, 75, 85, 55, 60).
4. Enter **ExcelData2** into cell A2.
5. Add some sample values in the cells below (B2-B9: 80, 35, 45, 75, 40, 85, 55, 60).
6. Enter **ExcelData3** into cell A3.
7. Add some sample values in the cells below (C2-C9: 40, 60, 75, 85, 80, 55, 45, 35).
8. Enter **ExcelData4** into cell A4.
9. Add some sample values in the cells below (D2-D8: 60, 75, 85, 80, 55, 45, 35).
10. Rename this first sheet to **MyTable** (can be done with a double click on the sheet name at the bottom).
11. Follow one of the following methods to make Viz Engine can read this Excel data file:
    a. **Save and close** the Excel document.
    b. **Share** the Excel document.
        i. On the Review tab, in the Changes group, click the **Share Workbook** button.
        ii. The Share Workbook dialog box will appear, and you select the **Allow changes by more than one user at the same time.** check box on the Editing tab.
        iii. Click **Ok** button of the dialog.
        iv. Save the Excel document.

> **IMPORTANT!** You must use the same platform (x64 / x86) of Microsoft Excel and Viz Engine.

After the data sets are created, the following steps are to create a Data Animation scene.

1. Follow the "To create a Area Chart" instruction.
2. Modify the following parameters of Area Chart plugin.
   - Change **Shared Mem.** to **Scene.**
   - Set **Kay DataY** to **DataY.**
   - Set **Data Delim.** to **#.**
   - Set **Values Animation** to **Timer.**
   - Set **Data Nodes Animation** to **Timer.**
3. Add Data Import plugin into the same container.
   a. Set **File** to above prepared Excel file.

   b. Set **Table / Sheet** to **MyTable**.
   c. Set **Column(,Col...)** to **ExcelData1**.
   d. Set **Data Delim.** to **#.**
   e. Set **Shared Mem.** to **Scene.**
   f. Set **Key** to **DataY.**
4. Click the **GetIt** button and animation chart shows from zero to seven nodes.
5. Add one data node in the middle of chart.
   a. Set **Column(,Col...)** to **ExcelData2.**
   b. Click the **GetIt** button.
6. Change values of eight data nodes.
   a. Set **Column(,Col...)** to **ExcelData3.**
   b. Click the **GetIt** button.
7. Remove the first data node.
   a. Set **Column(,Col...)** to **ExcelData4.**
   b. Click the **GetIt** button.

⚠ **Note:** You can add more data columns and play with animation on data change.

Chart Animation using Control Chart

1. Follow the "To create a Area Chart" instruction.
2. Modify the following parameters of Area Chart plugin.
   - Set **Values Animation** to **Timer.**
   - Set **Data Nodes Animation** to **Timer.**
3. Add Control Chart plugin into the same container and save the scene.
4. Open Viz Trio.
   - Import the scene.
   - Click "1 (AreaChart)" in Tab Fields.
   - Edit values in Editing Template.

See Also

- Control Chart
- Data Fit
- Data Import
- Data Label
- Data Storage
- Area Stack

13.7.4 Bar Chart

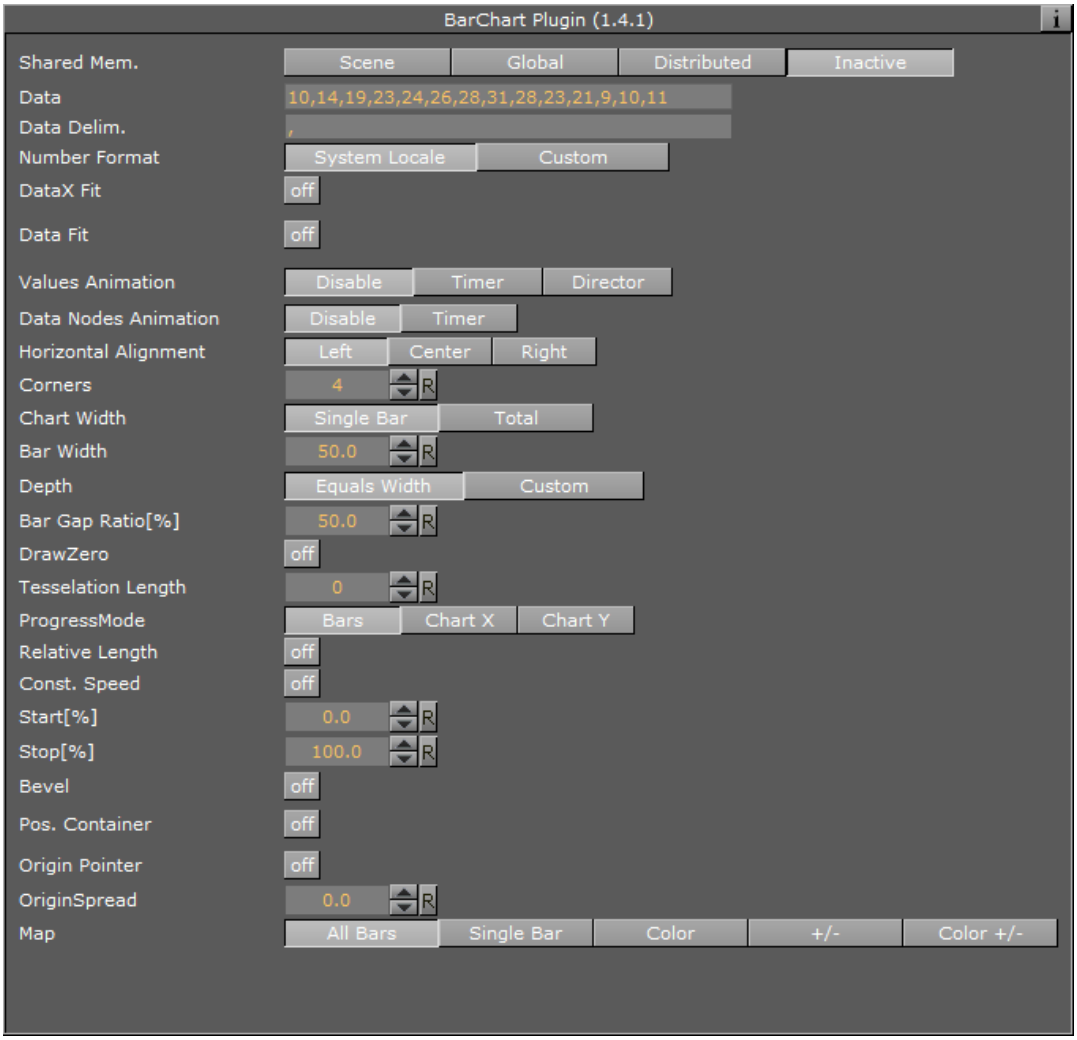
This plug-in draws a bar chart, filled with data out of a Shared Memory Map. You can use delimited strings or arrays for data transfer via Scene-, Global- or Distributed-Shared Memory.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> VisualDataTools

This section contains the following topics:

- Bar Chart Properties
- To Create a Bar Chart
- To Create a Bar Chart with Data Animation
  - Chart Animation using Data Import
  - Chart Animation using Control Chart

Bar Chart Properties

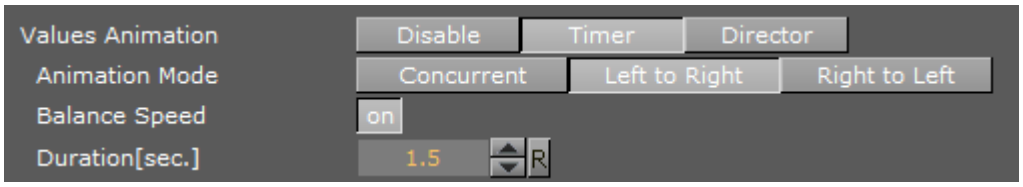


- **Shared Mem.:** Changes between **Scene**, **Global** and **Distributed** Shared Memory. Use **Inactive** memory to not forward any values via Shared Memory and the data is taken from the **Data** field.
- **Data**: Inputs chart data (available when **Inactive** is selected).
- **Key Data:** Determines shared Memory key name (available when **Scene**, **Global** or **Distributed** is selected).
- **Transfer Mode:** Sets **String** or **Array** based data transfer.

- **Data Delim.:** Defines the value separator sign(s).
- **Number Format**: Defines if Viz should get the decimal point separator from the `System Locale` or `Custom` settings. If Custom is enabled:
  - **Decimal Symbol**: Defines which symbol is used as decimal point when **Custom Number Format** is set.

> ⚠ **Note:** If the Bar Stack plug-in is used, the following Bar Chart properties need to be defined in the Bar Stack plug-in.

- **DataX Fit:** Enables data normalization on the horizontal:
  - **DataX Scale:** Scales input by the selected factor.
  - **DataX Offset:** Adds an offset to the incoming data.
  - **DataX Auto Scale:** Enables automatic data normalization.
  - **DataX Fit Size**:
    - **Total:** Scales complete chart as per the defined Start and Stop settings.
    - **Current:** Scales the current chart segment to the set borders.
- **Data Fit:** Enables data normalization:
  - **Data Scale:** Scales input by the selected factor.
  - **Data Offset:** Adds an offset to the incoming data.
  - **Data Origin Offset**: Adds an offset to the smallest/minimum value.
  - **Data Auto Scale:** Enables automatic data normalization.
  - **Data Fit Size**: Scales complete chart in dependency of the defined Start and Stop settings (**Total**) or scales the **Current** chart segment to the set borders.
  - **Data Detect Limits:** Detects minimum and maximum of all values and scales them to adjusted Start and Stop.
  - **Data Threshold**: Adds a definable offset to the detected limit.
  - **Data Start:** Lower Auto Scale edge.
  - **Data Stop:** Upper Auto Scale edge.
- **Compare:** Specifies another BarChart for comparison. You have to set this parameter also in the other chart(s) so that each chart can react on changes of the other(s).
- **Value Animation:**



Sets the vertical animation type to be created when data change is detected:

  - **Disable:** No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**. Also select an **Animation Mode**.
  - **Director**: Creates data transition animation with the Stage Director, that controls the Animation Progress parameter from 0.0% (old data) to 100.0% (new data). Also select an **Animation Mode**.
- **Animation Mode**:

- **Concurrent**: Changes all nodes concurrently.
- **Left to Right**: Transitions from left to right.
- **Right to Left**: Transitions from right to left.

- **Balance Speed**: Relates node values to the position of transformed data when set to **On** and Animation Mode is set to **Left to Right** or **Right to Left**.

> ⚠ **Example: Balance Speed**: If the values of four nodes change from 0,0,0,0 to 1,9,90,900 in **Left to Right** mode at 50% of the transformation. **Balance Speed Off**: Only the first two nodes will be transformed to new values. **Balance Speed On**: The first three nodes are already transformed to new values, and the last node also transformed to 44% of the new values, because the data change from 0 (0+0+0+0) to 1000 (1+9+90+900), at 50% of transformation (in this case the values is changed for 500 of 1000), the first 3 nodes are already finished and the last node is changed to 400.

- **Duration sec.**: Sets the duration of the animation in seconds for **Timer** mode.

- **Data Nodes Animation:** Sets the horizontal animation type to be created when number of data nodes change is detected.

- **Disable**: No animation.
- **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**

- **Duration sec.:** Sets the duration of the animation in seconds for **Timer** mode.

> ⚠ When the number of nodes is changed, bar chart looks for added and removed nodes from their values, adds new nodes to the chart from zero width and resizes all nodes to their new size, specified in duration.

- **Horizontal alignment**: Sets horizontal orientation to left, center or right.
- **Corners:** Sets the bar's corners.
- **Chart Width:** Defines width for a single bar or for all bars.
- **Bar Width:** Value of the previous parameter.
- **Max Width:** Allows bars to grow only to a certain width.
- **Bar Gap Ratio**: Adjusts the gap between the single bars. 0% means no gap.
- **Depth**: Defines if the bar depth should equal its width or lets you set a custom depth.
- **Bar Depth**: Adjusts bar depth in custom depth mode.
- **Draw Zero:** If the Bar value is equivalent to `0`, then a very flat bar will be drawn instead of nothing.
- **Tessellation Length:** Tessellates bar length.
- **Progress Mode**:
  - **Bars**: Allows each bar grows to its value.
  - **Chart X**: The chart grows horizontally. Each bar comes in at its final height.

    - **Chart** Y: The chart grows vertically. All bars grow together to their final height.
- **Relative Length:** If activated, each bar will have it's own 100% length (e.g. 7 bars equal 700%).
- **Const. Speed:** Sets the same animation duration for each bar.
- **Start%**: Determines starting point of the graph.
- **Stop%**: Determines stop point of the graph.
- **Bevel:** Activates Bevel mode.
    - **Bevel Top:** Chamfers bar's top.
    - **Bevel Bottom:** Chamfers bar's bottom.
    - **Bevel Size%:** Adjusts bevel's size from 0 to 100.
    - **Bevel Steps:** Sets roundness via the number of segmentation steps.

      ⚠ **Note:** A chamfer is a beveled edge connecting two surfaces.

- **Pos. Container:** Translates every child container to a bar's top.
    - **Container Offset:** Adds a certain offset to each container.
    - **Container Offset X:** Adds a X Axis offset to each container.
    - **Container Offset Y**: Adds a Y Axis offset to each container.
    - **Container Offset Z:** Adds a Z Axis offset to each container.
    - **Center Container:** Centers each translated container.
    - **Size Compensation**: Repositions containers if they can change their size. E.g. Text geometries used as chart labels. The labels will vary in their width which requires a repositioning if a horizontal BarChart (rotated 90 deg) is used.
    - **None**: Deactivates container size compensation.
    - **Width**: Compensates width changes.
    - **Height**: Compensates height changes.
- **Map:** Calculates texture coordinates for all bars, a single bar or defines vertex colors instead. Colors can be set individually for each bar or value dependent (+/- color). In the Color +/- mode you can override value dependent colors.
    - **Color ID:** Moves between the available vertex colors.
    - **Color:** Chooses color for the current ID.
    - **Positive:** Chooses color for positive values.
    - **Negative:** Chooses color for negative values.
    - **Color Override**: Activates or deactivates overriding of a positive or negative color with a defined alternative color.

To Create a Bar Chart

1. Create a new group and add the Bar Chart plug-in to it.
2. Open the Transformation Editor and set Position X and Y to -100.0 and Rotation Y to -25.0.
3. Add a material and/or a texture to it.
4. Open the Bar Chart editor and set the following parameters:
    - Set the following DataY values: 10,20,30,20,50,60,80,45,20.
    - Set Corners to 3.

- Set Bar Width to 20.0.
- Set Bar Gap Ratio[%] to 10.0.

To Create a Bar Chart with Data Animation

Chart Animation using Data Import

The following steps are to prepare data sets for animation in Excel file.

1. Start Microsoft Excel.
2. Enter **ExcelData1** into cell A1.
3. Add some sample values in the cells below (A2-A8: 80, 35, 45, 75, 85, 55, 60).
4. Enter **ExcelData2** into cell A2.
5. Add some sample values in the cells below (B2-B9: 80, 35, 45, 75, 40, 85, 55, 60).
6. Enter **ExcelData3** into cell A3.
7. Add some sample values in the cells below (C2-C9: 40, 60, 75, 85, 80, 55, 45, 35).
8. Enter **ExcelData4** into cell A4.
9. Add some sample values in the cells below (D2-D8: 60, 75, 85, 80, 55, 45, 35).
10. Rename this first sheet to **MyTable** (can be done with a double click on the sheet name at the bottom).
11. Follow one of the following methods to make Viz Engine can read this Excel data file
    a. **Save and close** the Excel document.
    b. **Share** the Excel document.
        i. On the Review tab, in the Changes group, click the **Share Workbook** button.
        ii. The Share Workbook dialog box will appear, and you select the **Allow changes by more than one user at the same time.** check box on the Editing tab.
        iii. Click **Ok** button of the dialog
        iv. Save the Excel document.

> **IMPORTANT!** You must use the same platform (x64 / x86) of Microsoft Excel and Viz Engine.

After the data sets are created, the following steps are to create a Data Animation scene.

1. Follow the "To create a Bar Chart" instruction.
2. Modify the following parameters of Bar Chart plugin.
    - Change **Shared Mem.** to **Scene.**
    - Set **Kay Data** to **Data.**
    - Set **Data Delim.** to **#.**
    - Set **Values Animation** to **Timer.**
    - Set **Data Nodes Animation** to **Timer.**
3. Add Data Import plugin into the same container.
    a. Set **File** to above prepared Excel file.
    b. Set **Table / Sheet** to **MyTable**.
    c. Set **Column(,Col...)** to **ExcelData1**.
    d. Set **Data Delim.** to **#.**
    e. Set **Shared Mem.** to **Scene.**
    f. Set **Key** to **Data.**
4. Click the **GetIt** button and animation chart shows from zero to seven nodes.

5. Add one data node in the middle of chart (Horizontal Animation).
    a. Set **Column(,Col...)** to **ExcelData2.**
    b. Click the **GetIt** button.
6. Change values of eight data nodes (Vertical Animation).
    a. Set **Column(,Col...)** to **ExcelData3.**
    b. Click the **GetIt** button.
7. Remove the first data node (Horizontal Animation).
    a. Set **Column(,Col...)** to **ExcelData4.**
    b. Click the **GetIt** button.

⚠ **Note:** You can add more data columns and play with animation on difference data change.

Chart Animation using Control Chart

1. Follow the "To create a Bar Chart" instruction.
2. Modify the following parameters of Bar Chart plugin.
    - Set **Values Animation** to **Timer.**
    - Set **Data Nodes Animation** to **Timer.**
3. Add Control Chart plugin into the same container and save the scene.
4. Open Viz Trio.
    - Import the scene.
    - Click "1 (BarChart)" in Tab Fields.
    - Edit values in Editing Template.

See Also

- Control Chart
- Data Fit
- Data Import
- Data Label
- Data Storage
- Presenter
- Bar Stack
- Advanced Bar Chart Creation

13.7.5 Line Chart

This plug-in draws a line chart, filled with data out of a Shared Memory Map. You can use delimited strings or arrays for data transfer via Scene-, Global- or Distributed-Shared Memory.

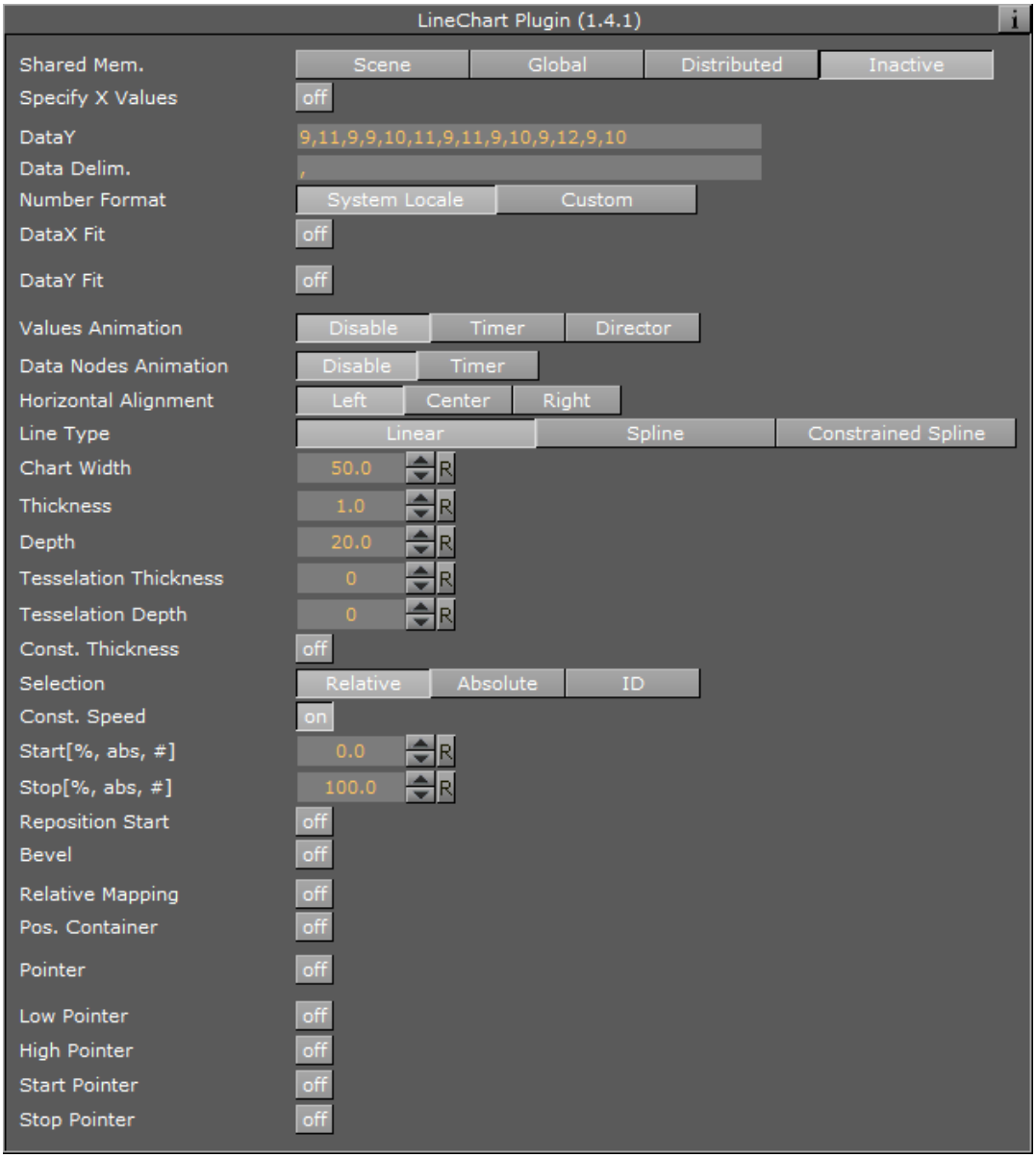
⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> VisualDataTools

This section contains the following topics:

- Line Chart Properties

- To Create a Line Chart
- To Create a Line Chart with Data Animation
  - Chart Animation using Data Import
  - Chart Animation using Control Chart

Line Chart Properties



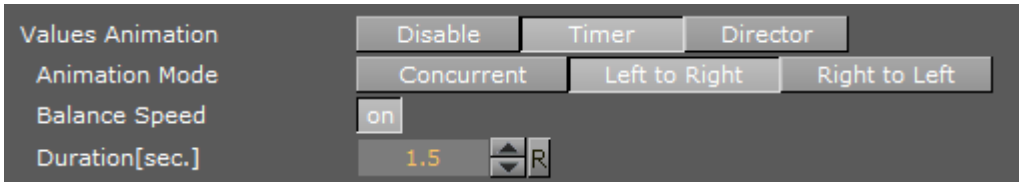
- **Shared Mem.:** Changes between Scene, Global and Distributed Shared Memory. Use Inactive memory to not forward any values via Shared Memory.
- **Specify X Values**: Enables DataX input.
  - **DataX:** Determines Shared Memory key name for X values. DataX is the default input parameter for X values.
  - **Key DataX:** Determines Shared Memory key name for X values.

- **DataY:** Determines Shared Memory key name for Y values. DataY is the default input parameter for Y values.
- **Key DataY**: Determines Shared Memory key name for Y values.
- **Transfer Mode**: Sets string- or array-based data transfer.
- **Data Delim.:** Defines the value separator sign(s).
- **Number Format**: Defines if Viz should get the decimal point separator from the `System Locale` or `Custom` settings. If Custom is enabled:
  - **Decimal Symbol**: Defines which symbol is used as decimal point when **Custom Number Format** is set.

> ⚠ **Note:** If the Line Stack plug-in is used, the following Line Chart properties need to be defined in the Line Stack plug-in.

- **DataX, -Y Fit:** Enables data normalization.
  - **DataX, -Y Scale**: Scales input by the selected factor.
  - **DataX, -Y Offset:** Adds an offset to the incoming data.
  - **DataX, -Y Auto Scale:** Enables automatic data normalization.
  - **DataX, -Y Fit Size**:
    - **Total**: Scales the whole chart to the defined borders.
    - **Current**: Scales the current chart segment to the set borders.
  - **DataX, -Y Detect Limits:** Detects minimum and maximum of all values and scales them to adjusted Start and Stop. This option is used to upscale the interesting part of a chart - especially if there are only little changes between the data values.
  - **DataX, -Y Threshold**: Adds a definable offset to the detected limit.
  - **DataX, -Y Start:** Lower Auto Scale edge.
  - **DataX, -Y Stop**: Upper Auto Scale edge.

- **Values Animation:** Sets the vertical animation type to be created when data change is detected:



  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**. Also select an **Animation Mode**.
  - **Director**: Creates data transition animation with the Stage Director, that controls the Animation Progress parameter from 0.0% (old data) to 100.0% (new data). Also select an **Animation Mode**.
- **Animation Mode**:
  - **Concurrent**: Changes all nodes concurrently.
  - **Left to Right**: Transitions from left to right.
  - **Right to Left**: Transitions from right to left.

- **Balance Speed**: Relates node values to the position of transformed data when set to **On** and Animation Mode is set to **Left to Right** or **Right to Left**.

  > ⚠ **Example: Balance Speed**: If the values of four nodes change from 0,0,0,0 to 1,9,90,900 in **Left to Right** mode at 50% of the transformation. **Balance Speed Off**: Only the first two nodes will be transformed to new values. **Balance Speed On**: The first three nodes are already transformed to new values, and the last node also transformed to 44% of the new values, because the data change from 0 (0+0+0+0) to 1000 (1+9+90+900), at 50% of transformation (in this case the values is changed for 500 of 1000), the first 3 nodes are already finished and the last node is changed to 400

- **Duration sec.**: Sets the duration of the animation in seconds for **Timer** mode.

- **Data Nodes Animation:** Sets the horizontal animation type to be created when number of data nodes change is detected:

  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**. Also select an **Animation Mode**
- **Duration sec.**: Sets the duration of the animation in seconds for **Timer** mode.

  > ⚠ When number of nodes is changes, line chart will looking for added and removed nodes from their values, add new nodes to the chart from zero width and resize all nodes to their new size in specified duration.

- **Horizontal Alignment:** Sets horizontal orientation to left, center or right.
- **Line Type:**
  - **Normal**: Direct connections between entered values.
  - **Spline**: Interpolates extra values to chamfer the graph.
  - **Constrained Spline**: Same as Spline mode but with a different algorithm to prevent overshooting.
- **Chart Width:** Adjusts line width (0 for 2D mode).
  - **Tessellation Width:** Sets tessellation length in spline modes.
- **Thickness**: Adjusts line thickness (0 for 2D mode).
- **Depth:** Adjusts line depth (0 for 2D mode).
- **Tessellation Thickness:** Tessellates line thickness.
- **Tessellation Depth:** Tessellates line depth.
- **Const. Thickness:** Builds the geometry with a constant width.
  - **Miter:** Applies a miter cut-off to avoid huge peaks for small data point angles.
- **Selection:** Specifies the Start- and Stop-type. Relative: 0%: 100%. Absolute: depends on your specified data. ID: Value ID starting from 0.
- **Start%, abs,:** Determines starting point of the graph.

- **Stop%, abs,:** Determines stop point of the graph.
- **Reposition Start:** Translates the whole chart always to the same starting point.
- **Bevel**: Activates Bevel mode.
  - **Bevel Start:** Chamfer start.
  - **Bevel Stop:** Chamfer stop.
  - **Bevel Size%:** Adjusts bevel's size from 0 to 100.
  - **Bevel Steps:** Sets roundness via the number of segmentation steps.

    ⚠ **Note:** A chamfer is a beveled edge connecting two surfaces.

- **RelativeMapping:** Stretches texture to fit graph length.
- **Pos. Container:** Translates every child container to a bar's top.
  - **Container Offset:** Adds a certain offset to each container.
  - **Value Dependent Pos.:** Translates the container above or below the line in dependency on the data values.
  - **Container Offset Z:** Adds a Z axis offset to each container.
  - **Center Container:** Centers each translated container.
- **Pointer:** You can use a container as a pointer and move it along the chart's shape.
  - **Container:** This container represents your pointer.
  - **Selection:** Specifies the position-type. Relative: 0%: 100%. Absolute: depends on your specified data. ID: Value ID starting from 0.
  - **Position%, abs,:** Translates pointer by this parameter.
  - **Offset:** Adds an offset to the container's position.
  - **Normal Offset:** Rotates the offset so that it is perpendicular to the chart's surface.
  - **Rotate:** Rotates the container in dependency on the chart's shape.
  - **Offset Z:** Adds a Z axis offset to the pointer.
  - **Center Container:** Uses the container's center for translation and rotation.
- **Low Pointer**: Activates the low pointer.
  - **Container:** Chooses a container for the pointer.
- **High Pointer**: Activates the high pointer.
  - **Container**: Chooses a container for the pointer.
- **Start Pointer**: Activates the Start Pointer.
  - **Container**: Chooses a container for the pointer.
- **Stop Pointer**: Activates the stop pointer.
  - **Container:** Chooses a container for the pointer.

To Create a Line Chart

1. Create a new group and add the Line Chart plug-in to it.
2. Open the Transformation Editor and set **Position X** and **Y** to `-100.0` and **Rotation Y** to `-25.0`.
3. Add a material and/or a texture to it.
4. Open the Line Chart editor and set the following parameters:
   - Set **Shared Mem.** to **Inactive**
   - Set the following **DataY** values: `10,20,30,20,50,60,80,45,20`

- Set **Line Type** to **Spline**
- Set **Chart Width** to `250.0`
- Set **Depth** to `10.0`
- Enable **Bevel**, **Bevel Start**, **Bevel Stop**
- Set **Bevel Size%** to `100.0`
- Set **Bevel Steps** to `10.0`

To Create a Line Chart with Data Animation

Chart Animation using Data Import

The following steps are to prepare data sets for animation in Excel file.

1. Start Microsoft Excel.
2. Enter **ExcelData1** into cell A1.
3. Add some sample values in the cells below (A2-A8: 80, 35, 45, 75, 85, 55, 60).
4. Enter **ExcelData2** into cell A2.
5. Add some sample values in the cells below (B2-B9: 80, 35, 45, 75, 40, 85, 55, 60).
6. Enter **ExcelData3** into cell A3.
7. Add some sample values in the cells below (C2-C9: 40, 60, 75, 85, 80, 55, 45, 35).
8. Enter **ExcelData4** into cell A4.
9. Add some sample values in the cells below (D2-D8: 60, 75, 85, 80, 55, 45, 35).
10. Rename this first sheet to **MyTable** (can be done with a double click on the sheet name at the bottom).
11. Follow one of the following methods to make Viz Engine can read this Excel data file
    a. **Save and close** the Excel document.
    b. **Share** the Excel document.
        i. On the Review tab, in the Changes group, click the **Share Workbook** button.
        ii. The Share Workbook dialog box will appear, and you select the **Allow changes by more than one user at the same time.** check box on the Editing tab.
        iii. Click **Ok** button of the dialog
        iv. Save the Excel document.

> **IMPORTANT!** You must use the same platform (x64 / x86) of Microsoft Excel and Viz Engine.

After the data sets are created, the following steps are to create a Data Animation scene.

1. Follow the "To create a Line Chart" instruction.
2. Modify the following parameters of Line Chart plugin.
    - Change **Shared Mem.** to **Scene.**
    - Set **Kay DataY** to **DataY.**
    - Set **Data Delim.** to **#.**
    - Set **Values Animation** to **Timer.**
    - Set **Data Nodes Animation** to **Timer.**
3. Add Data Import plugin into the same container.
    a. Set **File** to above prepared Excel file.
    b. Set **Table / Sheet** to **MyTable**.

c. Set **Column(,Col...)** to **ExcelData1**.
d. Set **Data Delim.** to **#.**
e. Set **Shared Mem.** to **Scene.**
f. Set **Key** to **DataY.**

4. Click the **GetIt** button and animation chart shows from zero to seven nodes.
5. Add one data node in the middle of chart.
   a. Set **Column(,Col...)** to **ExcelData2.**
   b. Click the **GetIt** button.
6. Change values of eight data nodes.
   a. Set **Column(,Col...)** to **ExcelData3.**
   b. Click the **GetIt** button.
7. Remove the first data node.
   a. Set **Column(,Col...)** to **ExcelData4.**
   b. Click the **GetIt** button.

⚠ **Note:** You can add more data columns and play with animation on difference data change.

Chart Animation using Control Chart

1. Follow the "To create a Line Chart" instruction.
2. Modify the following parameters of Line Chart plugin.
   - Set **Values Animation** to **Timer.**
   - Set **Data Nodes Animation** to **Timer.**
3. Add Control Chart plugin into the same container and save the scene.
4. Open Viz Trio.
   - Import the scene.
   - Click "1 (LineChart)" in Tab Fields.
   - Edit values in Editing Template.

See Also

- Control Chart
- Data Fit
- Data Import
- Data Label
- Data Storage
- Line Stack

13.7.6 Pie Chart

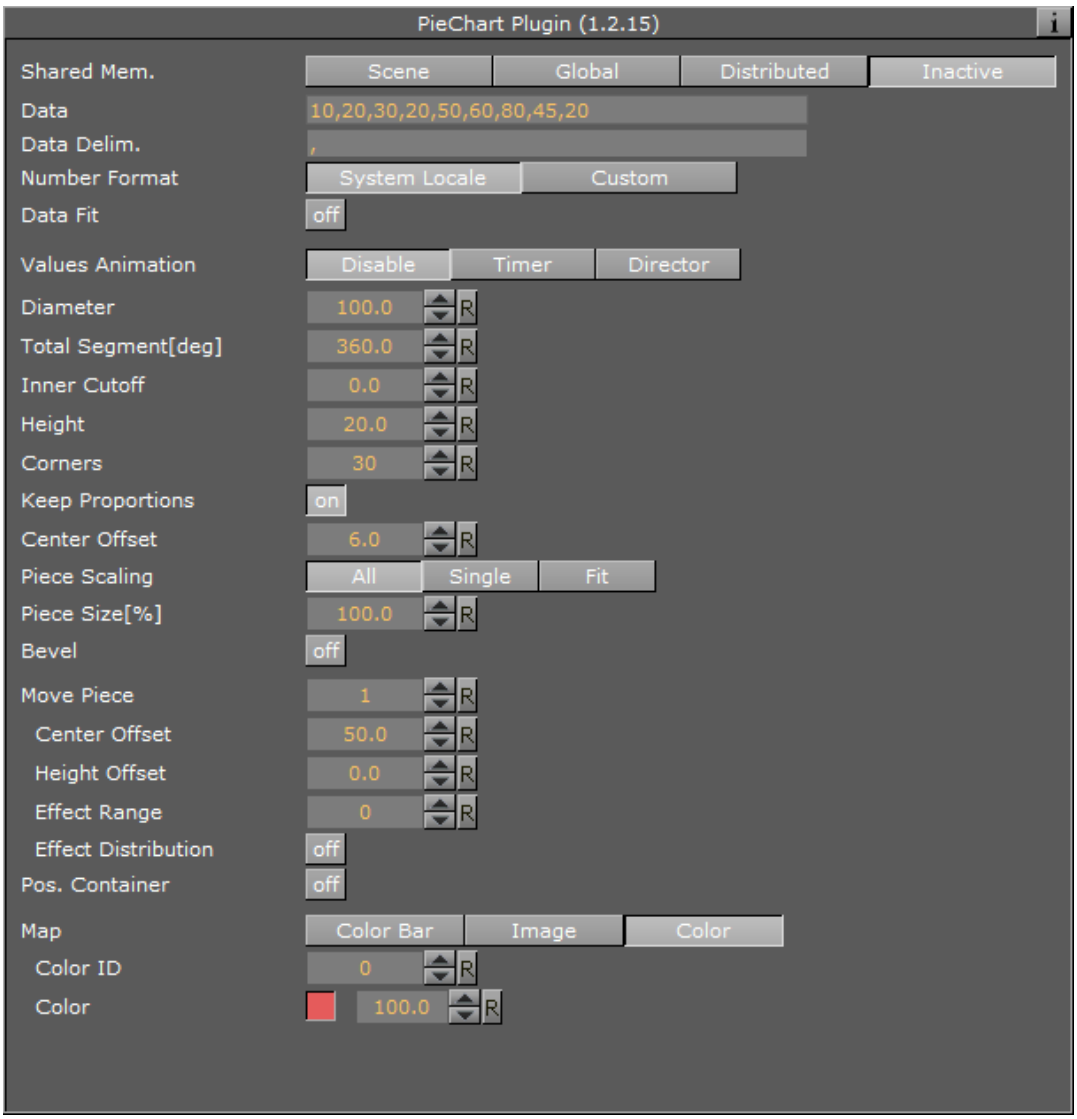
This plug-in draws a pie chart, filled with data out of a Shared Memory Map. You can use delimited strings or arrays for data transfer via Scene-, Global- or Distributed-Shared Memory.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> VisualDataTools

This section contains the following topics:

- Pie Chart Properties
- To Create a Pie Chart
- To Create a Pie Chart with Data Animation
  - Chart Animation using Data Import
  - Chart Animation using Control Chart

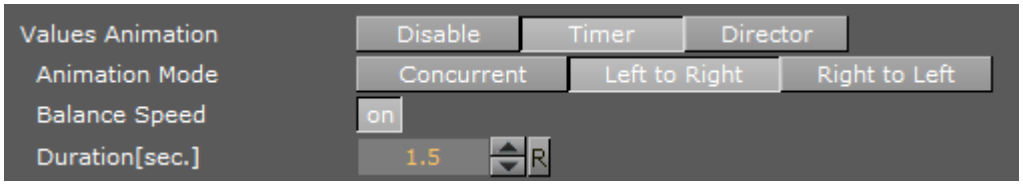
Pie Chart Properties



- **Shared Mem.:** Changes between **Scene**, **Global** and **Distributed** Shared Memory. Use **Inactive** memory to not forward any values via Shared Memory and the data is taken from the **Data** field.
- **Data:** Determines Shared Memory key name. Data is the default input parameter.
- **Transfer Mode:** Sets string- or array-based data transfer.
- **Data Delim.:** Defines the value separator sign(s).

- **Number Format**: Defines if Viz should get the decimal point separator from the `System Locale` or `Custom` settings. If Custom is enabled:
  - **Decimal Symbol**: Defines which symbol is used as decimal point when **Custom Number Format** is set.
- **Data Fit:** Enables data normalization.
  - **Data Offset:** Adds an offset to the incoming data.
- **Values Animation**:

  Sets the animation type to be created when data change is detected.



  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**. Also select an **Animation Mode**.
  - **Director**: Creates data transition animation with the Stage Director, that controls the Animation Progress parameter from 0.0% (old data) to 100.0% (new data). Also select an **Animation Mode**.
- **Animation Mode**:
  - **Concurrent**: Changes all nodes concurrently.
  - **Left to Right**: Transitions from left to right.
  - **Right to Left**: Transitions from right to left.
- **Balance Speed**: Relates node values to the position of transformed data when set to **On** and Animation Mode is set to **Left to Right** or **Right to Left**.

> ⚠ **Example: Balance Speed**: If the values of four nodes change from 0,0,0,0 to 1,9,90,900 in **Left to Right** mode at 50% of the transformation. **Balance Speed Off**: Only the first two nodes will be transformed to new values. **Balance Speed On**: The first three nodes are already transformed to new values, and the last node also transformed to 44% of the new values, because the data change from 0 (0+0+0+0) to 1000 (1+9+90+900), at 50% of transformation (in this case the values is changed for 500 of 1000), the first 3 nodes are already finished and the last node is changed to 400.

  - **Duration sec.**: Sets the duration of the animation in seconds for **Timer** mode.
- **Diameter:** Sets size of pie chart.
- **Total Segmentdeg:** Sets the pie chart's maximum angle in degrees (default: 100% are equal to 360 degrees).
- **Inner Cutoff**: Cuts off a round piece from the cake's center.
- **Height:** Adjusts height.
- **Corners:** Defines the minimum amount of corners at the pie's edge. In dependency on the number of pieces an algorithm adds corners to always give the pie chart the same appearance.

- **Keep Proportions:** If enabled, the pie chart would look real. Otherwise each piece's radius is recalculated.
- **Const. Offset:** Sets a constant gap width.
- **Center Offset:** Determines each pieces' distance from the pie's center.
- **Piece Scaling:**
  - **All:** All pieces equal 100%.
  - **Single:** Every piece's size is described by it's own 100%(e.g. 3 pieces mean 300%).
  - **Fit:** All pieces equal 100%, but they are scaled relatively to fit their new area with the correct values.
- **Piece Size%:** Parameter for the previous option.
- **Bevel:** Activates Bevel mode.
  - **Bevel Top:** Chamfers cake's top.
  - **Bevel Bottom:** Chamfers pie's bottom.
  - **Bevel Size%:** Adjusts bevel's size from 0 to 100.
  - **Bevel Steps**: Sets roundness via the number of segmentation steps.
- **Move Piece:** Defines the piece for the next parameter operation. Starts with 0, -1 means nothing selected.
  - **Center Offset:** Offsets center for a certain piece.
  - **Height Offset:** Offsets height for a certain piece.
  - **Effect Range:** Sets the range of pieces to move.
  - **Effect Distribution:** Sets the distribution of the effect range.
- **Pos. Container:** Translates every child container to a piece of cake.
  - **Container Offset:** Adds a certain center offset to each container.
  - **Center Container**: Centers each translated container.
  - **PosX, Y, Z:** Activates container translation on the particular axis.
  - **Container Pos. Z%**: Sets the relative position on the z axis for each child container.
- **Map**:
  - **Color Bar:** Sets a single color (V texture coordinate) for a certain piece(e.g. 3 pieces: V = 0.0, V = 0.5 and V = 1.0).
  - **Vertex:** Uses vertex colors for each piece starting with ID 0. **Color ID** moves between the available vertex colors. **Color** lets you choose color for the current ID.

> ⚠ **Note:** A chamfer is a beveled edge connecting two surfaces.

To Create a Pie Chart

1. Create a new group and add the Pie Chart plug-in to it.
2. Open the Transformation Editor and set Position X and Y to -100.0 and Rotation Y to -25.0.
3. Add a material and/or a texture to it.
4. Open the Pie Chart editor and set the following parameters:
   - Set the following DataY values: 10,20,30,20,50,60,80,45,20.
   - Enable Keep Proportions.
   - Set Center Offset to 6.0.

- Set Move Piece to 1.
- Set Center and Height Offset to 20.0.
- Enable Vertex and set different colors for all Color IDs.

To Create a Pie Chart with Data Animation

Chart Animation using Data Import

Prepare data sets for animation in Excel file.

1. Start Microsoft Excel.
2. Enter **ExcelData1** into cell A1.
3. Add some sample values in the cells below (A2-A8: 80, 35, 45, 75, 85, 55, 60).
4. Enter **ExcelData2** into cell A2.
5. Add some sample values in the cells below (B2-B9: 80, 35, 45, 75, 40, 85, 55, 60).
6. Enter **ExcelData3** into cell A3.
7. Add some sample values in the cells below (C2-C9: 40, 60, 75, 85, 80, 55, 45, 35).
8. Enter **ExcelData4** into cell A4.
9. Add some sample values in the cells below (D2-D8: 60, 75, 85, 80, 55, 45, 35).
10. Rename this first sheet to **MyTable** (can be done with a double click on the sheet name at the bottom).
11. Follow one of the following methods to make Viz Engine can read this Excel data file
    a. **Save and close** the Excel document.
    b. **Share** the Excel document.
        i. On the Review tab, in the Changes group, click the **Share Workbook** button.
        ii. The Share Workbook dialog box will appear, and you select the **Allow changes by more than one user at the same time.** check box on the Editing tab.
        iii. Click **Ok** button of the dialog
        iv. Save the Excel document.

IMPORTANT! You must use the same platform (x64 / x86) of Microsoft Excel and Viz Engine.

Create a Data Animation scene

1. Follow the "To create a Pie Chart" instruction.
2. Modify the following parameters of Pie Chart plugin.
    - Change **Shared Mem.** to **Scene.**
    - Set **Data Delim.** to **#.**
    - Set **Values Animation** to **Timer.**
3. Add Data Import plugin into the same container.
    a. Set **File** to above prepared Excel file.
    b. Set **Table / Sheet** to **MyTable**.
    c. Set **Column(,Col...)** to **ExcelData1**.
    d. Set **Data Delim.** to **#.**
    e. Set **Shared Mem.** to **Scene.**
    f. Set **Key** to **Data.**
4. Click the **GetIt** button and chart shows from zero to seven nodes.
5. Change to the another data set.

a. Set **Column(,Col...)** to **ExcelData2** or **ExcelData3** or **ExcelData4.**
b. Click the **GetIt** button.

6. Repeat step 5 with another data set including **ExcelData1**.

> ⚠ **Note:** You can add more data columns and play with animation on difference data change.

Chart Animation using Control Chart

1. Follow the "To create a Pie Chart" instruction.
2. Modify the following parameters of Pie Chart plugin.
3. Change **Values Animation** property to **Timer.**
4. Add Control Chart plugin into the same container and save the scene.
5. Open Viz Trio.
   - Import the scene.
   - Click "1 (PieChart)" in Tab Fields.
   - Edit values in Editing Template.

See Also

- Control Pie
- Data Fit
- Data Import
- Data Label
- Data Storage
- Presenter

13.7.7 Scatter Chart

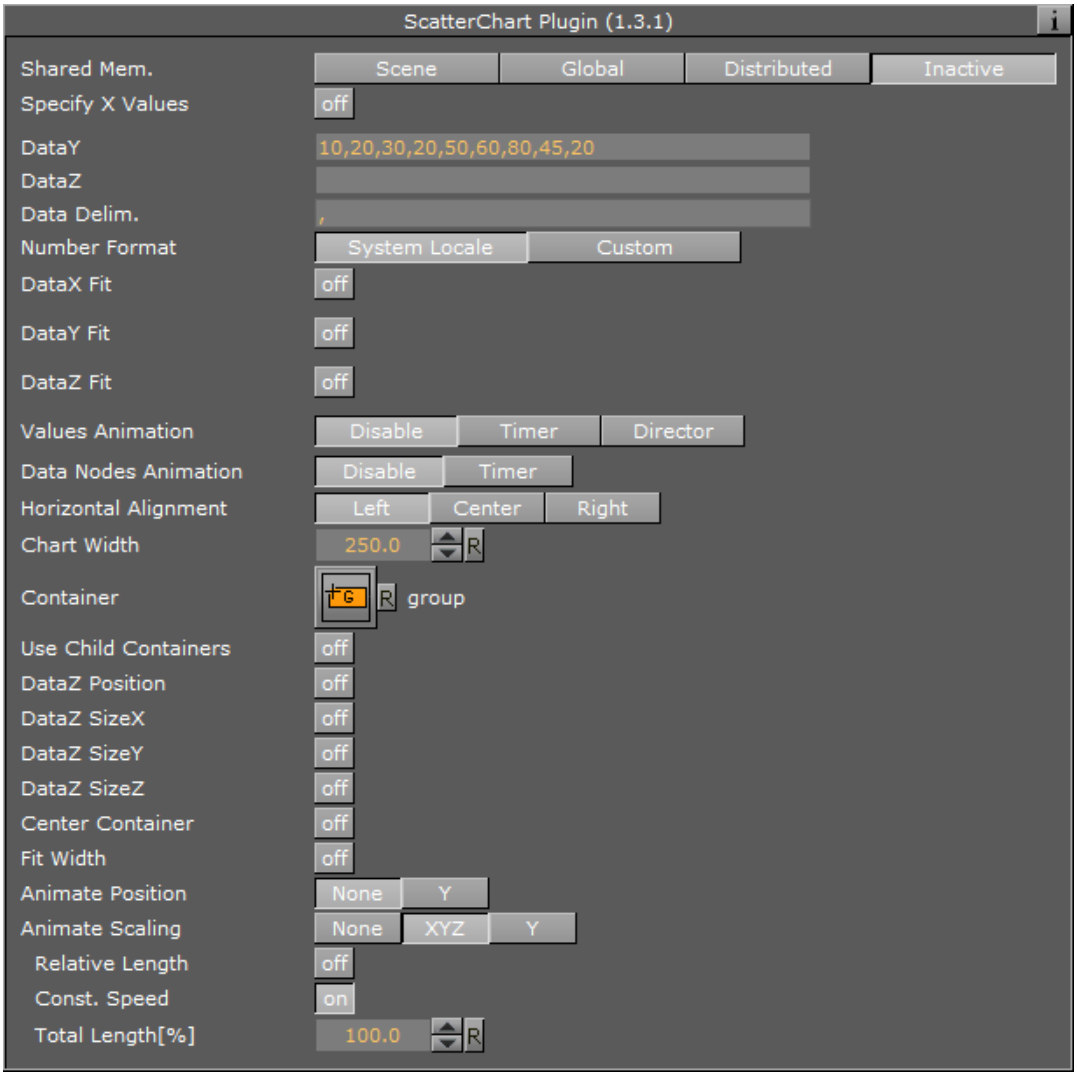
This plug-in draws a scatter chart, filled with data out of a Shared Memory Map. You can use delimited strings or arrays for data transfer via Scene-, Global- or Distributed-Shared Memory.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> VisualDataTools

This section contains the following topics:

- Scatter Chart Properties
- To Create a Scatter Chart
- To Create a Scatter Chart with Data Animation
  - Chart Animation using Data Import
  - Create a Data Animation scene
  - Chart Animation using Control Chart

Scatter Chart Properties

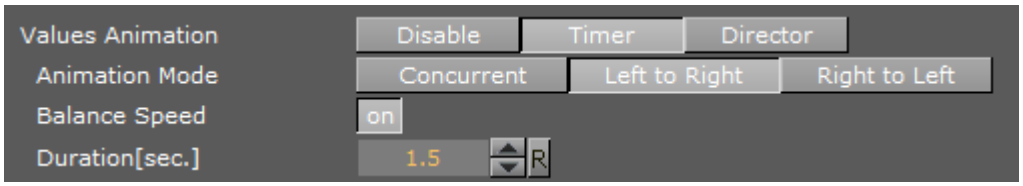


- **Shared Mem.:** Changes between **Scene**, **Global** and **Distributed** Shared Memory.
  Use **Inactive** memory to not forward any values via Shared Memory and the data is taken from the **Data** field.
- **Key DataX:** Determines Shared Memory key name for X values. DataX is the default input parameter for X values.
- **Key DataY:** Determines Shared Memory key name for Y values. DataY is the default input parameter for Y values.
- **Key DataZ:** Determines Shared Memory key name for Z values. DataZ is the default input parameter for Z values.
- **Transfer Mode:** Sets string- or array-based data transfer.
- **Data Delim.**: Defines the value separator sign(s).
- **Number Format**: Defines if Viz should get the decimal point separator from the `System Locale` or `Custom` settings. If Custom is enabled:

- **Decimal Symbol**: Defines which symbol is used as decimal point when **Custom Number Format** is set.

- **DataX, -Y, -Z Fit:** Enables data normalization.
- **DataX, -Y, -Z Scale:** Scales input by the selected factor.
- **DataX, -Y, -Z Offset:** Adds an offset to the incoming data.
- **DataX, -Y, -Z Auto Scale:** Enables automatic data normalization.
- **DataX, -Y, -Z Detect Limits:** Detects minimum and maximum of all values and scales them to adjusted Start and Stop.
- **DataX, -Y, -Z Threshold**: Adds a definable offset to the detected limit.
- **DataX, -Y, -Z Start:** Lower Auto Scale edge.
- **DataX, -Y, -Z Stop:** Upper Auto Scale edge.
- **Values Animation**:

  Sets the vertical animation type to be created when data change is detected:

  - **Disable**: No animation.
  - **Timer**: Creatse data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**. Also select an **Animation Mode**.
  - **Director**: Creates data transition animation with the Stage Director, that controls the Animation Progress parameter from 0.0% (old data) to 100.0% (new data). Also select an **Animation Mode**.
- **Animation Mode**:
  - **Concurrent**: Changes all nodes concurrently.
  - **Left to Right**: Transitions from left to right.
  - **Right to Left**: Transitions from right to left.
- **Balance Speed**: Relates node values to the position of transformed data when set to **On** and Animation Mode is set to **Left to Right** or **Right to Left**.

  > ⚠ **Example: Balance Speed**: If the values of four nodes change from 0,0,0,0 to 1,9,90,900 in **Left to Right** mode at 50% of the transformation. **Balance Speed Off**: Only the first two nodes will be transformed to new values. **Balance Speed On**: The first three nodes are already transformed to new values, and the last node also transformed to 44% of the new values, because the data change from 0 (0+0+0+0) to 1000 (1+9+90+900), at 50% of transformation (in this case the values is changed for 500 of 1000), the first 3 nodes are already finished and the last node is changed to 400.

- **Duration sec.**: Sets the duration of the animation in seconds for **Timer** mode.

- **Data Nodes Animation:**

  Sets the horizontal animation type to be created when number of data nodes change is detected.

  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**
- **Duration sec.:** Sets the duration of the animation in seconds for **Timer** mode.

> ⚠ When number of nodes is changes, scatter chart will looking for added and removed nodes from their values, add new nodes to the chart from zero width and resize all nodes to their new size in specified duration.

- **Horizontal Alignment**: Sets horizontal orientation to left, center or right.
- **Chart Width**: Adjusts the chart width.
- **Container:** Drag drop the container which will be used for the scatter nodes.
- **DataZ Position:** Uses DataZ for Z-axis positioning.
- **DataZ SizeX**: Allows Z Values to influence X scaling.
- **DataZ SizeY:** Allows Z Values to influence Y scaling.
- **DataZ SizeZ:** Allows Z Values to influence Z scaling.
- **Center Container**: Centers each translated container.
- **Fit Width**: Considers the width of the scatter node containers which means that the containers do not reach outside of the specified range.
- **Animate Position:** Sets the translation type: None: no animation, Y: animates the container along the Y axis (suggestion: Scale Y + DataLabel for BarChart alike labeling).
- **Animate Scaling:** Sets the scaling type: None: no scaling at all, XYZ: linear scaling in all directions, Y: scales in y direction only (->BarChart).
  - **Relative Length:** If activated, each scatter node will have its own 100% size (e.g. 7 nodes equal 700%).
  - **Const. Speed:** Sets the same animation duration for each bar.
  - **Total Length%:** Sets the accumulated size of all scatter nodes in percent.

To Create a Scatter Chart

1. Create a group and add the Sphere plug-in to it.
2. Open the transformation editor and set the Scaling (locked) to 0.2.
3. Add a material and/or a texture to the sphere.
4. Set the group with the sphere to hidden.
5. Create a new group and add the Scatter Chart plug-in to it.
6. Open the Transformation Editor and set Position X and Y to -100.0.
7. Open the Scatter Chart editor and set the following parameters:
   - Set the following DataY values: 10,20,30,20,50,60,80,45,20.
   - Set Width to 250.0.
   - Drag and drop the container with the Sphere to the Container placeholder.

To Create a Scatter Chart with Data Animation

Chart Animation using Data Import

Prepare data sets for animation in Excel file.

1. Start Microsoft Excel.
2. Enter **ExcelData1** into cell A1.
3. Add some sample values in the cells below (A2-A8: 80, 35, 45, 75, 85, 55, 60).
4. Enter **ExcelData2** into cell A2.
5. Add some sample values in the cells below (B2-B9: 80, 35, 45, 75, 40, 85, 55, 60).
6. Enter **ExcelData3** into cell A3.
7. Add some sample values in the cells below (C2-C9: 40, 60, 75, 85, 80, 55, 45, 35).
8. Enter **ExcelData4** into cell A4.
9. Add some sample values in the cells below (D2-D8: 60, 75, 85, 80, 55, 45, 35).
10. Rename this first sheet to **MyTable** (can be done with a double click on the sheet name at the bottom).
11. Follow one of the following methods to make Viz Engine can read this Excel data file
    a. **Save and close** the Excel document.
    b. **Share** the Excel document.
        i. On the Review tab, in the Changes group, click the **Share Workbook** button.
        ii. The Share Workbook dialog box will appear, and you select the **Allow changes by more than one user at the same time.** check box on the Editing tab.
        iii. Click **Ok** button of the dialog
        iv. Save the Excel document.

> **IMPORTANT!** You must use the same platform (x64 / x86) of Microsoft Excel and Viz Engine.

Create a Data Animation scene

1. Follow the "To create a Scatter Chart" instruction.
2. Modify the following parameters of Scatter Chart plugin.
    - Change **Shared Mem.** to **Scene.**
    - Set **Key DataY** to **DataY.**
    - Set **Data Delim.** to **#.**
    - Set **Values Animation** to **Timer.**
    - Set **Data Nodes Animation** to **Timer.**
3. Add Data Import plugin into the same container.
    a. Set **File** to above prepared Excel file.
    b. Set **Table / Sheet** to **MyTable**.
    c. Set **Column(,Col...)** to **ExcelData1**.
    d. Set **Data Delim.** to **#.**
    e. Set **Shared Mem.** to **Scene.**
    f. Set **Key** to **DataY.**
4. Click the **GetIt** button and animation chart shows from zero to seven nodes.
5. Add one data node in the middle of chart.

   a. Set **Column(,Col...)** to **ExcelData2.**
   b. Click the **GetIt** button.
6. Change values of eight data nodes.
   a. Set **Column(,Col...)** to **ExcelData3.**
   b. Click the **GetIt** button.
7. Remove the first data node.
   a. Set **Column(,Col...)** to **ExcelData4.**
   b. Click the **GetIt** button.

⚠ **Note:** You can add more data columns and play with animation on difference data change.

Chart Animation using Control Chart

1. Follow the "To create a Scatter Chart" instruction.
2. Modify the following parameters of Scatter Chart plugin.
   - Set **Values Animation** to **Timer.**
   - Set **Data Nodes Animation** to **Timer.**
3. Add Control Chart plugin into the same container and save the scene.
4. Open Viz Trio.
   - Import the scene.
   - Click "1 (ScatterChart)" in Tab Fields.
   - Edit values in Editing Template.

See Also

- Control Chart
- Data Fit
- Data Import
- Data Label
- Data Storage

13.7.8 Stock Chart

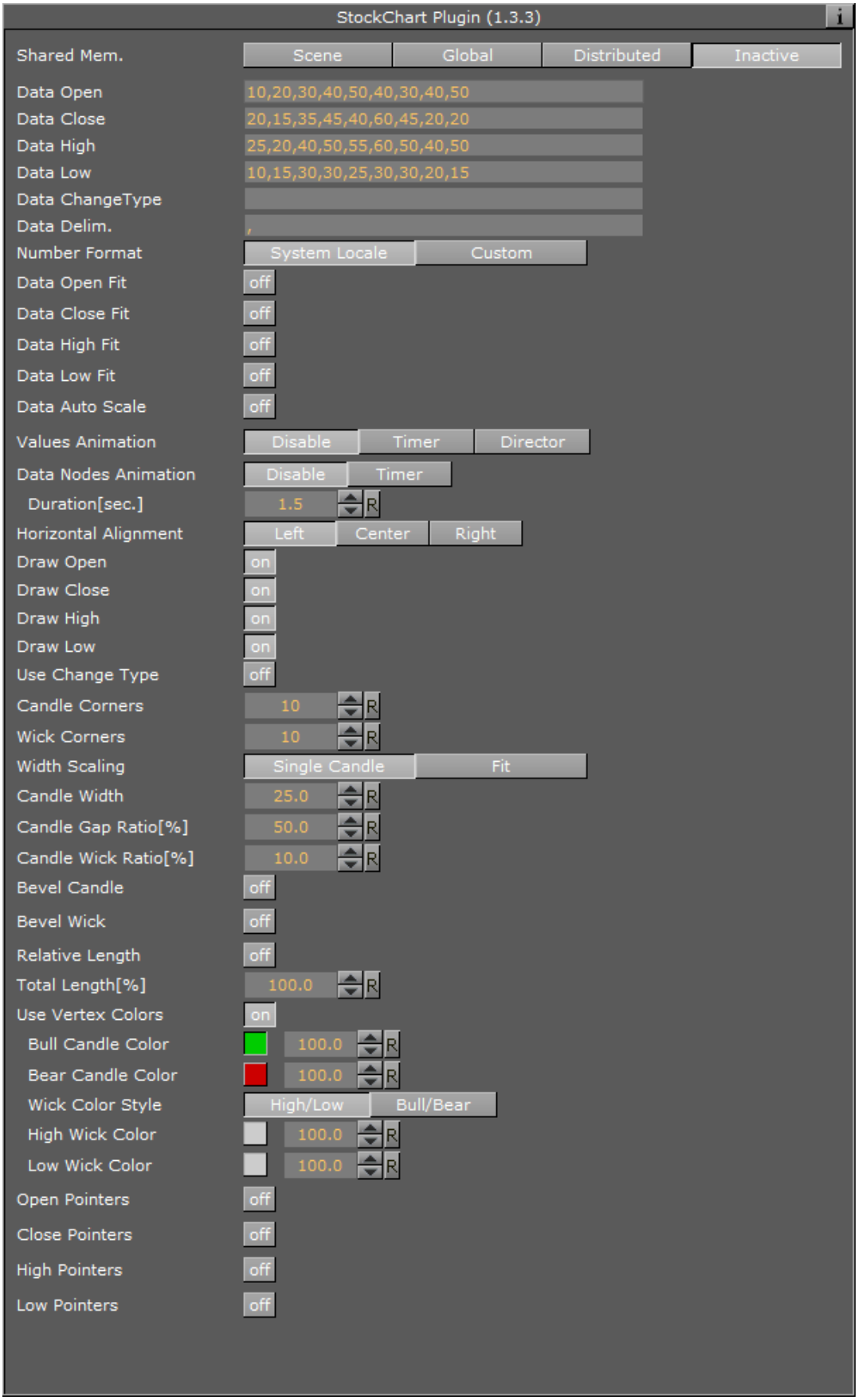
This plug-in draws a stock chart, filled with data out of a Shared Memory Map. You can use delimited strings or arrays for data transfer via Scene-, Global- or Distributed-Shared Memory.

⚠ **Note:** This plug-in is located in: Built Ins -> Geom plug-ins -> VisualDataTools

This section contains the following topics:

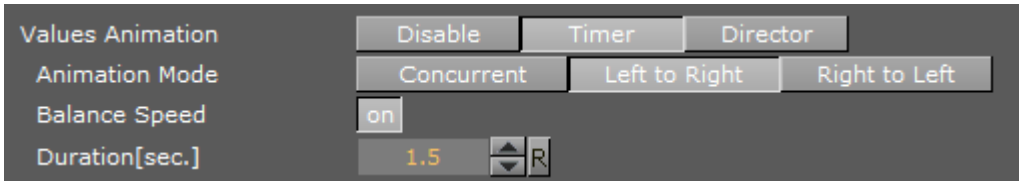
- Stock Chart Properties
- To Create a Stock Chart
- To Create a Stock Chart with Data Animation
  - Chart Animation using Data Storage
  - Chart Animation using Control Chart

Stock Chart Properties



- **Shared Mem.**: Changes between Scene-, Global- and Distributed-Shared Memory. Use Inactive memory to not forward any values via Shared Memory.
- **Key Open:** Determines Shared Memory key name for open values. DataOpen is the default input parameter for open values.
- **Key Close**: Determines Shared Memory key name for close values. DataClose is the default input parameter for close values.
- **Key High:** Determines Shared Memory key name for high values. DataHigh is the default input parameter for high values.
- **Key Low**: Determines Shared Memory key name for low values. DataLow is the default input parameter for low values.
- **Transfer Mode:** Sets string- or array-based data transfer.
- **Data Delim.**: Defines the value separator sign(s).
- **Number Format**: Defines if Viz should get the decimal point separator from the `System Locale` or `Custom` settings. If Custom is enabled:
  - **Decimal Symbol**: Defines which symbol is used as decimal point when **Custom Number Format** is set.
- **Data Open, -Close, -High, -Low Fit**: Enables data normalization.
  - **Data Open, -Close, -High, -Low Scale:** Scales input by the selected factor.
  - **Data Open, -Close, -High, -Low Offset**: Adds an offset to the incoming data.
  - **Data Auto Scale:** Enables automatic data normalization.
- **Value Animation**:

  Sets the vertical animation type to be created when data change is detected:



  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**. Also select an **Animation Mode**.
  - **Director**: Creates data transition animation with the Stage Director, that controls the Animation Progress parameter from 0.0% (old data) to 100.0% (new data). Also select an **Animation Mode**.
- **Animation Mode**:
  - **Concurrent**: All nodes change concurrently
  - **Left to Right**: Transition from left to right
  - **Right to Left**: Transition from right to left
- **Balance Speed**: Set to **On**, when Animation Mode is set to **Left to Right** or **Right to Left**, and the position of transformed data will be related to node values as well

  ⚠ **Example: Balance Speed**: If the values of four nodes change from 0,0,0,0 to 1,9,90,900 in **Left to Right** mode at 50% of the transformation. **Balance Speed Off**: Only the first two nodes will be transformed to new values. **Balance Speed On**: The first three nodes are already transformed to new values, and the last

node also transformed to 44% of the new values, because the data change from 0 (0+0+0+0) to 1000 (1+9+90+900), at 50% of transformation (in this case the values is changed for 500 of 1000), the first 3 nodes are already finished and the last node is changed to 400.

- **Duration sec.**: Sets the duration of the animation in seconds for **Timer** mode.

- **Data Nodes Animation:**

  Sets the horizontal animation type to be created when number of data nodes change is detected.

  - **Disable**: No animation.
  - **Timer**: Creates data transition animation (from an old set of data to a new set) within the specified time set in **Duration sec.**
- **Duration sec.:** Sets the duration of the animation in seconds for **Timer** mode.

  ⚠ When number of nodes is changes, stack chart will looking for added and removed nodes from their values, add new nodes to the chart from zero width and resize all nodes to their new size in specified duration.

- **Data Detect Limits:** Detects minimum and maximum of all values and scales them to adjusted Start and Stop.
- **Data Start:** Lower Auto Scale edge.
- **Data Stop:** Upper Auto Scale edge.
- **Draw Open**: Draws open data.
- **Draw Close**: Draws close data.
- **Draw High:** Draws high data.
- **Draw Low:** Draws low data.
- **Candle Corners**: Sets candle segments.
- **Wick Corners:** Sets wick corners.
- **Width Scaling:**
  - **Single Candle:** Adjusts size for a single candle.
  - **Fit:** Adjusts all candles to fit a certain width.
- **Candle Width:** Value for the previous parameter.
- **Candle Gap Ratio%:** 100% means that the gaps have the same size as the candles.
- **Candle Wick Ratio%:** 100% means that the wicks have the same size as the candles.
- **Bevel Candle**: Activates Bevel mode for the candle geometry.
  - **Bevel Candle Top**: Chamfers candle's top.
  - **Bevel Candle Bottom:** Chamfers candle's bottom.
  - **Bevel Candle Size%:** Adjusts bevel's size from 0 to 100.
  - **Bevel Candle Steps:** Sets roundness via the number of segmentation steps.
- **Bevel Wick**: Activates Bevel mode for the wick geometry.
  - **Bevel Wick Top:** Chamfers wick's top.

- **Bevel Wick Bottom:** Chamfers wick's bottom.
- **Bevel Wick Size%:** Adjusts bevel's size from 0 to 100.
- **Bevel Wick Steps:** Sets roundness via the number of segmentation steps.

- **Relative Length**:
  - **On:** Sets all accumulated candle lengths to equal 100%.
  - **Off:** Sets each candle to its own 100% (e.g. 3 candles: 300%).
- **Total Length:** Value for the previous parameter.
- **Use Vertex Colors**: Activates the following parameters.
  - **Bull Candle Color**: Defines the color for all candles where the close value is higher than the open value.
  - **Bear Candle Color:** Defines the color for all candles where the close value is lower than the open value.
  - **High Wick Color:** Sets high wick's color.
  - **Low Wick Color:** Sets low wick's color.

> ⚠ **Note:** A chamfer is a beveled edge connecting two surfaces.

To Create a Stock Chart

1. Create a new group and add the Stock Chart plug-in to it.
2. Open the Transformation Editor and set Position X and Y to -100.0.
3. Open the Stock Chart editor and set the following parameters:
   - Set the Data Open parameter to the following values: 10,20,30,40,50,40,30,40,50.
   - Set the Data Close parameter to the following values: 20,15,35,45,40,60,45,20,20.
   - Set the Data High parameter to the following values: 25,20,40,50,55,60,50,40,50.
   - Set the Data Low parameter to the following values: 10,15,30,30,25,30,30,20,15.
   - Set Candle and Wick Corners to 10.
   - Set Candle Width to 25.0.

To Create a Stock Chart with Data Animation

Chart Animation using Data Storage

1. Follow the "To create a Stock Chart" instruction.
2. Modify the following parameters of Stock Chart plugin.
   - Change **Shared Mem.** to **Scene.**
   - Set **Values Animation** to **Timer.**
   - Set **Data Nodes Animation** to **Timer.**
   - Set **Width Scaling** to **Fit.**
   - Set **Candle Width** to 325.
3. Add Data Storage plugin into the same container.
   - Set **Key Data1** to **DataOpen.**

- Set **Data1** to 10,20,30,40,40,30,40,50.
- Set **Key Data2** to **DataClose.**
- Set **Data2** to 20,15,35,45,60,45,20,20.
- Set **Key Data3** to **DataHigh.**
- Set **Data3** to 25,20,40,50,60,50,40,50.
- Set **Key Data4** to **DataLow.**
- Set **Data4** to 10,15,30,30,30,30,20,15.

4. Change **Shared Mem.** to **Scene** and animation chart shows from nine to eight nodes.
5. Update data with the same number of nodes.
   - Change **Shared Mem.** to **Inactive.**
   - Set **Data1** to 40,30,50,25,20,40,55,45.
   - Set **Data2** to 25,45,20,20,30,45,50,40.
   - Set **Data3** to 55,50,50,40,40,55,55,50.
   - Set **Data4** to 10,15,10,15,15,20,40,30.
6. Change **Shared Mem.** to **Scene** and values of nodes change in the animation.
7. Update data by adding two nodes.
   - Change **Shared Mem.** to **Inactive.**
   - Set **Data1** to 40,30,**20**,50,25,20,40,**30**,55,45.
   - Set **Data2** to 25,45,**45**,20,20,30,45,**45**,50,40.
   - Set **Data3** to 55,50,**50**,50,40,40,55,**55**,55,50.
   - Set **Data4** to 10,15,**20**,10,15,15,20,**25**,40,30.
8. Change **Shared Mem.** to **Scene** and animation chart shows from eight to ten nodes.

IMPORTANT! You must use the same platform (x64 / x86) of Microsoft Excel and Viz Engine.

Chart Animation using Control Chart

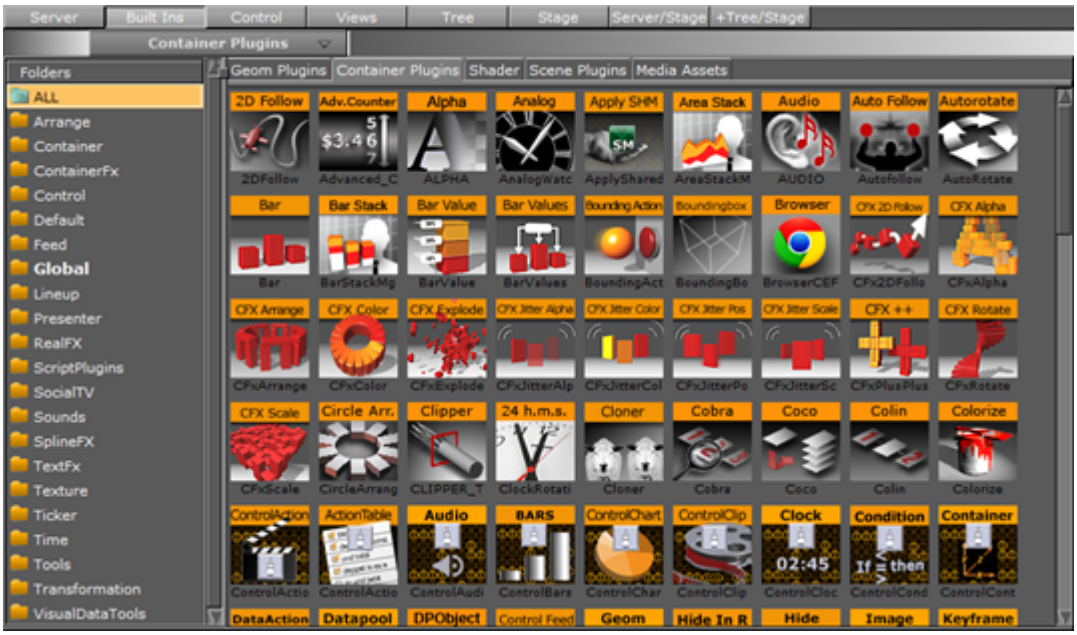
1. Follow the "To create a Stock Chart" instruction.
2. Modify the following parameters of Stock Chart plugin.
   - Set **Values Animation** to **Timer.**
   - Set **Data Nodes Animation** to **Timer.**
   - Set **Width Scaling** to **Fit.**
   - Set **Candle Width** to 325.
3. Add Control Chart plugin into the same container and save the scene.
4. Open Viz Trio.
   - Import the scene.
   - Click "1 (StockChart)" in Tab Fields.
   - Edit values in Editing Template.

See Also

- Control Chart
- Data Fit
- Data Import
- Data Label

- Data Storage

14 Container Plug-Ins



The default path for Container plug-ins is `<viz install folder>\plugin`.

14.1 Arrange

These Arrange Container plug-ins are located in the Arrange folder:

- Circle Arrange
- Grid Arrange
- Time Displacement

14.1.1 Circle Arrange

Circle Arrange is a tool for arranging a set of containers into a circular structure.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> Arrange*

This section contains information on the following topics:

- Circle Arrange Properties
- To Arrange Containers in a Circle

Circle Arrange Properties

- **Number of items:** Allows you to define how many items the structure is to made up of.
- **Number of Rows:** Sets the number of rows/rings for the circular structure.
- **Distribution**: Allows you to define how the containers are to distributed in the circular shape.
  - **Monospace:** Sets the same number of items in each circle row. The spacing between objects in the outer rows will then be bigger than in the inner rows.
  - **Equal**: Distributes the objects between the rows so the distance between the objects in the different rows are equal. The number of containers will then be larger in the outer rows.
- **Inner Radius**: Sets the radius from the center to the first container ring.
- **Outer Radius**: Sets the radius from the center to the outer ring.
- **Start Angle**: Setting this above -180° will create an open section in the circle structure. The opening is created in a clockwise direction.
- **End Angle**: Setting this below 180 will create an open section in the circle structure. The opening is created in a counterclockwise direction.
- **Orientate**: Orients the containers with their bottom towards the center of the circular structure when enabled.
- **Rebuild Containerlist**: Rebuilds the structure after having added or removed containers form the group.

To Arrange Containers in a Circle

1. Create a group container, and name it **Circle**.
2. Add a **Sphere** as a Sub-Container to the Circle container.
3. Add a material and/or a texture to the Sphere container.
4. Open the Transformation Editor for the Sphere container, and set **Scaling** (locked) to **0.2**.
5. *Optional*: Animate the Sphere object.
6. Make 15 copies of the Sub-Container, totalling the number of Sphere containers to 16. All copies should be placed as Sub-Containers of the Circle container and have the same position as the original.
7. Add the Circle Arrange container plug-in to the Circle container.
8. Open the Circle Arrange editor and do the following:
   - Set **Number of items** to **16**.
   - Set **Number of Rows** to **2**.
9. Click the **Rebuild Containerlist** button.
   - The plug-in will position all Sub-Containers in a circular structure.
     The order of the containers in the group decides their placing in the circular structure. If more than one row is selected, the first containers end up in the outer circle and the last containers in the inner circle.

See Also

- Grid Arrange

14.1.2 Grid Arrange

The Grid function is an effective tool for arranging a set of containers into a grid or tabular structure.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> Arrange*

This section contains information on the following topics:

- Grid Arrange Properties
- To Arrange Containers in a Grid

Grid Arrange Properties

- **Number of Rows:** Sets the number of rows the containers are to be distributed over.
- **Number of Columns**: Sets the number of columns the containers are to be distributed over.
- **Row Offset**: Sets the distance between the rows.
- **Column Offset**: Sets the distance between the columns.
- **Start In Corner:** Selects which corner to start the arrangement.

> ⚠ **Note:** The product of number of rows and number of columns should normally be equal number of containers. If it is higher, all rows will not be equal, if it is lower, some containers will not be shown, but they will still be rendered.

To Arrange Containers in a Grid

1. Add container to the Scene Tree
2. Name it **Grid**.
3. Add a **Sphere** as a Sub-Container to the **Grid** container.
4. Add a material and/or a texture to the Sphere container.
5. Open the Transformation Editor for the Sphere container.
6. Set **Scaling** (locked) to **0.2**.
7. *Optional*: Animate the Sphere object.
8. Create 15 copies of the Sub-Container (total of 16 Sphere containers). All copies should be placed as Sub-Containers of the Grid container and have the same position as the original.
9. Add the Grid Arrange plug-in to the **Grid** container.
10. Open the Grid Arrange editor and set these parameters:
    - **Number of Rows**: 4
    - **Number of Columns:** 4
    - **Row Offset**: 40.0

- **Column Offset:** 40.0
  The plug-in will position all Sub-Containers 4in a grid structure.
  The order of the containers in the group decides their placing in the grid structure. If more than one row is selected, the first containers end up in the first row(s) and the last containers in the last row(s).

See Also

- Circle Arrange

14.1.3 Time Displacement

Time Displacement allows you to use a gray-scaled alpha image for setting animation time offsets for all the objects in a group container. This is typically useful if you want to create some kind of randomized animation effect or that should follow a certain pattern.

What the function does is to create links between the animated containers and the pixels of the gray-scale image and then use the alpha level of each pixel to set the offset of the containers. The range from 100% alpha to 0% alpha on the pixels sets the offset level. By default 100% alpha sets no offset and 0% sets maximum offset.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Arrange

This section contains information on the following topics:

- Time Displacement Properties
- To Set Animation Time Offsets with a Grayscaled Alpha Image

Time Displacement Properties

- **Image**: Drag the alpha image you want to use for creating the containers offsets onto the drop zone here.
- **Invert**: Inverts the link between the image pixels and offsets. 0% gives no offset and 100% gives maximum.
- **Image Width:** Sets the height of the alpha image.
- **Image Height**: Sets the height of the alpha image. The product of **Image Width** multiplied with **Image Height** should normally be the same as the number of containers that are to be used in the function.
- **Pixel Reference**: Defines the way the pixels should be associated to the animated containers. You can choose between:
  - **Order in Tree**: The container order in the transformation tree decides how the pixels should be associated.
  - **Name of Container:** The container name sets the order in which the containers will be associated with the pixels.

- **Search Tree:** Sets the search level of the function, when it decides which Sub-Containers to include in the function:
  - **One Level:** Only the first level of Sub-Containers is being included in the function.
  - **Recursive:** All levels of Sub-Containers are being included.
- **Threshold Action:** Executes if the pixel value is lower than the threshold value.
  - **None**: No action is performed.
  - **Skip Pixel**: Skips the pixel in the offset process.
  - **Visible**- Makes objects visible when the pixel value is lower than the threshold value.
  - **Invisible**- Makes objects invisible when the pixel value is lower than the threshold value.
- **Threshold**: Sets the Threshold value where one of the above selected actions will be performed.
- **Stage Range (frames)**: Sets the range of the offset distribution. A high value will result in bigger differences in offset between containers that are associated with different pixel alpha values.
- **Stage Offset (frames):** Adds a static offset value to all containers.
- **Filter Type**- Filters the containers in or out. Available options are; **None**, **Including** and **Excluding**.
- **Filter String**- Sets the name of the containers that are to be included in the filter. Use of asterisks is supported.
- **Execute**: Starts the operation.

To Set Animation Time Offsets with a Grayscaled Alpha Image

1. Add a group to the scene tree and name it TimeDisplace, and the TimeDisplace plug-in to it.
2. Create a Sub-Container of the TimeDisplace container and name it Sphere.
3. Create a new group container as a Sub-Container of Sphere.
4. Add a Sphere geometry plug-in and material to the group container.
5. Animate the group container from Position X -200.0 to 200.0.
6. Make 6 copies of the Sphere container under the TimeDisplace container.
7. Add the Grid Arrange plug-in to the TimeDisplace container.
8. Open the Grid Arrange editor and set Number of Rows to 6 and Row Offset to 30.0.
9. *Optional*: Remove the Grid Arrange plug-in.
10. Add the Corena plug-in and rename the containers to Sphere, starting at index 0. This will allow you to filter out specific containers using the TimeDisplace filter.
11. *Optional*: Remove the Corena plug-in.
12. Open the TimeDisplace editor and add an alpha image to the Image drop zone.
13. Click **Execute**.

    
14. Click the Stage to see the offsets being added to the containers.
15. Run the animation to see the time offset in action!

See Also

- Image Editor
- Grid Arrange
- Corena

14.2 Container

These Container plug-ins are located in the Container folder:

- BoundingBox
- Cobra
- Coco
- Colin
- Cora
- Corena
- Toggle

14.2.1 BoundingBox

This plug-in allows to override the standard Bounding Box of containers independently of the type of the container.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Container

This section contains information on the following topics:

- BoundingBox Properties

BoundingBox Properties

- **X Left Bottom Back:** Changes the dimension in X-direction starting from left.
- **Left Bottom Back:** Changes the dimension in Y-direction starting from bottom.
- **Left Bottom Back:** Changes the dimension in Z-direction starting from flipside.
- **Right Top Front:** Changes the dimension in X-direction starting from right.
- **Right Top Front:** Changes the dimension in Y-direction starting from top.
- **Right Top Front:** Changes the dimension in Z-direction starting from front.
- **Set to Standard:** Initializes values to enclose the entire container object.

14.2.2 Cobra

The Cobra function is an easy to use function replace (parts) of container names. Enter a search pattern and a string the search should be replaced with and execute.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Container

This section contains information on the following topics:

- Cobra Properties
- To Rename Container(s)

Cobra Properties

- **Search pattern**: Sets the text string to be replaced.
- **Search pattern position:** Defines the position within the string. It can be set to Begin, End and Any.
- **Replace with**: Sets the text string to be used instead of the search pattern.
- **Search level:**
  - **Same**: Affects only containers on the same hierarchy level.
  - **Down**: Affects only Sub-Containers of the current container.
  - **Tree**: Searches the whole Scene Tree.
- **Add Index:** Adds a numerical index to the new name when set to **On**.
  - **Starting Index**: Sets the number to start the index.
  - **Reverse Index**: Uses a reverse index list when set to **On**.
- **Replace**: Starts the replace process.

To Rename Container(s)

1. Add the Cobra plug-in to a container that, for example, has Sub-Containers that need be renamed.
2. Open the Cobra editor.
3. Enter a **Search pattern** text string (e.g. Copy)
4. Set the **Search pattern position** to **Begin**.
5. Enter the **Replace with** text string (e.g. Sphere).
6. Set the **Search level** to **Down**.
7. Click **Replace**.

See Also

- Coco
- Corena

14.2.3 Coco

The Coco plug-in allows you to create copies of a container with great ease. This function is typically used together with the Arrange plug-ins.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Container

This section contains information on the following topics:

- Coco Properties
- To Create Copies of Containers

Coco Properties

- **Number of Copies:** Sets the number of copies to be created when the function is executed.
- **Scene Tree Level:** Allows you to select the hierarchic position of the copies.
  - **same**: Sets the copies on the same level as the source container.
  - **down**: Sets the copies as Sub-Containers to the source container.
- **Clear Transformation**: Sets all transformation values on the new copies, like scaling, position and rotation, back to zero. All copies will then be stacked at a initial position with all transformation values set to zero.
- **Starting Index**: Allows you to alter the starting point of the index numbering of the copies. By default the numbering starts on 1.
- **Container Name:** Allows you to change the name the containers will be given by default.
- **Execute**: After having set the required parameters, click this button to execute the duplicating process.

To Create Copies of Containers

1. Add the Coco plug-in to the container you want to copy.
2. Open the Coco editor and enter the Number of Copies (e.g. 10)
3. Set Scene Tree Level to **down**.
4. Click the **Execute** button.

See Also

- Arrange
- Cobra

14.2.4 Colin

The Colin plug-in allows you to align a container in the scene using two other containers as reference. This makes it easy to maintain a good and controlled symmetry in the scene.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Container

This section contains information on the following topics:

- Colin Properties
- To Align Containers

Colin Properties

- **Container 1:** Is the drop zone for the first referential container.
- **Container 2**: Is the drop zone for the second referential container.
- **HOffset %:** Sets the horizontal offset between the referential containers in percent. Horizontal means here an invisible line between the referential containers regardless of the actual boning of the line.
- **VOffset**: Sets the vertical offset from the horizontal line.
- **ZOffset**: Sets the Z offset from the line between the referential containers.
- **Align container rotation**
  - **2D**: When enabled, this option will position and rotate the container in the X- and Y-axis to keep itself aligned on the invisible line between the referential containers as they move.
  - **3D**: When enabled, this option will position and rotate the container in the X-, Y- and Z-axis to keep itself aligned on the invisible line between the referential containers as they move.
- **Automatic label orientation:** This parameter only works if the **Align container rotation** parameter is switched to **2D** or **3D**. It adapts automatically the rotation angles.
- **Autoscale object:** Unlocks the **Scaling** value.
- **Scaling**: Lets you enter the value the container should be scaled to.

To Align Containers

1. Add the Cube, Sphere and Icosahedron geometries to the scene tree.
2. Open the transformation editor for Cube and Icosahedron and set them to Position X -200.0 and 200.0, respectively, and Position Y 100.0.
3. Add the Colin plug-in to the Sphere container.
4. Open the Colin editor and drag and drop the Cube and Icosahedron containers onto the Container1 (left) and Container2 (right), respectively.
   - The container is then aligned between the two referential containers.

14.2.5 Cora

The Cora plug-in allows you to sort a set of Sub-Containers based on multiple criteria. It is a useful tool when your scene starts getting complex and it is difficult to maintain the overview. To make sorts by different criteria it can help you reorganize the scene tree structure. It works on all containers under the container it is added to.

The plug-in is often used together with the Time Displacement plug-in which enables many creative possibilities.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Container

This section contains information on the following topics:

- Cora Properties
- To organize the scene tree

Cora Properties

- **Criteria**: Selects the wanted sort criteria.
  - **Name**: Performs a case sensitive sort. In normal sort order, capitalized letters are listed first. Be aware, use letter case consistently if you want to have a correct result when you perform a sort by name.
  - **X**: Sorts by X-values.
  - **Y**: Sorts by Y-values.
  - **Z**: Sorts by Z-values.
  - **Vis**: Sorts all the visible containers first.
  - **Alpha**: Sorts by the alpha value among those containers who have an alpha function attached. Alpha values set on the material of a container are ignored.
  - **Key:** Sorts containers by key function. Among those with key signal attached, the key alpha value is used for the further sorting.
  - **Rand**: Sorts by random.
- **Mode**: Sets the order of the sorting. Choose between setting the **Highest** or the **Lowest** values.
- **Random Seed**: Is relevant if you have chosen random as your sort criteria. It specifies a seed for the random number generator. Even though Viz Artist use random numbers, the animation for a specific random seed will always look the same.
- **Random Weight:** Defines how random the randomize function will sort. Random 1.0 means random, random 0.0 does nothing and random 0.5 gives you a 50% chance if a container is resorted or stays in place.
- **Execute:** Starts the operation.

To organize the scene tree

- Add the Cora plug-in to the container that holds the Sub-Containers you want to sort, select the sort criteria and press the Execute button.

See Also

- Time Displacement

14.2.6 Corena

The Corena plug-in renames all Sub-Containers of the container you apply it to. The plug-in is often used together with the Time Displacement plug-in which enables many creative possibilities.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Container

This section contains information on the following topics:

- Corena Properties
- To Rename Container(s)

Corena Properties

- **Name:** Sets the name you want to rename the Sub-Containers to. You can define the number format of the new names by entering a hash in addition to the name for each digit you want in the numbering. If you enter ###newname, the containers will be named: 001newname, 002newname, and so on.
- **Add Index:** Enables/disables creation of a numbering index.
- **Starting Index:** Sets the number you want the numbering index to start from.
- **Reverse Index:** Creates the index in a reversed order when enabled.
- **Execute:** Starts the operation.

To Rename Container(s)

1. Add the Corena plug-in to the container that holds the containers you want to rename.
2. Enter the new name, set the indexing parameters and click **Execute**.

See Also

- Cobra

14.2.7 Toggle

Toggle is used to toggle between two objects. The plug-in is most commonly used with Transition Logic scenes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Container

This section contains information on the following topics:

- Description and Methodological Considerations
  - Toggle placeholder containers
  - Toggle directors and Key Frames
- Toggle Properties

Description and Methodological Considerations

Toggle plug-in is one of the main building blocks of Transition Logic, and it is in general a relatively simple plug-in. When the Toggle plug-in is added to a container it will create two child containers that act as placeholder containers for objects that can toggle on and off screen. In addition it will also create some utility directors in the stage.

Toggle placeholder containers

The two placeholder containers correspond to an **A**-side and a **B**-side. Each of the placeholders will again have a child container; one named **current** the other named **other**. The *current* is typically the visible container, while the *other* is not visible. At the time of a switch, these containers will switch names. This means that at any given time it is not possible to know if the *current* parent is the *A* side or the *B* side. It is the responsibility of the Toggle plug-in to keep track of this.

Toggle directors and Key Frames

The Toggle plug-in will create two directors each containing a default 1 second cross fade animation. These animations are placed in the **AX** and **BX** directors. *AX* can be interpreted as the *A*-side cross fade. The animations can be modified to show any kind of transition, but it is important that the *AX* and *BX* are the same.

In addition to the cross fade directors the Toggle plug-in will also create placeholder directors for the *object* animations. Merged objects that are loaded into these placeholder containers during playback may contain animations embedded in the geometry object. The object animations are automatically added as a sub-director to the *AO* or *BO* director depending on if the object is being loaded on the *A*-side or *B*-side.

> ⚠ **Caution:** It is NOT recommended to manually adjust the AO and BO directors.

There are two different scenarios of when the utility directors will be triggered:

1. When there is a state change.
   - When there is a state change the Toggle-Layer director will be animated from the current state to the required next state. In this scenario the utility directors will be triggered according to the toggle Key Frames in the stage. This means that it is the position of these Key Frames that will determine the timing of when the utility directors are triggered.
2. No state change, only changing the current object.
   - When there is no state change there will be no movement of the Toggle-Layer director. In this case the Toggle switch action will be triggered, which in effect will animate the

utility directors at the same time; animating the current object out and the next object in at the same time.

> ⚠ **Note:** Placeholder containers and directors are only generated if none exists from before.

The plug-in has a built-in support for object loading. Enter the path to an object (e.g. GEOM*myfolder/myobject) into the **Object** field, and the plug-in will load this object on the *other* container. The object will by default be invisible and can be loaded with data, before activating the Switch action to show the object.

An object loaded this way will be cached, so that requesting the same object again will not trigger loading again. The object cache can also be preloaded by specifying all the geometries that should be preloaded before initializing the Toggle plug-in. Preloading of the objects needed by the Toggle plug-in is done when a playlist is initialized. For this reason the *Preload Object* text field should normally not be manually modified in Viz Artist.

Before adding the plug-in to a container in a scene, please make sure that:

1. You give the container a unique and sensible name. This name will become the Toggle-Layer name. Also, as the Toggle plug-in will automatically generate directors, which will be named based on the container name and its parents names, it is worth considering if the location of the container that will host the Toggle plug-in is in the correct position in the scene tree.
2. The container the Toggle plug-in will be added to must not have any child containers.
3. A director with the same name as the container (Toggle-Layer name) is created.
4. The director has a minimum of two stop points and that these stop points are named correctly. The names must correspond with the Toggle-Layer states.

> ⚠ **IMPORTANT!** One of the Transition Logic requirements is to have a stop point named "O". Best practice is to always name the first stop point "O" (frame 0).

After the Toggle plug-in has been added to the container, the plug-in icon can be clicked to see its properties. Clicking the Default Key Frames button will create the most commonly used Key Frames for triggering the toggling between the current and the other objects. Adjusting the timing can easily be done by moving the Key Frames in the stage.

Toggle Properties

- **Object:** Shows the name of the currently loaded object (front scene GEOM). This field changes depending on the object loaded in the layer the toggle plug-in resides.
- **Show advanced:** Shows the advanced parameters.
  - **Auto-continue other object on Switch:** Toggles a continue action to the other placeholder when a switch action is triggered when enabled.
  - **State control only:** Ignores all commands sent to the Toggle will be ignored when enabled. This mean that no object will be loaded and only state changes on the layer director will be run.

- **Controls video:** Controls the toggling of video channel 3 and 4. This applies when using the Control Video plug-in.
- **Preload objects:** Specifies the objects that are to be pre-loaded on initialization.
- **Director commands:** Forms parameters into a valid set-director-position command, and applies them to either current or other placeholder object director. This field can be filled with certain director command parameters like for instance: SHOW 0.5, SHOW 50f, SHOW $pilot1.
- **Run director command on current** and **Run director command on other**: Forms and applies a director command based on the director_commands field as described above.

  > ⚠ **Note:** These options will be deprecated in future versions.

- **Switch without animation:** Switches the visible container without running any of the (A|B)X directors or starting the object animation. The AX and BX directors will be set to the correct in/out positions.
- **Dummy:** Is animated to create an animation that spans all the other toggle Key Frames. This will in some cases solve problems with toggle Key Frames not being triggered correctly.

- **Switch:** Animates out the current placeholder, then switches the other placeholder to current and animates it in.
- **Out:** Animates out the current placeholder.
- **In:** Animates in the current placeholder.
- **Switch in:** Switches the other placeholder to current and animates it in.
- **Switch in preview:** Hides the current placeholder without animating it, then switches the other placeholder to current. Afterwards it shows the new current placeholder by jumping to a director state suitable for generating a still preview.
- **Default Key Frames:** Inserts default Toggle Key Frames to invoke the toggles utility directors.
- **Continue:** Sends a continue action to the current object animation director.
- **Delete default Key Frames:** Removes the default Key Frames that were generated by the Default Key Frames button.
- **Initialize:** Clears the cache, and reloads pre-loaded geometries.
- **Position object director at preview location:** Positions the object director at given preview location in position field when enabled.
- **Move object director to next stop:** Moves the stage position to the next director stop point on a loaded object.
- **Delete containers for cached object:** Deletes all the containers used as the Toggles object cache.

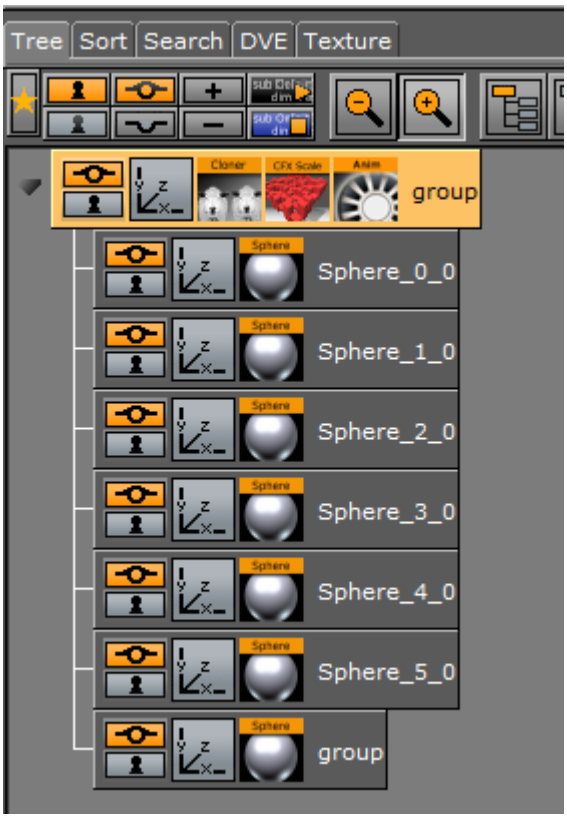
See Also

- Transition Logic

14.3 Container FX

Container FX is much like TextFX, except that instead of text, a set of containers are added under the plug-in. Instead of animating the containers, Container FX animates the containers below (aka the container-Set). Those containers can be circles and so on.

Like Text FX plug-ins use containers, a Container FX plug-in will use a set of containers under the plug-in and change their position like Text FX does. It is recommended to be familiar with Text FX before designing with Container FX to understand the idea. You can also use the Cloner plug-in to create the necessary containers under the plug-in.



In the above example the containers under the group container with the Sphere names is the container-set, and will be animated here.

Most of the Container FX plug-ins share a set of Common Properties.

The following plug-ins are in the Container FX folder:

- Common Container FX Properties
- CFX 2D Follow
- CFX Alpha
- CFX Arrange
- CFX Color
- CFX Explode
- CFX Jitter Alpha
- CFX Jitter Color
- CFX Jitter Position

- CFX Jitter Scale
- CFX Plus Plus
- CFX Rotate
- CFX Scale

14.3.1 Common Container FX Properties

The following properties are common to most of the Container FX plug-ins:

- **Progress %**: 0% progress is the beginning of the effect, 100% the end. Animate this value from 0% to 100% to see the effect or from 100% to 0% to animate the effect backwards.
- **Progress Type**
  - **Absolute**: 100% progress will animate all container's set, regardless of how many containers it has.
  - **Relative**: 100% progress animates ten containers. This is needed to adjust the timing of several containers with different sizes. The effect speed should be for example five containers per second, so the animation must be from 0% to 100% in two seconds. This will work for with ten containers or less. If you want to use more containers, animate the progress value over 100% (10% for each container).
- **Direction**: Sets the direction of the effect sequence, you can choose between the following options:
  - **Left**: Starts with the first container in the containers-set.
  - **Right**: Starts with the last container in the containers-set.
  - **Random**: Uses a random order.
  - **Static**: Processes all containers at the same time.
  - **Wave**: Starts with the first container, animates the effect from 0: 100% and then down again to 0%.
  - **Center**: Starts the effect from the center of the containers-set.
  - **2 Center**: Starts the effect at the same time from the beginning and the end of the containers-set. They meet at the center.
- **Interpolate**: Chooses between a soft or a linear interpolation of the transition from container to container.
- **Effect Range**: Defines how many containers are processed at the same time. If for example the Effect Range is set to 4, and you manually increase the progress value, you will see that when the fifth container starts to be processed, the first is finished, when the sixth starts, the second is finished, and so on.
- **Random Seed**: Specifies a seed for the random number generator when a random direction is chosen. Even though Viz Artist uses random numbers, the animation for a specific random seed will always look the same. This is typically useful if you combine two different container effects.
- **Container Order**: Sets the container order for the effect. Available options are As Is, Horizontal or Vertical. Horizontal and Vertical enables the containers -set Direction options.
- **Direction**: Sets the horizontal and vertical container-set direction (see Container Order). When the containers-set order is set to Horizontal, the container-set direction must be set to

Left to Right or Right to Left. When the container order is set to Vertical, the containers-set direction must be set to Up to Down or Down to Up.

14.3.2 CFX 2D Follow

The Container FX 2D Follow plug-in allows you to create an animation of geometry containers following a path defined by the 2D Ribbon geometry plug-in. The 2D Ribbon plug-in must be added as a parent container to the container holding the Container FX 2D Follow plug-in. The geometries that are to be animated needs to be added as sub-containers to the container holding the Container FX 2D Follow plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX 2D Follow Properties
- To Set Up the Container FX 2D Follow Plug-in

CFX 2D Follow Properties

- **Positioning**: *Absolute* positioning evenly distributes the followers across the entire length of the spline. *Relative* means that the followers will be added from left to right along the spline.
- **Path Position**: Moves the followers along the 2D Ribbon shape.
- **Position Wrap**: *Interpolate* means that text continues in the direction set by the last section of the spline; *Clamp* will stop the string on the spline end-point; *Repeat* will cause the string to move to the other spline end-point as soon as it moves over the end-point.
- **Kerning Scale**: Scales the kerning of the spline.
- **Align**: Rotates the followers by Z to align the X-Axis with the tangent of the spline at the followers' position.
- **Translation offset**: Moves the followers from the spline in the XY plane.
- **Rotation**: Rotates the followers using the spline as a rotation axis.

To Set Up the Container FX 2D Follow Plug-in

1. Add the 2D Ribbon geometry to the scene tree, then open the 2D Ribbon editor and enable **Show Control Point Values**.
2. Set alternating values of `30.0` and `-30.0` to the **Y axis** values, creating a wave shape.
3. Add a sub-container to the 2D Ribbon container, and add the **Container FX 2D Follow** plug-in to the newly created container.
4. Add the geometries that are to be animated along the 2D Ribbon's path as sub-containers to the Container FX 2D Follow container.
5. Open the **Container FX 2D Follow editor** and animate its **Path Position** value from **0.0** to **100.0**.

See Also

- 2D Ribbon

14.3.3 CFX Alpha

The Container FX Alpha plug-in is a container-set effect that creates a fade in effect for the containers in the set. The effect sequence can be set to go many different ways. A fade in and out effect can also be achieved.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFx Alpha Properties

CFx Alpha Properties

- **See**: Common Container FX Properties.
- **Alpha Begin %**: Sets the alpha level of the containers at 0% effect.
- **Alpha End %**: Sets the alpha level of the containers at 100% effect.

14.3.4 CFX Arrange

The CFX Arrange plug-in arranges containers in either a circular or a wave shape. The containers can be animated on the selected shape by animating the offset value.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Arrange Properties

CFX Arrange Properties

- **Offset %**: Moves the containers on the shape. 100% means one full rotation of the containers-set on the circle.
- **Scale %**: Sets the container-set on the shape. The parameter does not scale the containers, but the center of the bounding box of the containers.
- **Shape**: Changes the shape of the container layout. The options are Circle or Wave.

- **Diameter**: Sets the diameter of the circular shape.
- **Positioning**: Defines the position of the container-set on the circular shape. Relative means that the spacing of the container-set is maintained. Absolute means that the container-set is evenly distributed out on the circle.
- **Direction**: Sets the direction of the container-set on the circle to either Clockwise or Counterclockwise.
- **Rotation:** Rotates the container-set on the X-axis.
- **Align**: Aligns the container along the curve.

14.3.5 CFX Color

Adds a color effect to the containers-set.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Color Properties

CFX Color Properties

- **See**: Common Container FX Properties
- **Color Start**: Sets the initial color before applying the effect.
- **Color End**: Sets the color the effect should apply.

14.3.6 CFX Explode

The CFX Explode plug-in creates an explosion like function where the containers get thrown away from their initial position. The speed, direction and spread of the moving containers can be altered with parameters.

> ⚠ **Note:** Works only if the containers-set is set to texture.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Explode Properties

CFX Explode Properties

- **Duration**: Defines the duration of the progress for each of the containers. Simply increase the value if you want it last longer. You could achieve the same by making the gravity stronger and animating the progress slower, but it is easier to increase the duration instead if your explode effect is too short.
- **Opening Angle**: Sets the angle for the spread of the containers. 0 will send them straight up, 360 will spread them in a circular shape.
- **Angle Rotate X**: Rotates the opening angle around the X-axis.
- **Angle Rotate Y**: Rotates the opening angle around the Y-axis.
- **Force**: Sets the force that throws away the containers. A high force will make them go far away, conversely a low force will create only a small motion of the containers.
- **Force Spread %**: Sets a variation of the force among the containers.
- **Use Axis**: Allows you to select on which axis or combination of axes the containers are to spread along.
- **Gravity**: Sets a gravity force that influences the path of the containers to end up going downwards. The higher the value is set, the faster each container will divert from its initial path and start going downwards.
- **Use Rotation**: Rotates the container as they are being thrown away from their initial position when enabled.
- **Rotation Force**: Sets the degree of rotation as the containers are being thrown away.
- **Show Force**: Shows lines showing the containers path and speed in the Scene Editor when enabled.

See Also

- Common Container FX Properties

14.3.7 CFX Jitter Alpha

The CFX Jitter Alpha plug-in creates a jittering motion of the containers by randomly changing the alpha value of each container. The degree of change and the start sequence of the jittering can be altered. To use the plug-in, add it onto a container with a font. To create an animation, animate the progress. Other values can of course be animated as well.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Jitter Alpha Properties

CFX Jitter Alpha Properties

- **Effect What**: Defines where the effect should have an effect. Available options are background (BG), foreground (FG) or both.
- **Alpha Begin**: Sets the alpha level of the container at 0% effect.
- **Alpha End**: Sets the alpha level of the container at 100% effect.
- **Randomness**: Sets the intensity of the jittering alpha changes.

See Also

- Common Container FX Properties

14.3.8 CFX Jitter Color

The CFX Jitter Color animates a jittering effect on the color of the container.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Jitter Color Properties

CFX Jitter Color Properties

- **Anchor Horizontal**: Sets the anchor point for the containers on the horizontal plane.
- **Anchor vertical**: Sets the anchor point for the containers on the vertical plane.
- **Use Axis**: Defines on which axis or axes the containers scale to create the jittering effect.
- **Lock Axis**: Applies the same Min, Max and Randomness settings to all axes when enabled. If you disable randomness, these parameters will be visible for each of the axes and must be set individually.
- **Min**: Sets the minimum scaling for the containers.
- **Max**: Sets the maximum scaling for the containers.
- **Randomness**: Sets the intensity of the jittering movement.

See Also

- Common Container FX Properties

14.3.9 CFX Jitter Position

The CFX Jitter Position plug-in creates a jittering motion of the containers by randomly changing the position of each container. The degree of position change and the starting sequence of the

jittering can be altered. To create an animation, animate the progress. Other values can of course be animated as well.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Jitter Position Properties

CFX Jitter Position Properties

- **Use Axis**: Defines on which axis or axes the containers moves to create the jittering effect.
- **Lock Axis**: Gives the same values for Range and Randomness for all axes when enabled. If you disable it, Range and Randomness for each of the axes will be shown and you can set them individually.
- **Range**: Sets the range of the jittering movement.
- **Randomness**: Sets the intensity of the jittering movement.

See Also

- Common Container FX Properties

14.3.10 CFX Jitter Scale

The CFX Jitter Scale animates a jittering effect on the scale of the container.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Jitter Scale Properties

CFX Jitter Scale Properties

- **Anchor Horizontal**: Sets the anchor point for the containers on the horizontal plane.
- **Anchor vertical**: Sets the anchor point for the containers on the vertical plane.
- **Use Axis**: Defines on which axis or axes the containers scale to create the jittering effect.
- **Lock Axis**: Gives all axes get the same Min, Max and Randomness settings when enabled. If you disable randomness, these parameters will be visible for each of the axes and must be set individually.
- **Min**: Sets the minimum scaling for the containers.

- **Max**: Sets the maximum scaling for the containers.
- **Randomness**: Sets the intensity of the jittering movement.

See Also

- Common Container FX Properties

14.3.11 CFX Plus Plus

The Container FX Plus Plus plug-in allows you to set a number of effects on a container's set.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Plus Plus Properties

CFX Plus Plus Properties

- **Position Wrap**
  - **Extrapolate**: Continues the direction of the spline when set above 100% or below 0% of Path Position.
  - **Clamp**: Stops the string on the spline endpoints.
  - **Repeat**: Causes the string to move to the other spline endpoint as soon as it moves over the endpoint.
  - **ContainerFx**: Positions containers by the relative spline of each container using the ContainerFx parameters.
- **Scheme Type**: Defines how the container-set will look at 0% and 100% progress.
  - **In -> In**: at 0% first container will be at the beginning of the spline, at 100% last container will be at the end of the spline.
  - **In -> Out**: at 0% first container will be at the beginning of the spline, at 100% first container will be at the end of the spline.
  - **Out -> In**: at 0% last container will be at the beginning of the spline, at 100% first container will be at the end of the spline.
  - **Out -> Out**: at 0% last container will be at the beginning of the spline, at 100% last container will be at the end of the spline.
- **Progress (%)**: Animates the progress of the effect(s).

14.3.12 CFX Rotate

The CFX Rotate plug-in allows you to create an effect where the containers rotate on the X-, Y- or Z-axis.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Rotate Properties

CFX Rotate Properties

- **Anchor Horizontal**: Sets the anchor point for the containers on the horizontal plane.
- **Anchor Vertical**: Sets the anchor point for the containers on the vertical plane.
- **Use Axis**: Defines on which axis the containers rotate.
- **Begin**: Sets the initial rotated position of the containers.
- **End**: Sets the ending rotated position of the containers.

See Also

- Expert
- Common Container FX Properties

14.3.13 CFX Scale

The CFX Scale plug-in allows you to create a scaling animation of the containers.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> ContainerFx

This section contains information on the following topics:

- CFX Scale Properties

CFX Scale Properties

- **Anchor Horizontal**: Sets the anchor point for the containers on the horizontal plane.
- **Anchor Vertical**: Sets the anchor point for the containers on the vertical plane.
- **Use Axis**: Defines on which axis or axes the containers scale.
- **Lock Axis**: Scales the Begin- and End the same way for all the axes when enabled. If you disable the option, Begin and End must be set for all axes individually.
- **Begin**: Sets the initial size of the containers.
- **End**: Sets the ending size of the containers.

See Also

- Common Container FX Properties

14.4 Control

The following Container plug-ins are located in the Control folder:

- Common Control plug-in Properties
- Apply Shared Memory
- Control Action
- Control Action Table
- Control Audio
- Control Bars
- Control Chart
- Control Clip
- Control Clock
- Control Condition
- Control Container
- Control Data Action
- Control Datapool
- Control DP Object
- Control FeedView
- Control Geom
- Control Hide in Range
- Control Hide on Empty
- Control Image
- Control Key Frame
- Control List
- Control Map
- Control Material
- Control Multihop
- Control Num
- Control Object
- Control Omo
- Control Parameter
- Control Payload
- Control Pie
- Control Scaling
- Control Sign Container
- Control SoftClip
- Control Stoppoint
- Control Targa Alpha
- Control Targa Clip
- Control Text
- Control Video
- Control World
- Control Field Renamer
- Placeholder

14.4.1 Common Control plug-in Properties

Most control plug-ins have a set of common properties:

- **Field Identifier**: Used by the control client applications to identify the editable item in the scene. Only English characters (A-Z, a-z), numeric digits (0-1), dot (.) and dash (-) are allowed. When a dot is used in the identifier, e.g. "1.score", it means this is a sub-field. Multi-level sub-field is allowed. When making templates for Viz Trio, this should be a numeric value, starting at 1. The numbers are used to create the tab-order between the editable items. For Viz Pilot and Viz Ticker, this can also be a descriptive text string.
- **Description**: Describes the tab field or editable object. This is used as a description for the items when used in the control clients.
- **Show only in Expert Mode:** Hides the tab-field properties for the user if Expert Mode is enabled in Viz Trio. For more information, read about Viz Trio's macro commands, and the command *set_expert_mode_enabled*.

14.4.2 Apply Shared Memory

The Apply Shared Memory plug-in allows values to be transferred into Viz Engine using mechanisms provided by shared memory maps that can be applied to control plug-ins such as Control Text. Just as Control Object acts as a distribution hub for content provided by control applications, the Apply Shared Memory plug-in acts as a distribution hub for content provided by external systems feeding real-time data through the shared memory maps.

The control plug-ins that the Apply Shared Memory plug-in can apply content to resides in the same folder.

Just like the Control Object plug-in, the Apply Shared Memory plug-in only applies values to control plug-ins residing in the subtree of the scene tree where the Apply Shared Memory plug-in resides. Also like the Control Object plug-in, it matches the keys of incoming content with the field identifiers of individual control plug-ins to determine where to apply which incoming content value.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Apply Shared Memory Properties
- Visualize Composite Data Structures
- How Shared Memory Keys are Determined
- To Visualize Stock Data from Shared Memory
- To Test Stock Visualization

Apply Shared Memory Properties

- **Shared Memory Source:** Specifies which shared memory map to apply values from. The **Scene**, **Global** and **Distributed** choices select the shared memory maps that are referred to as *Scene.Map*, *System.Map* and *VizCommunication.Map* respectively in the section.
- **Memory Base Key:** Provides the key prefix path to combine with the field identifiers of control plug-ins to determine the key to look up in the chosen shared memory map when applying values to a particular field exposed by a control plug-in.
- **Expose Base Key:** Exposes the **Memory Base Key** property as a controllable field. Once this option is turned on the properties **Field Identifier** and **Description** as described in the Common Control plug-in Properties section can be specified.
  The **Expose Base Key** property is typically used to allow a control application to control the base key which determines where in the shared memory map to pull values from. However, since the exposed property is an exposed control field like any other, a parent Apply Shared Memory plug-in can be used to make the shared **Memory Base Key** property controllable by external systems through a shared memory map.

Choosing whether to allow the exposed field to be controlled by control applications or external systems feeding the shared memory map is done by placing either a Control Object or another Apply Shared Memory plug-in on a parent container, respectively.

Visualize Composite Data Structures

The **Memory Base Key** property combined with the field identifiers exposed by control plug-ins can be seen as a way to visualize sets of related key-value pairs stored in shared memory maps. While the **Memory Base Key** property specifies where in the shared memory map to locate the set of related key-value pairs, the exposed field identifiers specify which of the related key-value pairs from the selected set to visualize in a particular way in the scene subtree.

For example, the **Memory Base Key** property can specify a shared memory path representing a particular stock symbol, while the field identifiers of control plug-ins picks out particular values such as stock *name*, *value* and *change* from that location in the shared memory map.

Since the Apply Shared Memory plug-in only affects the scene subtree it is placed on, visualizing multiple composite data structure instances sharing the same structure and presentation can easily be achieved by copying subtrees and only change the Memory Base Key property.

For example, design a stock presentation design once, and copy it many times to create a scene showing several distinct stocks using the same visual style. Turning on the **Expose Base Key** property gives users of control applications control of which stocks to show in each presentation copy, while the data content for each stock is still supplied in real-time through the shared memory map.

How Shared Memory Keys are Determined

The shared memory keys used to look up values in a shared memory map is constructed by the combination of an absolute path specified in **Memory Base Key** property and an either relative or absolute path consisting of the field identifier of an exposed field. Any path starting with a slash (/) is considered an absolute path, while all other paths are considered relative.

If the field identifier is an absolute path, then that absolute path alone is the result of the resulting shared memory key used. For example, the field identifier “/weather/norway/bergen/temperature” will result in the shared memory key “/weather/norway/bergen/temperature” regardless of what the **Memory Base Key** property is.

If the field identifier is a relative path, the path is interpreted relative to the absolute base path provided in the **Memory Base Key** property, and the absolute path resulting from that interpretation is used as the shared memory key. For example, a **Memory Base Key** property of “/weather/sweden/stockholm” and a field identifier of “temperature” results in the shared memory key “/weather/sweden/stockholm/temperature”.

Like file system paths, a path segment consisting of a single dot (.) is considered referring to the current directory, while two dots (..) is considered referring to the parent directory. This notation is resolved to a regular absolute path before using the result as a shared memory key. For example, a **Memory Base Key** property of “/weather/./norway/trondheim” and a field identifier of “../bergen/windspeed” will resolve to the shared memory key “/weather/norway/bergen/windspeed”.

To Visualize Stock Data from Shared Memory

This procedure will visualize composite content fed through a shared memory map to the control plug-ins Control Text, Control Num and Control Sign Container.

1. Create a new container
2. Drag and drop a Apply Shared Memory plug-in into this container
3. Set the **Memory Base Key** property of the Apply Shared Memory plug-in to “/stocks/nyse/ibm”
4. Create two text geometries as children of this container
5. Drag and drop a Control Text plug-in into the first text geometry container
6. Set the **Field Identifier** property of the Control Text plug-in to “name”
7. Turn off **Use formatted text** property of the Control Text plug-in
8. Drag and drop a Control Num plug-in into the second text geometry container
9. Set the **Field Identifier** property of the of the Control Num plug-in to “value”
10. Create a third container as a child of the Apply Shared Memory plug-in container
11. Drag and drop a Control Sign Container plug-in into this container.
12. Set the Field Identifier property of the of the Control Sign Container plug-in to “value”
13. Create three text geometries as children of the Control Sign Container plug-in container
14. Drag and drop a Control Num plug-in into each of these three text geometry containers
15. For each of these three Control Num plug-ins set the **Field Identifier** property to “change”
16. Arrange the layout of the text geometries, and apply materials for styling

To Test Stock Visualization

To test that stock visualization commands can be sent to populate the global shared memory map, see the descriptions in the section.

While these commands are suitable for testing the designs, a real external system that feeds the Viz Engine with data would use a more efficient data transfer protocols to populate the shared memory map. See the External Data Input section for information about sending data efficiently using UDP.

When changing a value in the shared memory map that is visualized by a control plug-in the scene updates immediately to reflect the new value. The shared memory map can be populated with stock values for many stock symbols, including stocks that are not visualized at the moment. As soon as the **Memory Base Key** property of the Apply Shared Memory plug-in is changed to visualize the values for a different stock symbol, and the scene updates immediately to reflect the value previously populated into the shared memory map for that stock symbol.

14.4.3 Control Action

The Control Action plug-in executes an action, for example Viz Engine commands or Control Object commands, when receiving input on the “input” field.

The action can contain more than one command to be called, and the commands must be separated by semicolons. When “Notify Only When Value Change” is checked the actions will only be triggered if the input value differs from the one already stored on the “input” field from previous invocations.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Action Properties

Control Action Properties

- **Input Value:** Specifies the value input will be compared against.
- **Notify Only When Value Change:** Triggers a command when a value is changed.
- **Action Type:**
  - **Viz Command**: Triggers an internal Viz Engine command. For example: *“THIS_SCENE*STAGE*DIRECTOR*Audio START”*.
  - **ControlObject Command:** Triggers a ControlObject command. For example: *“ON 1 SET abc”*.
- **Action:** Actions to execute if values match.

See Also

- Common Control plug-in Properties
- Control Object

14.4.4 Control Action Table

The Control Action Table plug-in allows defining a table of actions similar to Control Action. The actions are associated with a given value. When Control Action Table receives input, it compares the received data to each of the values. If the data matches one of the values the plug-in starts the corresponding action.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Action Table Properties

Control Action Table Properties

- **Input Value**: Specifies the value input will be compared against.
- **Notify Only When Value Change:** Triggers a command when a value is changed.
- **Value 1/2:** The value that the given action should be associated with.
- **Action 1/2:** Actions to execute if value match.
- **Default Action:** Will be executed if the input does not match any of the specified values.

See Also

- Common Control plug-in Properties
- Control Object

14.4.5 Control Audio

The Control Audio plug-in enables the operator of a control application to choose a sound clip.

To expose the different properties, and change the audio file in one of the control applications, simply add the Control Audio plug-in to the same container as the Audio plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Audio Properties

Control Audio Properties

- **Director:** Sets the name of the stage director the audio channel is located below.
- **Audio Key Frame:** Sets the ID/name of the audio clip Key Frame.

  > ⚠ **Note:** It is mandatory to add Director and Audio Keyframe name.

- **Important:** Make sure the audio clip name does not contain any white spaces.

See Also

- Common Control plug-in Properties
- Control Object
- Audio

14.4.6 Control Bars

The Control Bars plug-in allows binding of tab field values to the Bar Value fields in the Bar values presenter plug-in. Each bar value property in the Bar Values editor is presented as a separate tab field in Viz Trio.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Bars Properties

Control Bars Properties

- **First Bar Field ID**: Sets the tab-field ID for the first Bar value.
- **Last Bar Field ID**: Sets the tab-field ID for the Last Bar value.
- **Expose Bars Max/Total Value**: Exposes the Bars Max/Total value as a separate tab field when enabled.

See Also

- Common Control plug-in Properties
- Control Object

14.4.7 Control Chart

The Control Chart plug-in binds control of chart data to a Control Object. It currently supports the graph geometry plug-in and the bar values, pie values and Visual Data Tools plug-ins. It serves the purpose to interface all kind of chart data. It covers graph charts, bar charts, pie charts, area charts and generally all kind of plug-in data, which has a table-like representation.

Control chart offers the same kind of interface like control list. That means it delivers a schema specification as type encoding. The chart data is communicated as an XML table value fully compatible to control list values. The supported control chart commands mimic the control list commands.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Chart Properties

Control Chart Properties

- **Custom Number of Exposed Rows:** Enables the Number of exposed Rows setting.
  - **Number of exposed rows:** Sets the default number of exposed rows for the scene when the number of rows can be changed by the operator. If the number of rows cannot be changed the number of exposed rows is fixed.
- **Mutable Number of Rows**: Allows the operator can add or delete rows if set to on.
  - **Custom minimum and maximum Number of Rows:** Enables the Minimum and Maximum Number of Rows settings. To the operator, the number of fields can only be added or deleted when inside the range (for example greater than > 2 and less than < 10).

IMPORTANT! If Control Chart is used in combination with Visual Data Tools plug-ins, make sure the Visual Data Tools plug-in Shared Memory is set to `Inactive`.

See Also

- Common Control plug-in Properties
- Control Object
- Visual Data Tools

14.4.8 Control Clip

The Control Clip binds control of an AVI clip played by MoViz or SoftClip to a Control Object. With the Control Clip plug-in the clips may be changed within the control application.

The MoViz plug-in enables the user to play media files or media streams inside Viz Engine. MoViz uses the Microsoft DirectShow Filtergraph framework to render the media. Therefore it is possible to play for example MPEG and QuickTime files or streams from a server. For the latter to work, the system must be appropriately equipped.

SoftClip is a Viz Artist/Engine plug-in that can show AVI clips, either projected on a texture, or rendered directly.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Clip Properties

Control Clip Properties

This plug-in does not have any properties or parameters except those that are common to all plug-ins.

See Also

- Common Control plug-in Properties
- Control Object

14.4.9 Control Clock

The Control Clock plug-in allows the user to set time for a clock as well as controlling several clock functions. The clock object is a text container in the scene with the clock function enabled.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Clock Properties

Control Clock Properties

- **Expose Limit:** Enables the user set a limit for the clock.
- **Expose Direction**: Enables the user to choose the direction for the clock.
- **Action:** Sets the clock action that will be the default value in the control client's clock editor.
- **Input Value**: Shows the current value for the plug-in. It is not necessary to set any value here. It is normally only used for debugging purposes.

See Also

- Common Control plug-in Properties
- Control Object

14.4.10 Control Condition

With the Control Condition plug-in action can be triggered based on the input given in a tab-field with the same field identifier.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Condition Properties
- Example Actions

Control Condition Properties

- **Input Value**: Shows the current input value to the plug-in.
- **Format**: Sets the format for the input value that it should conditionally trigger on.
    - **Condition**: Sets the type of condition if Format is set to number.
- **Arguments**: Sets the arguments for the conditional operation.
- **Action Type / Action**: Sets the action that is to be performed when the condition is met. It can be a control object command like “ON 2 SET mytext” or a Viz Engine command prefixed by a zero and a space.
- **Use Else Commands**: Triggers an *else* action when the input data falls within the conditions when enabled.
    - **Else Action**: Sets the else action that will be triggered when the input data does not fall within the conditions.

Example Actions

Actions can be Control Object commands in the form like this:

```
“ON 2 SET Mike Johnson”
```

or commands like:

```
0 RENDERER*STAGE START
```

Viz Engine commands must be prefixed with a zero followed by a space. This is to enable the plug-in to recognize that the command should be interpreted as a Viz Engine command.

See Also

- Common Control plug-in Properties
- Control Object

14.4.11 Control Container

The Control Container plug-in exposes a range of different transformation properties for a container.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Container Properties

Control Container Properties

The following properties can be exposed:

- Visibility
- Object
- X/Y/Z position
- X/Y/Z rotation
- X/Y/Z scaling
  For position, rotation, and scaling a stop point Key Frame can be specified. This enables the user to control a Key Frame in an animation, for instance the end Key Frame in a position animation.

See Also

- Common Control plug-in Properties
- Control Object

14.4.12 Control Data Action

Binds control of a control parameter to a Data Action.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Data Action Properties

Control Data Action Properties

- **Input value:** Shows the current input value.
- **Prefix:** Adds a prefix, if required
- **Postfix:** Adds a postfix, if required

See Also

- Common Control plug-in Properties

- Control Object

14.4.13 Control Datapool

Control Datapool is used with the *DataPool* plug-in repository, available separately. The plug-in is used to bind a tab field value to a DataPool plug-in variable. Any DataPool plug-in used in a scene must have its variable listed in the **DataPool** scene plug-in. Control Datapool links a tab-field to these variables.

For more information about the DataPool plug-ins, refer to the DataPool documentation.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Datapool Properties

Control Datapool Properties

- **DataPool Variable**: The DataPool variable can be controlled by the user.
- **Input Value**: Shows the current input value.
- **Control Type**: Specifies if the tab-field value should be controlled through a DataPool variable or a field name.
- **Variable Type**: Allows the user to define the kind of data the referenced variable is linked to. The type selected here will decide what kind of editor will be opened in a Viz Trio client.

See Also

- Common Control plug-in Properties
- Control Object

14.4.14 Control DP Object

Allows copying or linking of Data Pool objects.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control DP Object Properties

Control DP Object Properties

- **Input value is:** Select from Object to COPY, Object to LINK or Direct VALUES
- **Input value**: Shows the current input value.

See Also

- Common Control plug-in Properties
- Control Object

14.4.15 Control FeedView

Set the Feed View locator to a given value.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control FeedView Properties
- Example Input Values

Control FeedView Properties

- **Locator**: Sets the FeedView locator to a given value.
- **Prefix**: If a prefix is set, it will be added to the front of the locator key.

Example Input Values

```
/nyse/nasdaq/cisco /nyse/nasdaq/intel
```

If a prefix is set it will be added to the front of the locator key, a prefix of:

```
/nyse/nasdaq
```

With an input of:

```
IBM
```

Will resolve to:

```
/nyse/nasdaq/IBM
```

The locator field is the value as received from ControlObject. It will be trimmed of CR/LF before concatenated into a FeedView key.

See Also

- Common Control plug-in Properties
- Control Object

14.4.16 Control Geom

The Control Geom plug-in exposes the control of geometry objects to the user.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Geom Properties

Control Geom Properties

- **Geom location Prefix**: Sets a path to a folder. The geometry location exposed to the control client can be a simple string, instead of a full path. This is relevant if connected to a newsroom system. For example if a user wants to edit a text and geometry on the same tab-field by just entering a text string. If for instance the user have a folder of weather symbols and the control client receives “sunny” from the external system it will load the geometry object with the path: “geom location prefix + sunny”, which then typically would be an image of a shining sun.
- **Input Value**: Shows the current input value. Enter a value to test the relation of max/min. input and scale values.
- **Deny Browsing Upwards**: Restricts the user from browsing for objects in folders other than the selected folder and its sub-folders when enabled.

See Also

- Common Control plug-in Properties
- Control Object

14.4.17 Control Hide in Range

The Control Hide in Range plug-in hides or shows a container if the input value given by the user is

within or outside a specified value range. The input value must be from a font container that holds a numeric value, typically with a Control Num plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Hide in Range Properties

Control Hide in Range Properties

- **Input Value**: Shows the current input value.
- **Lower Range Bound**: Sets the upper bound for the range.
- **Upper Range Bound**: Sets the lower bound for the range.
- **Visibility in Range**: Sets if the container should be visible or hidden when the input value is within the range.

See Also

- Common Control plug-in Properties
- Control Object

14.4.18 Control Hide on Empty

The Control Hide on Empty plug-in performs a very simple function: It hides the container it sits on if the input value is blank (text) or zero (numbers). Use this plug-in to hide the graphics items if the user does not enter any value in the connected tab-field. Together with the Autofollow plug-in, the plug-in can be used to make intelligent scenes that hide objects that receive no value from the Viz Trio user and also rearrange those items that have a value or text and are visible. For example when using a lower third scene with a bug and a breaking news bar, the bug and the breaking news bar should not be shown all the time. So instead of having to make several explicit variants simply make one scene and use the plug-in to hide the bug and the breaking news only if the page in Viz Trio have no value in those tab-fields.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Hide on Empty Properties

Control Hide on Empty Properties

- **Input Value**: Shows the current input value
- **Treat Input as Numerical**: Hides the containers with an input value of 0 or 0.0 when enabled .

See Also

- Common Control plug-in Properties
- Control Object

14.4.19 Control Image

The Control Image plug-in creates an image control in the control clients. The plug-in must be placed on the container that holds the image/texture. If an image/texture is invalid or Viz Engine for some reason cannot load it, then the current image/texture remains visible.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Image Properties

Control Image Properties

- **Expose Image Position**: Allows the user to change the position of the image/texture.
- **Expose Image Scaling**: Allows the user to change the scaling of the image/texture.
- **Deny Browsing Upwards**- Restricts the user from browsing for images/textures in folders other than the selected folder and its sub-folders when enabled.
- **Image location Prefix**: If the full path to a folder is entered, the image/texture location exposed to the control clients can be a simple string, instead of a full path. This is relevant if connected to a newsroom system. For example if a user wants edit a text and an image/texture on the same tab-field by just entering a text string. This is relevant if a user has a folder containing flag images/textures, and the control client receives “US” from the external system it will load the image/texture with the path: “image location prefix + US”, which then typically would be the “Stars and stripes” (flag of the U.S.A).
- **Input Value**: Shows the current value for the plug-in. It is not necessary to set any value here. It is used for debugging purposes.

See Also

- Common Control plug-in Properties
- Control Object

14.4.20 Control Key Frame

The Control Key Frame plug-in can be used to give the user control over the value of a “single value Key Frame” unlike position values for instance where all three axes must be specified. Create an animation, for instance an alpha animation, and give the Key Frame to control a name. Put the Control Key Frame plug-in on the object’s container and set the parameters.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Key Frame Properties
- Input Value Example

Control Key Frame Properties

- **Expose Time**: Enables the user to adjust the Key Frame along the time-line.
- **Control Parent**: When enabled, this option allows for two Control Key Frame plug-ins to be within one container.

  > ⚠ **Note:** In this case the control Key Frame plug-ins must be added as Sub-Containers of the container.

- **Minimum Input**: Sets the minimum allowed input value.
- **Maximum Input**: Sets the maximum allowed input value.
- **Minimum Value**: Sets the minimum value, which will be set relative to the minimum input value.
- **Maximum Value**: Sets the maximum value, which will be set relative to the maximum input value.
- **Input Value**: Shows the current input value. Enter a value to test the relation of max/min. input and scale values.
- **Key Frame Time**: Sets the time value of the Key Frame.
- **Key Frame**: Sets the name of the Key Frame that is to receive a value by the control client user.

Input Value Example

Example for the input values: If the Key Frame is an alpha Key Frame which in Viz Artist can be float values from 0 to 100, and the following values are set:

- Minimum input = 1
- Maximum input = 10
- Minimum value = 10

- Maximum value = 90.
  This means that if the user gives the following input:
- 1, which is the lowest input value allowed, the alpha value in Viz Artist will be set to 10.
- 4, which is somewhere in between, the relative alpha value will be 36,666... .
- 10, which is the highest value allowed, the alpha value is set to 90.

See Also

- Common Control plug-in Properties
- Control Object

14.4.21 Control List

The Control List plug-in allows creation of table controls. By following some design conventions when designing the table in Viz Artist, Viz Trio will show a table editor to the user. This editor supplies a more effective way for filling tables than having one tab field for each table cell.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control List Properties
- Tables and Preview Points
- To Design a Table Scene

Control List Properties

- **Number of exposed rows**: Sets the default number of exposed rows for the scene when the number of rows can be changed by the operator. If the number of rows cannot be changed the number of exposed rows is fixed.
- **Mutable Number of Rows**: Allows the operator to add or delete rows when set to **On**.
- **Minimum and Maximum Number of Rows**: Sets the minimum and maximum number of rows. To the operator, the number of fields can only be added or deleted when inside the range (for example greater than > 2 and less than < 10).

Tables and Preview Points

A table scene might be hard to read if you need to show a large table with many rows (e.g. 20) and/or columns. In such cases it is not uncommon to only show a subset of the information (e.g. five rows), and rather animate the scene to show the next five rows and so on.

However, in such a situation, while editing table data in Viz Trio, you cannot use a basic jump to Key Frame event for preview, as this will only show the Key Frame that relates to the table's Field identifier (e.g. 3), and not the rows or columns of the table.

To enable operators to preview the data in the scene, it is therefore possible to add a preview point that extends to the next starting point (i.e. the next five rows of the table), using the table's (Control List) Field identifier in addition to the row number.

So, if the identifier of Control List is 03, you have to define tags with names 03.0, 03.5, 03.10 and 03.15 (zero based row index) for Viz Trio to detect the tags and jump to them on focusing the according rows in Viz Trio's Control List table editor.

- **Basic jump to Key Frame**: When adding Key Frame events (stop points, tags,...) and giving them the name of a control plug-in identifier, Viz Trio jumps the time-line to that Key Frame event when selecting the tab field with that identifier.
- **Extended jump to Key Frame**: If the Key Frame event name is e.g. `3_005` and there is a control list with identifier `3`, then Viz Trio jumps the time-line to that Key Frame event. In this case when the sixth row of the table is focused.

To Design a Table Scene

To design a table scene that will be correctly interpreted by the Control List plug-in there are some design conventions that needs to be followed:

1. Add a group as the root container for the table. Give it a descriptive name such as "table"
2. Add the Control List plug-in to the table group container
3. Add a new group as a Sub-Container to the table container. This container will be the design for the rows in the table. Give it a name such as "row_1".
4. Add a Control Object plug-in to the newly added row container.
5. As Sub-Containers of the row container, add text containers with backgrounds to form the design for a single table row with the required number of columns.
    - Each of the text containers must get a Control Text plug-in and the field identifiers for each must be set from 1-n depending on how many columns the table should have.
    - The Description property in the ControlText plug-ins will be picked up by the Control List plug-in and used as column headers.
6. When the single row looks as it should, duplicate it to make the needed total number of table rows.
    - The *container* plug-in Coco can be used to automate this task.
    - Add the Coco plug-in to the row container, and set the number of copies, for example 9 (1+9=10), and click *Execute*.
7. Arrange the rows
    - The *arrange* plug-in Grid Arrange can be used to automate this task.
    - Add the Grid Arrange plug-in to the table container, and set the number of rows to the same as above and press ENTER.
    - Set the row offset and press ENTER.
      The table item of the scene is ready and when imported into Viz Trio and read, the table tab-field will be shown with a special table editor. When imported into Viz Template Wizard, a template with a custom table editor will be generated.

See Also

- Common Control plug-in Properties
- Control Object

14.4.22 Control Map

The Map plug-in binds the control of the Viz World Client plug-in XML property to a ControlObject which allows an operator to use the Viz World Client Editor to select maps.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Map Properties

Control Map Properties

This plug-in does not have any properties or parameters except those that are common to all plug-ins.

See Also

- Common Control plug-in Properties
- Control Object

14.4.23 Control Material

The Control Material plug-in exposes the material control to the Viz Trio user.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Material Properties

Control Material Properties

- **Material**: Sets the path to the material that should be used when the page is imported into the control client.
- **Deny Browsing Upwards**: Restricts the user from browsing for materials in folders other than the selected folder and its sub-folders when enabled.

See Also

- Common Control plug-in Properties

- Control Object

14.4.24 Control Multihop

When the scene containing the Control Multihop plug-in is opened from Viz Trio or Viz Pilot, the operator can use the Multihop Editor (within the World Map Editor) to edit the hops.

The Control Multihop plug-in should be placed next to the HopsManager plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

Control Multihop Properties

This plug-in does not have any properties or parameters except those that are common to all plug-ins.

14.4.25 Control Num

The Control Number plug-in (also known as Control Num) is used to be able to decide how a number input is to be formatted. It can be a value given by the control client user or by any external source. It should be used instead of Control Text when numbers are to be the input value. To make it work, add the plug-in to the container that holds the text object.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Num Properties

Control Num Properties

- **Minimum Input**: Sets the minimum allowed input value.
- **Maximum Input**: Sets the maximum allowed input value.
- **Input Value**: Shows the current input value. Enter a value to test the number formatting.
- **Show Verbatim**: Disables all formatting when enabled.
- **Thousand Separator**: Sets the required thousand separator for the output.
- **Plus Prefix**: Applies the prefix entered here in front of positive values.
- **Minus Prefix**: Applies the prefix entered here in front of negative values.
- **Suffix**: Applies the string entered here after all values.
- **Decimal Point**: Defines the decimal point for the output.

- **Decimal Places**: Sets the number of decimals that will be shown by the output.
- **Leading Zero**: Places a zero in front of decimal values between 1 and -1 when enabled.
- **Zero Replacement**: Sets the string to show instead of zero values.

See Also

- Common Control plug-in Properties
- Control Object

14.4.26 Control Object

For each Scene that is to become a Viz Trio page template, a Viz Pilot template or a Viz Ticker template, it must have one instance of the Control Object plug-in.

Control object provides control channels for subtrees of Viz Scenes. The control channel information is stored in the subtree itself, so the Scene subtree can be merged and stored as objects in the object pool.

When a Control plug-in is added to a Scene, a control object plug-in is automatically added to the root container of the scene subtree to be controlled. The plug-in stores and gathers field definitions which describe how values are processed and applied to various properties of the subtree.

The easiest way define fields is to place one or more Control plug-ins on containers within the subtree. These Control plug-ins will find the Control Object and register properties related to the container they are placed on.

⚠ **Note:** For Viz Ticker it is possible to have more than one Control Object plug-in. It should be on the root container for the tab objects in the scene.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Object Without Transition Logic
- Control Object With Transition Logic Enabled
- Control Object Properties
  - Transition Logic Off
  - Transition Logic On
- Control Object Advanced Properties

Control Object Without Transition Logic

In the simple scene based mode, the plug-in should be put on the root container for the page tab-field containers. A description, which will be used as template description in the control clients, should also be entered. To load the graphics in another rendering layer than the Middle (default),

choose Front or Back. By using different layers up to three items can be on screen simultaneously without having to design a Transition Logic setup.

> ⚠ **Note:** Objects in the Back layer will be rendered behind objects in the Middle and Front layer and so on, so the scenes must then be designed with that in mind.

Control Object With Transition Logic Enabled

With Transition Logic enabled, logical layer and state can be defined for the scene. These will be used by the control clients to find out which transitions and animations to trigger when a page is put on-air or off-air. These transitions and layers must all be defined in the background scene. A background scene name can also be set if a background scene other than the default (named default) is used.

Control Object Properties

Transition Logic Off

- **Description:** Provides the description of the object the scene subtree represents. This description is typically presented to users in GUI applications that allow the user to fill in data that should be presented by the controlled object.
- **Transition Logic:** Specifies whether the controlled object uses transition logic. Transition logic loads objects onto layers in a background scene, and runs animations to change the background scene layer into a required state.
- **Viz Layer:** Determines the Viz layer to load the scene which contains the scene tree which contains this Control Object. This parameter is only used when transition logic is set to Off
- **Use All Directors:** Controls whether to play every director or only the default director in a non-transition logic scene, and takes effect when the Vizrt Media Sequencer triggers the `TAKE` command. When loading the scene into a control application such as Viz Trio, all directors in the stage will start automatically when the scene is taken in. By changing this to **Off**, only directors with `Auto Start` enabled in the **Director Editor** will start when the scene is taken in. This parameter is only used when **Transition Logic** is set to **Off**.

  > ⚠ **Example:** The scene design incorporates an animated background which starts automatically when the scene is loaded. To avoid the background animation restarting when the scene is played out, set **Use All Directors** to **Off**.

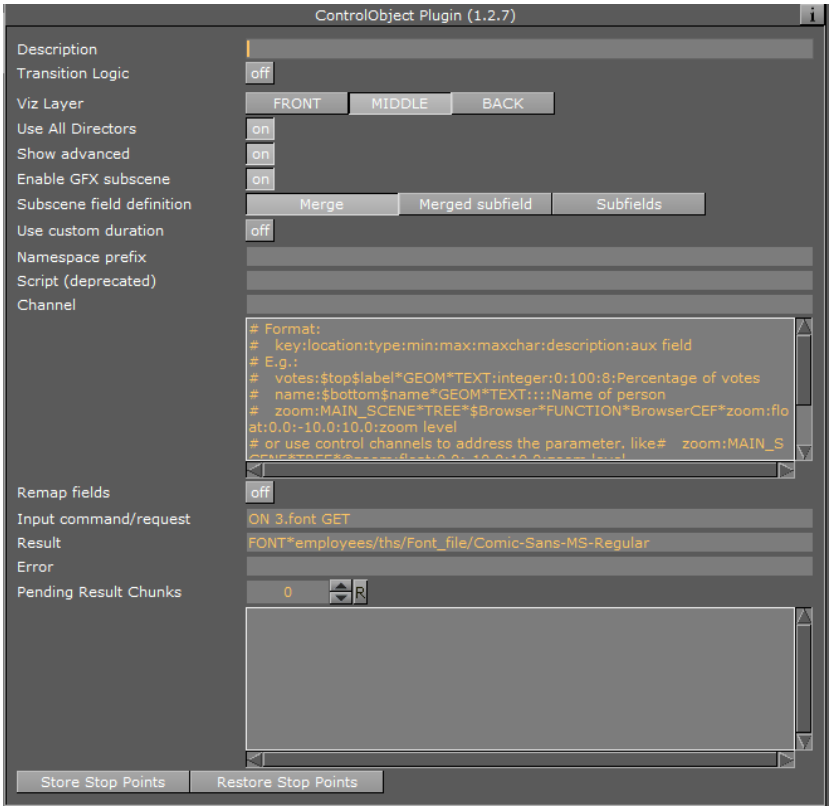
- **Show advanced:** See Control Object Advanced Properties.

Transition Logic On

- **Layer identifier:** Selects the layer the controlled object should be loaded onto in the background scene. This parameter will only be used when transition logic is set to On. For the transition logic to work, the identifier given here must be identical to the names of the following components of the background scene:
  - **A container with a Toggle plug-in:** The transition logic will use the toggle plug-in on this layer-container to load the object
  - **A director:** The transition logic will animate this layer-director into the state requested by the object
- **State identifier:** The state which the layer-director of the background scene should be animated to, when the object is loaded into its target layer. The identifier given here must be identical to one or more stop-points defined on the time-line of the layer-director of the background scene. This parameter will only be used when transition logic is set to On.
- **Background scene:** The scene which has the layer-container and layer-director needed by the transition logic to load and animate in the object. This parameter will only be used when transition logic is set to On.
- **Store Stop Points**: This mechanism can be used to work around the default behavior of Viz Artist which discards stop-points when sub-trees are merged into objects. It stores information about the stop-points of directors used by the sub- tree in a way that will survive when merging the sub-tree into an object stored in the object pool.
- **Restore Stop Points**: Recreates the stop-points used by the director of an object that has been loaded from the object pool. This uses the information previously stored by Store Stop Points. When used in conjunction with the Toggle plug-in, it will operate on the object's animation director.

Control Object Advanced Properties

When the Show Advanced button is set to On, a channel property is shown as well as a text box for manual field definitions. Most of these parameters are used for system integration purposes and are rarely relevant in a normal scene design for Viz Trio, Viz Pilot and Viz Ticker.



- **Enable GFX subscene:** Includes the control field definition from the plugins inside the GFX subscenes as part of the main scene's control field definition when enabled for ControlObject of the main scene.
  - **Subscene field definition:** Chooses how the control fields of the subscene are included in the main scene's control field definition.

Option	Result
Merge	All subscene control fields are merged with main scene control field at the same level. For example, the field "1.name" in a subscene will be shown as is in the control application.
Merged subfield	All subscene control fields are merged together and put under a specified field identified with the value in **Subscene field id/prefix** option. For example, the field "1.name" of a subscene assigned to any GFX channel will be shown as "gfx.1.name" in the control application.
Subfields	All subscene control fields of each scene are separately put under subfield with identifier prefixed with the value in **Subscene field id/prefix** option. For example, the field "1.name" of a subscene assigned to GFX channel 5 will be shown as "gfx5.1.name" in the control application.

  - **Subscene field id/prefix:** Names the subfield identifier for the subscene's merged subfields, or the prefix for the subfield of each subscene.
- **Use custom duration**: Sets the duration parameter:

    - **Duration:** Determines the amount of time the graphics should be shown when triggered as an interstitial item, such as sponsor items in a ticker.
- **Namespace prefix:** Allows the fields defined by control plug-ins below, to propagate to control objects further up the Scene Tree. When definitions are propagated upwards, the field identifier will prefixed with the name prefix
- **Script (deprecated):** Identifies the name of a script that contains functions that will be executed when various operations are completed in Viz Trio.
- **Channel**: Sets the name of the default playback channel for the template.
- **Manual field definitions**: In the text box, custom fields can be defined by using a fixed syntax: Each entry is a line of text consisting of colon (:) separated fields. The fields for each entry are:

Field description	Example
Field identifier/ key	votes
Property location	toplabel*GEOM*TEXT
Type	There are many type identifiers and are used by the control applications to determine which content editor control to show for a given field.e.g.:integerboolfloatinteger textrichtextduplettriplet
Minimum value	0
Maximum value	100
Maximum number of characters	8
Field description	Percentage of votes
Auxiliary field	Conveys various type specific type constraints and auxiliary information needed by the control application to provide a convenient editing experience. E.g. limiting text to a single text line, allowing only upper case characters, or limiting images to only images found in a certain image pool folder.

- **Remap fields**: For use in scenes with nested control object plug-ins. This feature is mainly for advanced integration use. When ON, Control Object will give the control clients the list of entries in map, rather than the list of entries registered by control plug-ins below the Control Object in the scene tree, and manually entered entries from entries. When actions are performed on fields, the Control Object will use map to rename the field identifier before dispatching the action.
- **Input command/request:** Field which receives commands and request actions. When this parameter is changed, the command described by the new value of the parameter is executed. Return values for requests are stored in the **Result** parameter.
- **Result:** Stores the result of requests given through the **Input command/request** parameter. Applications which complete the requests typically read the result text from parameter immediately after setting the request text in the **Input command/request** parameter.
- **Error:** In the event of the **Input command/request** returning an error, this field will be populated by the error message. This will be read by the control application, which then presents the user with an appropriate error message.

  > ⚠ **Note:** The API requires the Control Application to request the contents from both the **Result** and **Error** fields when sending an **Input command/request**. However, the plug-in will only populate one of the fields, depending on the returned result. This means that one of the fields will always be empty.

- **Pending Result Chunks:** If the contents of the **Result** or **Error** fields exceeds the size allowed, the information is split into “chunks” to allow more information to be passed through several transactions. This field is automatically updated by the plug-in itself.

See Also

- Common Control plug-in Properties
- Director Editor
- Toggle
- Transition Logic

14.4.27 Control Omo

The Control Object Moving (Omo) plug-in gives the possibility to add a group of containers and reveal one at the time. This is done by adding an Omo plug-in on the root container and in its editor specify which of the Sub-Containers to show. The Omo value can be made accessible for the control client by adding the Control Omo plug-in on the same container as the Omo plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Omo Properties

Control Omo Properties

- **Minimum Value**: Sets the minimum input value.
- **Maximum Value**: Sets the maximum input value.

See Also

- Common Control plug-in Properties
- Control Object
- Omo

14.4.28 Control Parameter

The Control Parameter plug-in works with function plug-ins, and is a very flexible plug-in that can be used to control properties that are not exposed in a specific plug-in, and requires some knowledge in the Viz Artist command language. The parameter to control must be identified by its command path relative to the container the plug-in is located at. To find the correct command path, open the Show Commands window (button in the lower left corner in Viz Artist) and make changes to the parameter to control. The command path will then be shown among the commands. For instance if the value to be changed is the alpha value in an alpha editor the command window will show the following:

```
receive <-1 SCENE*noname*TREE*#805*ALPHA*ALPHA SET 99.0>
```

#805 is the internal name for the container and the part needed to expose the alpha parameter (ALPHA * ALPHA). The SET part is added by the plug-in. To expose the alpha parameter for a container write ALPHA * ALPHA in the Parameter section in the plug-in and then set the correct data type which in this case is “Float”.

Another example would be to use Control Parameter to expose and control additional values like controlling both Pie Max and Max Value. These parameters are dependent on each other, but both values are not exposed unless the Control Parameter is used. Adjusting the Pie Max value gives the following:

```
receive <5679 SCENE*noname*TREE*#378*FUNCTION*PieValues*DATA GET>
```

In this case the FUNCTION*PieValues*DATA is the parameter that will expose the object property Pie Max such that it can be controlled through a template.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Parameter Properties

Control Parameter Properties

- **Parameter**: Sets the parameter path to the property to enable control for.
- **Input Value**: Shows the current input value.
- **Data Type**: Sets correct data type depending on the type of parameter to enable control for.
- **Normalize**: Enables normalization options, to normalize the input range for the Viz Trio user.
- **Minimum Input**: Sets the minimum allowed input value.
- **Maximum Input**: Sets the maximum allowed input value.
- **Minimum Value**: Sets the minimum value, which will be set relative to the minimum input value.
- **Maximum Value**: Sets the maximum value, which will be set relative to the maximum input value.
- **Prefix**: Adds a prefix to all values that are sent to the renderer. This can be used to show specific text or it can be used to add a path to an object pool for instance.
- **Control Parent**: Controls the parent container when enabled.

See Also

- Common Control plug-in Properties
- Control Object

14.4.29 Control Payload

Control Payload distributes a whole payload to children Control plug-ins.

> ⚠ **Example:** Social TV (Feed Streamer) uses this plug-in to send payload fields from a Data Hub to Viz Engine.

If the payload value is empty, the Container is hidden.

This section contains the following topics:

- Control Payload Properties

Control Payload Properties

- **Text Entry Box**: Data from external application (for example Viz Trio, Social TV (Feed Streamer)) to send the payload. Example:

```
<payload xmlns="http://www.vizrt.com/types">
```

```
        <field name="topic">
                <value>It\qs Valentine\qs Day, how you doin\q? </value>
        </field>
        <field name="option">
                <value>Hate it! Worst holiday ever|Love it! Wish it were every day|Thanks for the reminder,
need to buy roses</value>
        </field>
        <field name="count">
                <value>238|189|46</value>
        </field>
</payload>
```

See Also

Common Control plug-in Properties

14.4.30 Control Pie

The Control Pie plug-in is able to bind a tab-field to values from PieSlice plug-ins.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Pie Properties
- To Build a Pie Chart

Control Pie Properties

- **First slice ID**: Sets the tab-field number for the first slice (the first container below the group that holds the ControlPie plug-in).
- **Last Slice ID**: Sets the tab-field number for the last slice (the difference between Last and First should correspond to the total number of slices in the pie chart).
- **Expose Pie Max/Total Value**: Allows a Viz Trio user to set Max value for a pie slice and Max/Total value for the whole pie chart when enabled.

To Build a Pie Chart

1. Add a group container and name it, for instance “pie”. Add the presenter plug-in Pie Values and the control plug-in Control Pie to the group.
2. Add a Cylinder object as a Sub-Container to the group. Add material and the presenter plug-in Pie Slice to the new container. Name the container “slice” or similar.
    - The Cylinder object is not visible.

3. Open the Pie Values editor and increase the Value0 property a bit. The cylinder becomes visible and its size changes.
4. Open the Cylinder editor and set Centre to “Top”.
5. Open the Transformation Editor for the pie group, and rotate the pie group around 90 degrees on the X-axis to make the cylinder top face the camera.
6. Add a font as Sub-Container to the slice container, and add a material to make it visible against the background.
7. Open the Text Editor and set the horizontal justification to Centre. Open the Transformation Editor and rotate the font around the X-axis to -90 degrees and set Y position to 1.0.
8. Scale the font down a bit so it fits over the slice (for example *0.2*).
9. On the slice container, open the PieSlice editor and set Control Text Values to a data type, e.g. None, Integer, Float or Formatted. By choosing Integer or Float the slice text will change, which shows the value for the slice.
10. In the pie slice editor, adjust the Text Offset to position the value label relative to the center of the pie (for example *70.0*).
11. A single pie slice should be ready. Make the number of slices needed by making copies of the slice container and change their material so they can be differentiated from each other.
12. Set the different values in the PieValues editor to test.

See Also

- Control Object
- Pie Slice
- Pie Values
- Common Control plug-in Properties

14.4.31 Control Scaling

The Control Scaling plug-in enables the Viz Trio user to edit transformation scaling values for objects. Normalization between input and output values can be set by parameters.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Scaling Properties

Control Scaling Properties

- **Minimum & Maximum Input**: Sets minimum and maximum allowed input values.
- **Minimum Scaling**: Sets the minimum transformation value which will be set relative to the minimum input value.
- **Maximum Scaling**: Sets the maximum transformation value which will be set relative to the maximum input value.

- **Control X, Y, Z**- Controls the X, Y, and Z-axis independently when enabled. The Viz Template Wizard TripletEditor component has three onChange events available that accommodates for these parameters.
- **Input Value**: Shows the current input value. Enter a value to test the relation of max/min. input and scale values.
- **Stop Key Frame**: Forces the scaling values entered to be put into the end Key Frame for the animation if the scaled object is animated. For instance if a user is to create a bar graph that animates from zero to a value specified, then this is the case. Name the end Key Frame in the Viz Artist stage and enter that name in the “stop Key Frame” field in the control scaling plug-in.

See Also

- Common Control plug-in Properties
- Control Object

14.4.32 Control Sign Container

The Control Sign Container value can show one of three containers based on the value of a numerical input. The container that holds the plug-in must have three Sub-Containers:

- A negative value will trigger show the first container.
- A positive value will trigger show the second container.
- A zero value will trigger show the third container.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Sign Container Properties

Control Sign Container Properties

- **Input Value**: Shows the current input value. Enter a value here to test that the correct Sub-Container is shown.

See Also

- Common Control plug-in Properties
- Control Object

14.4.33 Control SoftClip

The Control SoftClip plug-in allows the user to control and browse for video clips played in Viz

Artist using the Soft Clip plug-in. Simply add the control plug-in to the same container as the SoftClip plug-in. Further, specify a Field Identifier and a Description.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control SoftClip Properties

Control SoftClip Properties

This plug-in does not have any properties or parameters except those that are common to all plug-ins.

See Also

- Common Control plug-in Properties
- Control Object

14.4.34 Control Stoppoint

Control Stop point enables control of the time value on a stop point. This can typically be used for setting the length of an animation.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Stoppoint Properties

Control Stoppoint Properties

- **Stoppoint**: Sets the name of the stop point to control.
- **Director**: Sets the name of the animation director where the stop point is placed.
- **Stop Time**: Sets the stop time in seconds. This is the value that is set by the Trio operator.

> ⚠ **Note:** This Control plug-in does not work with transition logic based scenes.

See Also

- Common Control plug-in Properties
- Control Object

14.4.35 Control Targa Alpha

The Control Targa Alpha plug-in binds control of Targa video alpha properties to a Control Object.

> ⚠ **IMPORTANT!** The Control Targa Alpha plug-in is deprecated. It is installed because of backward compatibility, and is set to inactive by default. It is not a supported plug-in.

This section contains information on the following topics:

- Control Targa Alpha Properties

Control Targa Alpha Properties

- **Channel**: Sets which channel to control.

See Also

- Common Control plug-in Properties
- Control Object

14.4.36 Control Targa Clip

The Control Targa Clip plug-in exposes control over a Targa clip channel. The Viz Trio user will get a Windows file browser, where video clips to play can be browsed for.

> ⚠ **IMPORTANT!** The Control Targa Alpha plug-in is deprecated. It is installed because of backward compatibility, and is set to inactive by default. It is not a supported plug-in.

This section contains information on the following topics:

- Control Targa Clip Properties

Control Targa Clip Properties

- **Global Clip:** When set to **On**, the clip specified will not be added to the Stage, but will play globally in the Scene. If set to **Off**, two more fields show, Director and Device key frame. The clip will then be loaded at the named key frame in the specified Director and will run when the Director is triggered.

See Also

- Common Control plug-in Properties
- Control Object

14.4.37 Control Text

The Control text plug-in binds the control of text to a ControlObject.

This plug-in can both be used to control regular text geometries and also text revealed by TFxWrite plug-ins. The plug-in will automatically detect and use the appropriate technique to set the text on its container.

The plug-in will also auto detect the Convert Case plug-in and route text through it if present.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Text Properties

Control Text Properties

- **Maximal characters**: Sets the maximum number of characters which can be typed into the text field.
- **Upper Case**: Converts lower case input into upper case when set to **On**.
- **Expose kerning**: Changes the minimum and maximum kerning parameters when set to **On**.
- **Expose line spacing**: Sets the minimum and maximum parameters for line spacing when set to **On**.
- **Expose Font:** Selects different Font when set to **On**.
- **Expose justification:** Changes the text justification when set to **On**.
- **Single Line**: Disables text wrap. Text is shown on one single line.
- **Use Formatted Text**: Enables use of character specific formatting in supported client applications. Character specific formatting includes kerning, position, rotation, scaling, alpha, and color. Formatted text cannot be used together with the Text FX Write plug-in.
  - To use the characters '<' and '>' correctly in Viz Multichannel, make sure that **Use Formatted Text** is set to `OFF`.
  - Character specific formatting is not supported for Viz Pilot.
  - In Viz Trio, the operator enter a required formatting mode via an assigned shortcut key. To change the format of one or more characters, hold down `<Alt>` and toggle the right and left arrow keys to shift mode. See the Viz Trio User Guide for more information on *Character Formatting*.
- **Input Value**: Shows the current value for the plug-in. It is not necessary to set any value here. It is used for debugging purposes.

See Also

- Common Control plug-in Properties
- Control Object

14.4.38 Control Video

The Control Video plug-in exposes control over a video codec channel. Add this plug-in to give the Client Application operator an editor, such as the Media Search in Viz Trio or Viz Pilot, in which to search Viz One for video clips.

The plug-in works in combination with the Video Clip plug-in. This allows you to merge Key Frames with the object for Transition Logic scenes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Video Properties
- To run the clip at a specific Key Frame on a specific Director

Control Video Properties

- **Loop Clip**: Enables or disables continuous looping for a video clip in graphics.
- **Auto Play**: Starts playout on scene load.
- **Toggle Channel**: Enables toggling between two clips in a Transition Logic scene. Enable to create a smooth transition between two clips when loading a new clip.
- **Codec Channel**: Sets the codec channel for the video board in use:
  - **1**: Is normally used as the default codec channel for full screen video.
  - **2**: Is normally used as the default codec channel for video in graphics.
- **Global Clip**: Plays the selected clip as a Media Asset, instead of adding the clip to the stage when set to **On**.

  > ⚠ **Tip:** Set **Global Clip** to `Off` to access the **Director** and **Clip Key Frame** fields.

- **Director**: Specifies the name of the director that the clip loads to.
- **Clip Key Frame**: Specifies the name of the Key Frame the clip loads to.
- **Viz One**: Transfers Clips From Viz One.
- **Use CUE**: Cues up clips by name only, and does not affect the involved clip channel. Requires **Global Clip** to be `Off`.
- **Expose cuing field**: Setting this to `On` lets client applications control the **Use CUE** option through the sub-field `cue`. Requires **Global Clip** to be `Off`.

To run the clip at a specific Key Frame on a specific Director

1. Set **Global Clip** to `Off`.
2. Enter the name of the director that controls playout of the clip in the **Director** field.
3. In the **Key Frame** field, enter the name of the Key Frame that triggers playout.

See Also

- Common Control plug-in Properties
- Control Object
- Video Clip
- Media Assets
- Media Asset Panel

14.4.39 Control World

The ControlWoC plug-in allows you to expose a set of properties to the operator. The control is a replacement for the Control Map plug-in with much more options and on-the-fly feedback from Viz Artist/Engine.

> ⚠ **Note:** The two controls are not compatible. The new control with not work with Viz Pilot, and is only a part of Viz Artist/Engine 3.3 and above.

The control can expose different fields based on the container it resides on. When tabbing to the control (in a navigator scene) the camera will jump to map location and all feedback (exact camera position) will be immediate.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control World Properties

Control World Properties

- **Client Mode**: Sets the client mode to Full or Simple. This will, when selecting a map, open the full or simple map client in the control application.
- **Input**: Sets the exposed properties. The field is based on the input from the NavFinder plug-in; however, you may also manually change this field.

See Also

- Common Control plug-in Properties
- Control Object

14.4.40 Control Field Renamer

The **Field Renamer** plug-in allows for renaming the field identifier of multiple control plug-ins in a single operation.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Control Field Renamer Properties
- To Rename Control Fields

Control Field Renamer Properties

- **Begin from ID**: Enter the numeric value of the first ID in the renaming range.
- **End at ID:** Enter the numeric value of the last ID in the renaming range.
  - **Follow Changes**: Changes the Begin and End IDs will also be changed if set to **On**.
- **Padding**: Sets the zero padding for the desired output. Zero padding adds one or more prefixing zeros if the ID consists of fewer digits than specified.

  > ⚠ **Example:** If **Padding** is set to `4`, the ID '14' will be changed to `0014`, and the ID '103' will be changed to `0103`.

- **Method**: Choose the renaming method:

  - **Set:** In the field **Set lowest ID to**, enter the desired value to change all IDs in the specified range by the same value

  - **Shift:** Shifts all the IDs in the specified range by the provided amount
- **Execute**: Click the Execute button to execute the rename process.

To Rename Control Fields

1. Drag the **Field Renamer** plug-in onto a container. When executing the renaming process, all control plug-ins in that container and all of its' child containers will be affected.

   > ⚠ **Tip:** To rename all field identifiers of an entire scene, place all containers in the scene inside a parent container and add the **Control Field Renamer** plug-in to that parent container.

2. Set the range of IDs to be changed in **Begin from ID** and **End at ID**.

> ⚠ **Note:** If the **Follow Changes** option is set to on, the above two range fields will also be changed.

3. Use **Padding** to set the length for zero padding. Set to zero for no padding.
4. Choose the renaming **Method**.
5. Click **Execute** to execute renaming.

See Also

- Common Control plug-in Properties
- Control Object

14.4.41 Placeholder

The Placeholder plug-in is used to mark placeholder containers when designing library objects for Viz Trio compositing. A placeholder container will be available as a parent to other compositing objects in the scene tree.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Control

This section contains information on the following topics:

- Placeholder Properties

Placeholder Properties

- **Sort Order**: Integral sorting order identifier (lower number is sorted first).
- **Description**: A description of the tab field or editable object. This will be used as a description for the items when used in the control clients.

See Also

- Common Control plug-in Properties
- Control Object

14.5 Default

The following container plug-ins are located in the Default folder:

- Global Magnifier Controller
- Screen2World
- Trio Scroll item

14.5.1 Global Magnifier Controller

The Global Magnifier Controller plug-in gives control over the global magnifier from inside a plug-in. When the controller is assigned to a container, the global magnifier is applied to this container. The magnifier is positioned and scaled according to the positional and scale values of the host container. The plug-in additionally provides access to the global magnifier properties **Scale**, **Ellipsis** and **Alpha**.

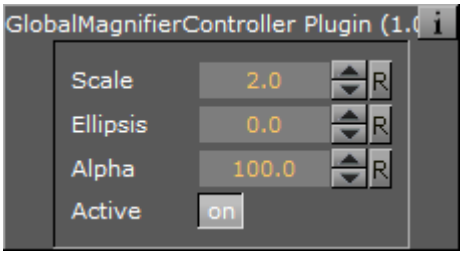
Please observe that only one Magnifier Control plug-in may be active within a scene at any given time. Because of this, if there are several instances of the Global Magnifier Controller, the global magnifier will only be applied to the last instance activated. However, when the currently applied controller is deactivated, the plug-in automatically applies the global magnifier to the next container with an active controller plug-in. The global magnifier is fully deactivated only when there are no active controllers left.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Default

This section contains information on the following topics:

- Global Magnifier Controller Properties

Global Magnifier Controller Properties



- **Scale:** Sets the scale of magnification. The minimum value is `1.0`, the maximum value is `100.0`. The default value is `2.0`.
- **Ellipsis:** Creates an ellipsis effect, mimicking the distortion created by a concave or convex lens. The minimum value is `-10.0`, the maximum value is `10.0`. The default value is `0.0`.
- **Alpha:** Sets the alpha value for the magnification. The minimum value is `0.0`, the maximum value is `100.0`. The default value is `100.0`.
- **Active:** Enables or disables magnification. Can be set to `On` or `Off`. By default, this is set to `On`.

14.5.2 Screen2World

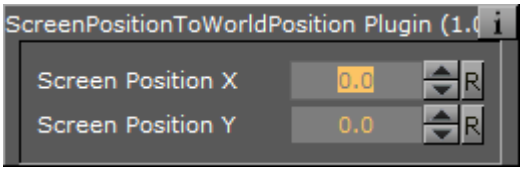
The Screen2World plug-in converts the passed screen positions X and Y to world positions X, Y and Z, and sets them for the container the plug-in is applied to.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Default

This section contains information on the following topics:

- Screen2World Properties

Screen2World Properties



- **Screen Position X:** Sets the X value for the screen position. Accepted values are floating point numbers ranging from `-100000.0` to `100000.0`. The default value is `0.0`.
- **Screen Position Y:** Sets the Y value for the screen position. Accepted values are floating point numbers ranging from `-100000.0` to `100000.0`. The default value is `0.0`.

14.5.3 Trio Scroll item

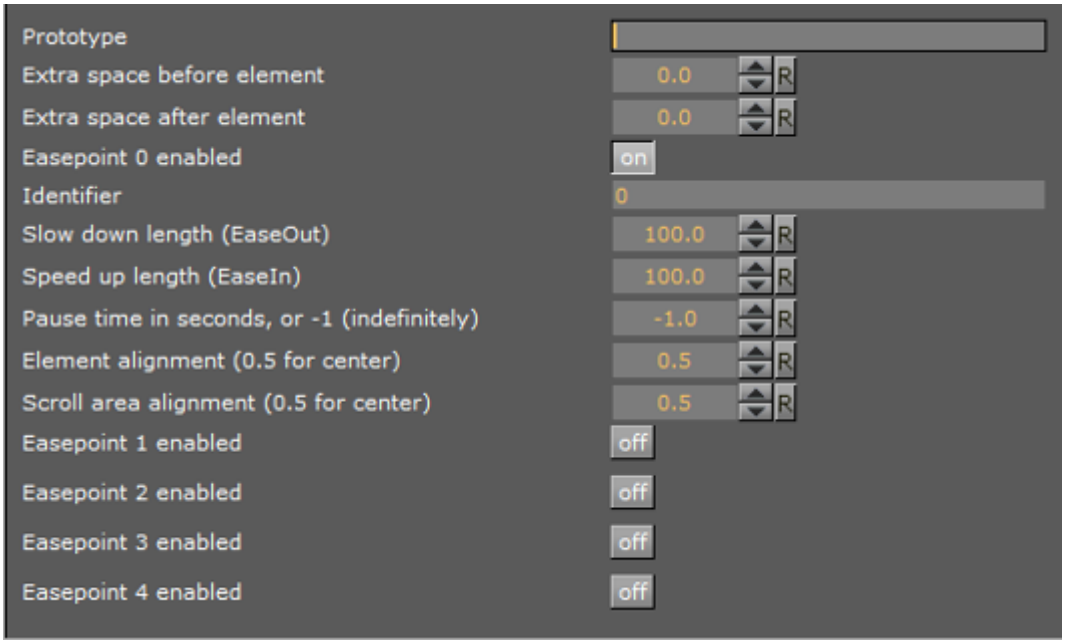
The Trio Scroll item plug-in is used for data storage by the Trio Scroll Geometry plug-in. It is automatically added to each base item of a Viz Trio scroll scene.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Default

This section contains information on the following topics:

- Trio Scroll Properties

Trio Scroll Properties



- **Prototype:** Sets the path and the name of the scene.
- **Extra space before item:** Sets the space before the item.
- **Extra space after item:** Sets the space after the item.
- **Easepoint 0-4 enabled:** Enables the easepoint(s).
  - **Identifier:** Sets the identifier for the easepoint.
  - **Slow down length (EaseOut)**: Sets the size in pixels that will be used to ease out the scroll item.
  - **Speed up length (EaseIn)**: Sets the size in pixels that will be used to ease in the scroll item.
  - **Pause time in seconds, or -1 (indefinitely):** Sets the time in seconds for how long the scroll should wait before continuing the scroll.

    > ⚠ **Note:** This will affect the total time the scroll is scheduled to use.

  - **Item alignment (0.5 for center):** Sets the alignment of the item relative to the height of the item. 0.5 (equal to 50%) is the default value and represents the middle of the item's bounding box area.
  - **Scroll area alignment (0.5 for center):** Sets the scroll alignment for the item in the scroll area. 0.5 (equal to 50%) is the default value and represents the middle of the scroll's bounding box area.

See Also

- **Create New Scroll** in the Viz Trio User Guide
- Trio Scroll Geometry plug-in

14.6 Feed

The purpose of the Feed plug-ins is to receive real-time data and apply it to the state of a Viz Artist Scene in various ways. They are mostly used to create Scenes for use in financial data and stock feed visualizations.

> ⚠ **Note:** They are now mostly replaced by a combination of the shared memory map, Viz scripting, Control plug-ins and the Apply Shared Memory plug-in as the preferred way to bring real-time data into Viz Artist Scenes.

The method of use for the plug-ins is as follows:

1. Identify text geometry, or container visibility in a scene that an external real-time data feed source is to control.
2. Apply the appropriate Feed plug-in to the container of the identified property.
3. In the plug-in properties, specify a data feed key to look for in the incoming data feed.
4. Send UDP messages to carry the content of the data feed in a YAML format which carries key-value pairs to a specific port on the Viz Engine host, which the plug-in infrastructure listens to.

The following container plug-ins are located in the Feed folder:

- Feed Activate
- Hide in Range
- Feed View

14.6.1 Feed Activate

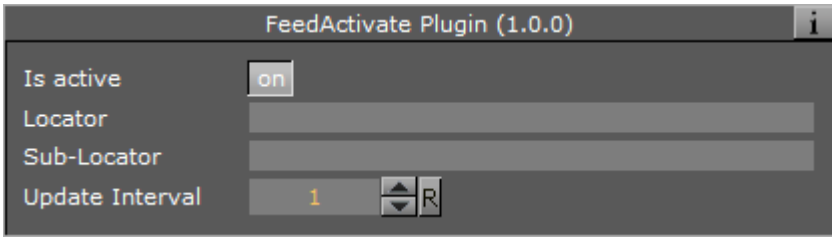
Feed Activate activates and deactivates plug-ins based on a given value in a FeedRelay hierarchy.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Feed

This section contains information on the following topics:

- Feed Activate Properties

Feed Activate Properties



- **Is active:** Updates visibility of sub-containers based on container name and value retrieved location.
- **Locator:** Gets data for text from this location.
- **Sub-Locator:** Appends this to locator path when retrieving text.
- **Update Interval:** Determines how often (in number of fields) an update occurs (increase to gain performance).

14.6.2 Hide in Range

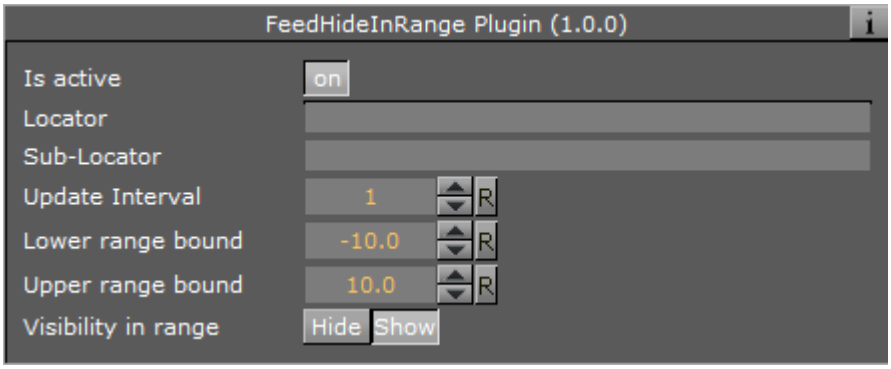
Makes container either visible or invisible depending on whether value from real-time feed is within a specified range of numerical values.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Feed

This section contains information on the following topics:

- Hide in Range Properties

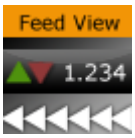
Hide in Range Properties



- **Is active:** Updates text geometry of container.

- **Locator:** Gets data for text from this location.
- **Sub-Locator:** Appends this to locator path when retrieving text.
- **Update Interval:** Determines how often (in number of fields) an update occurs.

14.6.3 Feed View



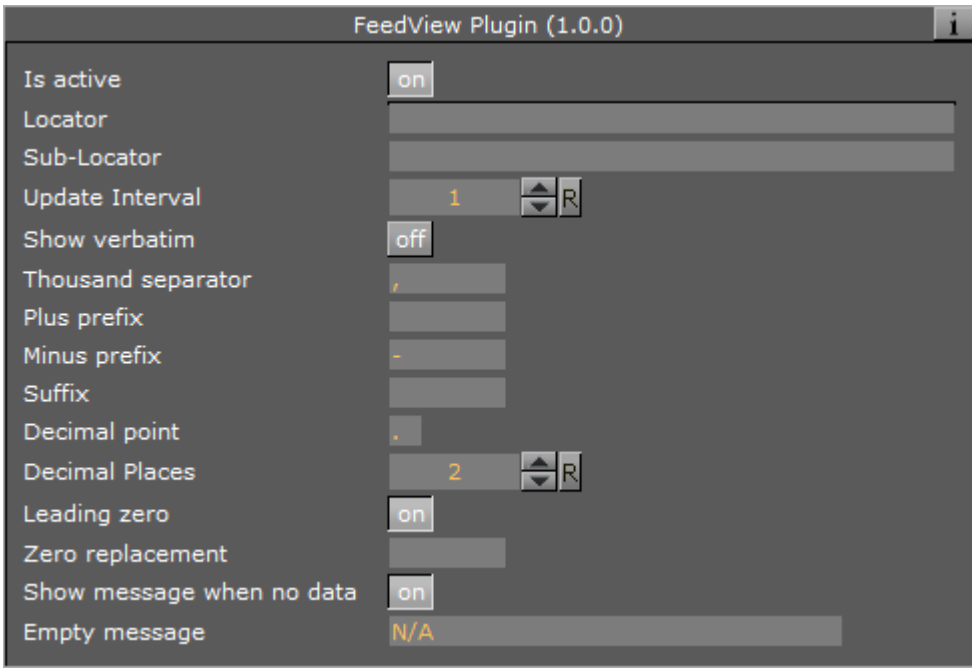
Applies value from a real-time feed onto text geometry with various forms of optional formatting.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Feed

This section contains information on the following topics:

- Feed View Properties

Feed View Properties



- **Is active**: Updates text geometry of container.
- **Locator:** Gets data for text from this location.
- **Sub-Locator:** Appends this to locator path when retrieving text.
- **Update Interval:** Determines how often (in number of fields) an update occurs.
- **Show verbatim:** Shows data as given, does not interpret and reformat.
- **Plus prefix:** Uses this prefix for positive values.

- **Minus prefix:** Uses this prefix for negative values.
- **Suffix:** Appends this suffix.
- **Decimal point:** Uses these characters as a decimal point.
- **Leading zero:** Uses leading zero for values less than one.
- **Zero replacement:** Uses this string when value is zero.
- **Show message when no data:** Sets the contents of the 'Empty Message' field as text if the locator is not found in the cache if checked. If this is unchecked, the text geometry will be left unmodified.
- **Empty Message:** Sets text if the locator is unregistered.

14.7 Global

The following container plug-ins are located in the Global folder:

- Alpha
- Audio
- Clipper
- Expert
- Extrude
- Glow
- HDR
- Key
- Look-At
- Mask Source and Mask Target
- Lighting
- Z-Sort
- Projector Source and Projector Target
- Script
- Shadow Caster and Shadow Receiver
- Synchronized Properties
- Video Clip
- Window Mask

14.7.1 Alpha

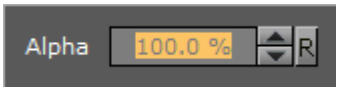
The Alpha plug-in adds an alpha channel to the container. The alpha channel defines the degree of transparency for the container and its Sub-Containers. This feature can be used to easily fade in or out a complete part of the scene-tree.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section contains information on the following topics:

- Alpha Properties

Alpha Properties



- **Alpha:** Sets the alpha value for the container.

14.7.2 Audio

The Audio plug-in allows a designer to configure audio channels. The Audio plug-in can be applied to any Container. The plug-in has three different audio clip mixing modes, and each mode panel has a test function.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> Global*

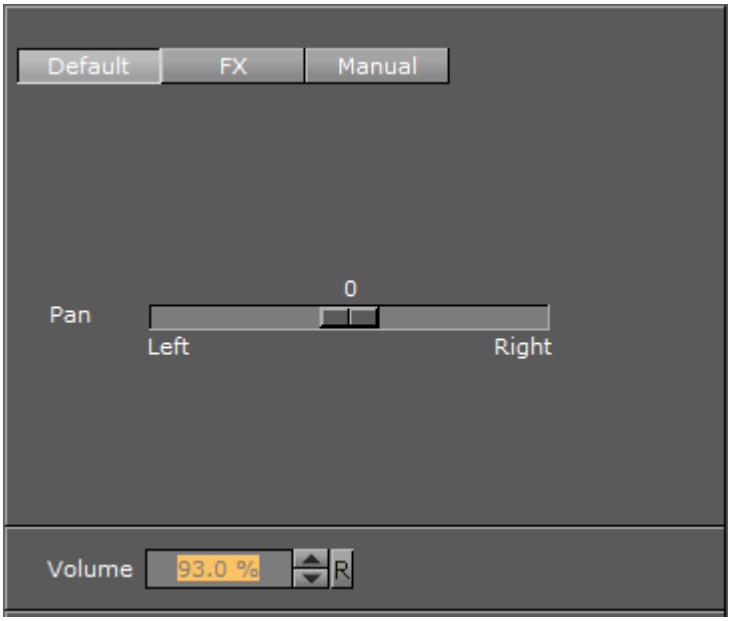
This section contains information on the following topics and procedures:

- Default Mode
- FX Mode
- Manual Mode
- To Configure Audio Channel Aliases
- To Test Audio Channel Setup

Default Mode

The Default mode uses the Channel settings stored in the clip. For example FrontLeft and FrontRight are mixed to the aliases FRONT_LEFT, FRONT_RIGHT and so on.

If Viz Engine is configured with fewer channels, Viz Engine mixes the remaining clip channels according to the channel geometry. For example if Viz Engine only has stereo configured, but the clip used has 7.1 surround sound, Viz Engine mixes LeftBack and LeftMiddle to FRONT_LEFT and so on.



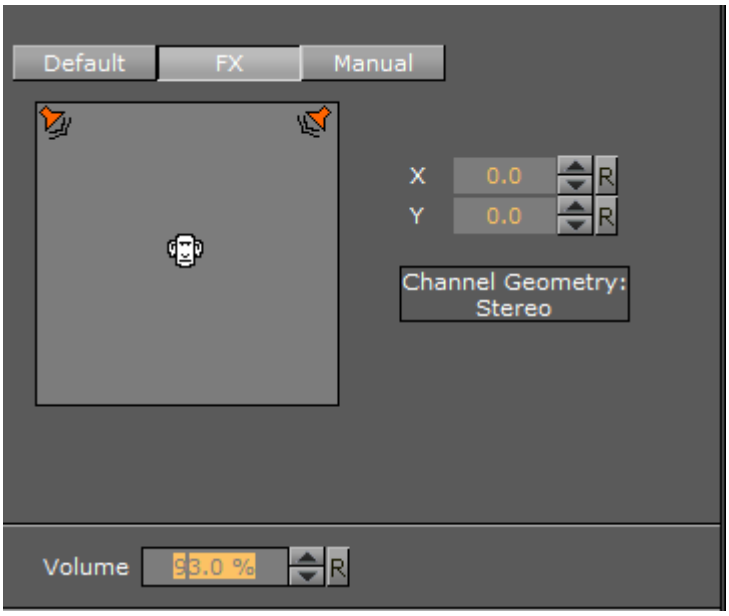
The **Pan** slider uses audio aliases when adjusting the volume of the left and right speakers. This is valid for multichannel configurations as well. If the slider is moved to the right in a 7.1 configuration LeftBack, LeftMiddle and LeftFront will be muted.

The **Volume** field controls the overall audio volume of all clip channels.

FX Mode

In the FX mixing mode Viz Engine mixes the clip in relation to the listener position which can be defined in the fields X and Y. Viz Engine takes the channel geometry into account.

The **Volume** field controls the overall audio volume of all clip channels.



Manual Mode

In Manual mixing mode it is possible to enter the Out Channel alias for every clip channel. The mask can hold more than one channel. This allows the mix of every clip channel to an arbitrary amount of out channels.

The **Volume** field controls the overall audio volume of all clip channels.

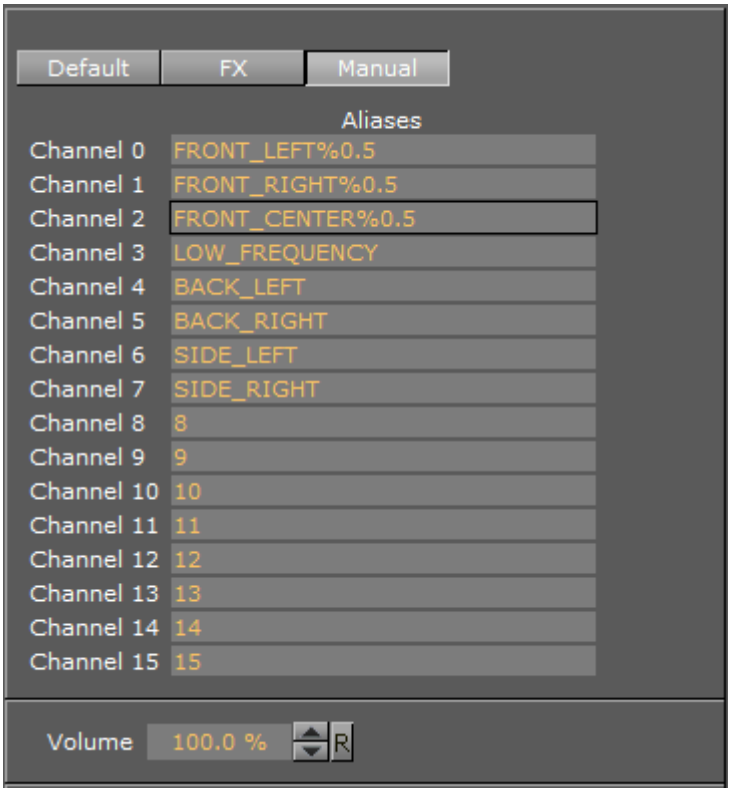
The syntax for the string is as follows:

```
CHANNEL_ALIAS%VOLUME+CHANNEL_ALIAS%VOLUME…
```

or

```
[CHANNEL_ALIAS%VOLOUME]+
```

⚠ **Example:** [FRONT_LEFT%0.5]+



To Configure Audio Channel Aliases

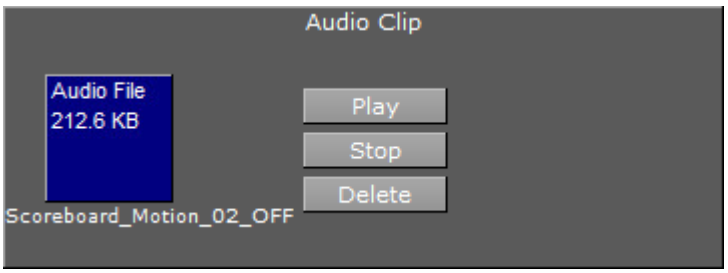
1. Add an Audio clip to a Scene.
2. Click the **Audio** plug-in icon (opens the Audio plug-in editor).
3. Click **Manual**.

4. Enter these channel configurations (as entered in **Audio Settings** in Viz Configuration (see the Viz Engine Administrator Guide):
   - Out Channel 0: FRONT_LEFT
   - Out Channel 1: FRONT_RIGHT
   - Out Channel 2: FRONT_CENTER
   - Out Channel 3: LOW_FREQUENCY
   - Out Channel 4: BACK_LEFT
   - Out Channel 5: BACK_RIGHT
   - Out Channel 6: SIDE_LEFT
   - Out Channel 7: SIDE_RIGHT

   > ⚠ **Note:** You can use aliases as configured in the Channels tab (see the **Audio Settings** page in the **Configuring Viz** section of the Viz Engine Administrator Guide

5. Save the scene.
   Depending on the settings in the channel configuration a clip will now play the different languages.

To Test Audio Channel Setup



1. Add a group container to the scene tree
2. Add the Audio plug-in to the group container
3. Open the Audio plug-in editor, and add an audio clip to the *Audio Clip* drop-zone, and click Play

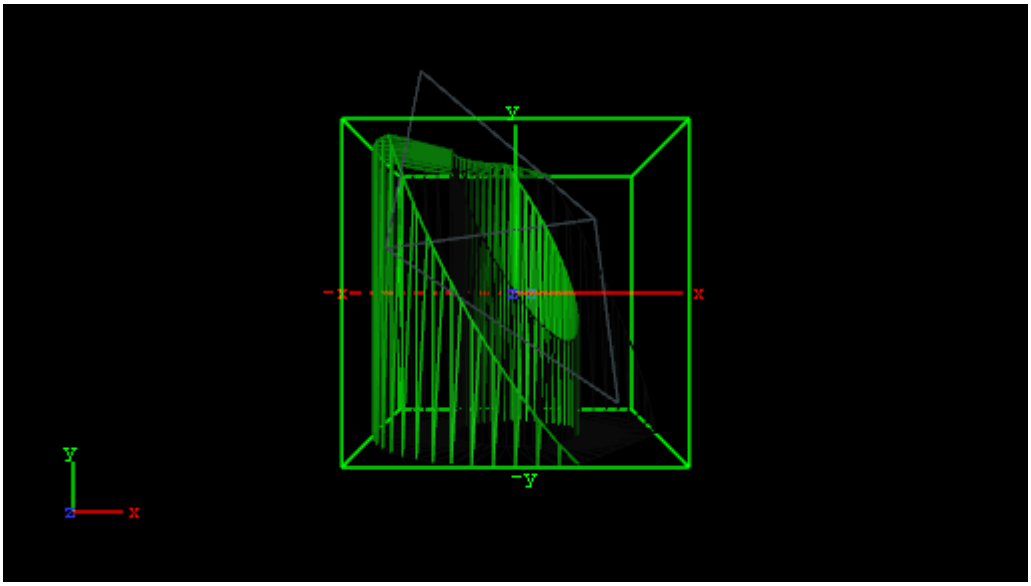
   > ⚠ **Tip:** Always have a set of test clips that will provide audio for the different channel setups.

See Also

- Audio Settings (see the **Configuring Viz** section of the Viz Engine Administrator Guide)

14.7.3 Clipper

The Clipper plug-in is an alternative to the mask function. Objects with the clipper plug-in attached that are placed behind the clipper plane and will be masked/clipped. If an object is only partially behind a clipper plane, only the part of the object that are behind the plane will be clipped out, since the clipping is done in true 3D space. The function uses the OpenGL clipping planes. Up to 6 of them may be used. To define a clipper plane use the Properties Panel (see Modify Container Properties).



All the containers that you want to be affected by the clipping plane, you must have the clipper plug-in applied and thereafter in the editor specify which of the clipping plane is to mask out/clip the container. It is possible to let a container be affected by multiple clipping planes.

The clipper plug-in can replace mask in many situations. Some of the advantages are compared to normal mask are:

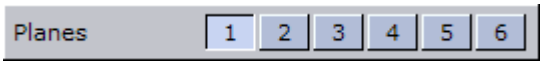
- Clipping is done in true 3D space, while masking is done in 2D (although the 2D mask is created by a 3D object).
- There is no performance hit. In fact there is even a performance gain if parts of the object are clipped. Regular masks have a quite high performance cost.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics and procedures:

- Clipper Properties
- To Create a Clipping Plane Effect

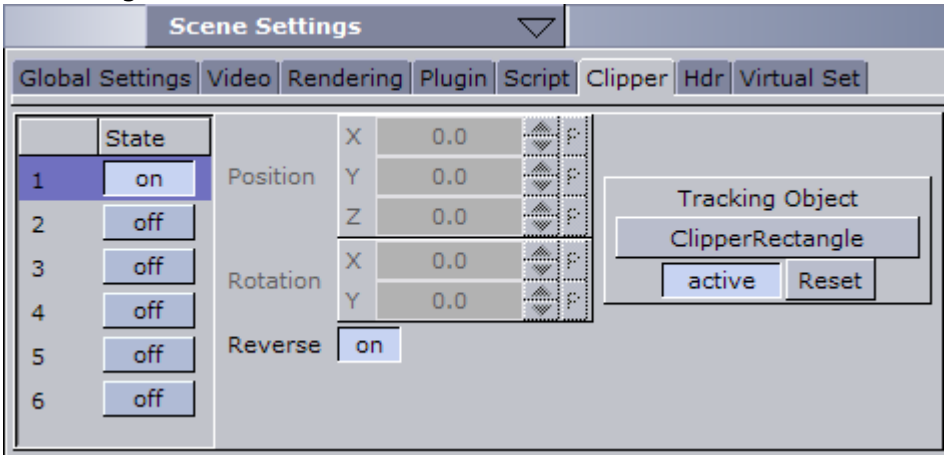
Clipper Properties



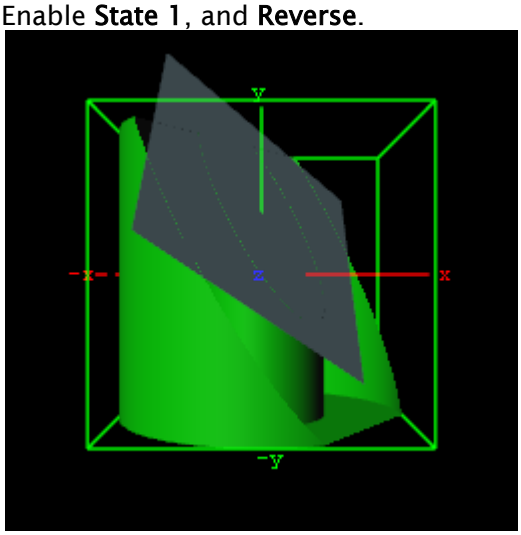
- **Planes:** Enables the planes. By default 1 is enabled.

To Create a Clipping Plane Effect

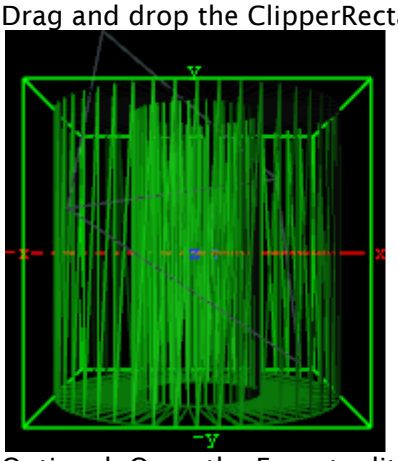
1. Add two group containers to the scene tree (same level) and name them **Cylinder** and **ClipperRectangle**.
2. Add the following to the **Cylinder** container:
   - Cylinder geometry plug-in
   - Clipper container plug-in
   - Material
   - Optional: Expert container plug-in
3. Add the following to the **ClipperRectangle** container:
   - Rectangle geometry plug-in
   - Material
   - Optional: Expert container plug-in
4. Open the editor for the Cylinder container and set **Scaling** (locked) to 2.0.
5. Open the Cylinder editor and set Hole to 45.0.
6. Open the Clipper editor and enable **Plane 1** (enabled by default).
7. Optional: Open the Expert editor(s) and enable **Back Face** (On) and **Twosided Lighting**.
8. Open the editor for the ClipperRectangle and set the following parameters for the following properties:
   - Set Position Y to 45.0.
   - Set Rotation X to -45.0.
   - Set Rotation Y to 45.0.
   - Set Scaling (locked) to 2.0.

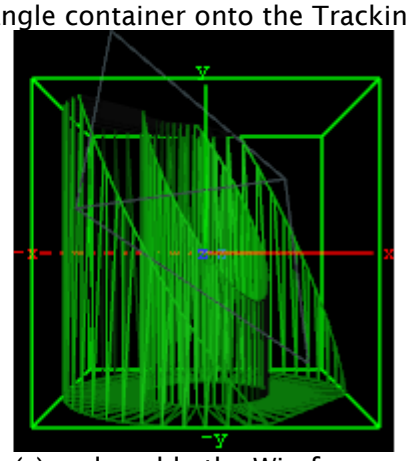


9. Open Scene Settings, and select the Clipper Panel tab.
10. Enable **State 1**, and **Reverse**.



11. Drag and drop the ClipperRectangle container onto the Tracking Object drop zone.





12. Optional: Open the Expert editor(s) and enable the Wireframe mode to see rendering result, and the effect clipping has on the polygons.

See Also

- Clipper Pane

14.7.4 Expert

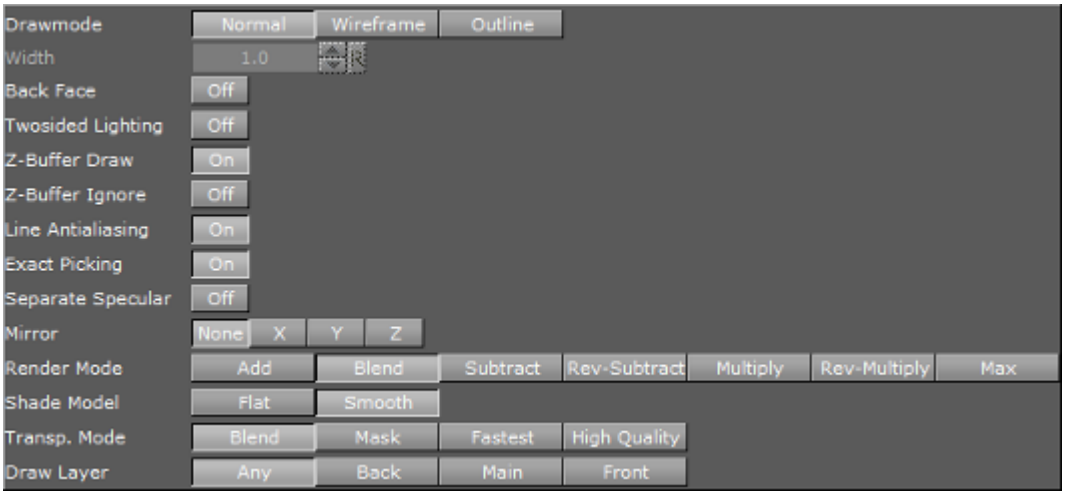
The Expert plug-in sets some special properties and adds some advanced functions to a Container.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topic:

- Expert Properties

Expert Properties



- **Draw Mode:** Selects how to have the container drawn:
  - **Normal**: Shows the rendered object with material and/or textures.
  - **Wireframe:** Shows the polygons that are drawn to create the object.
  - **Outline:** Shows the outline of the object.
- **Width**: Sets the width of the wires if **Wireframe** or **Outline** draw mode is selected.
- **Back Face:** Draws a backface for an image or a 2D font when enabled. This will of course take more performance to render.
- **Twosided Lighting:** Enables two-sided lighting for a single container instead for a whole scene.
- **Z-Buffer Draw**: Disables the OpenGL z-buffer draw when set to **Off**. This is in some cases the last possibility for solving some z-sort problems. The technical explanation for this is very complex and will not be described here.
- **Z-Buffer Ignore**: Ignores the OpenGL z-buffer draw option when set to **Off**, and draws the object no matter if it is in the back or not.
- **Line Anti aliasing:** Renders an anti-aliased outline for 2D objects on machines that do not have anti-aliasing with multi-sampling. This extra outline can be removed by disabling this option.
- **Exact Picking:** Enables or disables Exact picking for Events. If Exact Picking is disabled, the bounding box is used instead of a pixel readback.
- **Separate Specular:** Enables specular highlights for textured geometry.
- **Mirror**: Enables you to mirror an image/texture or a 2D font over the X-, Y- and Z-axis.
- **Render Mode:**
  - **Add:** Symbolically: C*A + FC. The source color gets added to the target color. The amount of color that is added depends, as we see from the formula, on the alpha value. However, it is always an addition, so the end result will always be a lighter color than the initial frame content. If the frame color has high values on all three color channels (RGB), you might experience that the addition of the new color takes all

channels to values above 255 (saturation). The values will be clamped at 255, which is white.

- **Blend**: Symbolically: C*A+(1-A)*FC. The new color value gets created as a weighted average of the source and the target. The weight factor is the alpha value of the rendered color. That means that if the incoming color has a very low alpha value, its influence on the new color will be small, and conversely, if the new color has a high alpha value, its influence on the new color will be bigger.
- **Subtract**: Symbolically: FC-C*A. The new color is the result of the incoming color being subtracted from the color in the frame buffer and the result is written back into the frame buffer. The alpha value of the incoming color decides how much who gets subtracted. If the incoming color has high values on all three color channels (RGB), you might experience that the subtraction of the new color takes all channel values to 0, and the result is a black color.
- **Rev-Subtract**: Symbolically: C*A-FC. This is a reversed version of subtractive. The color in the frame buffer gets subtracted from the incoming color and the result is written back into the frame buffer.
- **Multiply:** Symbolically: CA*FC. The new color and alpha get multiplied with the existing values in the frame buffer. The formula presupposes that the colors and alpha re described as values between 0 and 1. A color rendered with multiply will always result in a darker color than both the color being rendered and the color in the frame buffer.
- **Rev-Multiply:** This is a reversed version of multiply.
- **Max:** Higher color values will overwrite lower color values (for example, white draws over black - independently from the Z-Sort).

- **Shade Model**
  - **Flat**: Shows the single polygons the texture is made of.
  - **Smooth:** Calculates a smooth texture.
- **Transp. Mode**
  - **Blend**: Is the standard option.
  - **Mask**: Uses a raster.
  - **Fastest**: Uses the transparency mode which takes the least time to render.
  - **High Quality:** Uses the transparency mode which delivers the best looking result.
- **Draw Layer:** Defines if the objects are drawn in all layers or only if it shows in a certain layer:
  - **Any:** The object is always drawn.
  - **Main:** The object is drawn only if it is in a scene, located in the Main Layer.
  - **Back:** The object is drawn only if it sits in the Back Layer.
  - **Front:** The object is drawn only if it is in the Front Layer.

> ⚠ **Note:** The Draw Layer option can be used, for example, for a menu in a Scene which should be only visible if it is loaded in the front layer.

14.7.5 Extrude

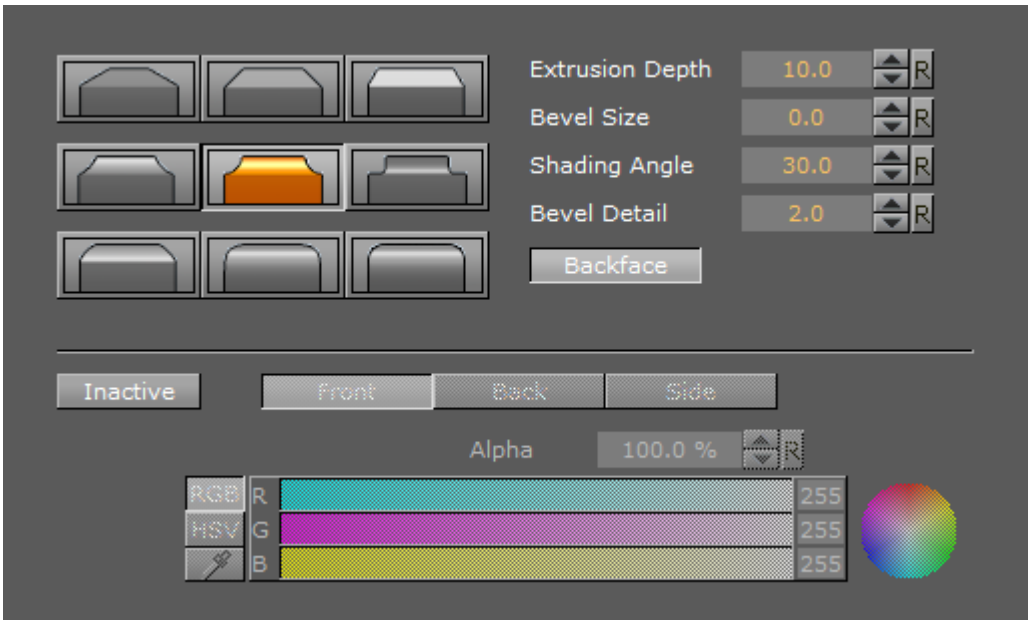
An Extrude plug-in can be explained simply as extruding or sweeping the 2D item through space along its Z-axis. The sweeping path the item is followed during this process is used to create a surface. A a result a 3D item with front, back and sides is created from the 2D item.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

- Extrude Properties
- To Extrude a 2D object

Extrude Properties

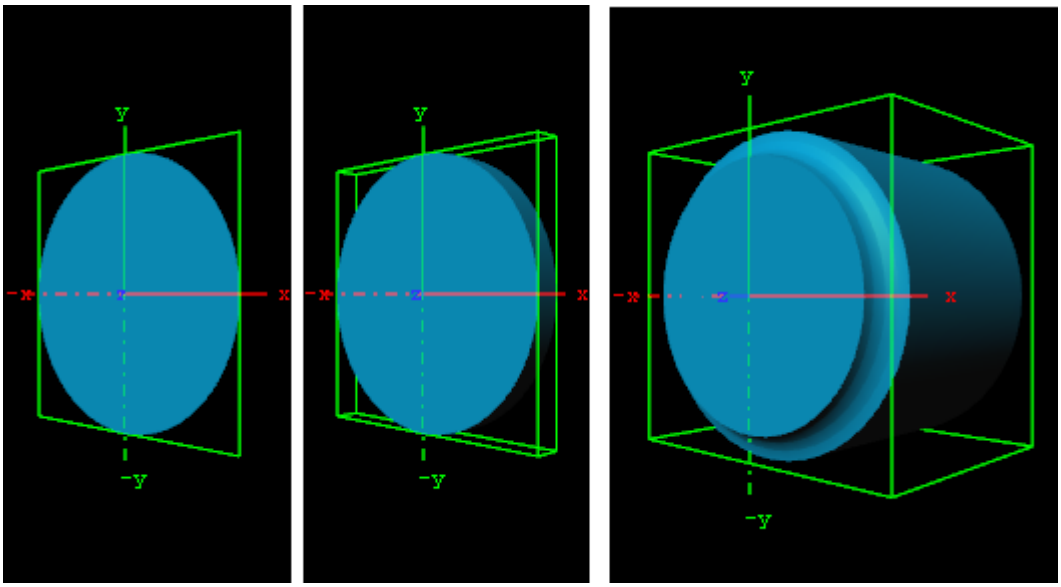


- **Bevel Size:** The buttons marked with different bevel types are used to set the type of bevel to use on the figure.
- **Extrusion Depth:** Sets the extrusions depth on the Z-axis.
- **Bevel Size**: Sets the size of the bevel. This must be set to some value to enable the selection of bevel type.
- **Shading Angle:** Sets the angle of the shading function. To “erase out” unevenness, try increasing the shading angle.

- **Bevel Detail**: Sets the degree of detail of the bevel. The lover the value is set the more detailed the bevel is constructed. The more detailed the bevel is, the smaller tilings its constructed from.
- **Backface:** Allows you toenable or disable visualization of the backface.
- If colors are set to **Active**, you can edit the items **Front**, **Back** and **Side** color by using the sliders or changing the values.

> ⚠ **Note:** Adding the Extrude plug-in to a font will set the font's quality to `Normal` and disable the **Quality** selector in the Text Editor

To Extrude a 2D object



1. Create a group and add the Circle geometry plug-in and material to it.
2. Add the Extrude plug-in to the same container.
3. Open the Extrude editor and select one of the bevel effects, and set Extrusion depth to 100.0 and Bevel Size to 10.0.

14.7.6 Glow

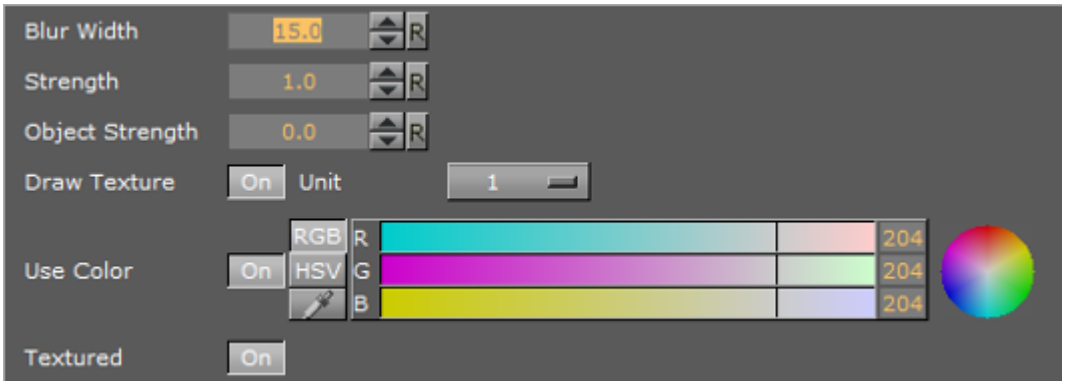
The Glow plug-in makes the lit faces of objects glow.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

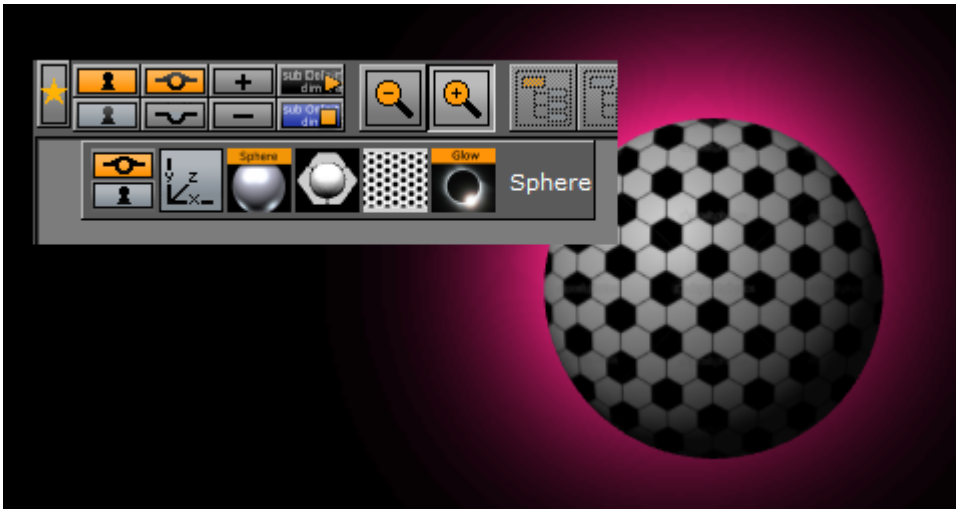
- Glow Properties
- To Add a Glow

Glow Properties



- **Blur Width:** Sets the width of the blur (the size of the glow).
- **Strength:** Sets the intensity/strength of the glow.
- **Object Strength:** Sets the intensity/strength of the glow on the faces of the object.
- **Draw Texture**: Uses the selected Texture for the Glow Effect. When a multi-textured object is used, use the **Unit** drop down box to select which of the textures to use.
- **Use Color:** Changes the color of the glow when set to **On**. If set to **Off** the glow will be based on the Color (Material and Texture) of the Object.
- **Textured:** Applies the Texture, or Textures, of the object on the glow.

To Add a Glow

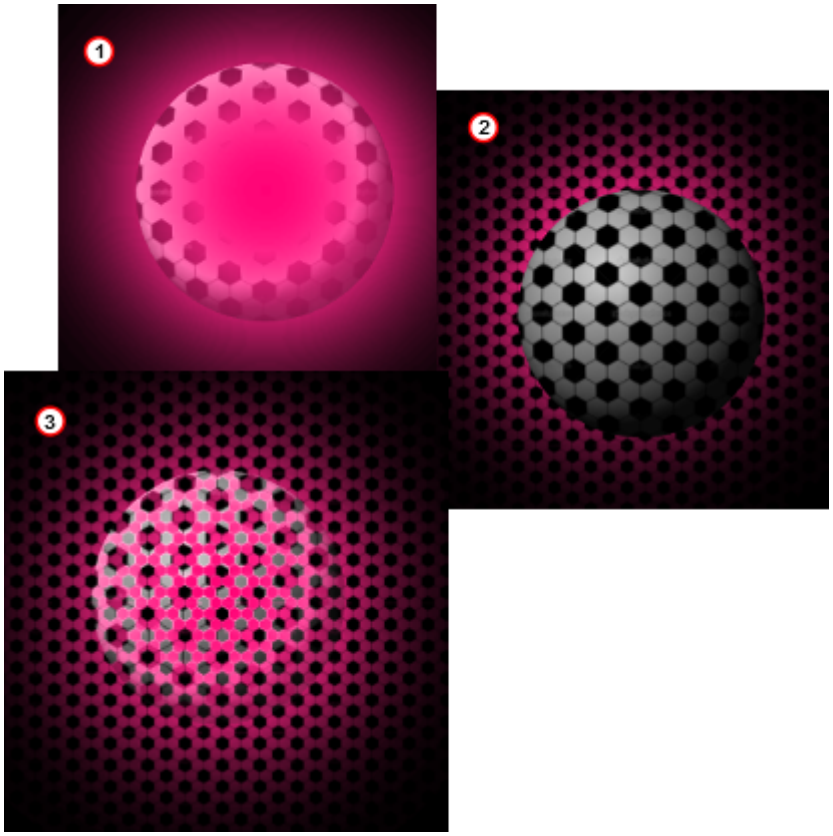


1. Add a Sphere plug-in to the Scene Tree.
2. Add a Material and a Texture to the Sphere container.
3. Add the Glow plug-in.
4. Open the Glow editor and set these parameters:
    - **Blur Width**: 50.0.

- **Strength**: 8.0.
- **Object Strength**: 0.0.
- Enable **Use Color** and set the color parameter.

Examples with the same parameters and:

- **Draw Texture** set to **On** (1)
- **Texture** set to **On** (2)
- **Draw Texture** and **Texture** set to **On** (3)



14.7.7 HDR

Enables High Dynamic Range Rendering/Imaging for the selected container. This improves the contrast of the generated scene in a much more realistic way.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topic:

- HDR Properties

- To Add HDR

HDR Properties

This plug-in does not have any properties or parameters.

To Add HDR

- Drag the plug-in on a container and modify the HDR settings under Scene setting/hdr. Specify a diffuse and a Cubic Reflection image, as required.

14.7.8 Key

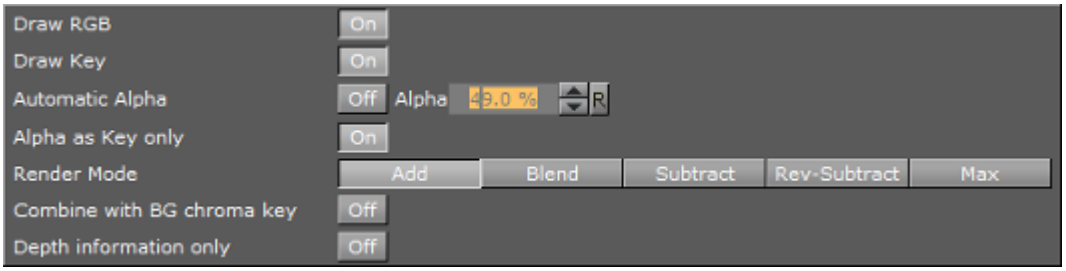
The Key plug-in adds a Key signal to a container.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topic:

- Key Properties
- To Add Key

Key Properties



- **Draw RGB**: Enables or disables the graphical object.
- **Draw Key**: Enables or disables the key signal.
- **Automatic Alpha:** Calculates the alpha value automatically when set to **On**. Takes the alpha value entered when set to **Off**. This is normally done to obtain some level of transparency so that, for example, video background is visible through the key object.
- **Alpha as Key only**: Takes the alpha value of the object and renders the key with this alpha instead of the alpha value of the key function attached. The object itself is rendered opaque. This is usually used to render transparent objects for keyed graphics. The color mixing of

the object in the scene and the, for example, video background is done by the external keyer so the object must be rendered non transparent.

- **Render Mode**: Sets the mode in which the alpha values of the key item to be rendered should be mixed with the alpha values already existing in the frame buffer.

> ⚠ **Note:** Be aware that with the subtractive and the rev-subtractive, mode the order in which the objects are being rendered then is crucial. Make sure the z-sort is correct.

The mixing is done on a per pixel basis (in the formulas the range of the alpha values is from 0-1, instead of 0-100 as in the value field).
If A is the current source alpha value being rendered now and FA is the alpha of the frame buffer contents (target), the different modes can be explained as following:

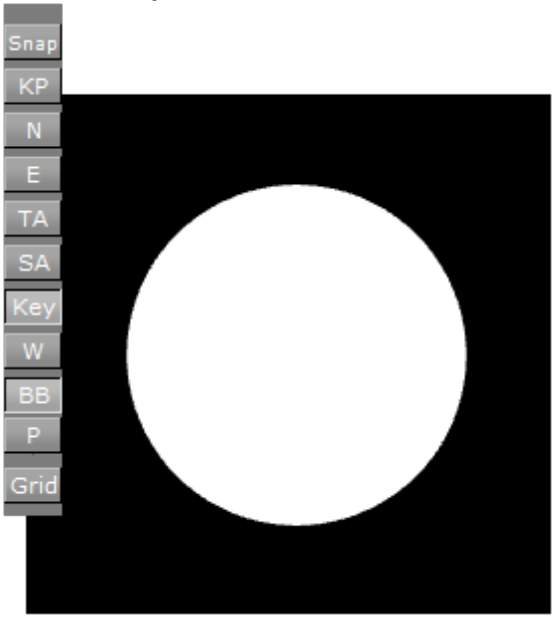
- **Add:** (A + FA) The source alpha gets added to the target alpha. As the formula shows it is an addition, so the end result will always be a higher alpha value than the initial frame content. If the frame alpha has a high value, you might experience that the addition gives a total value above 1. This will be clamped at 1 which is opaque.
- **Blend**: (A+(1-A)*FA) The new alpha value gets created as a weighted average of the source and the target. If we as an example say that the key being rendered has alpha 1 and the frame alpha has 0.5, the formula will be like this: 1+(1-0.5)*0.5 = 0.75.
- **Subtract:** (FA-A) The new alpha value is the result of the incoming alpha being subtracted from the alpha in the frame buffer and the result is written back into the frame buffer.
- **Rev-Subtract:** (A-FA) This is a reversed version of subtractive. The alpha in the frame buffer gets subtracted from the incoming alpha and the result is written back into the frame buffer.
- **Max:** Uses the maximum of all key signals.
- **Combine with BG chroma key**: Set to **On** to combine the alpha of the Container with the alpha of the background, before blending the foreground with the background (active in a virtual sports render sequence only).
- **Depth information only**: Depth only for occlusion rendering with live video. Containers which are marked for **Depth Only** are rendered before Containers which are marked for chroma or linear keying. This could be useful for sports, where real objects should occlude virtual items of the scene. This setting requires the **Render Sequence** to be set to `Arena` in the scene's Global Settings Panel.

To Add Key

1. Open a Scene.
2. Add an object.

3. Add Key plug-in.

4. Click the **Key** button on the left side of the Scene Editor.



See Also

- Global Settings Panel

14.7.9 Look-At

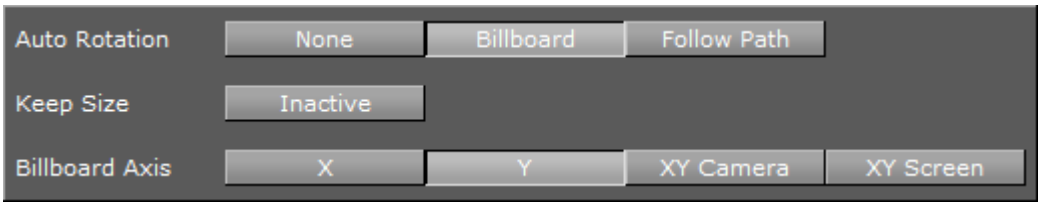
The Look-At plug-in is used to rotate a Container so that it faces a defined direction. This is especially valuable in animations.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topic:

- Look-At Properties

Look-At Properties



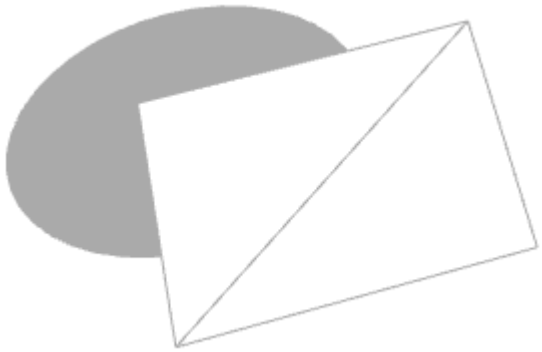
- **Auto Rotation**
  - **None:** No rotation.
  - **Billboard**: Will rotate the container to face the camera.
  - **Follow Path**: Will rotate the container to justify at the animation path.
- **Keep Size**:
- **Billboard Axis**: Only with **Billboard:**
  - **X:** Rotates the container around the X-axis to face the camera, even if camera moves.
  - **Y:** Rotates the container around the Y-axis to face the camera, even if camera moves.
  - **XY Camera:** Rotates the container around the X- and Y-axis to face the camera, even if it moves.
  - **XY Screen**: Rotates the container around the X and Y-axis to face the camera, if the camera orbits the container only.

See Also

- Create Animations

14.7.10 Mask Source and Mask Target

With the Mask (Source and Target) plug-in one Container can act as a mask for other Containers. The Mask Container is called the source and the Containers which are affected by the source mask are called targets. A Container affected by a mask becomes transparent where the mask covers it.



Due to graphics hardware limitations, currently Viz Artist supports a maximum of eight different target and source layers for one object. This means that any target within a layer will be affected by all masks for the same layer. For example, if a mask has the layers 1, 2 *and* 3 selected, it will affect all targets that have 1, 2 *or* 3 selected.

Enable shadows in the Global Settings Panel:

Layers 7 and 8 may not be available because global shadow settings are enabled by default. Shadows use two layers of the mask plug-in, but can be set to **off** in Global Settings Panel, to use all eight layers.

In the Global Settings Panel define if a mask layer is to be drawn inverted. Normally the mask cuts a hole in the mask target where it covers it. If a layer is selected to be drawn inverted the opposite situation is the case. Only where the mask covers the mask target will the target be visible.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

- Mask Source and Mask Target Properties
- To Add a Mask

Mask Source and Mask Target Properties

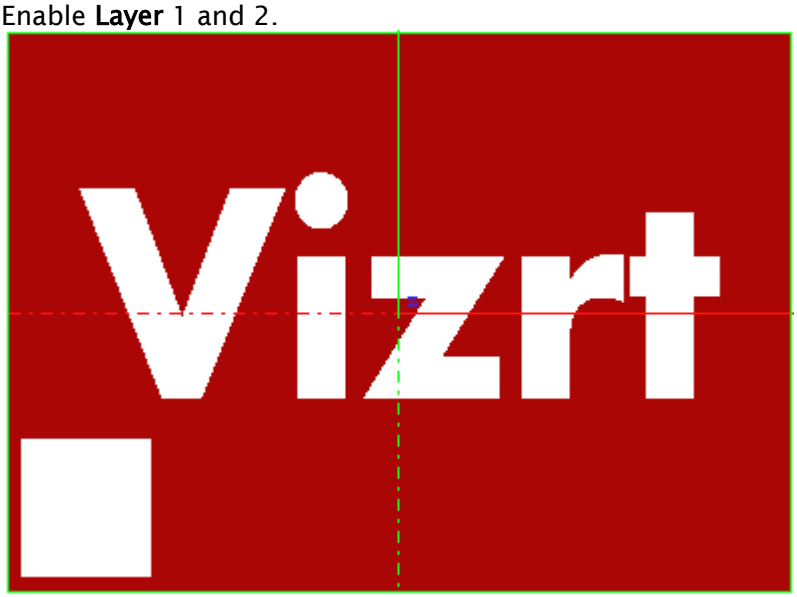
- **Layer:** Sets the layer the mask should have an effect on. This allows the creation of more than one mask that will have an effect on the target

- **Inherited**: (Mask Source Only) When enabled (On), all Sub-Containers of the mask source will mask the target. If disabled (Off) all Sub-Containers of the mask source will not mask the target as it no longer inherits the mask properties
- **Force Visible**: (Mask Source Only) When enabled (On), the geometry serving as mask source will be rendered in the scene editor, even when not selected. When disabled (Off), the mask source geometry will only be rendered when selected. This enables the designer to work more efficiently with multiple mask sources when designing the scene. The mask source geometry never gets rendered on the output.

To Add a Mask

1. Add three Containers to the Scene Tree.
2. Name them **source_1**, **source_2** and **target.**
3. Add a Font and the Mask Source plug-in to the **source_1** Container.
4. Add a Rectangle and the Mask Source plug-in to the source_2 group.
5. Add a Rectangle, material (e.g. red) and the Mask Target plug-in to the **target** Container.
6. Open the **source_1**Transformation Editor:
    a. Set **Axis Center X** and **Y** to C
    b. Reset **Position X** and **Y** to 0.0.
7. Open the **source_1**Text Editor:
    a. Enter some text (e.g. Vizrt).
8. Open the **source_1** Mask Source editor:
    a. Set **Layer** to 1.
9. Open the **source_2**Transformation Editor:
    a. Set **Position X** to -120.0 and **Y** to -70.0.
10. Open the **source_2** Rectangle editor:
    a. Set **Width** and **Height** to 50.0.
11. Open the **source_2** Mask Source editor:
    a. Set **Layer** to 2.
12. Open the **target** Rectangle editor:
    a. Set **Width** to 300.0 and **Height** to 200.0.
13. Open the **target** Mask Target editor

a. Enable **Layer** 1 and 2.



14. Add a Container as a Sub-Container of **source_2**.
15. Name it **source_inherited.**
16. Add a Circle to the **source_inherited** Container.
17. Open the **source_inherited**Transformation Editor:
    a. Set **Position X** to 240.0.
18. Open the **source_inherited** Circle editor:
    a. Set **Radius** to 25.0.

19. Open the **source_2** Mask Source editor
    a. Disable **Inherited** (Off).
    - This should make the Circle disappear as it no longer inherits the Mask Source properties.

14.7.11 Lighting

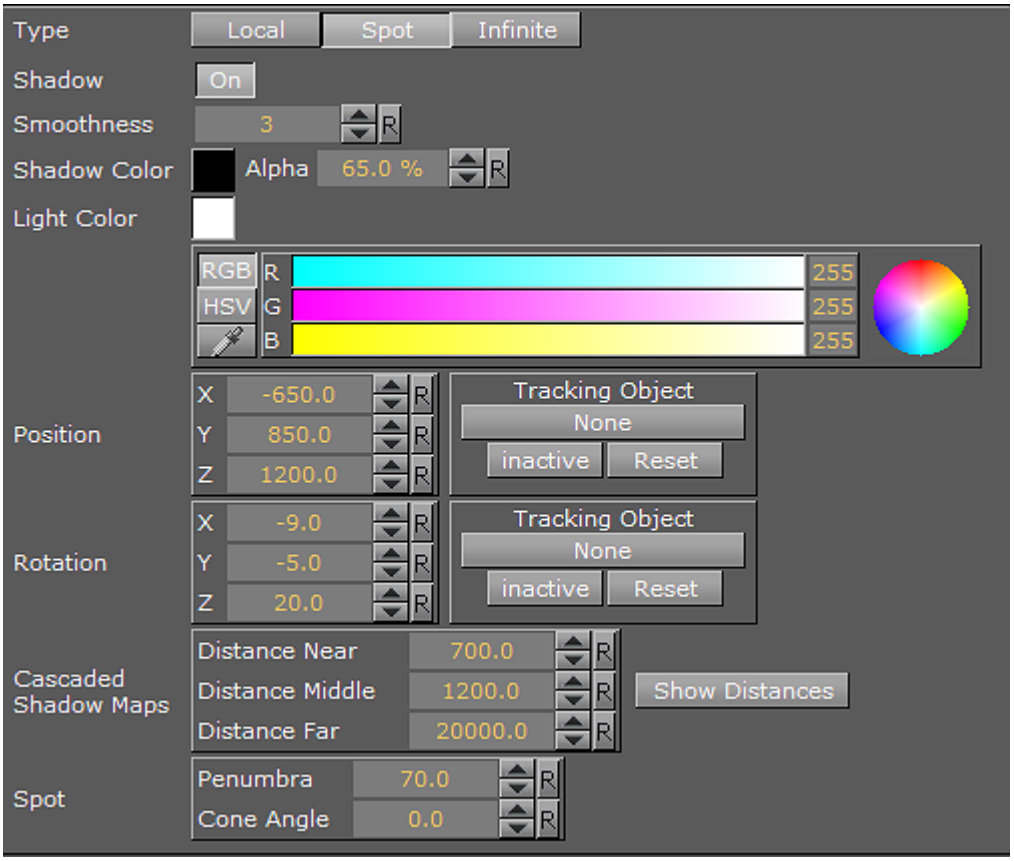
Adds individual lighting per object.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

- Lighting Properties
- To Add Light

Lighting Properties



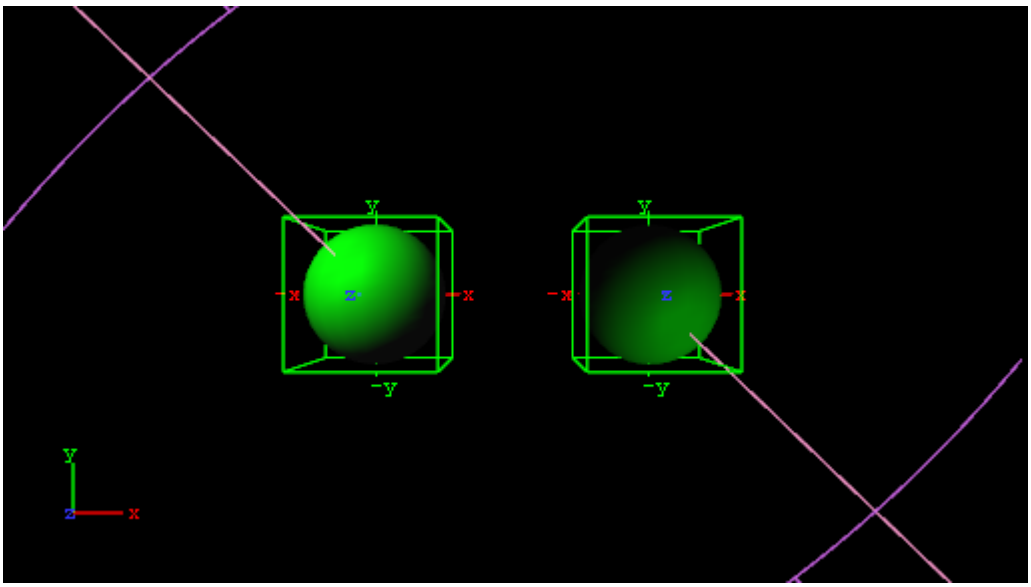
- **Type:** Determines the light source type:
  - **Local**: The Local light source is a positional light. It is near or within the scene, and the direction of its rays is taken into account in lighting calculations. Local lights have a greater performance cost than directional lights, due to the additional calculations. A

real life equivalent to a local light source is a light bulb. The Local light source has properties for color and position.

- **Spot:** The Spot light source emits a cone of light. The only objects that are illuminated are those within the cone. The Spot light source has properties for color, position, rotation, and attenuation.
- **Infinite:** The Infinite light source is a directional light. It is considered to be an infinite distance away from the objects in the scene. Because of the distance, the rays of light are considered parallel by the time they reach the object. A real life equivalent to an infinite light source is sunlight. The Infinite light source has properties for color, position, and rotation. Infinite is selected by default.

- **Shadow**: Turns shadows on or off.
- **Smoothness**: Affects how shadows blend. Value from 1 to 10. The smoother, the more blended (soft) the shadows will be.
- **Shadow Color**: Selects the color and alpha value of the shadow.
- **Color:** Sets the light source color.
- **Position:** Sets the position of the light source along the X, Y, and Z axis.
- **Rotation:** Sets the values for Pan, Tilt, and Twist for Spot or Infinite light sources.
- **Cascaded Shadow Mask**: Determines the frustum distances affected by shadows, i.e. the three-dimensional region with shadow effects specified as Near, Middle and Far regions.
- **Spot:** Sets the concentration of the light within the light cone of the Spot light source. When set to zero, the whole light cone has the same intensity. If set greater than zero, the intensity decreases away from the centre.

To Add Light



1. Open the Light Editor, and disable global light settings.
2. Add a Sphere geometry to the scene tree, and add material and the Lighting plug-in to it.
3. Duplicate the Sphere container and place it at the same level (root) as the other container.
4. Open the transformation editor and set Position X to -100.0 for the first container and Position X to 100.0 for the second.

5. Open the Lighting editors the first container one and set the following parameters:
    - Set Type to Spot.
    - Set Light Color to 0,255,0.
6. Open the Lighting editors the second container one and set the following parameters:
    - Set Type to Spot.
    - Set Light Color to 0,255,0.
    - Set Position X and Y to 650.0 and -850.0, respectively.
    - Set Rotation X and Y to -650.0 and 850.0, respectively.

See Also

- Shadow Maps

14.7.12 Z-Sort

When this plug-in is applied to a group, it disables back to front automatic sorting for the immediate children of that group. This enables designers to set the rendering order manually.

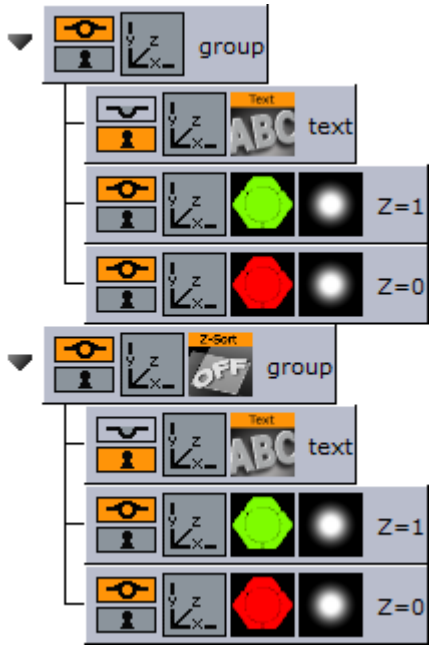
> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

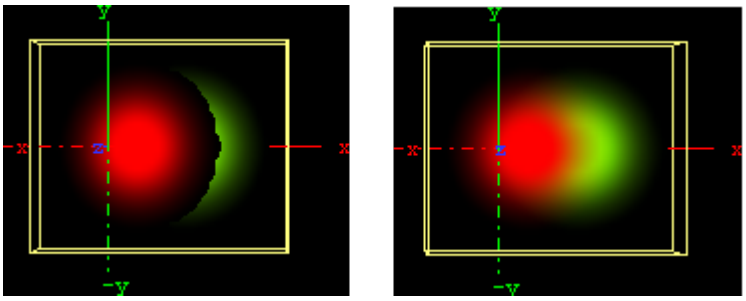
- Z-Sort Description
- Transparency, Rendering Order Logic and Priority

Z-Sort Description

To sort objects of a Scene on the Z-axis is essential for any 3D program to render correctly. The renderer starts by rendering the object furthest behind, and then mixing in objects closer and closer to the camera. Sometimes the machine gets “confused” with what should be rendered first and last because it calculates this from what is the center of the object’s Z position.



A typical example of a Z sort problem is when two colored objects have rotated their Y axis to for example -12.0. In this case the default behavior for a render engine is to render the first object (green) with a Z position 0.0 first, and then the second object (red) with Z position 1.0 last.



However, when objects are rotated the output will be wrong because the red object should be rendered before the green due to the rotation of the objects. This can be corrected by adding the Z-Sort plug-in, which lets users manually sort objects using the logic of the tree structure. Hence, what is placed at the top of a group in the scene tree gets rendered first, and what is placed below will be rendered after.

Sorting is solved by rendering the second (red) object first, and then the first (green) last. Be aware that although the red object has a Z position of 1.0 and the green 0.0, the red object must be rendered before the green because of the overlapping rotation (Y -12.0). Normally objects that are furthest behind should be placed at the top in the scene tree.

The Z-Sort plug-in disables the automatic Z-sort for a Container. How the items in the Container get sorted in the Z-buffer is then dictated by the order of the objects in the scene tree.

Transparency, Rendering Order Logic and Priority

Questions:

- Why does a scene look different when a parent container is set to 99% transparency?
- Why are do some Containers disappear?

- When should the Z-Sort Off Plug-In be used?
  Explanation:

The explanation to why changing an alpha plug-in to 99% on a parent container, which affects the way objects are rendered, is the Rendering Order. When rendering polygonal Scenes with transparencies (like those created in Viz Artist) the rendering order is important with the current architecture of graphics cards.

The common practice for 3D applications is to render semi-transparent objects from back to front (or rather from far to near).

⚠ **Example:** Think of a scene that shows a city skyline through of a semi-transparent window. Lets assume that the window is rendered **before** the skyline. Some background color is applied to the screen, then the window is rendered. The graphics card will blend the color of the window with the background color (say, blue). In addition to color the graphics card will also mark the distance of the window's pixels from the camera for each pixel, so that farther away pixels will not obstruct the window. This technique for achieving obstruction between objects on per-pixel basis is called Depth-Buffer or Z-Buffer. Now render the skyline. The color of the skyline should have been blended with the color of the window, but it will not. The graphics card will reject every skyline pixel because its distance from the camera is farther away than the distance of the window pixels. This is a case where the positive role of the Depth-Buffer produces undesirable results.

To avoid this undesirable result, Viz Artist splits the rendering of the containers in the Scene into three parts:

1. The mask objects are rendered first to 'mark' the areas of masking on the screen.
2. The opaque objects are rendered front to back.
3. The transparent objects are rendered back to front.
   Viz Artist/Engine decides whether an object is transparent based on some criteria, for example:

- If the texture of the object has an alpha channel
- If the material of the object has an alpha value less than 100%
- If the object has an inherited ALPHA function with an alpha value less than 100
- and more...

  With all this said, rendering back to front is not perfect because the rendering order is not done per pixel; rendering is not even done per polygon. Viz Artist/Engine renders a scene per complete container; container order is based on the center of the container.

⚠ **Example:** Take a tilted flat map with icons. The map container is large, and its center is in front of some of the icons, but in the back of other icons. Assuming the icons have transparency the map will actually be rendered in between the icons! This behavior is visually incorrect as the map should be rendered first, then all the icons.

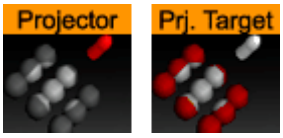
Solution

This behavior can be fine-tuned using the Z-Sort Off Plug-In. When this plug-in is applied to a group, it will disable back to front automatic sorting for the immediate children of that group. This enables designers to set the rendering order manually.

Notes/Exceptions

- If an alpha plug-in at 99% is applied on the top container it can force the whole collection of containers to be sorted back to front, and can solve the problem, but this is not a best practice. There is much more overhead rendering a scene like this. A more permanent solution would be to optimize the scene with the Z-Sort Off Plug-In.
- Opaque objects are better rendered from front to back because there is some performance gain for each obstructed pixel (it is not being processed after being rejected). With that being said opaque objects can be rendered at any order.

14.7.13 Projector Source and Projector Target



The projector and the projector target is used to project something on something else (e.g. an image onto a geometry).

A projector container creates a planar projection of their texture onto container(s) with the projection target function attached. In principle it works similar to a slide-projector. For objects that do not have texture coordinates, the projector function can be used to apply texture to such objects. It also gives the possibility to have two textures on a single object, the objects own texture and the projected texture.

The projector and the projection target can be set up to react on different layers similar to shadows and masks.

> ⚠ **Note:** The projector plug-in does not work on text.

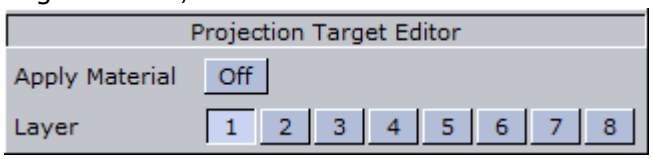
> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

- Projector Source and Projector Target Properties
- To Project a Texture onto an Object

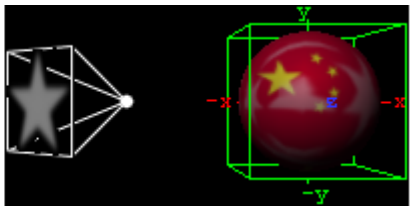
Projector Source and Projector Target Properties

- **fovy:** Sets the field of view for the projector container. The higher the fovy is set, the larger the projected texture on the receiver will be. You can picture it as the zoom function of a slide projector.
- **Show**: Allows you to hide/show the projector object.
- **Layer**: Lets you select by which layers the projector and receiver are to perform the function. You can use up to 8 different projector layers and source layers for one object. This means that targets with the layer 1 set will be affected by all projectors which have the corresponded layer selected. If a projector has the layers 1, 2 and 3 selected it will affect all targets with 1, 2 or 3 selected.



- **Layer:** Lets you select by which layers the projector and receiver are to perform the function. You can use up to 8 different projector layers and source layers for one object. This means that targets with the layer 1 set will be affected by all projectors which have the corresponded layer selected. If a projector has the layers 1, 2 and 3 selected it will affect all targets with 1, 2 or 3 selected.

To Project a Texture onto an Object



1. Add a group container to the scene tree, add the Projector and Alpha plug-in to it, an image/texture, and name it **Source**.
2. Open the Alpha editor and set the alpha value to 50.0%.
3. Open the transformation editor for the Source container and set Rotation Y to 135.0.
4. Add another group to the scene tree, and name it **Target**.
5. Open the transformation editor for the Target container, and set Position X to 130.0.
6. Add the Sphere geometry plug-in, material, image/texture and the Projector Target plug-in to the Target container.
7. *Optional*: Animate the sphere.
8. Open the Projector editor and **disable** the **Show** property.

14.7.14 Script

The script functionality is a plug-in. Scripts can be added to a Container and stored with the Scene. It is possible to archive and import scripts as part of a Scene Archive as with any other plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

A script can also be saved as its own script plug-in, and used in future Scenes (see Create Script-based plug-ins). To save a script as a script plug-in, drag a compiled script into the Script plug-ins folder. The Script plug-ins folder is a special folder that lets users create and name plug-ins.

> ⚠ **Note:** Script plug-ins are saved to *<viz data folder>\Scriptplug-ins*.

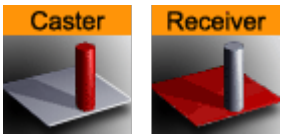
> ⓘ **WARNING**: A script plug-in is not stored in the database, and is not distributed with an archive, because it is code and not data.

There is an important difference between a script that is not converted into a plug-in, and a script that is. As long as it is not converted the content of the script code is data within the 'Script' plug-in, just like the width property is data within for example a geometry plug-in. Once the script is converted to a plug-in it is a plug-in itself, like any other plug-in.

See Also

- Script Editor
- Script plug-ins

14.7.15 Shadow Caster and Shadow Receiver



The Shadow Caster and Receiver plug-ins are used to enable an object's shadow to reflect on another object. To create a shadow item in a scene, two containers at a minimum must be used. The caster plug-in must be attached to one container and the receiver function to the other.

The container acting as the caster must be positioned between the receiver container and a light source. For correct lighting calculation, a material is also required on the container.

Objects that can function as a shadow receiver should be a built in 2D geometry or a font. An imported 2D will not work.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

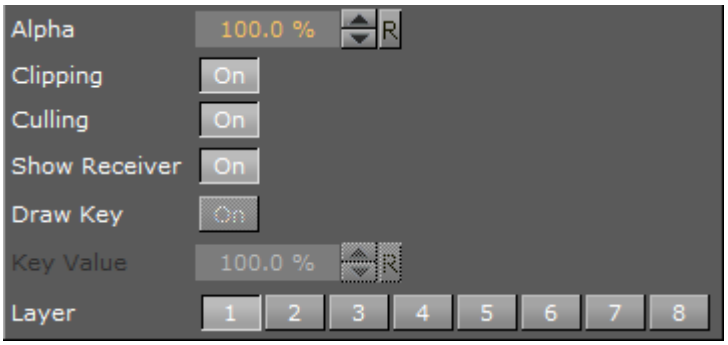
This section also contains information on the following topics:

- Shadow Caster Properties
- Shadow Receiver Properties
- To Cast a Shadow
- Shadow Receiver Known Limitations

Shadow Caster Properties

- **Lights:** Allows you to select which light sources are to shine on the caster object and thereby producing the shadow/shadows.
- **Alpha:** Sets the alpha value and hence the transparency of the shadow being cast on the receiver objects.
- **Color:** Allows you to define the shadows color.
- **Auto-Fading:** With this option enabled, if you fade out a caster object either by using the alpha function or by adding a material with a low alpha value, the shadows created by the caster will fade correspondingly. The option is enabled by default, since it looks right that the shadow fades out when the caster object fades out.
- **Layer:** Allows you to select by which layer(s) the caster will cast shadow. You can select up to 8 layers. Each layer corresponds to the layer selected in the Receiver objects. If you want all shadow casters to cast shadow on all receivers, you can set all layers on all casters and receivers to 1. If you want one caster to create shadow on some receivers and another caster to create shadow on different receivers, you select different layers for the sets of casters and corresponding receivers. Casters and receivers can have multiple layers enabled at the same time so you can set up a great number of combinations of caster/receiver combinations.

Shadow Receiver Properties



- **Alpha:** Sets the alpha value.
- **Clipping:** With clipping enabled the shadow will be cut on the border of the object. With clipping disabled, the shadow will be rendered onto the background outside the object, which normally looks wrong. Rendering with the clipping options set takes more resources than without. If you for instance have an infinite floor or a room with walls, you can switch

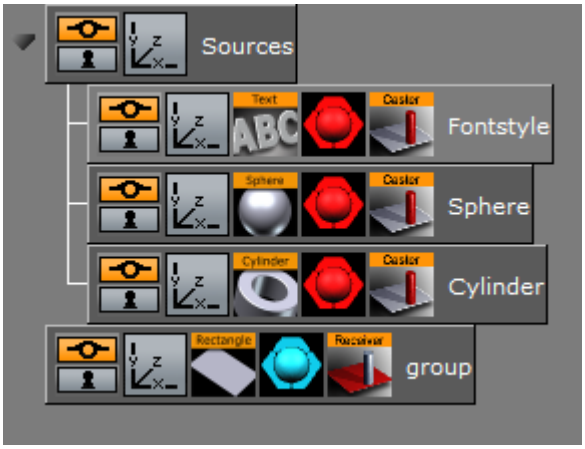
of clipping without seeing the “false” shadows, and thereby save some rendering performance.

- **Culling:** With culling enabled, Viz Artist will decide in advance of the rendering process if the shadow will be visible or if it will be hidden by other objects. If that is the case it will skip it in the render process. If you know for sure that the shadow for an object in a scene will be rendered visible for the whole object, you can disable culling to save some rendering time, but the effect is marginal.
- **Show Receiver**: Hides or shows the receiver object. If it is hidden, the shadow will still be visible.
- **Draw Key:** Adds key to the shadow. When enabled, the receiver must be hidden for the key to be drawn.
- **Key Value:** Sets the key value.

> ⚠ **Note:** The Key Value setting is not available when Shadow Type in the Global Settings Panel is set to Shadow Map.

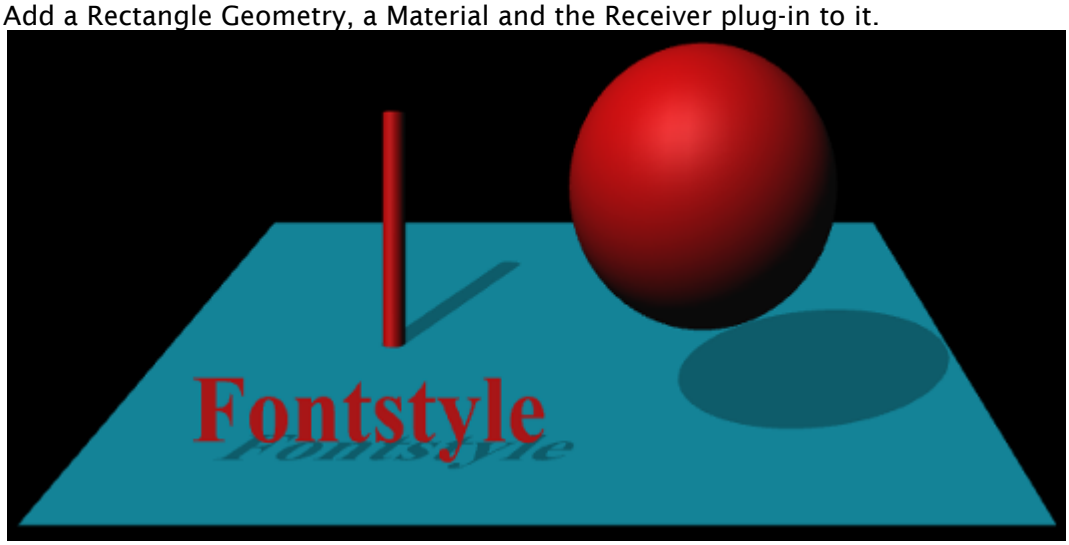
- **Layer:** Allows you to select by which layers the receiver will receive shadow. You can select up to 8 layers. Each layer corresponds to the layer selected in the caster objects. If you want all shadow casters to cast shadow on all receivers, you can set all layers on all casters and receivers to 1. If you want one caster to create shadow on some receivers and another caster to create shadow on different receivers, you select different layers for the sets of casters and corresponding receivers. Casters and receivers can have multiple layers enabled at the same time so you can set up a great number of combinations of caster- and receiver combinations.

To Cast a Shadow



1. Add a group container to the Scene Tree
2. Name it **Sources**.
3. Add a Fontstyle and the Sphere and Cylinder Geometries as Sub-Containers of Sources.
4. Add a Material and the Shadow Caster plug-in to each Sub-Container.
5. Add a Group Container at the same level as the **Sources** Container

6. Add a Rectangle Geometry, a Material and the Receiver plug-in to it.



7. Position the Rectangle and the other objects such that the Sphere, Cylinder and text casts a shadow on the rectangular surface.

> ⚠ **Caution:** Containers are rendered twice when shadows are used.

Shadow Receiver Known Limitations

With shadows in Stencil Mode, the current implementation of the shadow receiver assumes that the receiving surface is flat and a standard geometry object, such as a rectangle, orientated on the XY plane. Non-flat geometries, and planes which are not extending in the XY plane, will not produce a properly casted shadow. However, by activating Shadow Map for the desired light source, shadows will render correct. Always make sure that the receiving object has a material.

See Also

- Shadow Maps

14.7.16 Synchronized Properties

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

The Synchronized Properties plug-in can be used to synchronize container property changes within the same scene loaded on a cluster of Viz Engines. Synchronization needs to be configured for each Viz Engine taking part in the clustered setup. This plug-in works on a per container basis, which keeps network traffic low. Child containers do not inherit the plug-in, and requires their own

instance of the plug-in if they are to be synchronized. A scene plug-in is available for use-cases where the entire scene is to be synchronized.

⚠ **Example:** A renderer with touch screen controls a stereoscopic interactive scene which is rendered on 2 other Engines. All 3 Engines need to perform all property or transformation changes at the same time.

This feature works for all container properties and plug-in parameters, and requires that the Engines render the same scene, referencing the same UUIDs. This allows designers to work with different versions of the same scene, without any need to update the changed IDs in external applications. If a parameter change is transferred to another Engine, then the container UUID of the last loaded scene will be used for the update. If new UUIDs are required, a new scene will have to be created, either from scratch, or by merging the complete scene into a container which is then split into a new scene.

To Synchronize Plug-in Property Changes on Multiple Viz Engines

- Configure the involved Engines for synchronization. Please refer to the **Global Input** page in the **Configuring Viz** section of the Viz Engine Administrator Guide for further information.
- Create the scene that should be synchronized.
- Add the **Synchronize Properties** plug-in to those containers that should be synchronized. No scripting is required for the synchronization.

See Also

- Scene Synchronized Properties plug-in
- For more information about how to synchronize multiple Viz Engines, see the **Global Input** page in the **Configuring Viz** section of the Viz Engine Administrator Guide.

14.7.17 Video Clip

The Video Clip plug-in works in combination with the Control Video plug-in. It allows you to use clip Key Frames in the stage for clip playback. When used with Transition Logic, the Key Frames can be merged with the object.

When Video Clip is used in Transition Logic scenes, it allows a Toggle between the two clip channels. A single object will playback any clip. In a Transition Logic scene, the Toggle plug-in will make sure that the channels toggle correctly, which allows for transitions between two running clips. The plug-in also toggles the clip texture if texture mode is selected.

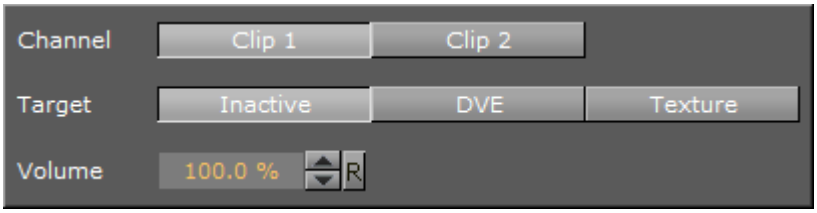
The plug-in is commonly used in a video workflow with Viz One. It automatically adds a Clip Channel plug-in. The clip channel has to be activated using the **Video Input** in Viz Configuration (see the Viz Engine Administrator Guide).

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

- Video Clip Properties
- To Add a Video Clip

Video Clip Properties



- **Channel:** Sets the clip channel to be used. Available options are; Clip channel 1 and 2.
- **Target:** Sets the clip mode. Setting target to DVE adds default Key Frames to the stage. Setting target to Texture adds the **Live Video** Media Asset texture plug-in and default Key Frames to the stage. Available options are; Inactive, DVE or Texture.
- **Volume:** Sets the clip volume in percentage.

To Add a Video Clip

1. Check the Viz Configuration to see if clip channels are activated.
2. Add the Video Clip plug-in to a geometry (e.g. a rectangle).
3. Select Clip Channel 1 and Type Texture.
4. Switch to the Stage and select the Clip under VideoClip.
    - On the right side the clip to be played can be selected.
    - Additional Key Frames can be added to play different clips.

See Also

- Video Clips
- The Stage for Animation
- Toggle
- Control Video
- Transition Logic

14.7.18 Window Mask

Window Mask limits the area into which a Sub-Container of the Window Mask plug-in is rendered to a user-defined rectangular section of the output.

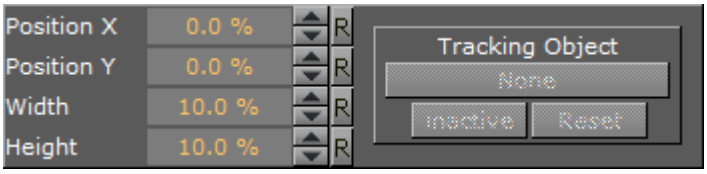
It is possible to add a tracking object. If a tracking object is used, it's bounding box is projected onto the screen and the resulting rectangle is used as a mask.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

This section also contains information on the following topics:

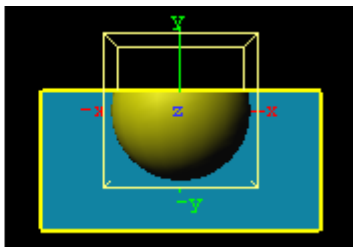
- Window Mask Properties
- To Use the Window Mask

Window Mask Properties



- **Position X**, **Position Y:** Specifies the upper left corner of the rectangle. The values are scaled in such a way that (0, 0) refers to the upper left and (100, 100) to the bottom right corner of the screen.
- **Width**, **Height**: Specifies the dimensions of the rectangle in percentages of the screen dimensions.
- **Tracking Object:** Provides an alternative to setting the position and dimensions manually. When you drag a container from the scene tree onto this drop zone, the rectangle will automatically be adjusted to match the position and size of that container. **Inactive** will toggle tracking on and off. To remove the container click on **Reset**.

To Use the Window Mask



1. Add a Rectangle geometry to the scene tree, and add material to it.
2. Open the Rectangle editor and set Width to 200.0.
3. Open the transformation editor and set Position Y to -36.
4. Add a new group, and add the Window Mask plug-in to it.
5. Add a Sphere geometry as a Sub-Container of the group container, and add material to it.
6. Open the Window Mask editor and drag and drop the Rectangle container onto the Tracking Object drop zone.
    - This should give you a sphere that is partly masked due to the rectangle object.

14.8 Lineup

Lineup container plug-ins are special plug-ins used to allow the Lineup template in Viz Pilot to collect information about the scene hierarchy.

The following Container plug-ins are located in the Lineup folder:

- Tree Props

14.8.1 Tree Props

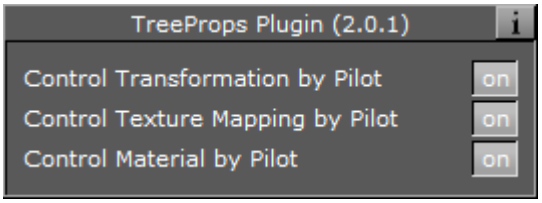
The Tree Props container plug-in is required for use of the Tree Status plug-in, needed by the Lineup template in Viz Pilot to collect information about the scene hierarchy. Move the Tree Props plug-in onto the group holding the transformation which is to be controlled by Viz Pilot. Viz Pilot will then be able to build its own internal tree properties list.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Lineup

This section also contains information on the following:

- Tree Props Properties

Tree Props Properties



- **Control Transformation by Pilot:** Allows Viz Artist control the transformation properties of the container holding the plug-in when enabled.
- **Control Texture Mapping by Pilot:** Allows Viz Pilot control the Texture Mapping of the container holding the plug-in when enabled.
- **Control Material by Pilot:** Allows Viz Pilot control the material on the container when enabled.

14.9 MultiTouch Plug-Ins

The MultiTouch plug-ins are used to take incoming touch events, process them and then trigger events, perform transformations on objects, telestrate, or trigger script events. The idea behind these plug-ins is to bundle often used or needed functionality and eliminate the need to write your

own scripts or plug-ins for that. They are great tools to bring MultiTouch interactively fast and easily into a Scene.

The Container plug-ins require the MtSensor plug-in to be available to work. If a MultiTouch plug-in is added to a Scene it will automatically add the MtSensor plug-in to the Scene if it is not there.

The MultiTouch plug-ins can communicate with Viz scripts, either through event handlers, Shared Memory variables, or Data Pool variables. The plug-ins can also be controlled by triggering the 'push buttons' and/or setting the 'parameter' variables exposed by the plug-in. Specifics on what features are available for each plug-in are documented in the plug-in specific sections below.

In the Viz Artist editor interface the 'E' button toggles between enabling and disabling emulated sensor input using the mouse as well as direct input from a sensor.

The plug-ins will work with multiple layers. Each Scene layer with MultiTouch objects installed in it should also have an instance of the MtSensor plug-in automatically installed.

The following Container plug-ins are located in the MultiTouchComp folder:

- Mt2D Control plug-in
- MtButton plug-in
- MtNavigator plug-in
- MtTelestrator plug-in
- Plug-in Event and Notification System
- Mt3D Control plug-in

Make sure you also read up on the Plug-in Event and Notification System.

14.9.1 Mt2D Control plug-in

The Mt2D Control plug-in moves, scales, and rotates an object in 2D.

The Mt2D Control plug allows multi-finger 2D manipulation of an object in the scene. When the object is touched, it will translate, rotate, and scale to keep the object underneath the user's fingers as closely as possible. It will move within the screen relative 2D plane through the object's center position defined by the objects orientation axes.

The container being manipulated must not have any local non-uniform scale. The plug-in will force y and z scale so they are equal to x scale when the object is touched. If non uniform scaling is required on object in the container, put the object's container in a group and scale the parent. Or put the plug-in on an identity group and insert the object to be moved as a child container. All transformations set on the manipulated container's parents will be preserved.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> MultiTouchComp

This section contains information on the following topics:

- Mt2D Control Configuration
- Events and Notifications
- To Create a Simple Scene with the Mt2D Control plug-in

Mt2D Control Configuration

- **Momentum:** Enables movement or rotation with momentum after the object is released.
  - **Friction:** Sets the amount of friction used when momentum is enabled.

    ⚠ **Note:** Using this feature will conflict with any externally scripted or other animations applied to the container.

- **Lock Translation:** Allows only rotation and scaling of the object.
- **Lock Translation (Screen X):** Prevents movement left-to-right on the screen.
- **Lock Translation (Screen Y):** Prevents movement up-and-down on the screen.
- **Lock Translation (Object X):** Prevents movement along the object's local X axis.
- **Lock Translation (Object Y):** Prevents movement along the object's local Y axis.
- **Lock Rotation:** Prevents rotation of the object.
- **Lock Scale:** Prevents resizing of the object.
- **Limit Position:** Limits the maximum and minimum position of the X and Y axis.
- **Limit Rotation:** Limits the maximum and minimum rotation of the X and Y axis.
- **Limit Scale:** Limits the maximum and minimum scale.
- **Id:** Gives additional context in the handler script, specify a string that will identify any notifications dispatched by this plug-in. This is often included as an argument for the event so a common script may handle events from a number of plug-ins.
- **Set Shared Memory:** Enables shared memory to be updated for the plug-in notifications when set to **On**.
  - **Shared Memory Prefix:** This sets a 'prefix name' to be prepended to the shared memory variables maintained by the plug-ins notifications. For plug-ins that maintain multiple fields each field name has the prefix prepended to it followed by a dot '.' so as to mimic member access to an object, i.e: if the prefix is 'Obj' the fields 'field1' and 'field2' would be identified with the strings 'Obj.field1' and 'Obj.field2'. The shared memory field 'Obj' is also maintained and is simply an integer that is modified every time any of its 'subfields' is updated.
  - **Shared Memory Type:** Click either Global, Scene, or Distributed. This selects the shared memory area to update.
- **Set Data Pool:** Shows 'you wish' plug-in notifications to set a Data Pool variable.
  - **Data Pool Variable:** Shows the name of the Data Pool variable to have set.
- **Active during screen wide telestration?**: If the button to is to stay active when the screen-wide telestrator is on, set to On. The default is off, which means the button cannot be triggered when telestration is on.
- **Hit Coordinate Type**: Selects the Coordinate system to use.Click either Local, World or Screen.
- **Produce delta values, don't move object:** Does not alter transform of object with attached controller, but instead provide 'deltas' in the event notifications.

> ⚠ **Note:** Rotation scale or position constraints do not operate in this mode.

Events and Notifications

The events dispatched by Mt2D Control are:

OnGrab (when object is 'grabbed'):

sub **ONGrab** (**HitContainer as String, Id as String, HitPoint as Vertex, plug-inContainer as String, FirstChildContainer as String**)

end sub

- **HitContainer as String**
- **Id as String**
- **HitPoint as Vertex**
- **plug-inContainer as String**
- **FirstChildContainer as String**

OnGrabUpdate (when object is moved):

sub **OnGrabUpdate** (**HitContainer as String, Id as String, Position as Vertex, Rotation as Vertex, Scale as Vertex, Pressure as Double**)

end sub

- **HitContainer as String**
- **Id as String**
- **Position as Vertex**
- **Rotation as Vertex**
- **Scale as Vertex**
- **Pressure as Double**

OnAnimation (dispatched when post grab animation (momentum) is started, followed by OnGrabUpdates):

sub **OnAnimation** (**HitContainer as String, Id as String**)

end sub

- **HitContainer as String**
- **Id as String**

OnGrabRelease (when both touch grab and momentum animation are complete):

sub **OnAnimation** (**HitContainer as String, Id as String**)

end sub

- **HitContainer as String**
- **Id as String**

The Shared Memory Field names updated are:

- **HitContainer** (the name of the container that was 'hit')
- **Id** (the user ID string entered in the notification interface)

- **HitPoint**
- **plug-inContainer**
- **FirstChildContainer**
- **Position**
- **Rotation**
- **Scale**
- **Pressure**
- **State**

The Data Pool structure updated is:

- **Mt2D Control:**
  - string HitContainer
  - string Id
  - string HitPoint
  - string plug-inContainer
  - string FirstChildContainer
  - string Position
  - string Rotation
  - string Scale
  - string Pressure
  - string State

To Create a Simple Scene with the Mt2D Control plug-in

⚠ **Note:** If a touch screen is not available, set **Multi Touch** input to **Mouse** in the **Communication** section of Viz Configuration (see the Viz Engine Administrator Guide).

1. Create a new Scene.
2. Add a new group Container.
3. Add a Cube Geometry to the Container.
4. Add the Mt2DControl plug-in to the Container.
5. Open the Scene in **On Air**. Move, rotate or scale the Rectangle with the touch input device.
6. Use the Mt2DControl plug-in editor to modify the plug-in behavior as required.

If ON

14.9.2 MtButton plug-in

The MtButton plug-in triggers an action when tapped or pressed.

The MtButton plug-in will make a container or its immediate children act as buttons. Several methods may be used to trigger the buttons. They are selected in the Viz Artist UI with the 'Trigger Type' radio button.

The MtButton can be enabled or disabled. When the button is disabled, any tap or stroke input it receives will not trigger the associated events. The button can be enabled or disabled through the

plug-in parameters, or through a command. A button which is disabled after a stroke has pressed on a button will NOT prevent generation of events for a stroke that is currently active.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> MultiTouchComp

This section contains information on the following topics:

- MtButton Configuration
- Events and Notifications
- To Create a Simple Scene with the MtButton plug-in

MtButton Configuration

- **Id:** Gives additional context in the handler script, specify a string that will identify any notifications dispatched by this plug-in. This is often included as an argument for the event so a common script may handle events from a number of plug-ins.
- **Set Shared Memory:** Enables shared memory to be updated for the plug-in notifications when set to **On**.
  - **Shared Memory Prefix:** This sets a 'prefix name' to be prepended to the shared memory variables maintained by the plug-ins notifications. For plug-ins that maintain multiple fields each field name has the prefix prepended to it followed by a dot '.' so as to mimic member access to an object, i.e: if the prefix is 'Obj' the fields 'field1' and 'field2' would be identified with the strings 'Obj.field1' and 'Obj.field2'. The shared memory field 'Obj' is also maintained and is simply an integer that is modified every time any of its 'subfields' is updated.
  - **Shared Memory Type:** Click either Global, Scene, or Distributed. This selects the shared memory area to update.
- **Set Data Pool:** Shows 'you wish' plug-in notifications to set a Data Pool variable.
  - **Data Pool Variable:** Shows the name of the Data Pool variable to have set.
- **Active during screen wide telestration?**: If the button to is to stay active when the screen-wide telestrator is on, set to On. The default is off, which means the button cannot be triggered when telestration is on.
- **Trigger type:**
  - **On Touch:** Sets the buttons to generate an 'onTrigger' event immediately when they are touched. No other events are generated in this mode.
  - **On Release**: Sets the buttons to generate an 'onTrigger' event when the last stroke present over a button is released. onStrokeEnter() and onStrokeLeave() events are generated as strokes enter and leave the button (see protocol below).
  - **On Tap:** Sets the buttons to generate an 'onTrigger' event when strokes recognized as 'taps' occur after a button is quickly pressed and released without movement or a finger is held down on a key on an object. onStrokeEnter() and onStrokeLeave() events are also generated in this mode.

    - **Taps Only:** Sets the buttons to emit an 'onTap' event when stokes recognized as a 'taps' occur. No other events are generated in this mode.
- **Hit Coordinate Type**: Selects the Coordinate system to use. Click either Local, World or Screen.

Events and Notifications

The MtButton generates an **OnTap** event when it is tapped in 'Taps Only' mode:

sub OnTap(**HitContainer as String, Id as StringHit, Point as VertexTaps as Integer, plug-inContainer as String, FirstChildContainer as String**)

end sub

- **HitContainer as String:** the name of the container that was 'hit'
- **Id as String:** the user ID string entered in the notification interface
- **HitPoint as Vertex:** the Point where the object was tapped in local coordinates
- **Taps as Integer:** the number of taps ie: 2 == double tap
- **plug-inContainer as String:** name of the container the plug-in is attached to
- **FirstChildContainer as String:** direct (first level) child container of the plug-in, that is a (indirect or direct) parent of the tapped container. If the plug-in container itself is tapped, the plug-in container is returned.

The MtButton generates an **OnTrigger** event when it is triggered in any other mode than 'Taps Only':

sub OnTrigger(**HitContainer as String, Id as String, HitPoint as Vertex, Taps as Integer, plug-inContainer as String, FirstChildContainer as String, StrokeId as Integer**)

end sub

- **HitContainer as String:** the name of the leaf container that was 'hit'
- **Id as String:** the user ID string entered in the notification interface
- **HitPoint as Vertex:** the Point where the object was tapped in local coordinates
- **Taps as Integer:** the number of taps if tapped, ie: 2 == double tap
- **plug-inContainer as String:** name of the container the plug-in is attached to
- **FirstChildContainer as String:** first child container below the plug-in the trigger occurred on or below
- **StrokeId as Integer:** Unique ID of stroke causing this event.

The MtButton generates an **OnStrokeEnter** event when it is triggered in any other mode than 'Taps Only':

sub OnStrokeEnter(**Id as String, plug-inContainer as String, FirstChildContainer as String, StrokeId as Integer, StrokeCount as Integer, Active as Integer**)

end sub

- **Id as String**: the user ID string entered in the notification interface
- **plug-inContainer as String:** name of the container the plug-in is attached to
- **FirstChildContainer as String:** first child container below the plug-in the stroke entered.
- **StrokeId as Integer:** Unique ID of stroke causing this event. Only valid from initial enter to final leave.

- **StrokeCount as Integer:** count of stroke on the button after entry. Will be ‘1’ when the first stroke enters
- **Active as Integer:** This parameter is deprecated and will always return '0'.

The MtButton generates an **OnStrokeLeave** event when it is triggered in any other mode than ‘Taps Only’:

sub OnLeave(**Id as String, plug-inContainer as String, FirstChildContainer as String, StrokeId as Integer, StrokeCount as Integer, Active as Integer**)

end sub

- **Id as String:** the user ID string entered in the notification interface
- **plug-inContainer as String:** name of the container the plug-in is attached to
- **FirstChildContainer as String:** first child container below the plug-in the stroke exited.
- **StrokeId as Integer:** Unique ID of stroke causing this event. Only valid from initial enter to final leave.
- **StrokeCount as Integer:** count of stroke on the button after exit. Will be ‘0’ when the last stroke exits
- **Active as Integer:** This parameter is deprecated and will always return '0'.

The shared memory field names are:

- **HitContainer:** the name of the container that was ‘hit’
- **Id:** the user ID string entered in the notification interface
- **HitPoint:** the Point where the object was tapped in the selected coordinate type.
- **Taps:** the number of taps, ie: 2 = double tap
- **plug-inContainer:** name of the container the plug-in is attached to
- **FirstChildContainer:** first child container below the plug-in that tap occurred on or below
- **StrokeId:** unique ID of last trigger or tap.

The Data Pool structure updated is:

- MtButton:
  - string Id
  - string HitPoint
  - string HitContainer
  - string Taps
  - string plug-inContainer
  - string FirstChildContainer
  - string StrokeId

To Create a Simple Scene with the MtButton plug-in

⚠ **Note:** If a touch screen is not available, set **Multi Touch** input to **Mouse** in the **Communication** section of Viz Configuration (see the Viz Engine Administrator Guide).

1. Create a new Scene.
2. Add a new group Container.

3. Name the group Container ButtonScript.
4. Add the Script plug-in to the ButtonScript Container (Container plug-ins > Global).
5. Open the Script Editor and type this script:

```
sub OnTap(HitContainer as String, Id as String,    HitPoint as Vertex,    Taps as Integer,    plug-
inContainer as String,    FirstChildContainer as String)
        dim bt as Container
        dim text as String
        bt = FindSubContainer("ButtonText")
text = "OnTap(\n   HitContainer = " & HitContainer
text &= "\n   Id = " & Id
text &= "\n   HitPoint = " & HitPoint
text &= "\n   Taps = " & Taps
text &= "\n   plug-inContainer = " & plug-inContainer
text &= "\n   FirstChildContainer = " & FirstChildContainer & "\n)"
        bt.Geometry.Text = text
end sub
```

6. Click **Compile and Run**.
7. Add three Sub-containers. Name them, in order:
   - Button1
   - Button2
   - ButtonText
8. Add a Circle Geometry to the Button1 and Button2 Containers.
9. Use the Circle Geometry Transformation editors to move the two circles so that both can be seen.
10. Right-click the Button1 and Button2 Containers, in turn, and select **Add Material...** from the context-menu. Open the Material Editor and define a color for both Containers.
11. Add the MtButton plug-in to the Button1 and Button2 Containers.
12. In the MtButton plug-in editor:
    - Type 'myButton1' in the Button1 **Id** field
    - Type 'myButton2' in the Button2 **Id** field
    - In **Trigger Type** field click **On Tap** for both containers
13. Add a Font to the ButtonText Container.
14. Open the Scene in **On Air**.
15. Click on each button, in turn, and see the text change with each selection.
16. Use the MtButton plug-in editor to modify the button's actions, as required.

14.9.3 MtNavigator plug-in

The MtNavigator plug-in controls the Viz World Navigator plug-in.

The MtNavigator plug-in allows MultiTouch free navigation over a map generated by the Curious World Maps plug-ins. Currently it only supports the 'Flat' map model, and not the globe or other 3D models. It will send commands to the Navigator plug-in to change the position and distance of the map based on MultiTouch input.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> MultiTouchComp

This section contains information on the following topics:

- MtNavigator Configuration
- Events and Notifications

MtNavigator Configuration

- **Georeference Container:** Specifies the container which owns the Navigator plug-in.
- **Momentum:** Enables movement or rotation with momentum after the object is released.
  - **Friction:** Sets the amount of friction used when momentum is enabled.
- **Limit Camera Distance:** Limits the maximum and minimum camera distance.
- **Is free navigation allowed:** Activates or deactivates navigation.
- **Id:** Gives additional context in the handler script, specify a string that will identify any notifications dispatched by this plug-in. This is often included as an argument for the event so a common script may handle events from a number of plug-ins.
- **Set Shared Memory:** Enables shared memory to be updated for the plug-in notifications when set to **On**.
  - **Shared Memory Prefix:** This sets a 'prefix name' to be prepended to the shared memory variables maintained by the plug-ins notifications. For plug-ins that maintain multiple fields each field name has the prefix prepended to it followed by a dot '.' so as to mimic member access to an object, i.e: if the prefix is 'Obj' the fields 'field1' and 'field2' would be identified with the strings 'Obj.field1' and 'Obj.field2'. The shared memory field 'Obj' is also maintained and is simply an integer that is modified every time any of its 'subfields' is updated.
  - **Shared Memory Type:** Selects the shared memory area to update. Click either Global, Scene, or Distributed.
- **Set Data Pool:** Shows 'you wish' plug-in notifications to set a Data Pool variable.
  - **Data Pool Variable:** Shows the name of the Data Pool variable to have set.

Events and Notifications

The events dispatched by MtNavigator are:

OnNavGrab (when navigable object is 'grabbed'):

sub **OnNavGrab** (HitContainer as String, Id as String, Offset as Vertex, Distance as Double, plug-inContainer as String)

end sub

- **HitContainer as String**
- **Id as String**
- **Offset as Vertex**
- **Distance as Double**

- **plug-inContainer as String**
  OnNavGrabUpdate (when navigable object is moved):

sub **OnNavGrabUpdate** (HitContainer as String, Id as String, Offset as Vertex, Distance as Double, plug-inContainer as String)

end sub

- **HitContainer as String**
- **Id as String**
- **Offset as Vertex**
- **Distance as Double**
- **plug-inContainer as String**
- **OnNavGrabRelease** (when navigable object is released):
  - HitContainer as String
  - Id as String

The Shared Memory Field names updated are:

- **HitContainer:** the name of the container that was 'hit'
- **Id:** the user ID string entered in the notification interface
- **Offset:** the current lon/lat position in the Navigator
- **Distance:** the current distance in the Navigator
- **plug-inContainer:** the name of the container this plug-in is attached to
- **Active:** 1 if 'grabbed' 0 if 'released'

The Data Pool structure updated is:

- MtNavigator:
  - string Id
  - string HitContainer
  - string Offset
  - string Scale
  - string plug-inContainer
  - string Active

14.9.4 MtTelestrator plug-in

The MtTelestrator plug-in is used to do telestration on a Container level (used with the GraffitiTex plug-in (added automatically to the same Container)).

For scene-wide telestration the MtSensor plug-in has a telestration feature.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> MultiTouchComp

This section contains information on the following topics:

- MtTelestrator Configuration
- Events and Notifications

- To Create a Simple Scene with Telestration

MtTelestrator Configuration

- **Active:** Enables or disables telestration input from the user.
- **Minimum Telestration Width:** Determines minimum brush width to assign based on pressure applied to sensor.
- **Maximum Telestration Width:** Determines maximum brush width to assign based on pressure applied to sensor.
- **Color:** Assigns the color for subsequent telestration drawing.
- **Id:** Gives additional context in the handler script, specify a string that will identify any notifications dispatched by this plug-in. This is often included as an argument for the event so a common script may handle events from a number of plug-ins.
- **Set Shared Memory:** Enables shared memory to be updated for the plug-in notifications when set to **On**.
    - **Shared Memory Prefix:** This sets a 'prefix name' to be prepended to the shared memory variables maintained by the plug-ins notifications. For plug-ins that maintain multiple fields each field name has the prefix prepended to it followed by a dot '.' so as to mimic member access to an object, i.e: if the prefix is 'Obj' the fields 'field1' and 'field2' would be identified with the strings 'Obj.field1' and 'Obj.field2'. The shared memory field 'Obj' is also maintained and is simply an integer that is modified every time any of its 'subfields' is updated.
    - **Shared Memory Type:** Selects the shared memory area to update. Click either Global, Scene, or Distributed.
- **Set Data Pool:** Shows 'you wish' plug-in notifications to set a Data Pool variable.
    - **Data Pool Variable:** Shows the name of the Data Pool variable to have set.
- **Init:** Clears current telestration and make Telestration active.
- **Clear:** Clears the contents of the attached telestration.

Events and Notifications

The events dispatched by MtTelestrator are:

- `OnTelestrationStateChange` (sent when telestration is activated or deactivated):
    - `plug-inContainer` as String
    - `Id` as String
    - `TelestrationOn` as Boolean
- `OnTelestrationColorChange` (sent when telestration is activated or deactivated):
    - `plug-inContainer` as String
    - `Id` as String
    - `PenColor` as Color
- `OnTelestrationClear` (sent when telestration is cleared):
    - `plug-inContainer` as String
    - `Id` as String

    - `TelestrationOn` as Boolean
      The shared memory field names are:

- **plug-inContainer:** the name of the plug-ins container or '' if attached to a scene
- **Id:** the user ID string entered in the notification interface
- **PenColor:** the current pen color
- **Active:** 1 if telestration is active, 0 if not

The Data Pool structure updated is:

- MtTelestrator:
    - string plug-inContainer
    - string Id
    - string PenColor
    - string Active

To Create a Simple Scene with Telestration

⚠ **Note:** If a touch screen is not available, set **Multi Touch** input to **Mouse** in the **Communication** section of Viz Configuration (see the Viz Engine Administrator Guide).

1. Create a new Scene.
2. Add a new group Container.
3. Add a Rectangle Geometry to the Container.
4. Add the MtTelestrator plug-in to the container. This automatically adds the GraffitiTex plug-in to the container.
5. Set the **Active** parameter for the MtTelestrator and GraffitiTex to **ON** in the Properties Panel.
6. Open the Scene in **On Air**. Use the touch input device to create freehand drawing.
7. Use the MtTelestrator and GraffitiTex plug-in editors to modify the telestration affect as required.

See Also

- GraffitiTex plug-in

14.9.5 Plug-in Event and Notification System

All the MultiTouch (Mt) plug-ins share a common event notification module and associated configuration interface.

For all plug-ins that generate events, events are dispatched up the Container hierarchy from the plug-in Container to the first script found with a handler present for the event being generated.

For plug-ins that optional update values in Shared Memory, select which one of the three maps to update:

- System
- Scene
- Distributed
  Each shared memory area updated by a plug-in is composed of one 'root' variable specified in the 'Shared Memory Prefix' field in the plug-ins user interface. This 'root' variable is

updated if any of its additional sub-fields are updated. The root variable contains an integer or 'update id'. Sub-fields are identified by adding 'dot delimited' suffix on the root name. For example, if 'Shared Memory Prefix' is set to 'Name', the sub-field 'field' would be identified as 'Name.field' in a Viz script.

For all plug-ins that can set a Data Pool variable, each has a 'structure' defined in a `.dp` file. There is one such file for each plug-in. Both the file and the name of the structure defined by it are the same as the name of the plug-in. For example, in the case of an MtButton, the file is `MtButton.dp` and the name of the structure for sub-item access is `MtButton`.

14.9.6 Mt3D Control plug-in

The Mt3D Control plug-in moves, scales, and rotates an object in 3D.

The Mt3D Control plug-in provides an arc ball-like controller for the object it is attached to. When used with one finger, the object can be rotated. When multiple fingers are used, the object can be moved, rotated and scaled in a 2D plane (the same as the Mt2D Control plug-in).

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> MultiTouchComp

This section contains information on the following topics:

- Mt3D Configuration
- 2D Configuration
- Output Configuration
- Events and Notifications
- To Create a Simple Scene with the Mt3D Control plug-in

Mt3D Configuration

- **Config:** Selects 3D Config
- **Axis Lock:** Constrains the rotation to a specific axis, or allow free rotation. Options are:
  - Off
  - X Axis
  - Y Axis
  - Z Axis
- **Axis Lock Local**: Uses the local axis instead of the world axis if axis lock is on.
- **Active during screen wide telestration?**: If the button to is to stay active when the screen-wide telestrator is on, set to On. The default is off, which means the button cannot be triggered when telestration is on.
- **Hit Coordinate Type**: Selects the Coordinate system to use. Click either Local, World or Screen.

2D Configuration

- **Config:** Select 2D Config
- **Momentum:** Enable movement or rotation with momentum after the object is released.
  - **Friction:** Set the amount of friction used when momentum is enabled

> ⚠ **Note:** Using this feature will conflict with any externally scripted or other animations applied to the container.

- **Lock Translation:** Allow only rotation and scaling of the object
- **Lock Translation (Screen X):** Prevent movement left-to-right on the screen
- **Lock Translation (Screen Y):** Prevent movement up-and-down on the screen
- **Lock Translation (Object X):** Prevent movement along the object's local X axis
- **Lock Translation (Object Y):** Prevent movement along the object's local Y axis
- **Lock Rotation:** Prevent rotation of the object
- **Lock Scale:** Prevent resizing of the object
- **Active during screen wide telestration**?: If the button to is to stay active when the screen-wide telestrator is on, set to On. The default is off, which means the button cannot be triggered when telestration is on.
- **Hit Coordinate Type**: Click either Local, World or Screen. This selects the Coordinate system to use.

Output Configuration

- **Config:** Select Output Config
- **Id:** To give additional context in the handler script, specify a string that will identify any notifications dispatched by this plug-in. This is often included as an argument for the event so a common script may handle events from a number of plug-ins.
- **Set Shared Memory:** If ON enables shared memory to be updated for the plug-in notifications.
  - **Shared Memory Prefix:** This sets a 'prefix name' to be prepended to the shared memory variables maintained by the plug-ins notifications. For plug-ins that maintain multiple fields each field name has the prefix prepended to it followed by a dot '.' so as to mimic member access to an object, i.e: if the prefix is 'Obj' the fields 'field1' and 'field2' would be identified with the strings 'Obj.field1' and 'Obj.field2'. The shared memory field 'Obj' is also maintained and is simply an integer that is modified every time any of its 'subfields' is updated.
  - **Shared Memory Type:** Click either Global, Scene, or Distributed. This selects the shared memory area to update.
- **Set Data Pool:** Shows 'you wish' plug-in notifications to set a Data Pool variable
  - **Data Pool Variable:** Shows the name of the Data Pool variable to have set.

- **Active during screen wide telestration?**: If the button to is to stay active when the screen-wide telestrator is on, set to On. The default is off, which means the button cannot be triggered when telestration is on.
- **Hit Coordinate Type**: Click either Local, World or Screen. This selects the Coordinate system to use.

Events and Notifications

The events dispatched by Mt3dControl are:

OnGrab (when object is 'grabbed'):

sub OnGrab (**HitContainer as String, Id as String, HitPoint as Vertex, plug-inContainer as String, FirstChildContainer as String**)

end sub

- **HitContainer as String**
- **Id as String**
- **HitPoint as Vertex**
- **plug-inContainer as String**
- **FirstChildContainer as String**

OnGrabUpdate (when object is moved):

sub **OnGrabUpdate** (**HitContainer as String, Id as String, Position as Vertex, Rotation as Vertex, Scale as Vertex, Pressure as Double**)

end sub

- **HitContainer as String**
- **Id as String**
- **Position as Vertex**
- **Rotation as Vertex**
- **Scale as Vertex**
- **Pressure as Double**

OnGrabRelease (when object is "released"):

sub OnGrabRelease (HitContainer as String, Id as String)

end sub

- **HitContainer as String**
- **Id as String**

The Shared Memory Field names updated are:

- HitContainer (the name of the container that was "hit")
- Id (the user ID string entered in the notification interface)
- HitPoint
- plug-inContainer
- FirstChildContainer
- Position
- Rotation

- Scale
- Pressure
- State

The Data Pool structure updated is:

- Mt3dControl:
  - string HitContainer
  - string Id
  - string HitPoint
  - string plug-inContainer
  - string FirstChildContainer
  - string Position
  - string Rotation
  - string Scale
  - string Pressure
  - string State

To Create a Simple Scene with the Mt3D Control plug-in

⚠ **Note:** If a touch screen is not available, set **Multi Touch** input to **Mouse** in the **Communication** section of Viz Configuration (see the Viz Engine Administrator Guide).

1. Create a new Scene.
2. Add a new group Container.
3. Add a Cube Geometry to the Container.
4. Add the Mt3D Control plug-in to the Container.
5. Open the Scene in **On Air**. Use the touch input device to transform the object.
6. Use the Mt3D Control plug-in editor to modify the plug-in behavior, as required.

14.10 PixelFX

The PixelFX plug-ins are part of the PixelFX plug-ins set.

The following Container plug-ins are located in the PixelFX folder:

- pxLensMulti

14.10.1 pxLensMulti

pxLensMulti serves as a parent container for all the other PixelFX plug-ins Geometry plug-ins. It simplifies the process of animating lens flares so that they resemble realistic behavior. In the child containers any number of lens flare shapes can be placed. pxLensMulti positions, colors, governs the opacity of all according to the configurable parameters.

The lens flare shapes at origin are a flat geometry. pxLensMulti will make sure that all the shapes it handles are constantly facing the camera.

⚠ **Note:** pxLensMulti creates a notional line along which all shapes are scattered.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> PixelFX

This section contains information on the following topics:

- pxLensMulti Properties

pxLensMulti Properties

Under the Spread tab, the following parameters can be configured to handle shapes' position:

- **Mode**: In Polar mode, the notional line can be defined by angle and radius originating from the screen's center. In Cartesian mode, the line is created between the screen's center and given an x-y location in the screen's coordinates. In Container mode, you need to drag a container from the scene tree into the Container box; the container imitates the light position so that the notional line starts from that position and goes through the screen's center.

  ⚠ **Note:** In Container mode, one point of the notional line is in the coordinates and the second point, through which the notional line passes, is defined in the screen coordinates. As such, camera movement in Container mode actually defines the shapes' movement.

- **Distance**: Defines the notional line that passes through the screen's center. The distance of this center point from the camera is defined by the this parameter's value.
- **Count**: Multiplies the quantity of each shape in each child container by the value entered for this parameter.
- **Radius Scale**: Scales the notional line from the origin onward.
- **Radius Rand**: Scales the notional line's length, regardless of origin.
- **Single Seed**: Allows only one seed to be used for each shape in the child containers when set to **On**. When set to Off, a different seed is used for each shape in the child containers.
- **Seed**: Defines the pseudo-random rule for randomly changed parameters. Once a value is chosen, although the original statement will be random, the effect will actually look the same on other computers as well.

Under the Attributes tab, the following can be configured to handle non-positional parameters:

- **Amount**: Determines the amount of light used in the shapes.
- **Amount Rand**: Randomizes the Amount parameter between different shapes.
- **Size Rand**: Randomizes the sizes between different shapes.
- **Color Rand**: Randomizes the color between different shapes.

See Also

- pxColorWorks
- PixelFX plug-ins in Geom plug-ins

- PixelFXLenseFlare in Shader plug-ins

14.11 Presenter

The following container plug-ins are located in the Presenter folder:

- Bar
- Bar Value
- Bar Values
- Pie Slice
- Pie Values

14.11.1 Bar

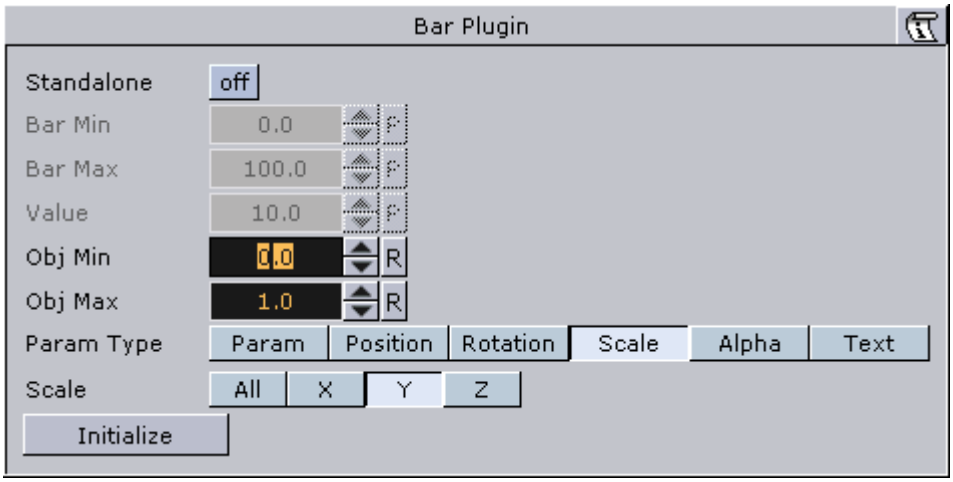
The Bar plug-in tells the Bar Values plug-in that the container is part of the bar chart and it also tells which items of the geometry on the container to control. Each container that is to be a bar, must hold a geometry and a Bar plug-in. Bar can be used as a standalone plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Presenter

This section contains information on the following topics:

- Bar Properties

Bar Properties



- **Standalone:** If the Bar plug-in is going to be used without a Pie Values plug-in to set the values, enable this button. It will enable you to set the value directly on the plug-in and you

can define normalization factors between the input values and the value that is being set on the geometry.

- **Bar Min / Max:** If in standalone, these values set the minimum and maximum values for the bar. These values are connected to the Obj Min/Max values and by adjusting these together you can get the required effect between input values and visual effect.
- **Value:** Sets the scaling value of the object.
- **Obj Min / Max:** Maps the input values to the native values on the object being controlled by the plug-in. Adjust these together with Bar Max/Min on the Bar Values plug-in or on the Bar plug-in itself if in standalone mode, to get the required effect between input values and visual effect.
- **Param Type:** Sets the type of value to control on the object.
  - **Param:** Controls a named parameter on the object. Use the console to find the correct name for the parameter: Enable “Show commands”, Open the object’s editor, change the parameter and see in the console what the parameter name is.
  - **Position:** Position values can be set to control the X, Y or Z axis.
  - **Rotation**: Rotation values can be set to control the X, Y or Z axis.
  - **Scale:** Scale values can be set to control All, X, Y or Z axis.
  - **Text:** Text values can be shown as Integer, Float or Formatted text.
- **Initialize:** Enables the plug-in to initialize and gain control of the parameters.

See Also

- Bar Value
- Bar Values
- Visual Data Tools

14.11.2 Bar Value

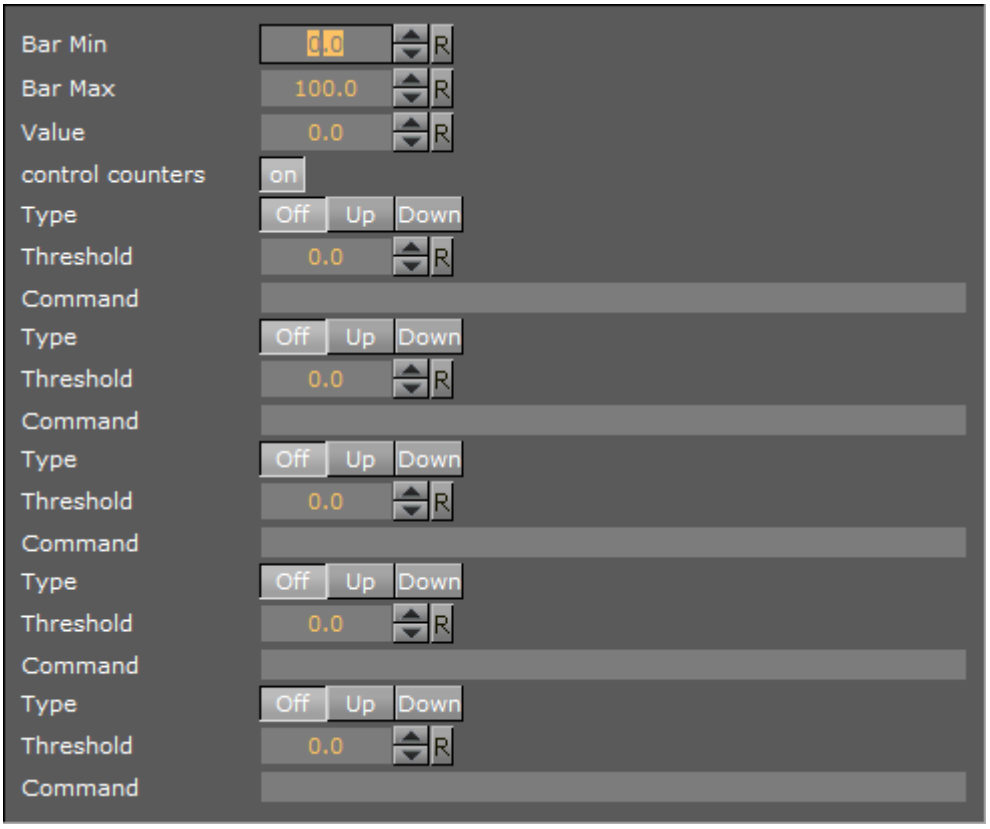
Bar Value controls the value of all Bar plug-ins and Counters of its child nodes. In addition a set of commands can be specified that are executed when the value passes certain thresholds.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Presenter

This section contains information on the following topics:

- Bar Value Properties
- To Create a Bar that Shows its Value

Bar Value Properties



- **Bar Min**: Minimum Value of the bar.
- **Bar Max**: Maximum Value of the bar.
- **Value**: The actual value of the bar.
- **control counters**: States if counters should be affected. If this is checked, Value will be shown in all Sub-Containers with Counter plug-ins.
- **Type**: Determines whether the command should fire if the threshold is passed from below, or from above. Off does not execute the command at all. Available options are Off, Up and Down.
- **Threshold**: The threshold against Value is compared.
- **Command**: The Viz Artist/Engine command that will be executed when the threshold is passed.

To Create a Bar that Shows its Value

1. Create a Container and add the Bar Value plug-in to it.
2. Open the Bar Value editor, and check control counters.
3. Set Threshold to 20, Type to Up and set a command (e.g. `MAIN*ONAIR GET_INFO`).
4. Add two Sub-Containers.
5. Add a Font and a Counter plug-in to the first Sub-Container.
6. Add a Cube and a Bar plug-in to the second Sub-Container.
7. Set the Cube's Y-Axis Center to (B)ottom.

8. Open the Bar plug-in editor and set Param Type to Scale and Scale to Y.
9. Modify Bar Value's Value field.
    - If the value exceeds 20 the information will be written to the console.

See Also

- Bar
- Bar Values
- Visual Data Tools

14.11.3 Bar Values

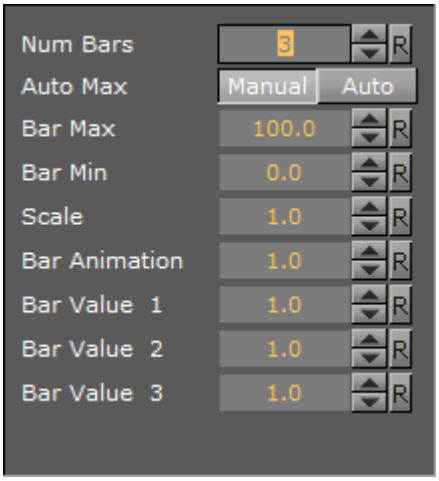
The Bar Values plug-in communicates with all Sub-Containers that hold a Bar plug-in and lets you set values for each of them.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Presenter

This section contains information on the following topics:

- Bar Values Properties
- To Create a Bar Chart with the Bar Values plug-in

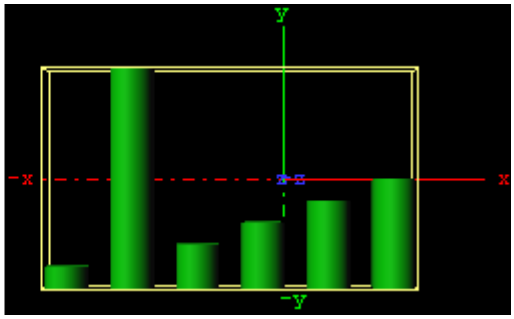
Bar Values Properties



- **Num Bars:** Sets the number of bars to be controlled by the Bar Values plug-in. Maximum number of bars is 25.
- **Auto Max:** Allows the user to switch between Manual and Auto mode. Auto mode disables the Bar Max value.
- **Bar Max:** Sets the maximum values of the bar.

- **Bar Min:** Sets the minimum value of the bar.
- **Scale:** Sets the scale of the object.
- **Bar Animation:** Sets the number of bars to be animated.
- **Bar Value 1-25:** Sets the value for the bar(s).

To Create a Bar Chart with the Bar Values plug-in



1. Add a group container to the scene tree, and add the Bar Values and Grid Arrange plug-in to it.
2. Add a group container as a Sub-Container to the first, and name it **Bar**.
3. Add material, the Cylinder and Bar plug-in to the Bar container.
4. Open the **Cylinder** editor and set Center to **Bottom**.
5. Open the **Bar** editor, set Param Type to **Scale** and Scale to **Y**, and click **Initialize**.
6. Create and place a number of copies of the Bar container (e.g. 5) at the same level as the Bar container.
7. Open the Grid Arrange editor and set **Number of Columns** to **6** and **Column offset** to **30.0**.
8. Open the Bar Values editor and set **Num Bars** to **6** and **Bar Animation** to **6**.

See Also

- Visual Data Tools
- Control Bars
- Bar
- Bar Value
- Pie Slice
- Pie Values

14.11.4 Pie Slice

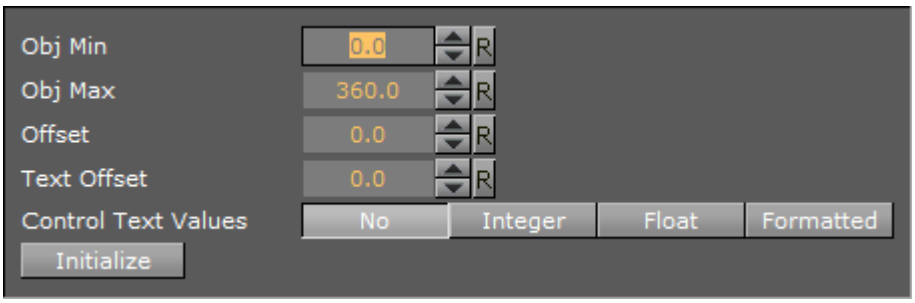
Use the Pie Slice plug-in with the Pie Values plug-in to create and animate a pie chart of up to 20 slices.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Presenter

This section contains information on the following topics:

- Pie Slice Properties

Pie Slice Properties



- **Obj Min / Max:** Maps the input values to the values on the cylinder object. Adjust these together with Bar Max/Min on the Pie Values plug-in to get the required effect between input values and visual effect.
- **Offset:** Offsets the slice from the center of the pie chart.
- **Text Offset:** Offsets the text label from the center of the pie chart.
- **Control Text Values:** Sets the type of text. Available options are No, Integer, Float and Formatted.
- **Initialize:** Initializes the value of the pie slice.

See Also

- Visual Data Tools
- Control Bars
- Bar
- Bar Value
- Bar Values
- Pie Values

14.11.5 Pie Values

Use the Pie Values plug-in with the Pie Slice plug-in to create and animate a pie chart of up to 20 slices.

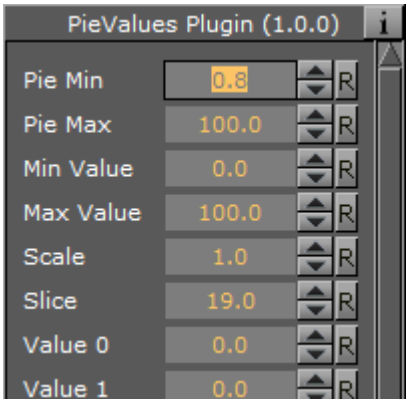
⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Presenter

This section contains information on the following topics:

- Pie Values Properties

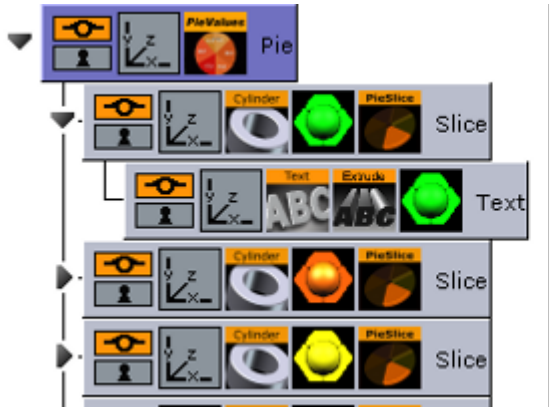
- To Create a Pie Chart with the Pie Values plug-in

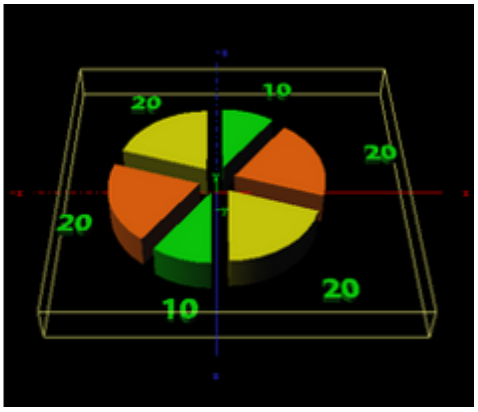
Pie Values Properties



- **Pie Min and Max**: Sets the minimum and maximum values for the pie size and the range of the size.
- **Min and Max Value**: Sets the range of the shown pie.
- **Scale:** Sets the scale of the pie. Default is 1.
- **Slice:** Sets the number of slices to be shown as part of the pie.
- **Value 0 - 19:** Sets the slice value. Available number of slices is 20.

To Create a Pie Chart with the Pie Values plug-in





1. Add a group container to the scene tree, and name it **Pie**.
2. Open the transformation editor and set **Rotation X** to **45.0**, and **Scaling Y** (single) to **0.2**.
3. Add the Pie Values plug-in to the Pie container.
4. Open the Pie Values editor and set **Slice** to **6.0**, and enter the following values in **Value 0** to 5; 10.0, 20.0, 20.0, 10.0, 20.0, 20.0.
5. Add a Sub-Container to the Pie container and name it **Slice**.
6. Add the Cylinder geometry plug-in, material and PieSlice plug-in to the Slice container.
7. Open the PieSlice editor and set Offset to 10.0 and Text Offset to 80.0.
8. Add a Sub-Container to the slice container and name it **Text**.
9. Open the transformation editor for the Text container and set **Scaling** (locked) to **0.2** and **Rotation X** to -90.0.
10. Add a font, material and the Extrude plug-in to the Text container.

11. Open the Extrude editor and set **Extrusion Depth** to **100.0**.
12. Create and place a number of copies of the Slice container (e.g. 5) at the same level as the Slice container.
13. Change the colors of the pie slices to distinguish each slice.

See Also

- Visual Data Tools
- Control Pie
- Bar
- Bar Value
- Bar Values
- Pie Slice

14.12 PxColorWorks

The color correction plug-ins, which are part of the PixelFX plug-ins set, are generally fast and efficient, however there is a certain performance penalty when animating them in a stacked scenario.

⚠ **Note:** The PixelFX plug-ins are a separate package from the standard Viz Artist installation, but function within the Viz Artist environment.

The following Container plug-ins are located in the pxColorWorks folder:

- PixelFX plug-ins
- pxAddSubtract
- pxBlackAndWhite
- pxBrightContrast
- pxColorMatch
- pxGamma
- pxHueRotate
- pxMask
- pxSaturation
- pxStack
- pxTint

See Also

- PixelFX in Shader plug-ins
- PixelFX plug-ins in Geometry plug-ins

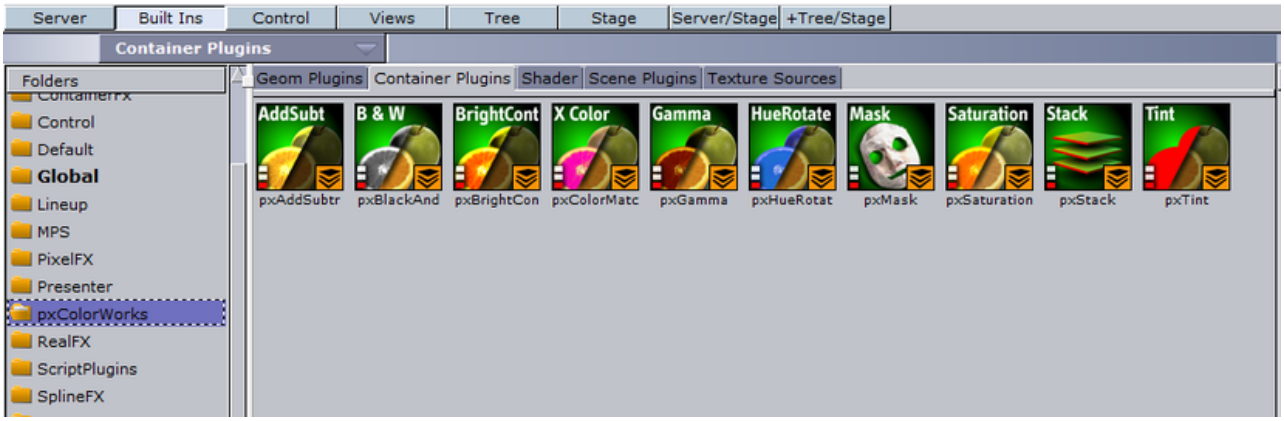
14.12.1 PixelFX plug-ins

The PixelFX plug-ins affect pixels and geometry, and produce various special effects. They do three main things:

- Color correction
- Lens flares
- Other pixel-based effects, such as noise, gradients, various distortion effects, transitions, and similar effects.
  The icon for each of the plug-ins have mini icons that show certain properties of the particular plug-in.

- At the bottom left, there is a performance bar, with three blocks (like a cellphone battery indicator). One block means the plug-in will not have a drastic performance effect, and three means the plug-in is taxing and should therefore be used with caution.
- At the bottom right the orange stack shows if the plug-in is stackable, meaning it can be used in conjunction with color correction plug-ins.
- Some plug-in icons themselves are divided in half with one side showing the visual effect with the plug-in and the other without it.
  Some of the PixelFX plug-ins work only when applied to a Container that also has a Texture. The rest however, function on the Renderer's pixel buffer. For example, you may apply color correction to the root Container in a Scene, therefore perform a color correction to the whole scene. Another example is pxNoise- apply it to the root level container of a scene or a whole tree branch to add a kind of film-grain effect.

One of the parameters that feature in a number of the PixelFX plug-ins is Seed. This defines the pseudo-random rule for changed parameters. Once a value is chosen, although the original state will be random, the effect will actually look the same on other computers as well.



14.12.2 pxAddSubtract

pxAddSubtract allows you to add or subtract a constant color value to each pixel. It can be applied to Everything, Shadows, Mid tones and Highlights.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

14.12.3 pxBlackAndWhite

pxBlackAndWhite allows you to gray scale an RGB image or a tree branch holding geometry. It can be applied to Everything, Shadows, Mid tones and Highlights.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

14.12.4 pxBrightContrast

pXBrightContrast allows you to adjust the brightness and the contrast on each of R, G and B on a given image or a tree branch holding geometry. It can be applied to Everything, Shadows, Mid tones and Highlights.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

14.12.5 pxColorMatch

pxColorMatch lets you replace one color with another. It works on images and tree branches holding geometry alike.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

Some important parameters:

- **Weight**: Defines the color range proximity to the selected color.
- **Inertia**: Defines the smoothness of the range proximity graph.
- **Count**: Defines the number of color replace pairs you wish to use.

14.12.6 pxGamma

pXGamma allows you to adjust the gamma correction on each of R, G and B on a given image. It can be applied to Everything, Shadows, Mid tones and Highlights.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

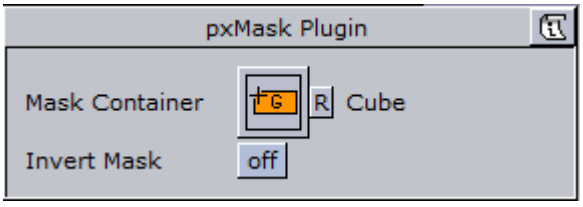
14.12.7 pxHueRotate

pxHueRotate allows you to adjust the hue rotation on an image. It can be applied to Everything, Shadows, Mid tones and Highlights.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

14.12.8 pxMask

pxMask allows you to supply a mask for the color correction operation. The mask is defined by a black and white image that is dragged into the pxMask plug-in parameters. pxMask, as indicated by the stackable icon, can function in conjunction with a mask.



> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

14.12.9 pxSaturation

pxSaturation allows you to adjust the color saturation on an image. It can be applied to Everything, Shadows, Mid tones and Highlights. You also have the option to preserve highlights, or not.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

14.12.10 pxStack

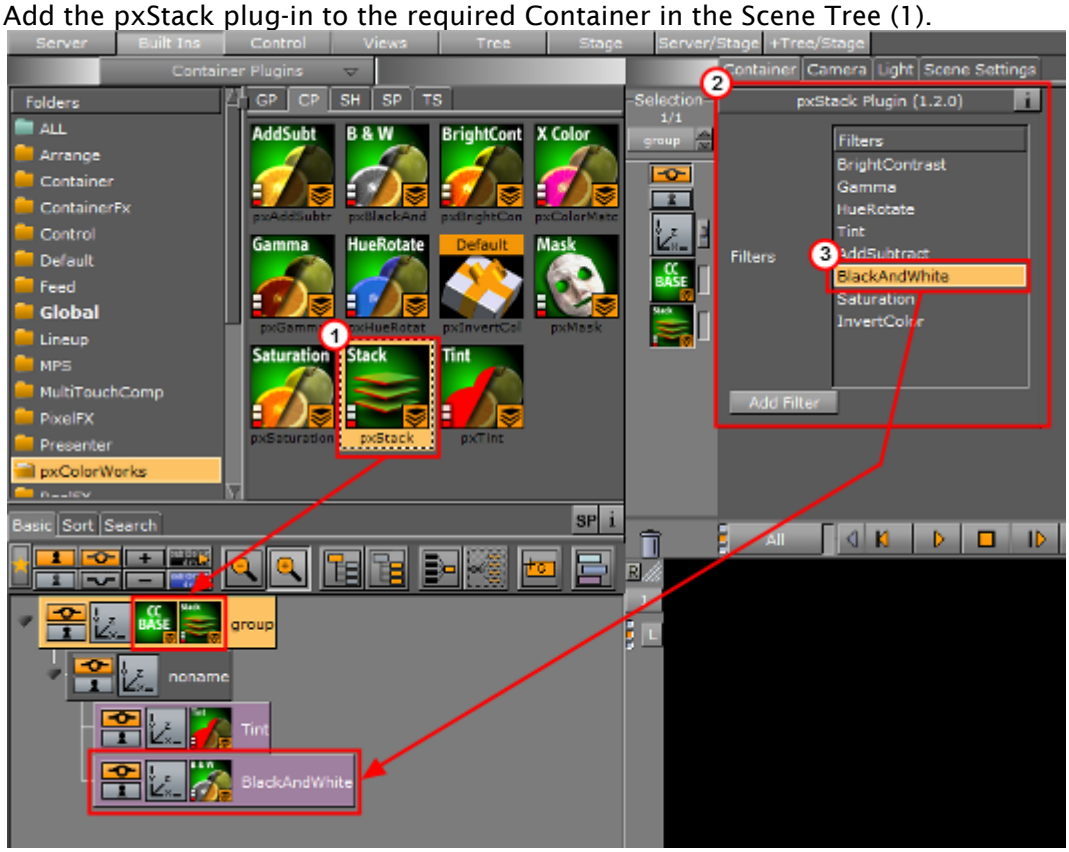
> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

This section contains information on the following topics:

- To apply to the pxStack plug-in

To apply to the pxStack plug-in

1. Add the pxStack plug-in to the required Container in the Scene Tree (1).



> ⚠ **Note:** When the pxStack plug-in is added to a Container, it automatically adds the pxCCBase plug-in to the same Container. Also, it automatically generates a child Container to host the color correction nodes you will define later.

2. Click the pxStack icon in the Container. The plug-inn editor (2) shows a list of color correction filters.
3. Click on a color correction filter.
4. Click **Add Filter**. The color correction filter is added as a new Container under the 'noname' Container in the Scene Tree (3). This termed as a 'Color Correction Node'.
5. Repeat to add more color correction filters. Each color correction filter applied is added (stacked) as a new Color Correction Node under the 'noname' Container in the Scene Tree.
6. Click on a Color Correction Node to open its editor. Modify the effect properties as required.
7. Change the order of the stacked properties, as required.

> ⚠ **IMPORTANT!** Selected properties act in the sequence in which they are ordered in the scene tree.

⚠ **Note:** To disable or enable a Color Correction Node click the Locked/Unlock icon in the top left of the Container.

⚠ **Note:** To remove a color correction function, delete the plug-in.

14.12.11 pxTint

pxTint allows you to add a color tint and apply it to an image in the required amount. It can be applied to Everything, Shadows, Mid tones and Highlights. You also have the option to preserve highlights, or not.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> pxColorWorks*

14.13 RealFX

The RealFX plug-in set enables you to create particle effects in Viz Artist.

Particle systems are a computer graphics technique to simulate certain physics-based effects, which are otherwise very hard to reproduce with conventional rendering techniques. Examples of such effects which are commonly replicated using particle systems include fire, explosions, smoke, weather effects, sparks, falling leaves, dust, meteor tails, or abstract visual effects like glowing trails, magic spells, etc.

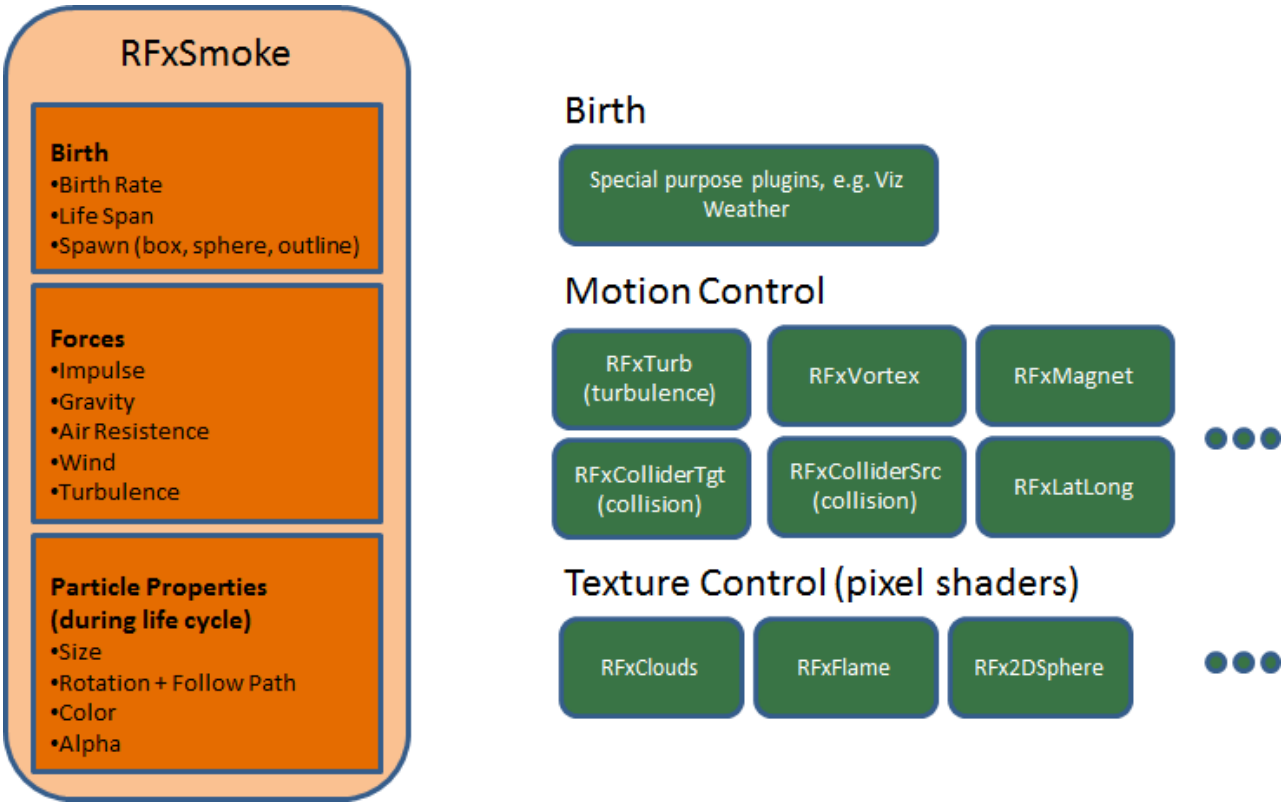
The particle effects in Viz Artist/Engine run in real-time, meaning that there are a few inherent constraints that must be taken into account when considering best practices for employing this plug-in set. For example, there is a trade-off between the number of particles and performance optimization; more generally there needs to be a considered balance between performance and visual quality.

RFxSmoke is the baseline plug-in within the RealFX plug-in set. The remaining plug-ins in this set are applied on top of RFxSmoke in any given Container. RFxSmoke includes built-in functionality and the ability to host the additional functionality contained in the other plug-ins in this set. Part of the built-in functionality, e.g. turbulence, is kept for compatibility with previous version of Viz Artist.

There are three categories of additional plug-ins:

- **Birth plug-ins:** refer to where the particles are spawned
- **Motion control plug-ins:** govern the position, direction, velocity, size and color of each particle
- **Texture control plug-ins:** affect the texture mapping and the “look” of each particle by using pixel shader technology

14.13.1 RealFX Architecture



The following Container plug-ins are located in the RealFX folder:

- RFxColliderSrc
- RFxColliderTgt
- RFxLatLong
- RFxMagnet
- RFxTurb
- RFxVortex

See Also

- RealFX in Shader plug-ins
- RealFX plug-ins in Geometry plug-ins

14.13.2 RFxColliderSrc

Located under the Container plug-ins tab, RFxColliderSrc is used in conjunction with RFxColliderTgt to create a collision and collision detection system. For example the source

could be a notional wall or floor, whereas the target would be the particles themselves. The particle target can work with a number of sources.

> ⚠ **Note:** When RFxCollisionSrc is added to a container, the RFxCollisionManager plug-in is automatically added at the scene level. This plug-in is used internally and not configurable.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> RealFX

Adjust the following parameters as required to achieve the required effect:

- Object Type (Auto, Sphere, Box)
- Bounce
- Random
- Proximity Fade and Fade Range: Allows you to fade particles based on their proximity to other objects (only spheres and boxes).
- Flag 1-8 (on/off)

See Also

- RFxSmoke
- RFxColliderTgt

14.13.3 RFxColliderTgt

Located under the Container plug-ins tab, RFxColliderTgt is used in conjunction with RFxTurb to create a collision and collision detection system. For example the source could be a notional wall or floor, whereas the target would be the particles themselves. The particle target can work with a number of sources.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> RealFX

Adjust the following parameters as required to achieve the required effect:

- Flag 1-8 (on/off)

See Also

- RFxSmoke
- RFxTurb

14.13.4 RFxLatLong

Located under the Container plug-ins tab, RFxLatLong fits any other RealFX motion onto a globe surface. The effects are mainly geographic and large-scale.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> RealFX

Adjust the following parameters as required to achieve the required effect:

- Radius
- Spawn Min Long
- Spawn Max Long
- Spawn Min Lat
- Spawn Max Lat
- Update from Globe (on/off)
- Spawn Min Alt
- Spawn Max Alt
- Orientation (Horizontal/Radial)
- Trim Radius

See Also

- pxColorWorks

14.13.5 RFxMagnet

Located under the Container plug-ins tab, RFxMagnet creates motion patterns that resemble a magnetic field.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> RealFX

Adjust the following parameters as required to achieve the required effect:

- Amount
- Center X
- Center Y
- Center Z
- Power Law

- Min Distance Clamp

See Also

- RFxSmoke

14.13.6 RFxTurb

Located under the Container plug-ins tab, RFxTurb applies turbulence-like forces to particles. It can be used to create the effect of random changes in wind force and direction.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> RealFX

Adjust the following parameters as required to achieve the required effect:

- Amount
- Inertia
- Wavelength
- Update Rate

See Also

- RFxVortex

14.13.7 RFxVortex

Located under the Container plug-ins tab, RFxVortex applies vortex-like forces to particles. It can be used to create the effect of a tornado-like twister.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> RealFX

Adjust the following parameters as required to achieve the required effect:

- Amount
- Inertia
- Center X
- Center Y
- Center Z
- Rot X
- Rot Y

See Also

- RFxSmoke

14.14 Script Plug-Ins

The Script plug-ins is a folder to save created Container Script plug-ins.

A script can be saved as its own script plug-in, and used in future Scenes (see Create Script-based plug-ins). To save a Script plug-in, drag a compiled script into the Script plug-ins folder.

> ⚠ **Note:** Script plug-ins are saved to *<viz data folder>\Scriptplug-ins*.

See Also

- Script plug-in
- Script Editor

14.15 Sounds

The following container plug-ins are located in the Sounds folder:

- Text2Speech

14.15.1 Text2Speech

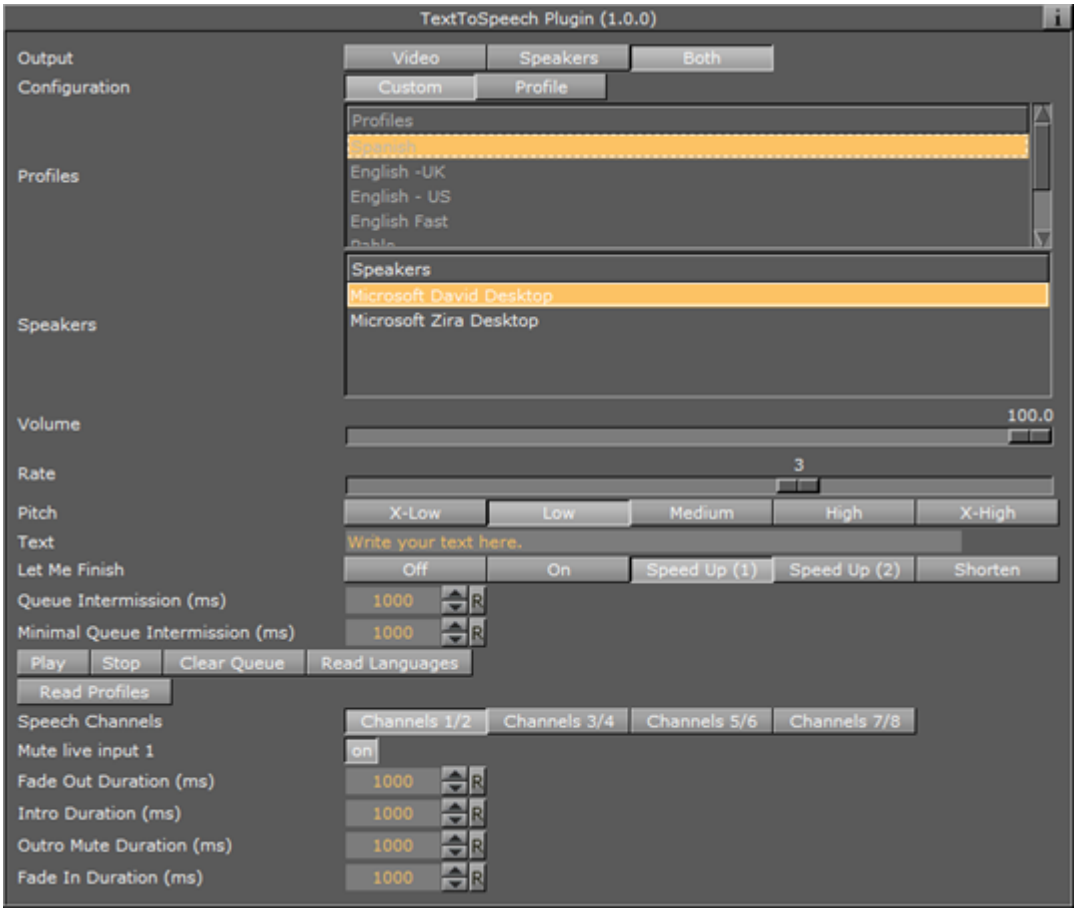
The Text2Speech plug-in is used to

> ⚠ **Note:** This plug-in is located in: *Built ins -> Container Plug-ins -> Sounds*

This section contains information on the following topics:

- Text2Speech Properties

Text2Speech Properties



- **Output:**
  - **Video:** Plays the output to the video out. For Video version only.
  - **Speakers:** Plays the output to the computer speakers. For both VGA and Video versions.
  - **Both:** Plays the output to both the video out and computer speakers. If running a Video version without a video card, this will cause double playback.
- **Configuration:** Selects predefined configurations for the **Speaker**, **Rate**, and **Pitch** parameters. This section is only visible when a profile file is located.
  - **Custom**: Sets the plug-in parameters manually. This is disabled if Configuration is set to `Profile`.
  - **Profile:** Sets the plug-in parameters based on the profile file selected in the **Profiles** pane. This pane is only visible when a profile file is located. If Configuration is set to `Custom`, the **Profile** button will be disabled.
- **Speakers**: Shows all speakers present in the system.
  - Lists all of the speakers currently present in the system, and allows the user to select one. Disabled if Custom configuration is selected.
- **Volume:** Sets the desired output volume.
- **Rate:** Sets the desired output rate.

- **Pitch:** Selects the output pitch. The available settings are:
  - X-Low
  - Low
  - Medium
  - High
  - X-High
- **Text:** Changes made to a text is played out immediately when in On Air mode. Play-out of changes to text during scene design in Viz Artist requires pressing the Play button.
- **Let Me Finish:** Defines the behavior when a new text triggers play-out. Used in On air mode only.
  - **Off:** New text will stop the current output and start playing the new text. When set to `Off`, **Queue Intermission** and **Minimal Queue Intermission** is disabled.
  - **On:** When text is playing out, adding new text will add this to a queue and play out automatically when the previous text is done, adjusted for the intermission value. When set to `On`, **Queue Intermission** is enabled.
  - **Speed Up (1):** When there is a play-out queue, the output **Rate** will be increased by one, and the **Intermission Minimal Value** will be used.
  - **Speed Up (2):** When there is a play-out queue, the output **Rate** will be increased by one. Depending on the queue size, the output rate can be increased by two. In both cases, the **Intermission Minimal Value** will be used.
  - **Shorten:** As when set to On, adding new text will add this to a queue to be played out automatically when the previous text is done, adjusted for the intermission value. However, the text will be shortened.
- **Queue Intermission (ms):** Defines the pause in milliseconds between text play-out when there is a text queue. The default value is `1000` (one second). This parameter is disabled when **Let Me Finish** is set to `Off`.
- **Minimal Queue Intermission (ms):** Defines the minimum pause in milliseconds between text play-out when there is a text queue and **Let Me Finish** is set to `Speed Up` or `Shorten`. The default value is `1000` (one second).
- **Play:** Starts play-out of the current text.
- **Pause:** Pauses the on-going audio play-out on the video out output.
- **Stop:** Stops play-out.
- **Read Languages:** Reads the Languages files from `%Programdata%\Vizrt\Viz3`.
  - Language files are used to improve the pronunciation of words. The Text2Speech plug-in utilizes the *Universal Phone Set (UPS)* machine-readable phonetic alphabet created by Microsoft for this purpose. The language files are also used to replace defined abbreviations, for example replacing *Av* with *Avenue.*
- **Read Profiles:** Reads profiles file from `%Programdata%\Vizrt\Viz3`.

14.16 SplineFX

The following container plug-ins are located in the SplineFX folder:

- 2D Follow

14.16.1 2D Follow

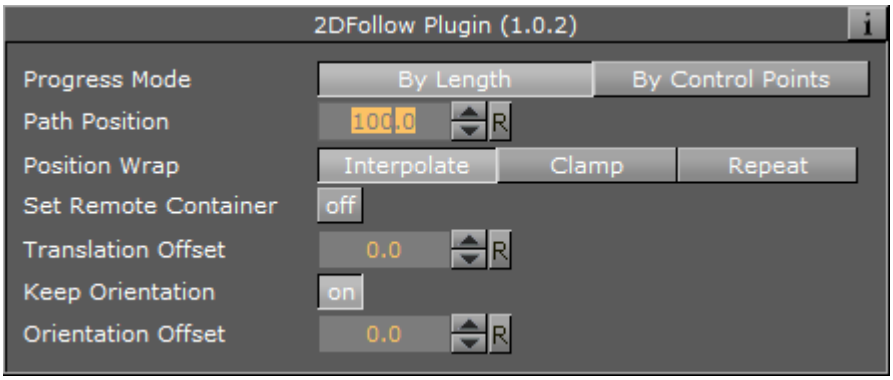
The 2D Follow function works together with the 2D Ribbon. What the plug-in basically does is to automatically create an animation path that follows the form of a 2D Ribbon. The 2DRibbon must be created first.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> SplineFX

This section contains information on the following topics:

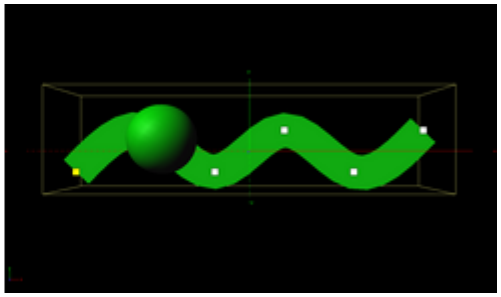
- 2D Follow Properties
- To Set up the 2D Follow plug-in

2D Follow Properties



- **Path Position**: Sets the position for the object along the path of the 2D Ribbon object. Animate this value to get an animation in the stage.
- **Position Wrap**: Can be set to **Interpolate**, **Clamp** or **Repeat**.
- **Translation Offset:** Sets an offset between the object and the 2D Ribbon path.
- **Keep Orientation**: Keeps the orientation between the object and the 2D Ribbon path as the object moves along the path when enabled.
- **Orientation Offset:** Offsets the orientation between the object and the 2D Ribbon path.

To Set up the 2D Follow plug-in



1. Add the 2D Ribbon geometry to the scene tree.
2. Open the 2D Ribbon editor and enable **Show Control Point Values**.
3. Set alternating values of 30.0 and -30.0 to the Y axis values, creating a wave shape.
4. Add the item(s) (e.g. a Sphere) you want to animate along the 2D Ribbon's path as a Sub-Container to the 2D Ribbon container.
5. Add the 2D Follow plug-in to the container.
6. Open the 2D Follow editor and animate its Path Position value from 0.0 to 100.0.

See Also

- 2D Ribbon
- Create Animations

14.17 TextFX

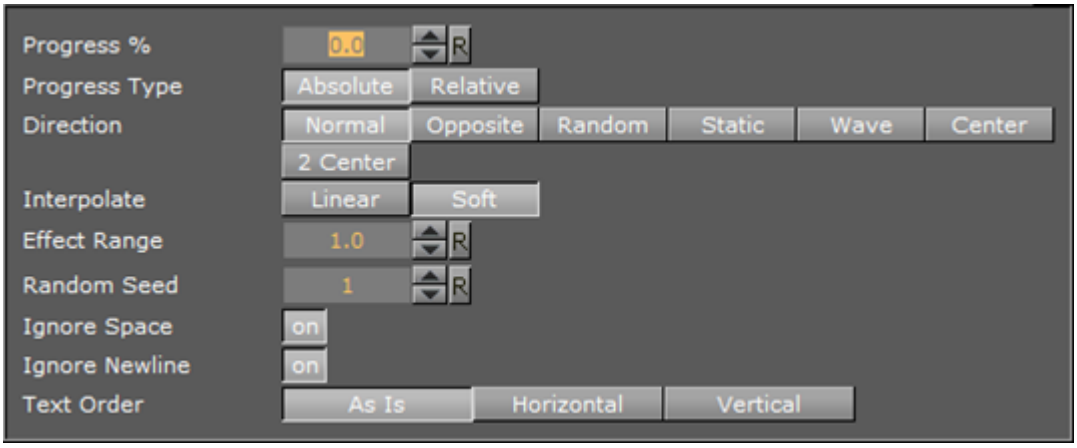
The following Container plug-ins are located in the TextFX folder:

- Common Text FX Properties
- Convert Case
- Mark Text
- Text FX Alpha
- Text FX Arrange
- Text FX Color
- Text FX Color Per Vertex
- Text FX Emoticons
- Text FX Explode
- Text FX Jitter Alpha
- Text FX Jitter Position
- Text FX Jitter Scale
- Text FX Plus Plus
- Text FX Rotate
- Text FX Scale
- Text FX Size
- Text FX Slide
- Text FX Vertex Explode
- Text FX Write

14.17.1 Common Text FX Properties

When in text editing mode, any TextFX effects are automatically deactivated to allow text input. As soon as the text editing is deactivated, either by selecting another container, or by just selecting a different container property within the selected container, such as transformation, the TextFX effect will be applied again in the scene preview.

The following properties are common to most of the Text FX plug-ins:



- **Progress %**: 0 percent progress represents the beginning of the effect, 100 percent the end. Animate this value from `0` to `100` to see the effect or from `100` to `0` to animate the effect backwards.
- **Progress Type**
  - **Absolute:** 100 percent progress will animate all characters of a text, regardless how many characters it has.
  - **Relative**: 100 percent progress animates 10 characters. This is needed to adjust the timing of several text objects with different sizes. The effect speed should be for example 5 characters per second, so the animation must be from 0 to 100 in two seconds. This will work for text with 10 characters or less. If you want to use longer texts, animate the progress value over 100 percent (10 percent for each character).
- **Direction**: Sets the direction of the text effect sequence, you can choose between the following options:
  - **Left**: Starts with the first character.
  - **Right:** Starts with the last character
  - **Random:** Uses a random order.
  - **Static**: All characters are processed at the same time.
  - **Wave**: Starts with the first character, animates the effect from 0 to 100 and then down again to 0.
  - **Center**: Starts the effect from the center of the text.
  - **2 Center**: Starts the effect at the same time from the beginning and the end of the text. They meet at the center.
- **Interpolate**: Choose between a **soft** or a **linear** interpolation of the transition from character to character.

- **Effect Range:** Defines how many characters are processed at the same time. If for example the Effect Range is set to 4, and you manually increase the progress value, you will see that when the fifth character starts to be processed, the first is finished, when the sixth starts, the second is finished, and so on.
- **Random Seed**: Specifies a seed for the random number generator when a random direction is chosen. Even though Viz Artist uses random numbers, the animation for a specific random seed will always look the same. This is typically useful if you combine two different text effects.
- **Ignore Space:** Ignores space when animating the effect.
- **Ignore Newline:** Ignores new lines when animating the effect.
- **Text Order**: Sets the text order for the effect. Available options are As Is, Horizontal or Vertical. Horizontal and Vertical enables the Text Direction options.

14.17.2 Convert Case

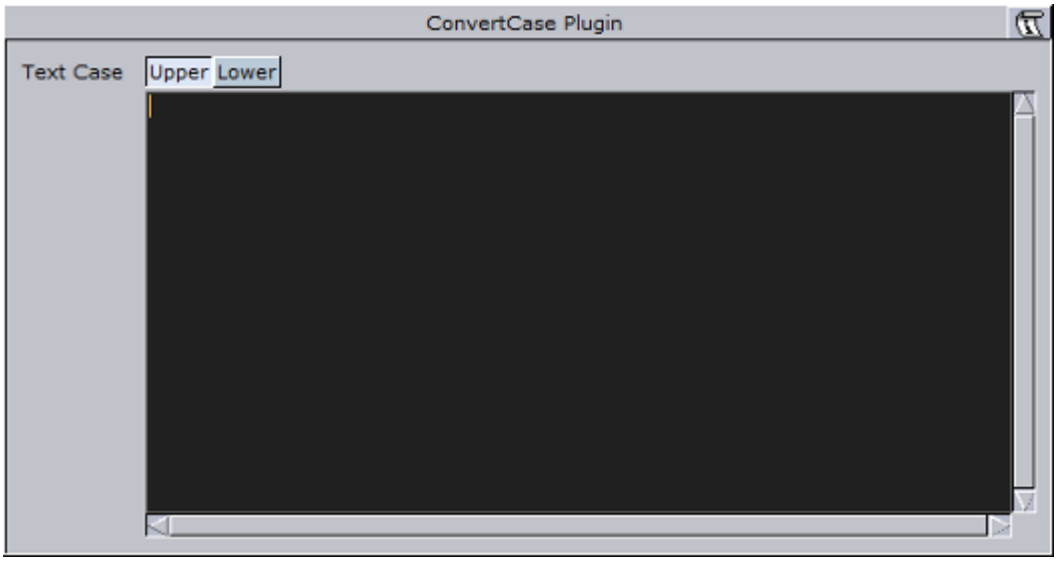
Converts ASCII strings to upper or lower case. Current container must hold a string.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFx

This section contains information on the following topics:

- Convert Case Properties
- To Convert a String to Upper or Lower Case

Convert Case Properties



- **Upper/Lower:** Converts the text for the specific container to upper or lower case.

```
FUNCTION*ConvertCase*text_case - int; 0: upper, 1: lower FUNCTION*ConvertCase*text - string
```

To Convert a String to Upper or Lower Case

1. Add a group to the scene tree, and add a font to it.
2. Add the Convert Case plug-in to the same container.
3. Open the Convert Case editor, and enter text in the editor.
4. Click the Upper button to convert to upper case text, or click the Lower button to convert to lower case text.

14.17.3 Mark Text

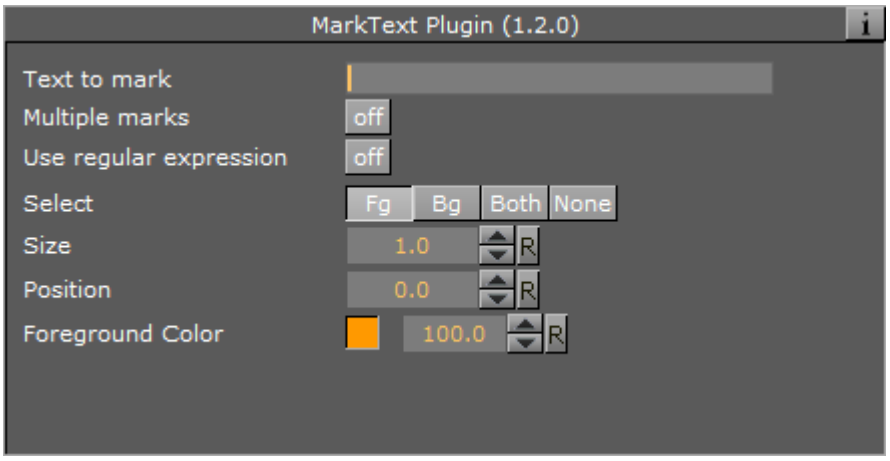
The Mark Text plug-in allows you to highlight or underline words in a text string.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Mark Text Properties
- To Mark Text

Mark Text Properties



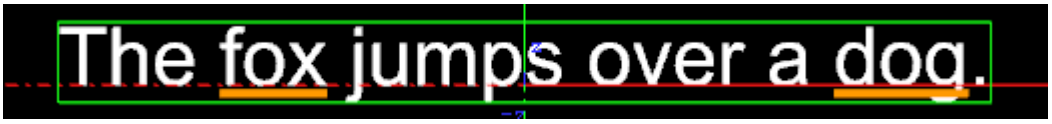
- **Text to Mark:** Sets the text that should be highlighted. The text is case sensitive.
- **Multiple marks:** Highlights all matched text.

- **Use regular expression**: Allows regex to be used for more sophisticated highlighting.

  - **Only matched group:** Highlights only on capture-groups `(value)`, another matched texts outside capture-groups or inside non-capture-groups `(?:value)` will not.
  - **Case insensitive:** Matches both upper and lower case.
  - **Grammar:** Selects one of the following supported grammars:
    - ECMA
    - Basic
    - Extended
    - Awk
    - Grep
    - Egrep
- **Mark:** Selects which text should be highlighted. Available options are foreground (Front), background (Shadow), Both and None. Background will highlight the text's shadow.
- **Size**: Sets the thickness of the highlight.
- **Position**: Sets the y-position of the highlight in relation to the text.
- **Depth:** Sets the z-position of the highlight in relation to the text. This allows the highlight to move behind the text.
- **Foreground Color**: Sets the color and alpha value of the highlight.

To Mark Text

1. Add a group container to the scene tree and name it Text.
2. Add a font to the Text container.
3. Open the font editor and enter a text string.
4. Add the Mark Text plug-in to the Text container.
5. Open the Mark Text Editor and enter a word from your text string or regex if enabled.



From this example:

- Text to mark: `(?:the|a) (\w+)`
- Use regular expression: on
- Only matched group: on
- Case insensitive: on
- Grammar: ECMA

⚠ **Note:** Do not change the font type between texture and geometry as this will not affect the whole string when using the Mark Text plug-in.

⚠ **Note:** Overlapping marks could occur with narrow character spacing or kerning. This would be visible in case of transparent marks.

14.17.4 Text FX Alpha

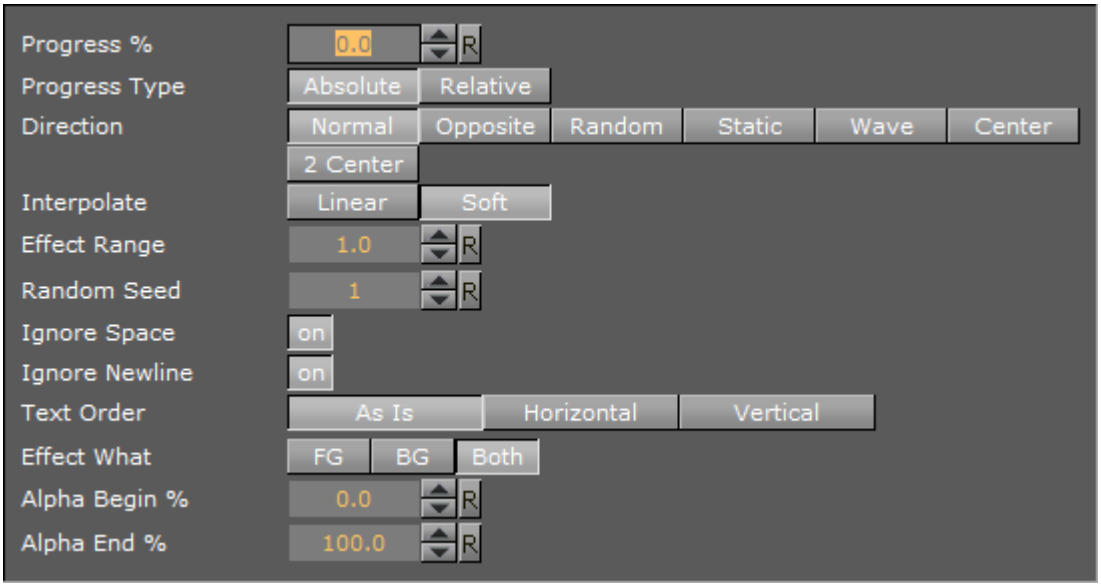
The Text FX Alpha plug-in is a text effect that creates a fade in effect for the text characters. The effect sequence can be set to go many different ways. A fade in and out effect can also be achieved.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Alpha Properties

Text FX Alpha Properties



- **Alpha Begin %**: Sets the alpha level of the character at 0% effect.
- **Alpha End %**: Sets the alpha level of the character at 100% effect.

See Also

- Common Text FX Properties

14.17.5 Text FX Arrange

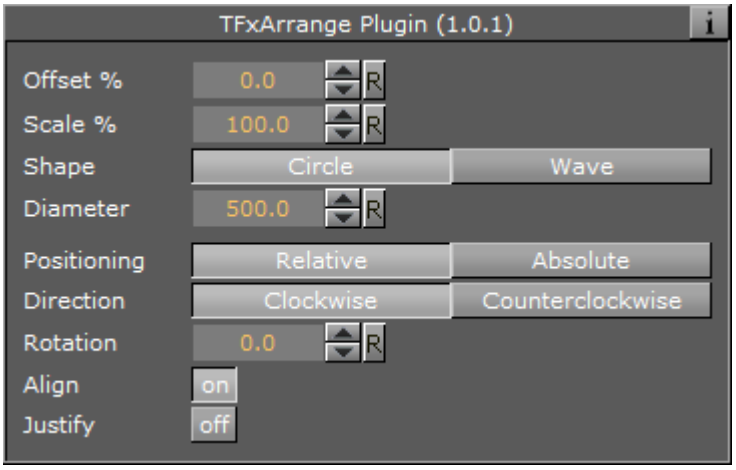
The Text FX Arrange plug-in allows you to arrange characters in either a circular- or a wave shape. The characters can be animated on the selected shape by animating the offset value.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Arrange Properties
  - Circle Parameters
  - Wave Parameters

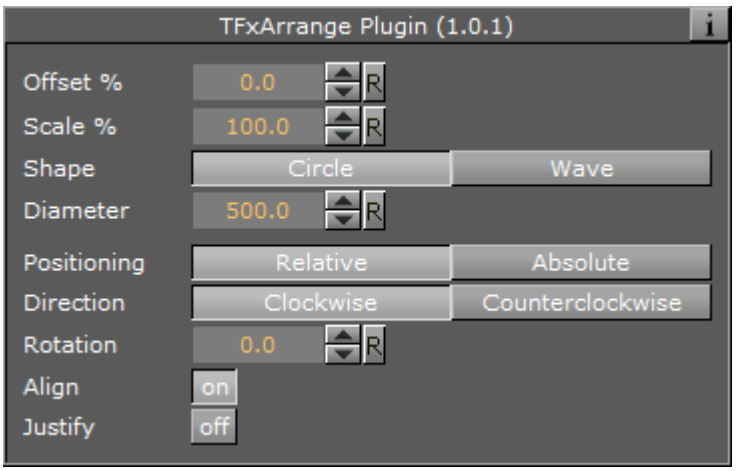
Text FX Arrange Properties



- **Offset %**: Moves the characters on the shape. 100% means one full rotation of the text on the circle.
- **Scale %**: Sets the text on the shape. The parameter does not scale the characters itself, but the kerning of the characters.
- **Shape**: Changes the shape of the character layout to **Circle** or **Wave**.

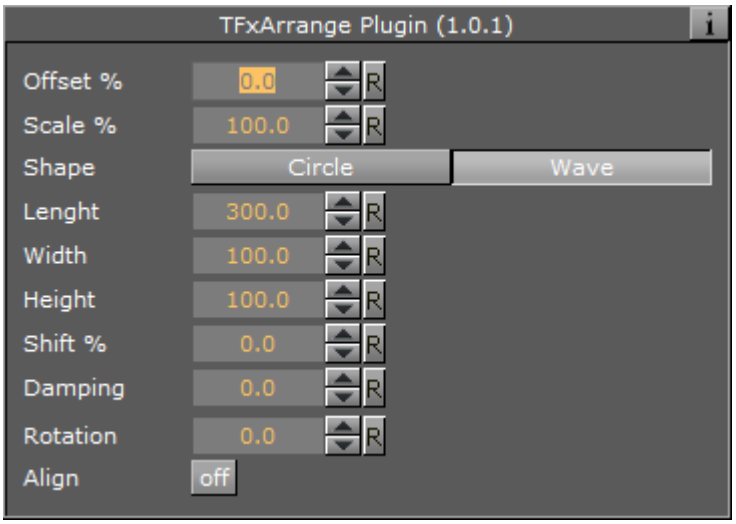
> ⚠ **Note:** The shape of the text transformation is not visible in the Scene Editor while the Text Editor is active.

Circle Parameters



- **Diameter**: Sets the diameter of the circular shape.
- **Positioning:** Defines the position of the text on the circular shape:
  - **Relative:** Maintains the text spacing of the text object.
  - **Absolute:** Distributes the text evenly onto the circle.
- **Direction**: Sets the direction of the text on the circle to either:
  - **Clockwise,** or
  - **Counterclockwise**.
- **Rotation:** Rotates the characters on the X-axis.
- **Align**: Rotates the characters on the Z-axis to align the X-axis with the tangent of the shape at the characters position.

Wave Parameters



- **Length**: Sets the wave length. A high value creates many and small waves, a low value creates fewer and bigger waves.
- **Width:** Sets the width of the wave.

- **Height:** Sets the height of the waves without altering the number of waves as the **Length** does.
- **Shift %:** The wave usually starts at the height 0. The shift value moves that point, thereby moving the texts position on the wave structure. Setting **Shift %** to 100% brings you back to the beginning.
- **Damping:** Modifies the amplitude of the wave curves.
- **Rotation:** Rotates the characters on the X-axis.
- **Align**: Rotates the characters on the Z-axis to align the X-axis with the tangent of the shape at the characters position.

14.17.6 Text FX Color

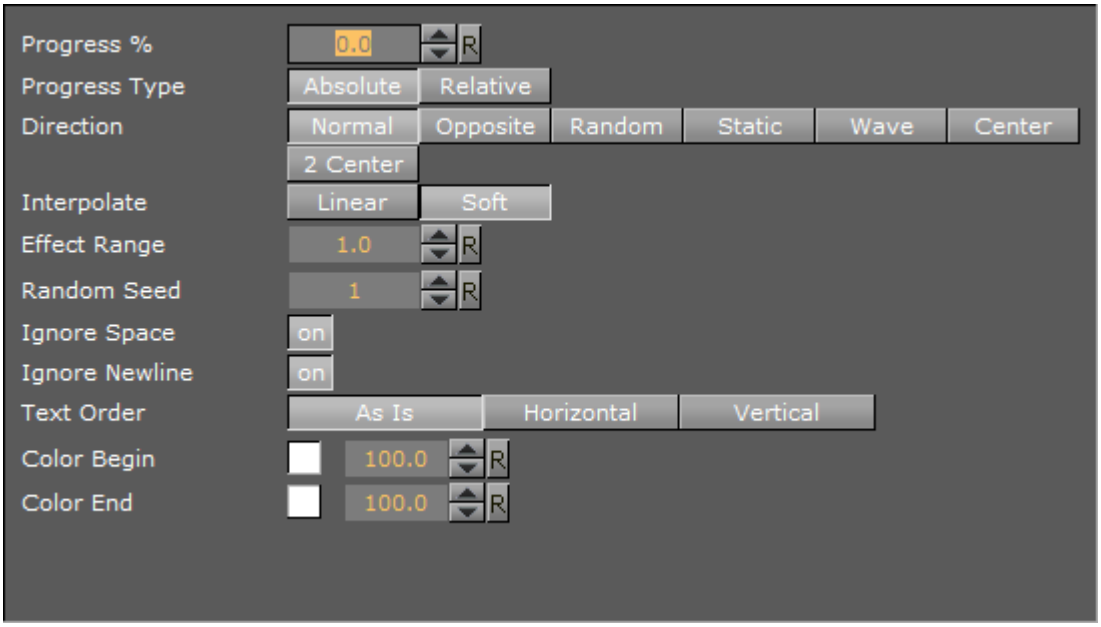
Adds a color effect to the text.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Color Properties

Text FX Color Properties



- **Color Begin:** Sets the initial color before applying the effect.
- **Color End:** Sets the color the effect should apply.

See Also

- Common Text FX Properties

14.17.7 Text FX Color Per Vertex

Adds a color per vertex effect to the text.

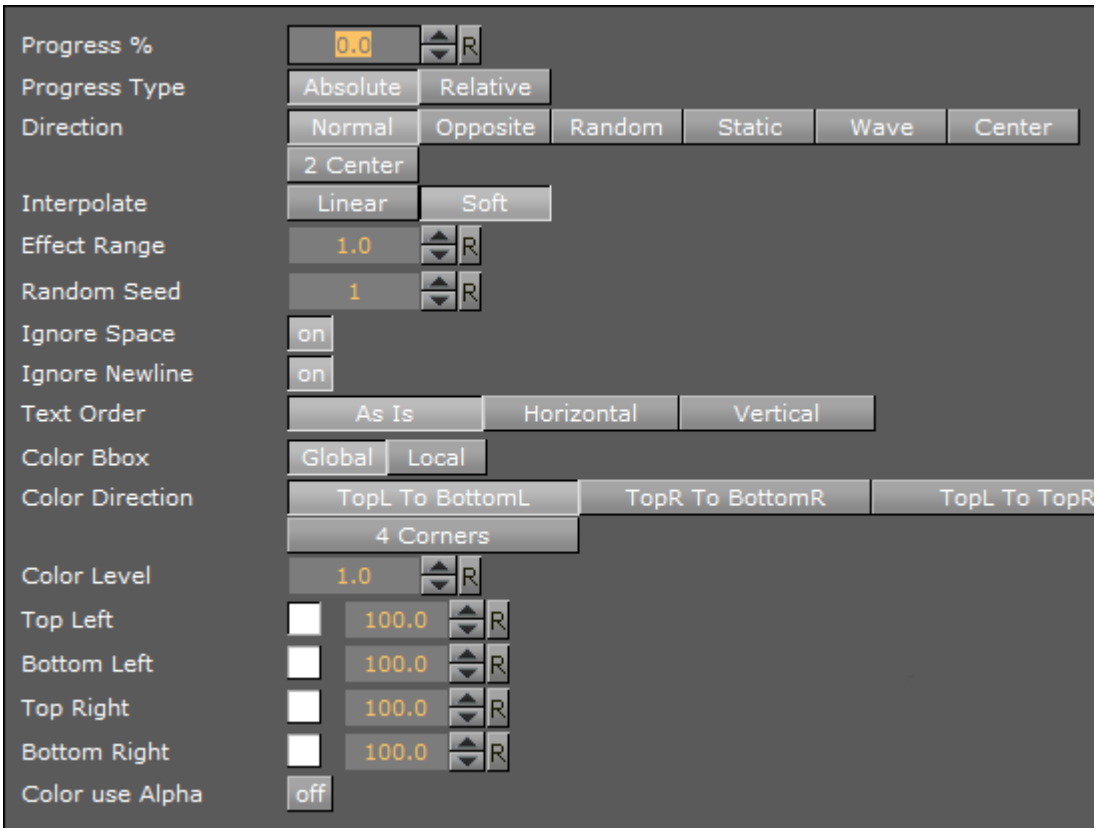
> **Note:** Works only if the text is set to texture (does not work with geometry text).

> **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Color Per Vertex Properties

Text FX Color Per Vertex Properties



- **Color Bbox:** Defines the reference bounding box, global for the entire text, or local for each character.
- **Color Direction:** Defines the color gradient shape. Available options are; top left to bottom left, top right to bottom right, top left to top right, bottom left to bottom right, top right to bottom left and 4 corners.
- **Color Level:** Sets the level of the color (0.1 to 1.9). Default is 1.0.
- **Top and Bottom Left and Right:** Sets the color and alpha value for each edge.
- **Color use Alpha:** Enables alpha.

See Also

- Common Text FX Properties

14.17.8 Text FX Emoticons

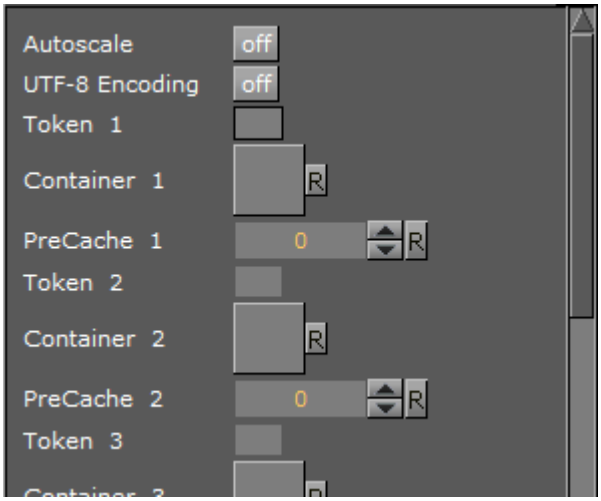
The Text FX Emoticons allows you to create an emoticon container, add it to the Emoticons plug-in and replace characters (tokens) in a text string with an assigned container.

> ⚠ **IMPORTANT!** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Emoticons Properties

Text FX Emoticons Properties



- **Autoscale:** Scales the emoticon containers so their height above the baseline (0) is identical to the height of character A.

- **UTF-8 Encoding:** Enables UTF-8 encoding for the text effect.
- **Token 1 - n:** The placeholder for an emoticon.
- **Container 1 - n:** References the container holding the emoticon. The container should be similar to the layout of a glyph.
- **Precache 1-n:**- Enables Viz Artist/Engine to cache the objects.

14.17.9 Text FX Explode

The Text FX Explode plug-in creates an explosion like function where the characters get thrown away from their initial position. The speed, direction and spread of the moving characters can be altered with parameters.

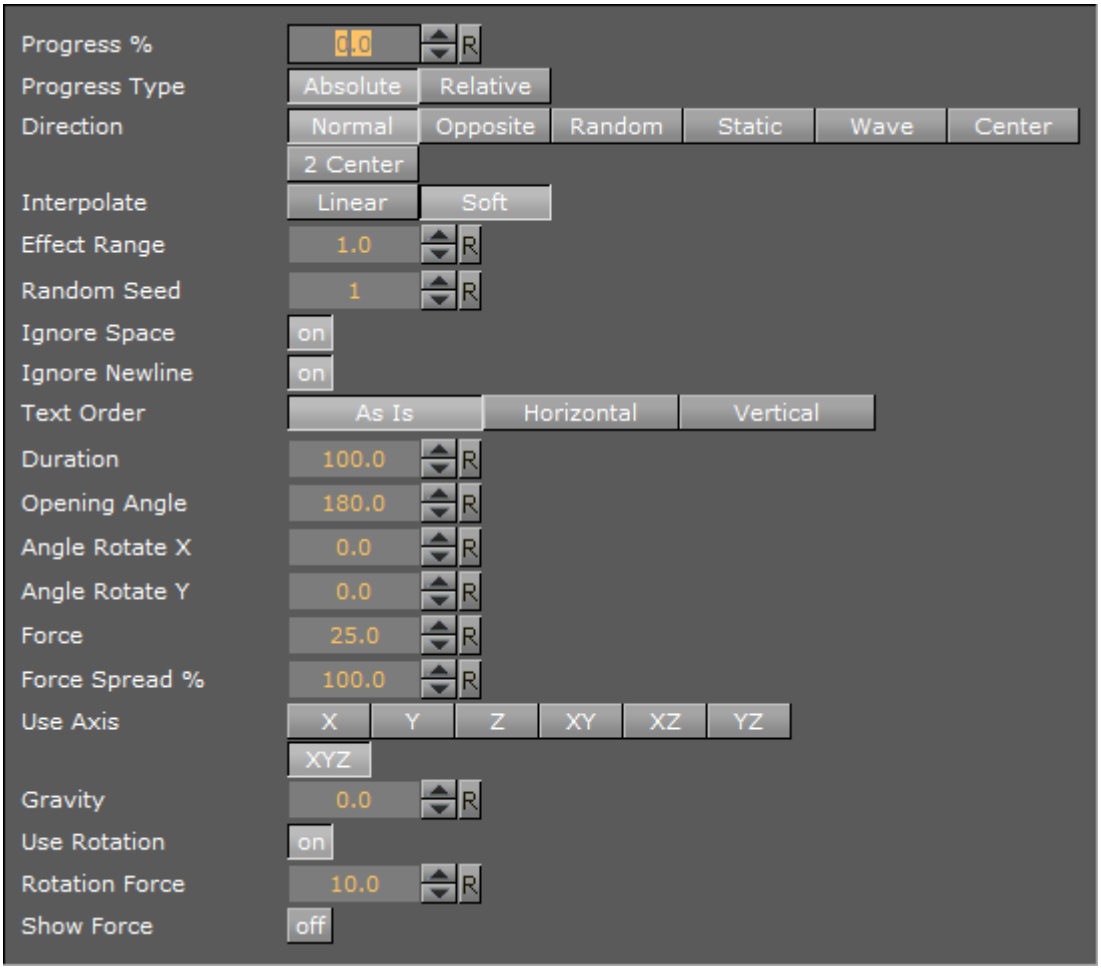
⚠ **Note:** Works only if the text is set to texture (does not work with geometry text).

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Explode Properties

Text FX Explode Properties



- **Duration:** Defines the duration of the progressing for each of the characters. If you want a longer progressing, increase this value. You could achieve the same by making the gravity stronger and animating the progress slower, but it is more easy to increase the duration instead if your explode effect is too short.
- **Opening Angle:** Sets the angle for the spread of the characters. 0 will send them straight up, 360 will spread them in a circular shape.
- **Angle Rotate X**: Rotates the opening angle around the X-axis.
- **Angle Rotate Y**: Rotates the opening angle around the Y-axis.
- **Force**: Sets the force that throws away the characters. A high force will make them go far away, conversely a low force will create only a small motion of the characters.
- **Force Spread %:** Sets a variation of the force among the characters.
- **Use Axis**: Allows you to select on which axis or combination of axes the characters are to spread along.
- **Gravity**: Sets a gravity force that influence the path of the characters to end up going downwards. The higher the value is set, the faster each character will divert from its initial path and start going downwards.

- **Use Rotation**: Rotates characters as they are being thrown away from their initial position when enabled.
- **Rotation Force:** Sets the degree of rotation as the characters are being thrown away.
- **Show Force**: Shows lines showing the characters path and speed in the Scene Editor when enabled.

See Also

- Text FX Vertex Explode
- Common Text FX Properties

14.17.10 Text FX Jitter Alpha

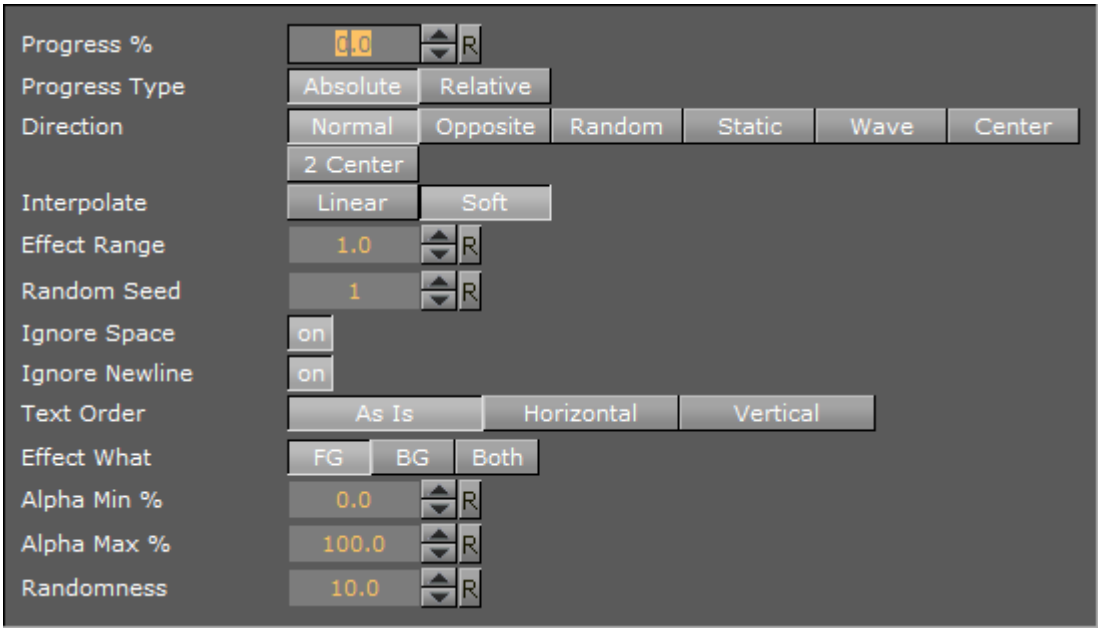
The Text FX Jitter Alpha plug-in creates a jittering motion of the characters by randomly changing the alpha value of each character. The degree of change and the start sequence of the jittering can be altered. To use the plug-in, add it onto a container with a font. To create an animation, animate the progress. Other values can of course be animated as well.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Jitter Alpha Properties

Text FX Jitter Alpha Properties



- **Effect What:** Defines where the effect should have an effect. Available options are background (BG), foreground (FG) or both.
- **Alpha Begin:** Sets the alpha level of the character at 0% effect.
- **Alpha End**: Sets the alpha level of the character at 100% effect.
- **Randomness:** sets the intensity of the jittering alpha changes.

14.17.11 Text FX Jitter Position

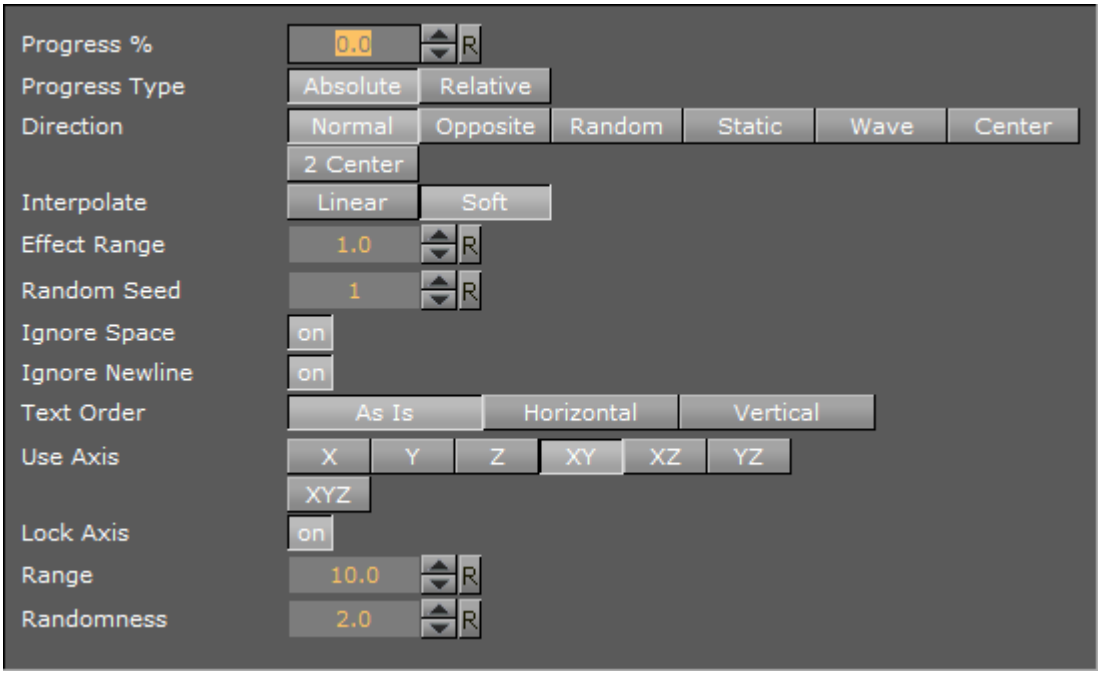
The Text FX Jitter Position plug-in creates an jittering motion of the characters by randomly changing the position of each character. The degree of position change and the starting sequence of the jittering can be altered. To use the function add it onto a container which holds a font. To create an animation, animate the progress. Other values can of course be animated as well.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Jitter Position Properties

Text FX Jitter Position Properties



- **Use Axis:** Defines on which axis or axes the characters moves to create the jittering effect.

- **Lock Axis:** Sets **Range** and **Randomness** to the same value for all axes when enabled. If you disable it, **Range** and **Randomness** for each of the axes will be shown and you can set them individually.
- **Range:** Sets the range of the jittering movement.
- **Randomness:** Sets the intensity of the jittering movement.

See Also

- Common Text FX Properties

14.17.12 Text FX Jitter Scale

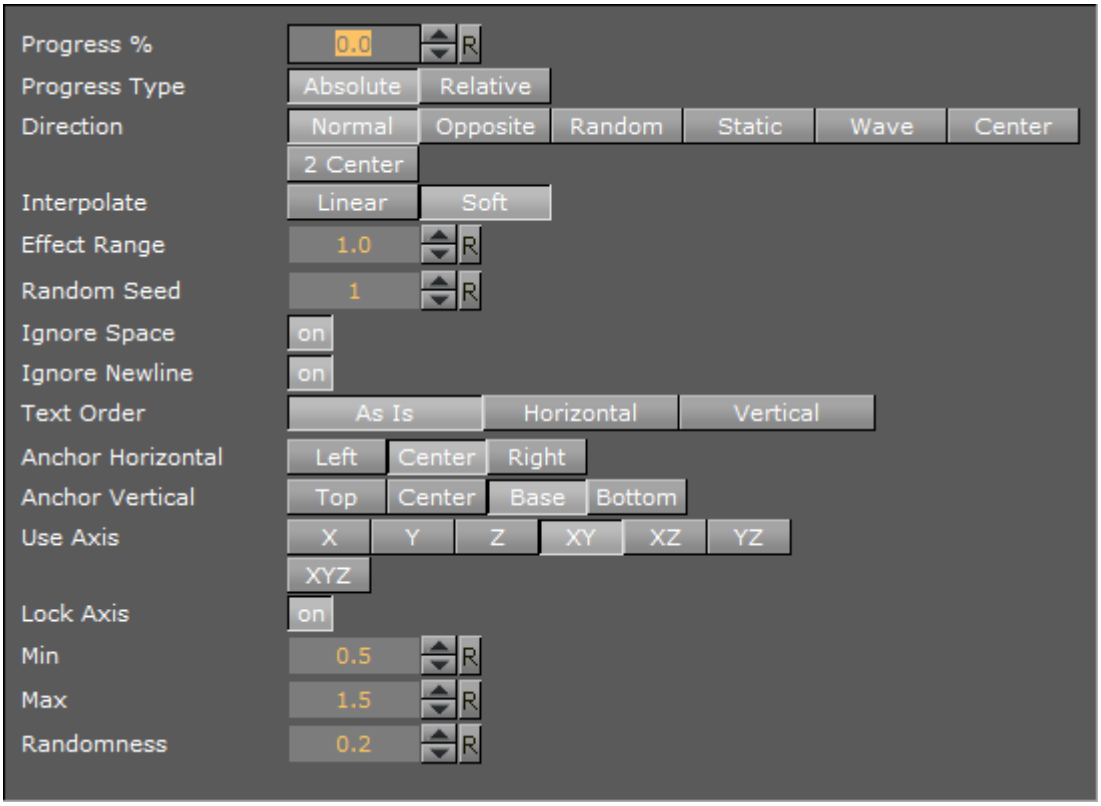
The Text FX Jitter Scale animates a jittering effect on the scale of the character.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Jitter Scale Properties

Text FX Jitter Scale Properties



- **Anchor Horizontal**: Sets the anchor point for the characters on the horizontal plane.
- **Anchor vertical**: Sets the anchor point for the characters on the vertical plane.
- **Use Axis**: Defines on which axis or axes the characters scale to create the jittering effect.
- **Lock Axis:** Gives all the axes get the same **Min**, **Max** and **Randomness** settings when enabled. If you disable randomness, these parameters will be visible for each of the axes and must be set individually.
- **Min**: Sets the minimum scaling for the characters.
- **Max**: Sets the maximum scaling for the characters.
- **Randomness:** Sets the intensity of the jittering movement.

See Also

- Common Text FX Properties

14.17.13 Text FX Plus Plus

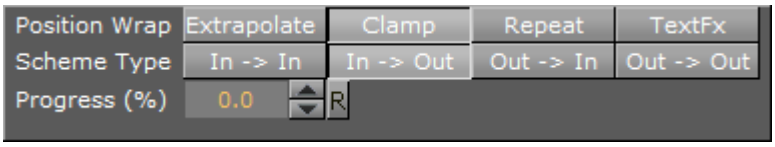
The Text FX Plus Plus plug-in allows you to set a number of effects on a text string.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

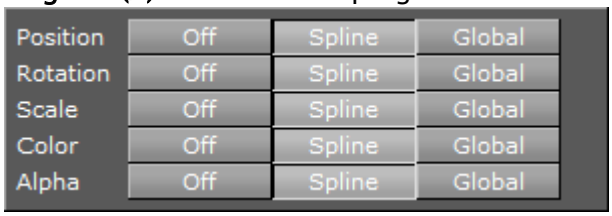
This section contains information on the following topics:

- Text FX Plus Plus Properties

Text FX Plus Plus Properties

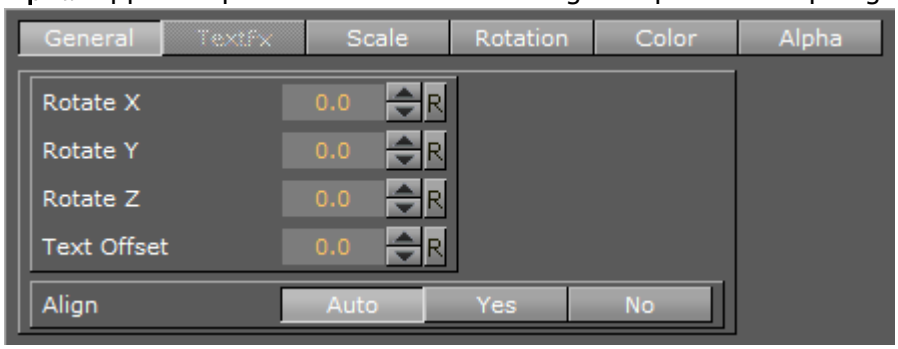


- **Position Wrap**
  - **Extrapolate:** Above 100% or below 0% of Path Position will continue the direction of the spline.
  - **Clamp:** Stops the string on the spline endpoints.
  - **Repeat:** Causes the string to move to the other spline endpoint as soon as it moves over the endpoint.
  - **TextFx**: Positions characters by the relative spline of each character using the common TextFx parameters.
- **Scheme Type**: Defines how the text will look at 0% and 100% progress.
  - **In -> In:** at 0% first character will be at the beginning of the spline, at 100% last character will be at the end of the spline.
  - **In -> Out:** at 0% first character will be at the beginning of the spline, at 100% first character will be at the end of the spline.
  - **Out -> In**: at 0% last character will be at the beginning of the spline, at 100% first character will be at the end of the spline.
  - **Out -> Out:** at 0% last character will be at the beginning of the spline, at 100% last character will be at the end of the spline.
- **Progress (%):** Animates the progress of the effect(s).



- **Position:** Positions the characters using the spline as a position axis.
- **Rotation:** Rotates the characters using the spline as a rotation axis.
- **Scale:** Scales the character's X, Y and Z-axis using the spline as a scaling graph.
- **Color:** Applies Color Change on character.

- **Alpha:** Applies Alpha on the characters using the spline as a alpha graph.



- **General:** Enables rotation, offset and align properties.
  - **Rotate X/Y/Z:** Adds extra Rotation to each character.
  - **Text Offset:** Sets the text offset.
  - **Align:** Rotates the characters to follow the spline movements.
- **TextFX:** See Common Text FX Properties.
  - **Direction:** Sets the direction of the text effect sequence. Normal starts with the first character (e.g. left). Opposite starts with the last character (e.g. right). Random uses a random order (see Random Seed). Static means that all characters are processed at the same time. Wave starts with the first character, animates the effect from 0 to 100% and then down again to 0%. Center starts the effect from the center of the text. 2 Center starts the effect at the same time from the beginning and the end of the text. They meet at the center.
- **Scale:** Defines the scale graph to apply on the characters for each axis.
- **Rotation**: Defines the rotation effect to apply on the characters for each axis.
- **Color**: Defines the color gradient effect to apply on the characters.
- **Alpha:** Defines the alpha effect to apply on the characters.

See Also

- Common Text FX Properties

14.17.14 Text FX Rotate

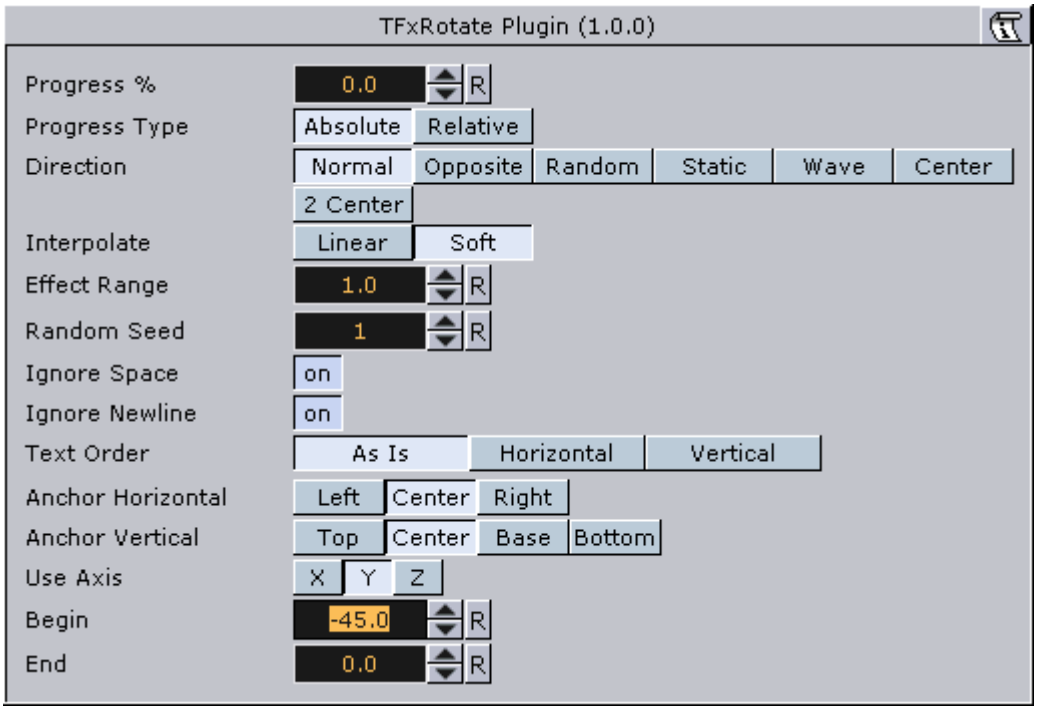
The Text FX Rotate plug-in allows you to create an effect where the characters rotate on the X-, Y- or Z-axis.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Rotate Properties

Text FX Rotate Properties



- **Anchor Horizontal:** Sets the anchor point for the characters on the horizontal plane.
- **Anchor Vertical:** Sets the anchor point for the characters on the vertical plane.
- **Use Axis**: Defines on which axis the characters rotate.
- **Begin**: Sets the initial rotated position of the characters.
- **End**: Sets the ending rotated position of the characters.

See Also

- Expert
- Common Text FX Properties

14.17.15 Text FX Scale

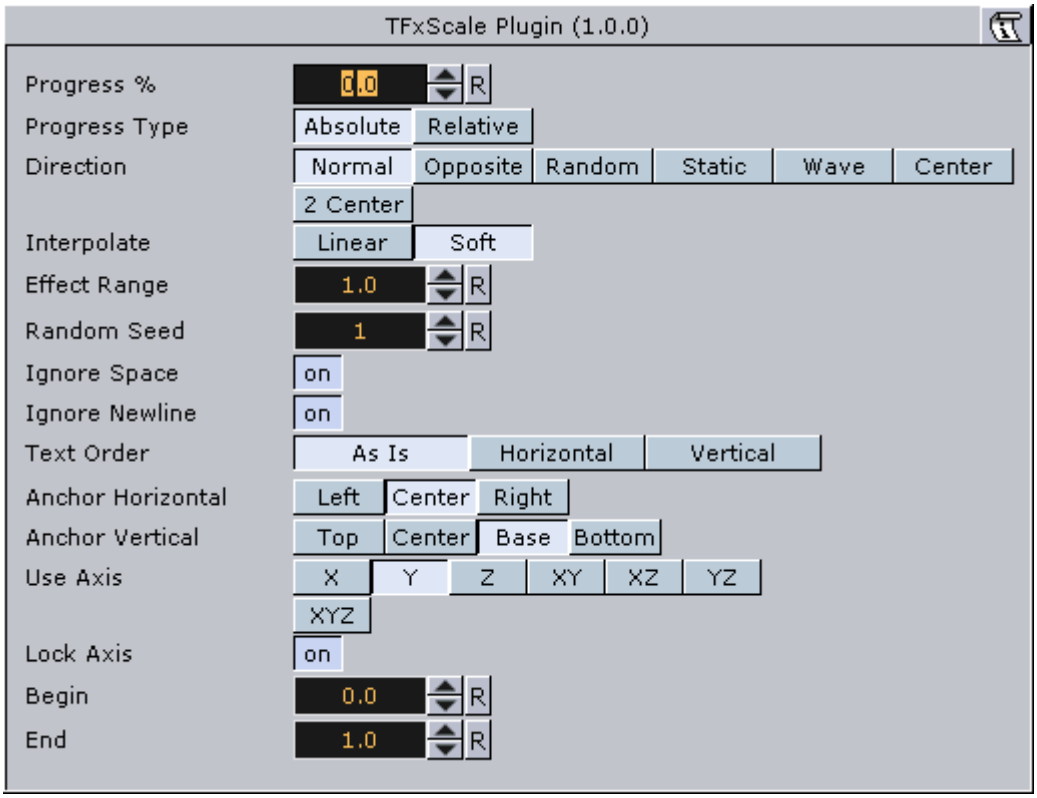
The Text FX Scale plug-in allows you to create a scaling animation of the characters.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Scale Properties

Text FX Scale Properties



- **Anchor Horizontal:** Sets the anchor point for the characters on the horizontal plane.
- **Anchor Vertical**: Sets the anchor point for the characters on the vertical plane.
- **Use Axis**: Defines on which axis or axes the characters scale.
- **Lock Axis:** Scales the **Begin**- and **End** the same for all the axes when enabled. If you disable the parameter, **Begin** and **End** must be set for all the axes individually.
- **Begin**: Sets the initial size of the characters.
- **End**: Sets the ending size of the characters.

See Also

- Text FX Size
- Common Text FX Properties

14.17.16 Text FX Size

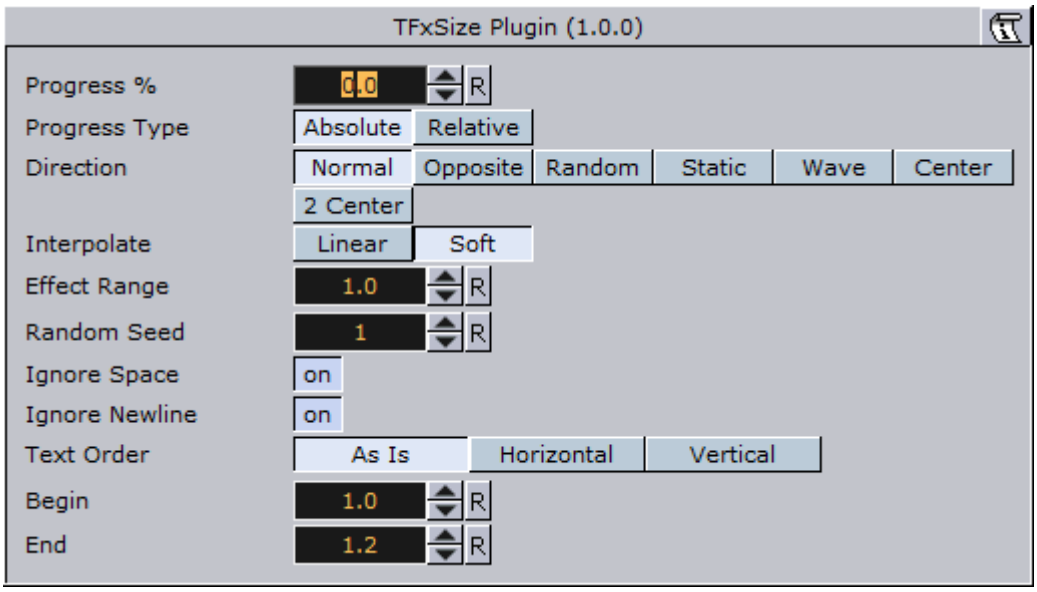
The Text FX Scale plug-in allows you to create a sizing animation of the characters.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Size Properties

Text FX Size Properties



- **Begin:** Sets the initial scale of the text before the effect.
- **End:** Sets the final scale of the text after the effect.

See Also

- Text FX Scale
- Common Text FX Properties

14.17.17 Text FX Slide

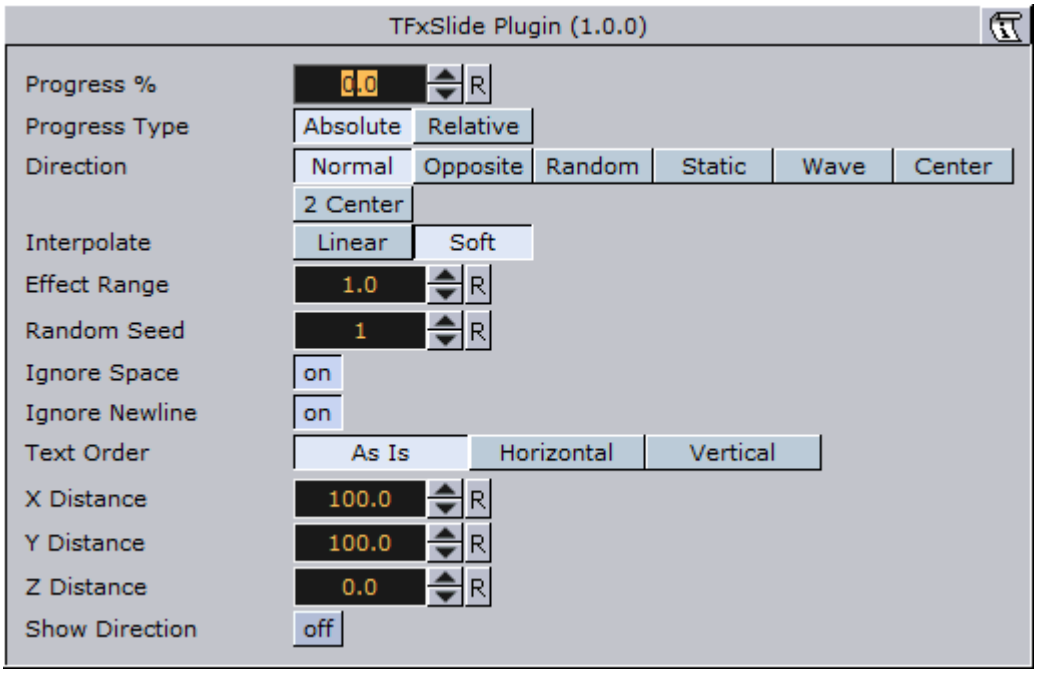
The Text FX Slide plug-in allows you to create a sliding animation of the characters on the X, Y and Z-axis.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Slide Properties

Text FX Slide Properties



- **X, Y and Z Distance:** Defines direction and distance of moved text.
- **Show Direction:** Enables and disables view of 'movement' vector.

> ⚠ **Note:** The direction will not be shown in On Air mode.

See Also

- Common Text FX Properties

14.17.18 Text FX Vertex Explode

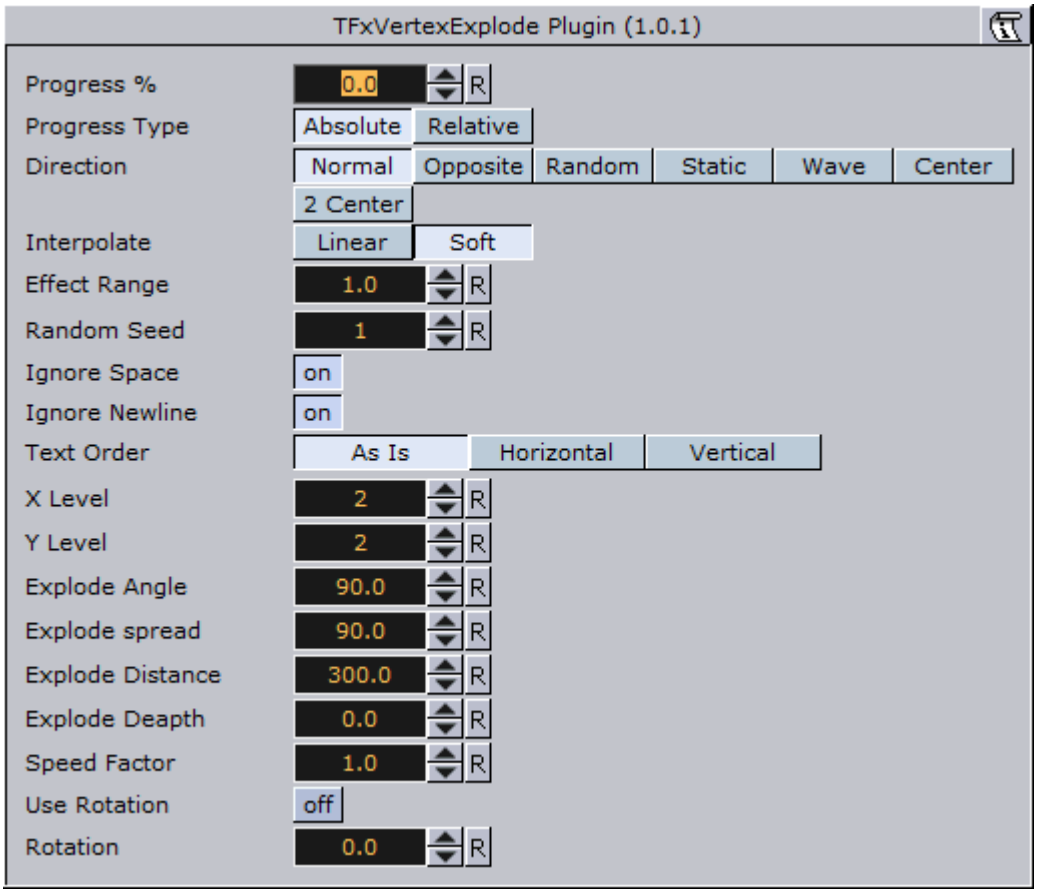
The Text FX Vertex Explode plug-in allows you to create an effect where text characters are exploded with a vertex effect.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Vertex Explode Properties

Text FX Vertex Explode Properties



- **X, Y and Z Distance:** Defines direction and distance of moved text.
- **Show Direction:** Enables and disables view of 'movement' vector.

> ⚠ **Note:** The direction will not be shown in On Air mode.

See Also

- Text FX Explode
- Common Text FX Properties

14.17.19 Text FX Write

The Text FX Write plug-in allows you to create an effect where text characters get shown as if they were written by the keyboard in real-time, a sort of Typewriter effect.

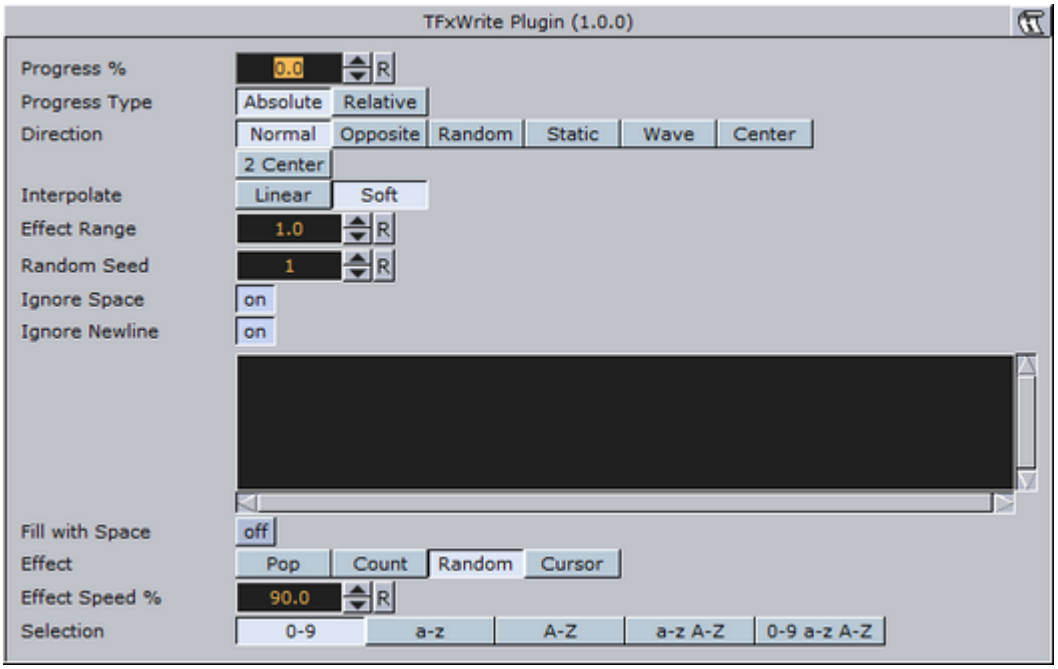
The text field in the font editor does not need to contain any text, as the text for this plug-in is entered in the plug-in editor. Text FX Write expects UTF-8 encoded text as input, if a font encoding other than *Default* is selected (see the **Font/Text Options** page in the **Configuring Viz** section of the Viz Engine Administrator Guide).

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> TextFX

This section contains information on the following topics:

- Text FX Write Properties

Text FX Write Properties



- **Fill with Space**: Adds a space for every character not yet written when enabled. The bounding box of the text is then sized correctly from the start. If you use an **Auto Follow** function connected to the text, enabling this option will create a better result.

- **Effect:** Allows you to select an effect to be executed as each character gets written on the screen. There are four effects to choose from and they have in addition their own parameters.
  - **Pop:** Creates the basic effect where the characters just pop up on the screen without any other sign being shown.
  - **Count:** Creates a counting sequence by showing a random symbol before showing the character to be shown.
  - **Random**: Writes random characters before the final character is shown (in combination with effect range). Use a non proportional font when you use this function to get the better looking effect.
  - **Cursor:** Shows a cursor that writes the characters. When you select this option, a text field opens below where you can choose which symbol to use as cursor.
- **Effect Speed:** Sets the speed of the effect that shows the preceding symbol. This is an additional parameter that is visible if you have selected **Count** or **Random**.
- **Selection**: Allows you to select between some preset character ranges to use for the effect. This is an additional parameter to the **Random** effect.
- **Cursor:** Allows you to enter the character which is to be used as a cursor.

See also

- Common Text FX Properties

14.18 Texture

The following container plug-ins are located in the Texture folder:

- BrowserCEF
- GeoGraffiti
- Grabbit
- GraffitiTex
- Image Clip
- Light Blur
- MoViz
- Noise
- SoftClip
- Tex Component

14.18.1 BrowserCEF

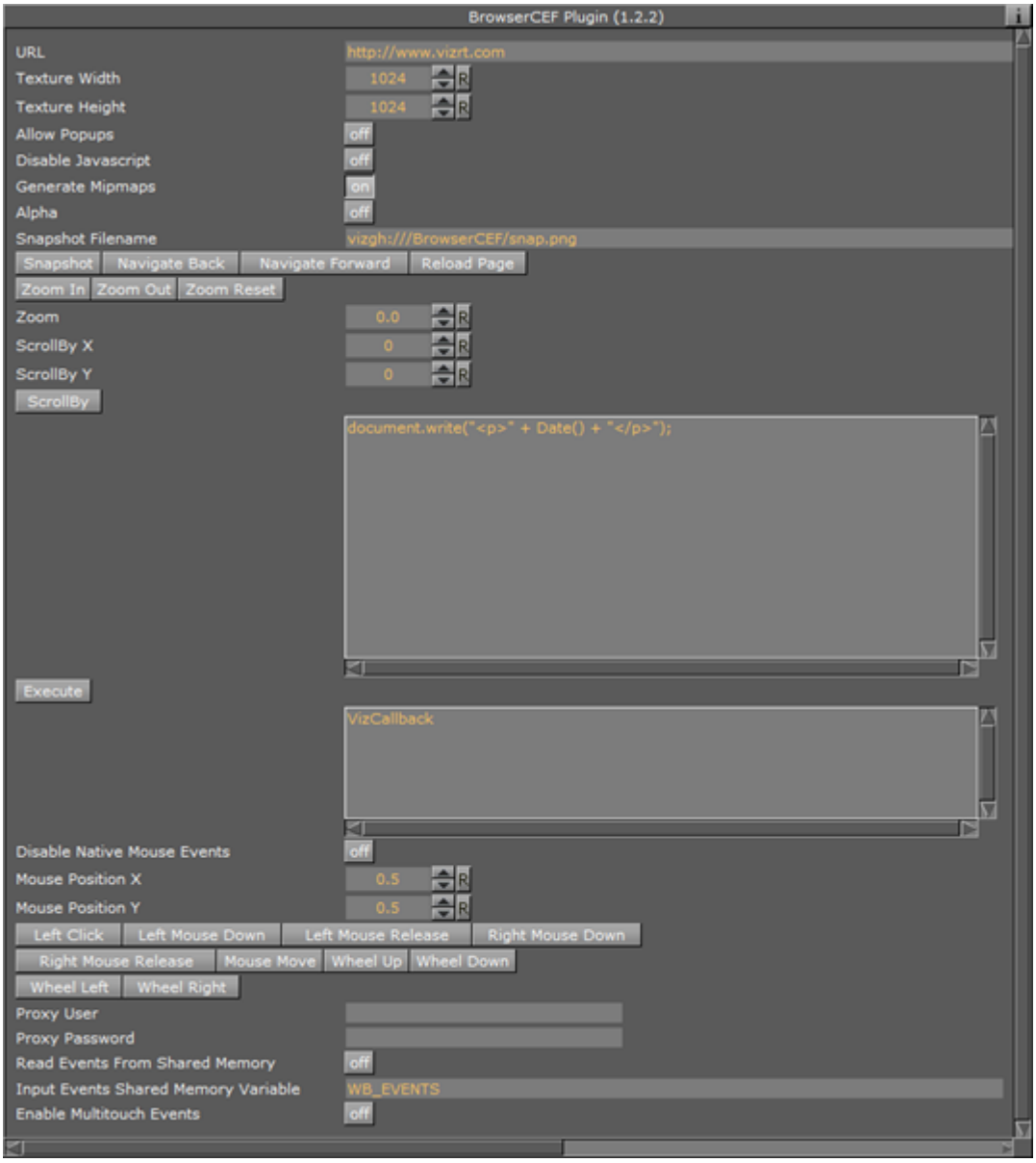
The BrowserCEF plug-in gives the ability to stream browser content on a Texture.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

This page contains the following topics and procedures:

- BrowserCEF Properties
- To Create a Simple Scene with BrowserCEF
- Interact with a Scene with JavaScript
- Example HTML Document

BrowserCEF Properties



- **URL:** Enter the URL address to be opened

- **Texture Width:** Width in pixels of the browser window
- **Texture Height:** Height in pixels of the browser window
- **Allow Popups:** Allow the browser to render popup windows. The popup will be rendered into the current texture. Close the popup by navigating back
- **Disable JavaScript:** Disable JavaScript execution
- **Generate Mipmaps:** Generate all mipmap levels.
  - Set to OFF for more performance
  - Set to ON for more quality when the texture is rendered on a smaller area on the screen than the width/height of the browser itself
- **Alpha:** Enable a transparent browser window
- **Snapshot Filename:** The full path and name for saving a snapshot. The location can be either a local file or on the Graphic Hub, and must be specified as a Uniform Resource Identifier (URI), for example:
  - **File**: `file:///c:/temp/BrowserCEF/example.png`
  - **Graphic Hub**: `vizgh:///BrowserCEF/example.png`
  - **Graphic Hub UUID**: `vizgh:///<F3F3F087-B85A-4AEE-B5019F78CB6AE5EC>/example.png`
- **Snapshot:** Take a snapshot of the current browser texture. The snapshot is saved as a file or to the Graphic Hub, as specified in **Snapshot Filename**
- **Navigate Back:** Navigate back to the previous page
- **Navigate Forward:** Navigate forward to the next page
- **Reload Page:** Reload the current page
- **Zoom In:** Zoom into the page, incrementing per click
- **Zoom Out:** Zoom out of the page, incrementing per click
- **Zoom Reset:** Reset zoom to default value
- Zoom: Enter the desired zoom value. The fields supported range is from `-10.0` to `10.0`. The default value is `0.0`.
  Like any other parameter, the value of this field can be controlled from a ControlObject. Please enable **Show Advanced** and then add the parameter under **Channel**:
  ControlObject exposure with direct address

  ```
  zoom:MAIN_SCENE*TREE*$Browser*FUNCTION*BrowserCEF*zoom:float:0.0:-10.0:10.0:zoom level
  ```

  ControlObject exposure with control channel address

  ```
  zoom:MAIN_SCENE*TREE*@zoom:float:0.0:-10.0:10.0:zoom level
  ```

  ⚠ **Note:** The BrowserCEF plug-in works asynchronously. Therefore, the zoom value needs to be applied after a successful load of an URL. This means that the value stored in the plug-in may not be the currently active zoom value until it has been applied.

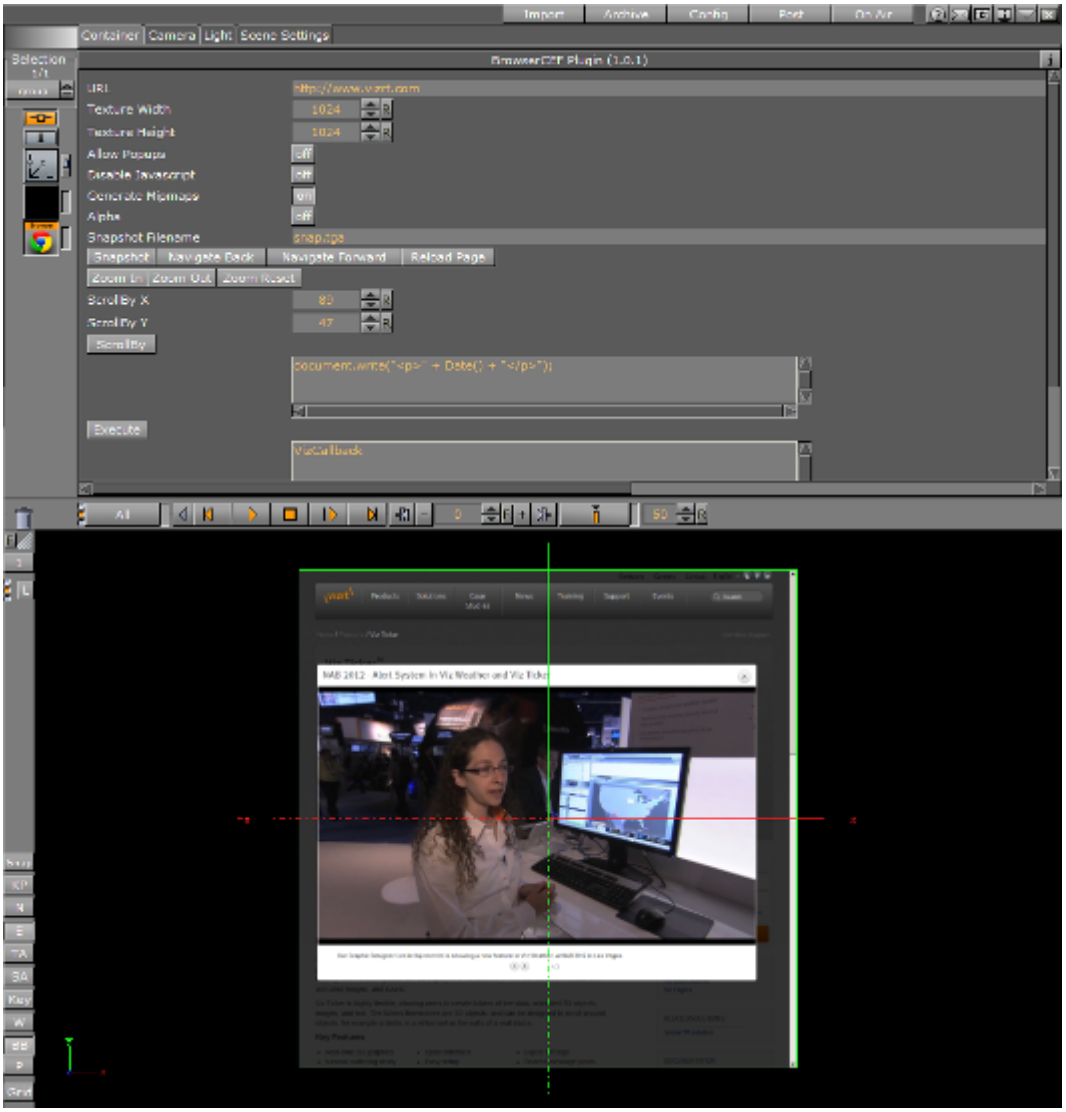
- **ScrollBy X:** Sets the scroll in the horizontal axis.
- **ScrollBy Y:** Sets the scroll in the vertical axis.
- **ScrollBy:** Executes ScrollBy X and Y values.
- **Execute:** Executes the code entered in the code entry field.

- **Disable Native Mouse Events:** Disables all mouse events, such as move, click and wheel events. Useful to control mouse events from outside, for example in scripts; or if it interferes with other plug-ins, such as Graffiti.
- **Mouse Position X and Y:** The u/v parameters control the position of the mouse cursor. The coordinates are normalized in [0,1], being (0,0) the lower left corner and (1,1) the upper right corner of the browser window. All mouse click and move events described below relate to this coordinate:
    - **Left Click:** Emulates a click (both mouse down and mouse up events) of the left mouse button.
    - **Left Mouse Down:** Emulates a left mouse button down event.
    - **Left Mouse Up:** Emulates a left mouse button up event.
    - **Right Mouse Down:** Emulates a right mouse button down event.
    - **Right Mouse Up:** Emulates a right mouse button up event.
    - **Mouse Move:** Emulates a mouse move event (might be triggered after the u/v coordinate has been changed.
    - **Wheel Up:** Emulates a mouse wheel up event.
    - **Wheel Down:** Emulates a mouse wheel down event.
    - **Wheel Left:** Emulates a mouse wheel left event.
    - **Wheel Right:** Emulates a mouse wheel right event.
- **Proxy User:** Provides the user name in case the system's http-proxy requires authentication.
- **Proxy Password:** Provides the user name password in case the system's http-proxy requires authentication.
- **Read Events From Shared Memory**: Enables the browser to receive events from shared memory callbacks.
- **Input Events from Shared Memory**: Prefix key for global shared memory. Valid commands are:
    - `WB_TEXT`: Can be used to emulate string input like 'Hello World!'
    - `WB_KEY`: Can be used for special key strokes. Possible values are ENTER, RETURN, BACKSPACE, DELETE, HOME, END, LEFT, RIGHT, UP, DOWN, PAGE_UP, PAGE_DOWN Sample for viz-scripting supposing ShmemVarName is WB_EVENTS:

      ```
      System.map["WB_EVENTS"] = "WB_TEXT Hello World!" System.map["WB_EVENTS"] = "WB_KEY ENTER"
      ```

      Note that a Shared Memory change event will only occur if the memory actually changes. Sending the same text twice, for example, will thus not trigger a Shared Memory event.
- **Enable Multitouch Events**: Enables native multitouch events (scroll, pinch to zoom, double tap).

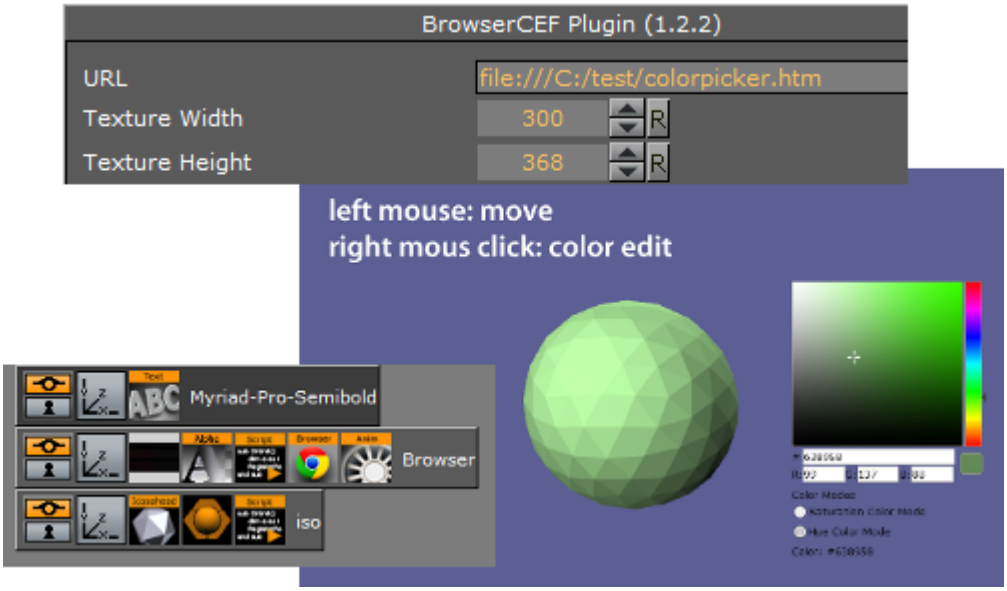
To Create a Simple Scene with BrowserCEF



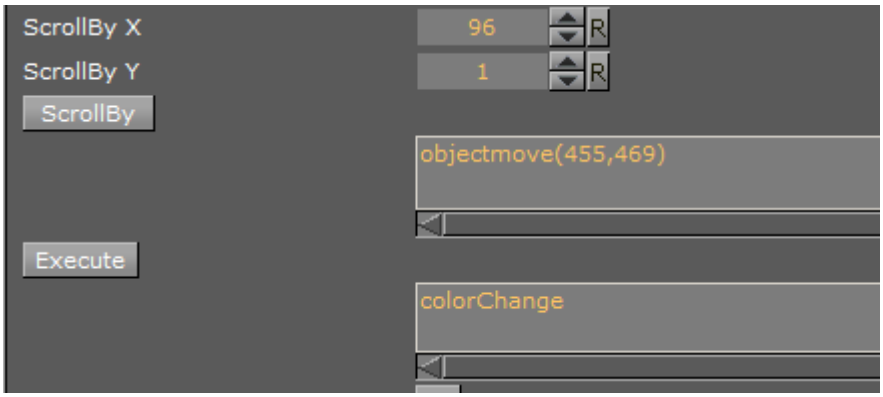
1. Add a new Container to the Scene tree.
2. Add a **rectangle** geometry to the Container
3. Add the BrowserCEF plug-in.
4. Open the BrowserCEF properties and enter a valid URL.
5. Click on E (enable the handling of interactive script/plug-in events (Scene Editor)).
6. Use the mouse or controls to navigate the web site.

Interact with a Scene with JavaScript

It is possible to interact with a Scene from within the JavaScript of a HTML document, for example, colorpicker.html.



In the JavaScript section of a HTML document interact with the Scene with the **window.vizrt** object. Append any function name on this object, for example, **window.vizrt.colorChange("255 100 100")**. The function **colorChange** must be defined in the **JavaScript function** section of the **BrowserCEF** plug-in.



To add more functions, add them in this GUI section, separated by new lines:

Only one argument is supported for each function, to add more arguments to a function use separators and encode all arguments into one string.

Calls from JavaScript are translated into Viz Shared Memory variable changes, on the **system.map**. To register these changes use the Shared Memory variable composed by the Container ID of the BrowserCEF plug-in and the string, for example, `" BROWSER STATUS"`.

⚠ **IMPORTANT!** There must be a space between the double quotation mark (") and BROWSER{{: " BROWSER STATUS"}}.

Script Example: When the script sits on the same Container:

```
sub OnInit()
	System.map.RegisterChangedCallback( this.VizId & " BROWSER STATUS" )
end sub

sub OnSharedMemoryVariableChanged(map As SharedMemory, mapKey As String)
	println( "shared map changed: " & map[mapKey] )

	Dim str As String = map[mapKey]
	Dim strstr As array[String]

	str.split(" ", strstr)
	if strstr[1] = "javascript" then
		if strstr[2] = "colorChange" then
			scene.findContainer("iso").Material.diffuse.red = cDbl(strstr[3])/255.0
		scene.findContainer("iso").Material.diffuse.green = cDbl(strstr[4])/255.0
		scene.findContainer("iso").Material.diffuse.blue = cDbl(strstr[5])/255.0
		end if
	end if
end sub
```

Example string passed to Shared Memory:

```
"0 JavaScript colorChange 0 53 4"
```

- The first number, either `0` or `1`, can be ignored
- `JavaScript`, shows that a JavaScript function was invoked
- The function name (`colorChange`) itself followed by the argument(s).
  Example call in the HTML document:

```
window.vizrt.colorChange(r + " " + g  + " " + b);
```

Example HTML Document

Entered in **URL**: file:///C:/test/colorpicker.htm (see BrowserCEF Properties):

```
<script type="text/javascript">
function colorAnim()
{
	window.vizrt.colorAnim()
}
```

```
function objectmove(x, y)
{
        PositionX.value = x;
        PositionY.value = y;
}
</script>

<!DOCTYPE html>
<html lang="en">
        <head>
                <title id='Description'>The jqxColorPicker widget allows you to easily pick a color.</
title>
                <link rel="stylesheet" href="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/styles/
jqx.base.css" type="text/css" />
                <script type="text/javascript" src="http://www.jqwidgets.com/jquery-widgets-demo/scripts/
jquery-1.11.1.min.js"></script>
                <script type="text/javascript" src="http://www.jqwidgets.com/jquery-widgets-demo/scripts/
demos.js"></script>
                <script type="text/javascript" src="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/
jqxcore.js"></script>
                <script type="text/javascript" src="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/
jqxcolorpicker.js"></script>
                <script type="text/javascript" src="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/
jqxradiobutton.js"></script>
        </head>
        <body class='default'>
                <div id='content'>
                        <script type="text/javascript">
                                $(document).ready(function ()
                                {
                                        // Create jqxColorPicker widgets.
                                        $("#jqxColorPicker").jqxColorPicker({ width: '250', height:
'250'});
                            $("#hueMode").jqxRadioButton({ width: '150', height: '25'});
                        $("#saturationMode").jqxRadioButton({ checked: true, width: '180', height: '25'});
                        $("#hueMode").on('change', function (event)
                                        {
                            if (event.args.checked)
                                                {
                                    $("#jqxColorPicker").jqxColorPicker({ colorMode: 'hue' });
                            }
                            else {
                                $("#jqxColorPicker").jqxColorPicker({ colorMode: 'saturation' });
                            }
                         });

                        $("#jqxColorPicker").on('colorchange', function (event)
                                        {
                                //console.log(event.args.color.r);
                                                window.vizrt.colorChange(event.args.color.r + " " +
event.args.color.g + " " + event.args.color.b);
                            $("#colorlog").html("<div>Color: #" + event.args.color.hex + "</div>");
                                        });
```

```
                        });
                    </script>
                    <div id='jqxWidget'>
                        <div id="jqxColorPicker"></div>
                        <div style="font-size: 13px; font-family: verdana; margin-top: 10px;">
                            <div style="margin-bottom: 5px;">Color Modes</div>
                            <div id="saturationMode">Saturation Color Mode</div>
                            <div id="hueMode">Hue Color Mode</div>
                </div>
                        <div style="font-family: Verdana; font-size:13px;" id="colorlog"></div>
                    </div>
            </div>
        </body>
</html>
```

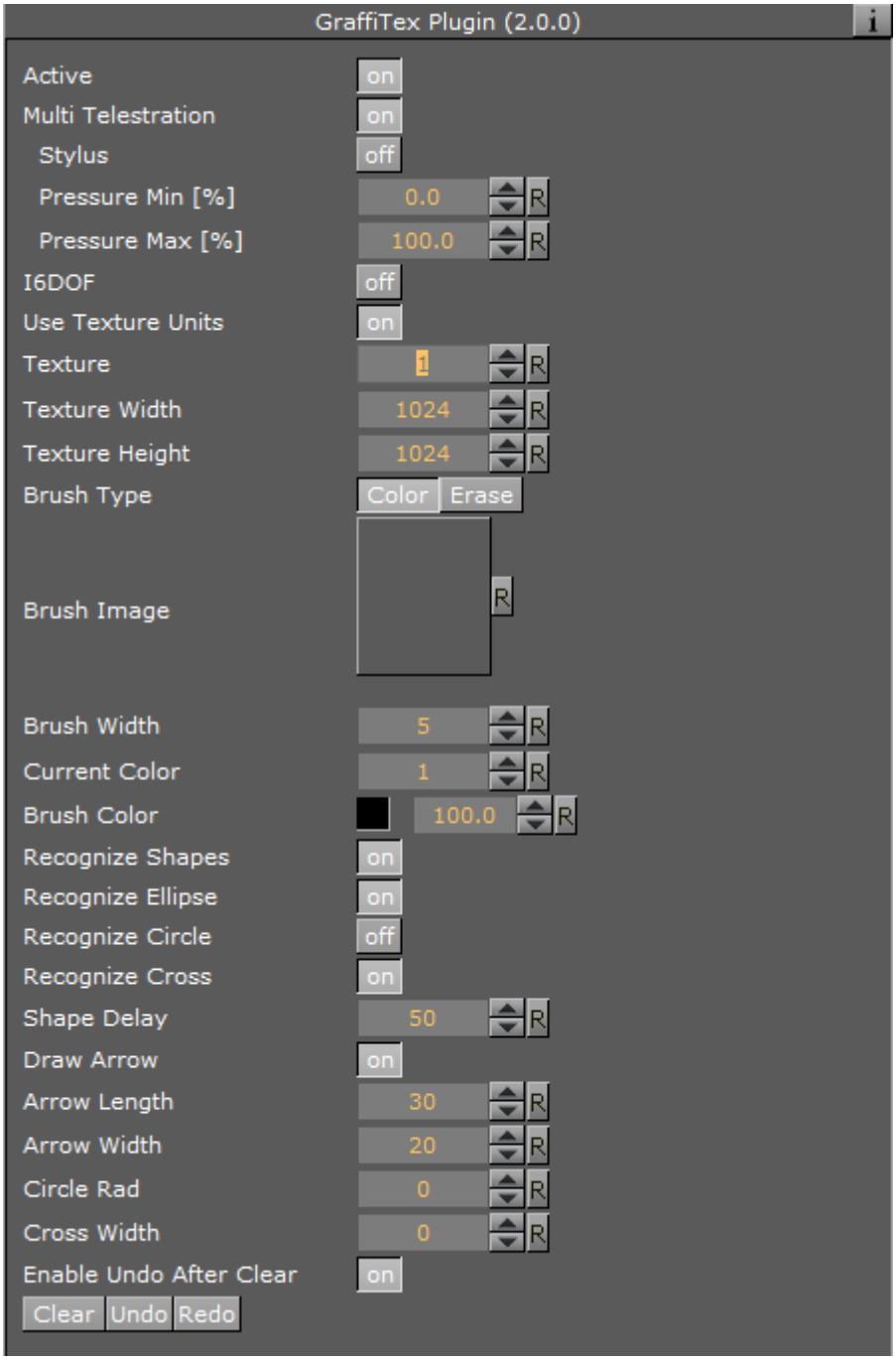
14.18.2 GeoGraffiti

The Graffiti Texture plug-in gives the ability to draw freehand on top of containers with the Globe plug-in. The freehand draw is created with a brush shape, used with a mouse. The plug-in can also recognize some rendered shapes, such as circles, ellipses, crosses, and arrows, and replaces the hand-drawn item with the recognized shape.

The plug-in can either work in combination with Viz Engine's multi-texturing technology or can add a standalone texture.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

GeoGraffiti Properties



- **Active**: Enables/disables drawing.
- **Texture Width**: Sets dimensions of the texture used for drawing canvas.
- **Brush Type**: Color or eraser brush
- **Brush Type**: Selects color or eraser brush.
- **Brush Image**: Determines the shape of the brush using an image. If empty, a round brush will be used.

- **Brush Width**: Sets the width of the brush in pixels. Visible only if the color brush is selected.
- **Eraser Brush Width**: Sets the width of the eraser in pixels. Visible only if the eraser brush is selected.
- **Brush Color**: Selects the color of the brush.
- **Recognize Shapes**: Toggles shape recognition mode on or off.
    - **Recognize Ellipse**: Specifies whether shape recognition will try to recognize ellipse shape.
    - **Recognize Circle**: Specifies whether shape recognition will try to recognize circle shape.
    - **Recognize Cross**: Specifies whether shape recognition will try to recognize cross shape.
    - **Shape Delay**: Determines the number of frames to wait from mouse up before trying to recognize shapes.
    - **Draw Arrow**: Specifies whether non-recognizable shapes will be converted to an arrow.
    - **Arrow Length**: Sets the length of an arrow head.
    - **Arrow Width**: Sets the width of an arrow head.
    - **Circle Rad**: Determines the radius of the circle replacing a recognized circle. If zero, the radius of the recognized circle will be used.
    - **Cross Width**: Determines the width of the cross replacing the recognized cross shape. If zero, the width of the recognized cross shape will be used.
- **Enable Undo After Clear:** Enables undo after the Clear button has been used.
- **Clear (button)**: Clears the canvas.
- **Undo (button)**: Undoes an action.
- **Redo (button):** Redoes an action.

14.18.3 Grabbit

The Grabbit plug-in gives the ability to use captured video inside Viz Engine. Grabbit uses the Microsoft DirectShow Filter graph framework to capture the video.

Devices which are supported by DirectShow and are self-contained such as capture cards or web cameras, can be used. Self-contained means the capture device does not need any further input.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

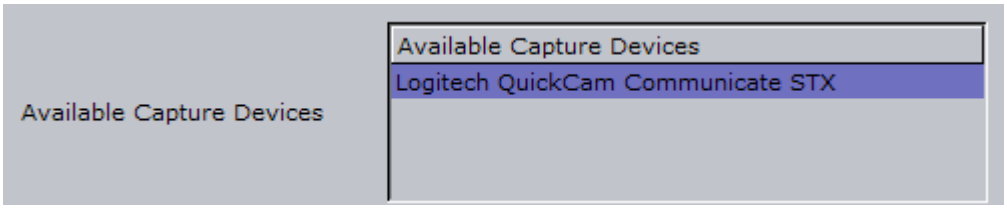
This page contains the following topics and procedures:

- Grabbit Properties
- Required Software
- Best Practices
- Known Limitations
- To Capture a Video Stream with Grabbit

- To Capture a Video Stream with Grabbit and a VfW Supported Device

Grabbit Properties

Following is a list of all parameters of the plug-in. Depending on the state of the plug-in, they may be hidden. Some of them are not visible at all.



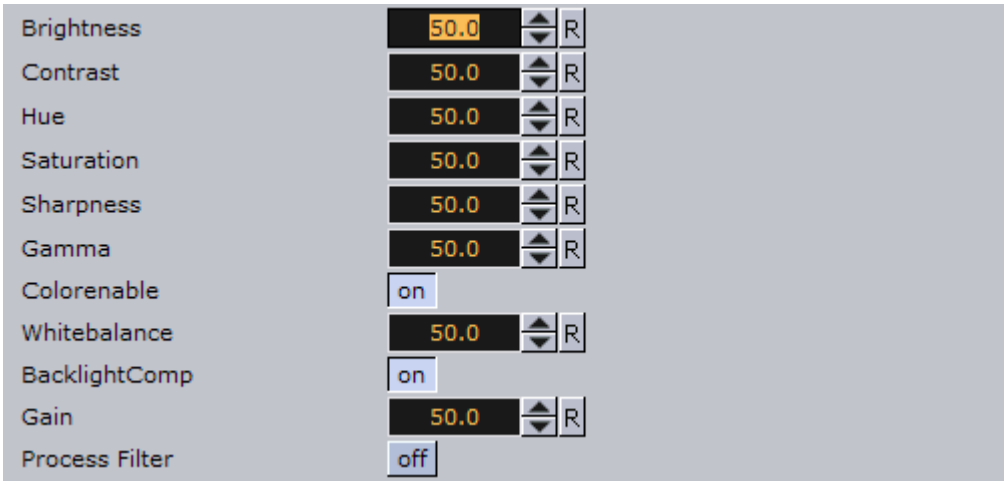
- **Available Devices:** Lets you decide which video capture device to use. This list is created during loading of the plug-in. That means if you have a capture device which is not known at startup of the renderer it will not show up. On the other hand it does not know if a capture device is already in use by some other plug-in. Since the list is static is just shows all capture devices which were available at startup. If the parameter is changed and the plug-in is already started it will stop the capturing before loading the new capture device. Changes are effective immediately.
  - Parameter name: `CaptureDevices (list)`.
- **Selected Capture Device:** Contains the simple name for the chosen capture device from the list. This parameter decides which DirectShow Device is taken for capturing. If the parameter is changed and the plug-in is already started it will stop the capturing before loading the new capture device.
  - Parameter name: `CaptureDevice (string)`.

  > ⚠ **Note:** This parameter is hidden from the user interface.

- **Capture Format:** Human readable string that represents the current capturing format. It shows the width by height, frame rate, video subtype, and bit depth. It is a read only resource and is changed with the configuration dialogs below. This should really be an opaque data type.

  > ⚠ **Note:** This parameter is hidden from the user interface.

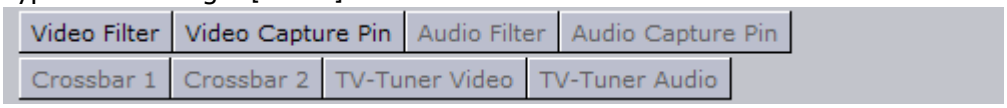
- Parameter name: `CaptureFormat (string)`.



Parameters are shown if the filter supports the ProcAmp interface.

- **Brightness:** Changes the brightness. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Brightness`.
  - Type: float. Range: [0-100]. Default: 50.
- **Contrast:** Changes the contrast. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Contrast`.
  - Type: float. Range: [0-100]. Default: 50.
- **Hue:** Changes the hue. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Hue`.
  - Type: float. Range: [0-100]. Default: 50.
- **Saturation**: Changes the saturation. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Saturation`.
  - Type: float. Range: [0-100]. Default: 50.
- **Sharpness:** Changes the sharpness. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Sharpness`.
  - Type: float. Range: [0-100]. Default: 50.
- **Gamma:** Changes the gamma. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Gamma`.
  - Type: float. Range: [0-100]. Default: 50.

- **Colorenable:** Enables or disables color. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Colorenable`.
  - Type: toggle button. Default: on.
- **Whitebalance:** Changes the white balance. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Whitebalance`.
  - Type: float. Range: [0-100]. Default: 50.
- **Backlight Compensation:** Enables or disables the back light compensation. Changes are effective immediately but not for already captured video.
  - Parameter Name: `BacklightComp`.
  - Type: toggle button. Default: on.
- **Gain:** Changes the gain. The range is relative and the value is re-calculated internally according to the possibilities (min/max/stepping) of the device. Changes are effective immediately but not for already captured video.
  - Parameter Name: `Gain`.
  - Type: float. Range: [0-100]. Default: 50.



Parameters are shown if the filter supports the DirectShow dialog interface. Since the dialogs are handled by Windows and not the Viz Artist GUI they may open in the background. If so please bring the dialog to the front and make your settings. Make sure that the dialog is closed before you change anything in the Viz Artist GUI.

The buttons are disabled if there is no such dialog. Some implementations hint that a dialog is available even if it is not. In that case the button is not disabled but no dialog window shows.

- **Video Filter:** The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up. Changes are effective immediately.
  - Parameter Name: `ButtonVideoFilter`.
  - Type: push button.
- **Video Capture Pin:** If available choose RGB24 or UYV as format. Other settings may not work. The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up. Changes are effective immediately.
  - Parameter Name: `ButtonVideoCapPin`.
  - Type: push button.
- **Audio Filter:** The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up. Changes are effective immediately.
  - Parameter Name: `ButtonAudioFilter`.
  - Type: push button.

- **Audio Capture Pin:** The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up. Changes are effective immediately.
    - Parameter Name: `ButtonAudioCapPin`.
    - Type: push button.
- **Crossbar 1:** The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up.
    - Parameter Name: `ButtonCrossbar1`.
    - Type: push button.
- **Crossbar 2:** The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up.
    - Parameter Name: `ButtonCrossbar2`.
    - Type: push button.
- **TV-Tuner Video:** The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up.
    - Parameter Name: `ButtonTvVideo`.
    - Type: push button.
- **TV-Tuner Audio:** The dialog window may be behind the renderer GUI after activation. Please be careful to close the dialog window. Otherwise the renderer and the GUI may lock up.
    - Parameter Name: `ButtonTvAudio`.
    - Type: push button.
      Video for Windows Dialogs:

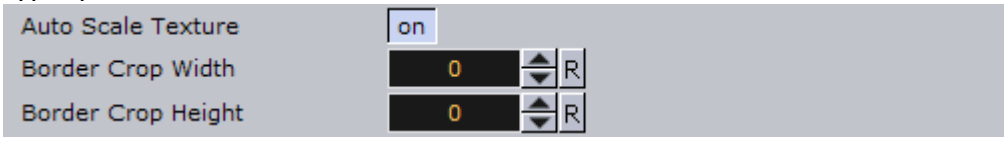
Parameters are shown if the filter supports the Video for Windows (VfW) dialog interface. Since the dialogs are handled by Windows and not the Viz Artist GUI they may open in the background. If so please bring the dialog to the front and make your settings. Make sure that the dialog is closed before you change anything in the Viz Artist GUI.

Buttons are disabled if there is no such dialog. Some implementations hint that a dialog is available even if it is not. In that case the button is not disabled but no dialog window shows.

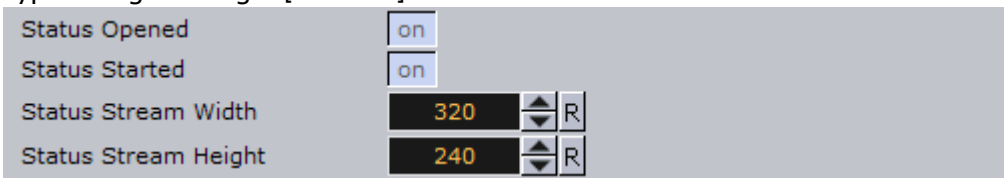
- **Cap Source:** If the filter supports the VfW Source dialog this button is present. It can be activated if the capturing is stopped otherwise the button is grayed out. The dialog window may be behind the renderer GUI after activation. please be careful to close the dialog window. otherwise the renderer and the GUI may act in a strange way. Changes are effective immediately.
    - Parameter Name: `ButtonCaptureSource`.
    - Type: push button.
- **Cap Format:** If the filter supports the VfW Format dialog this button is present. It can be activated if the capturing is stopped otherwise the button is greyed out. The dialog window may be behind the renderer GUI after activation. please be careful to close the dialog window. otherwise the renderer and the GUI may act in a strange way. Changes are effective immediately.
    - Parameter Name: `ButtonCaptureFormat`.
    - Type: push button.
- **Cap Display:** If the filter supports the VfW Display dialog this button is present. It can be activated if the capturing is stopped otherwise the button is grayed out. The dialog window

may be behind the renderer GUI after activation. please be careful to close the dialog window. otherwise the renderer and the GUI may act in a strange way. Changes are effective immediately.

- Parameter Name: `ButtonCaptureDisplay`.
- Type: push button.



- **Auto Scale Texture:** If enabled the captured video is scaled according to the underlying geometry. Changes are effective immediately but not for already captured video.
  - Parameter Name: `AutoScale`.
  - Type: toggle button. Default: on.
- **Border Crop Width:** Crop pixels on the horizontal sides of the video. This parameter is shown only if Auto Scale Texture is enabled. Changes are effective immediately but not for already captured video.
  - Parameter Name: `BorderCrop_x`.
  - Type: integer. Range: [0-0x7fff]. Default: 0.
- **Border Crop Height:** Crop pixels on the vertical sides of the video. This parameter is shown only if Auto Scale Texture is enabled. Changes are effective immediately but not for already captured video.
  - Parameter Name: `BorderCrop_y`.
  - Type: integer. Range: [0-0x7fff]. Default: 0.



- **Status Opened:** Shows if the capture graph has been built successfully.
  - Parameter Name: `StatusOpened`.
  - Type: toggle button (show only).
- **Status Started:** Shows if the capture graph has been started.
  - Parameter Name: `StatusStarted`.
  - Type: toggle button (show only).
- **Status Stream Width:** Shows the captured video width.
  - Parameter Name: `StatusStream_x`.
  - Type: integer (show only).
- **Status Stream Height:** Shows the captured video height.
  - Parameter Name: `StatusStream_y`.
  - Type: integer (show only).

- **Start:** Starts capturing. Feedback is given through Status Started and Status Stream Width/Height. If capturing does not start check the Status Opened and Connected. Capturing cannot commence if the filter graph is not connected. Changes are effective immediately.
  - Parameter Name: `ButtonStart`.

    - Type: push button.
- **Stop:** Stops capturing. You need to do this to select the Vfw Dialog buttons. Changes are effective immediately.
    - Parameter Name: `ButtonStop`.
    - Type: push button.
- **Clear:** Clears the texture to opaque white. You may need to hit the button twice for the change to show. Changes are effective immediately.
    - Parameter Name: `ButtonClear`.
    - Type: push button.

Required Software

- DirectShow version 9.0c. It is strongly recommended to have a clean DirectShow installation to start with. Additional source filters may be added later. Find more Information at the
- DirectShow of VfW compatible capture card.

Best Practices

GraphEdit is a tool from the Microsoft DirectShow SDK that allows you to visualize the default Filtergraph that DirectShow builds to render the Media. If you can render the Media inside GraphEdit it should play in Viz Engine, too. If it does not play in GraphEdit it will certainly not play in Viz Engine.

Known Limitations

- Dynamic listbox for capture devices.
- Handle the capture device list dynamically.
- ProcAmp settings should be updated from dialog settings as well.
- Query for availability in the various ProcAmps.
- The Capture Device must be known to the system at application start. When using USB devices this means they must be connected before Viz Engine is started.

To Capture a Video Stream with Grabbit

1. Add the Rectangle geometry to the Scene Tree.
2. Add the Grabbit plug-in to it.
3. Select the capture device to work with from the **Available Capture Devices** list.
    - You should get a positive **Status Opened** feedback.

    > ⚠ **Note:** Do not select a device which is currently in use (either by another instance of this plug-in or by another application).

4. If you have a device that supports the capture format enumeration and the ProcAmp interface you will get a screen similar to the following.
5. Set the **Capture Format** by using the dialogs behind the configuration buttons Video Filter, and so on.

6. Select **Auto Scale Texture** if you want the texture position and scaling set by this plug-in so that the video fills out the entire geometry. If you have selected this feature you may want to crop something on the horizontal or vertical sides of the video. Use the Border Crop Width and Border Crop Height for this effect.
7. Click the **Start** button to start capturing. You should get a positive *Status Started* feedback. In addition, the *Status Stream Width* and *Status Stream Height* should reflect the dimensions of the captured video.
8. Adjust the effect settings. You might want to experiment with them. Some parameters have discreet values as for example the *Gamma* parameter.

> ⚠ **Note:** The effect settings are highly dependent on the selected device.

To Capture a Video Stream with Grabbit and a VfW Supported Device

> ⚠ **Note:** If you have a device which supports the VfW Dialogs interface you will most probably see that your device is tagged with VfW.

Once you have selected the capture device it needs to be configured. You do this by clicking the buttons that leads to the VfW dialog windows.

1. Click **Cap Source** and **Cap Format** and set the parameters according to your needs.
   - The values you set here cannot be changed via the usual control mechanism. You need to pay close attention that the dialog window may open in the behind the Viz Artist GUI.
   - Make sure that each dialog window is closed before any other changes are made in the Viz Artist GUI.
   - You should get a positive **Status Connected** feedback. If you want to go back to change something here the graph must be in Stopped mode.
2. Select **Auto Scale Texture** if you want the texture position and scaling set by this plug-in so that the video fills out the entire geometry.
   - If you have selected this feature you can use the **Border Crop Width** and **Border Crop Height** to crop something on the horizontal or vertical sides of the video.
3. Click the **Start** button to start capturing.
   - You should get a positive **Status Started** feedback. In addition the **Status Stream Width** and **Status Stream Height** should reflect the dimensions of the captured video.
4. Click the **Stop** button to stop the capturing.
   - This will set it back to Stopped mode.
   - You should see the feedback in the **Status Started** toggle.
   - Once the capturing is stopped you can use the **Clear** button to set the underlying texture to opaque white.

14.18.4 GraffitiTex

The Graffiti Texture plug-in gives the ability to draw freehand on top of flat containers, such as rectangles. The freehand draw is created with a brush shape, used with a mouse, 6DOF device, or multi touch device. The plug-in can also recognize some rendered shapes, such as circles, ellipses, crosses, and arrows, and replaces the hand-drawn item with the recognized shape.

6DOF (6 Degrees of Freedom) events are triggered for all layers, but the Main Layer (see Layer Manager) has priority over the Front and Back Layers. This means that if the Front Layer scene grid is to be used for 6DOF, then the grid in the Front Layer must be set to **active,** and the grid in the Main Layer must to be set to **inactive**. If there is no grid defined in any of the three Layers, an orthogonal XY 2D grid will be used (see Grid Tool-bar).

The plug-in can either work in combination with Viz Engine's multi-texturing technology or can add a standalone texture.

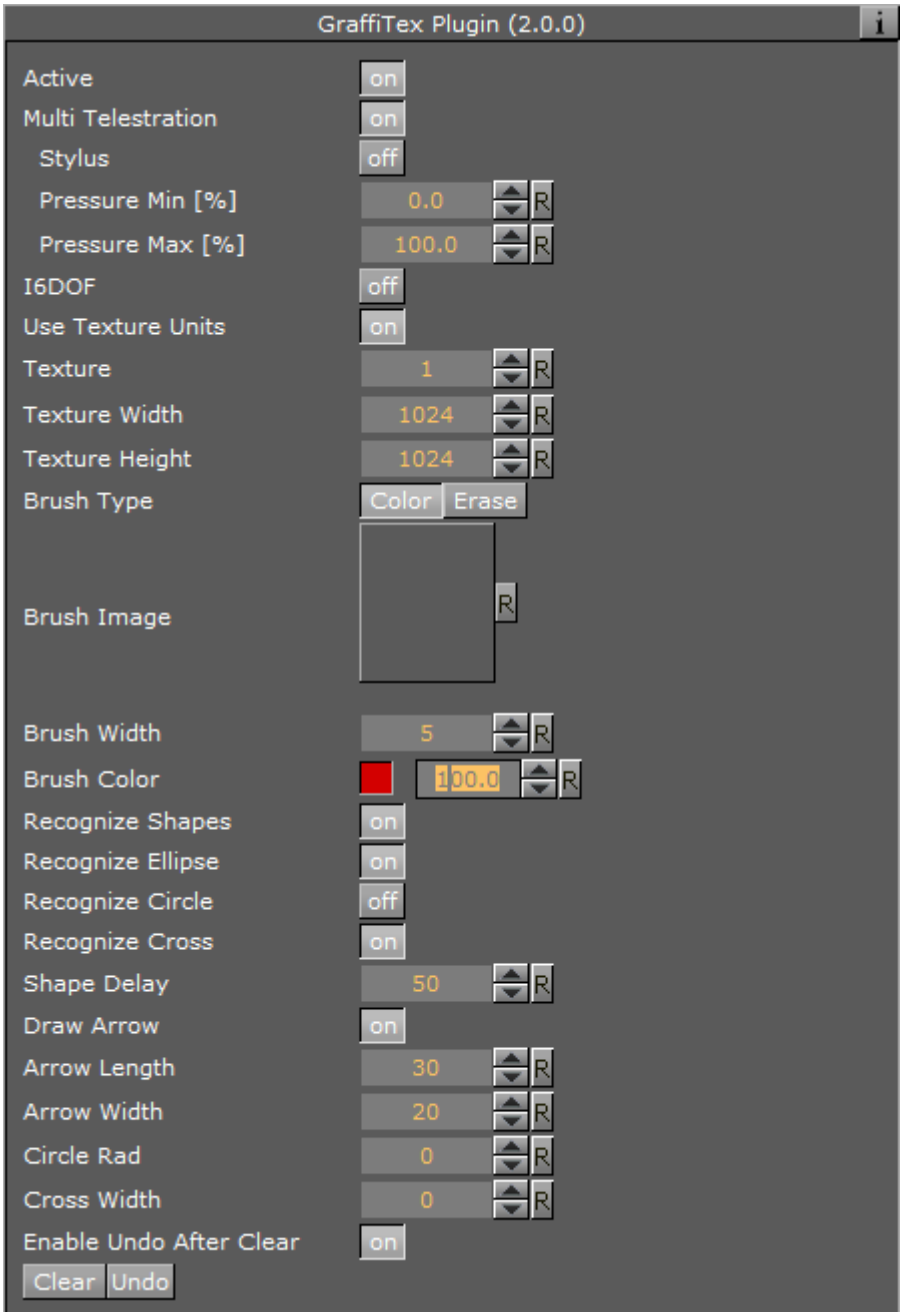
There are two graffiti plug-ins; the scene plug-in Graffiti which is used globally for the scene, and this one which is applied on the container level.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

This page contains the following topics and procedures:

- GraffitiTex Properties
- To Create a Container Level Graffiti Effect

GraffitiTex Properties



- **Active**: Enables/disables drawing.
- **Multi Telestration:** Allows handling of multiple touch events for multi-touch devices. Some devices incorporate a pressure factor; this influences the width of the brush:
  - **Pressure Min %:** Sets the minimum width of the brush.
  - **Pressure Max %:** Sets the maximum width of the brush.
- **I6DOF**: Specifies whether input comes from mouse of 6DOF.

- **Transparent Base**: Specifies whether the base color of the container will be the background of the rendered brush or whether the brush will be drawn on a transparent background.
- **Texture Width, Texture Height**: Sets dimensions of the texture used for drawing canvas.
- **Brush Type**: Selects color or eraser brush.
- **Brush Image**: Determines the shape of the brush using an image. If empty, a round brush will be used.
- **Brush Width**: Sets the width of the brush in pixels. Visible only if the color brush is selected.
- **Eraser Brush Width**: Sets the width of the eraser in pixels. Visible only if the eraser brush is selected.
- **Brush Color**: Selects the color of the brush.
- **Recognize Shapes**: Toggles shape recognition mode on or off.
  - **Recognize Ellipse**: Specifies whether shape recognition will try to recognize ellipse shape.
  - **Recognize Circle**: Specifies whether shape recognition will try to recognize circle shape.
  - **Recognize Cross**: Specifies whether shape recognition will try to recognize cross shape.
  - **Shape Delay**: Determines the number of frames to wait from mouse up before trying to recognize shapes.
  - **Draw Arrow**: Specifies whether non-recognizable shapes will be converted to an arrow.
  - **Arrow Length**: Sets the length of an arrow head.
  - **Arrow Width**: Sets the width of an arrow head.
  - **Circle Rad**: Determines the radius of the circle replacing a recognized circle. If zero, the radius of the recognized circle will be used.
  - **Cross Width**: Determines the width of the cross replacing the recognized cross shape. If zero, the width of the recognized cross shape will be used.
- **Enable Undo After Clear:** Enables undo after the Clear button has been used.
- **Clear (button)**: Clears the canvas.
- **Undo (button)**: Undoes an action.

To Create a Container Level Graffiti Effect

- First, add the plug-in to the container and set the plug-in properties.
- Then set Viz Artist in On Air Mode, and start drawing.

14.18.5 Image Clip

The Image Clip plug-in is designed to play back a sequence of still images (e.g. TGA or TIFF) rather than playing a movie file. It also supports alpha channel and various play modes (looping, swing, etc). The plug-in loads all images to RAM, then loads only one image at a time to texture memory. The download time is almost linear to the size of each frame. Preparation for this plug-in should be uncomplicated as most video applications has support for saving out TIFF. Since it plays the sequence from memory, one obvious benefit is that once loaded into memory it does not access the hard drive at all. So if it is needed to play back ten different sequences at once, ImageClip is pretty much the only solution. It is ideal for looping small animations such as logos. The drawback is that it consumes system memory.

Remember to keep track of how much memory each sequence needs, to avoid running out of memory on the Viz Engine machine. Running low on memory will make the system start caching parts of memory to the hard drive and it may not play the sequence and graphics in real-time.

Image clip should not be used with very large clips in Continuous and Array mode. Since all images are loaded into memory a large number of images or a large image size would require large amounts of memory.

Memory can be calculated as follows:

- Number of images * Image width * Image height * 3 (or 4 when using alpha)
  The use of Thread mode solves the memory issue, but it will take longer to request a frame. Therefore it should be used only in a slow motion image clip.

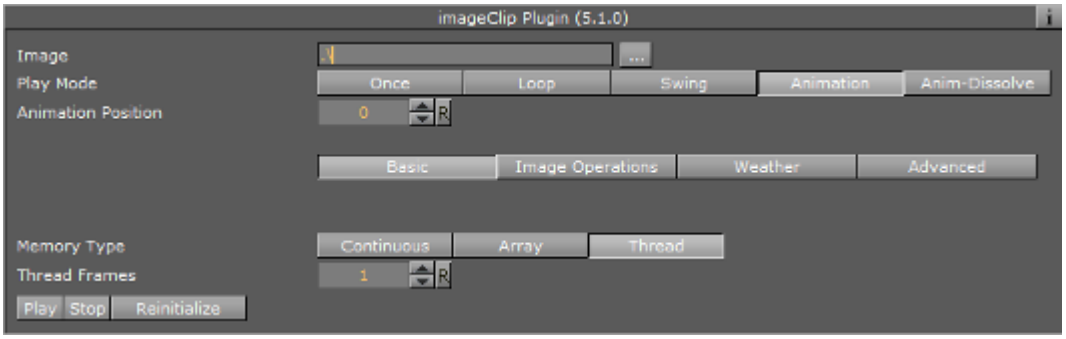
Images for an image sequence should be placed in specific folders. For example if you out many flag image sequences, you would have a separate folder for each flag image sequence. Make sure that the images are named sequentially e.g. “england001.tif”, ”england002.tif”, ”england003.tif” etc.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

This page contains the following topics and procedures:

- Image Clip Properties
- To Add an Image Clip

Image Clip Properties



- **Image**: Sets the image path and the first image you would like to use in the animation. Do not use clip names with numbers (except the counter (e.g. 000, 001, 002, etc.)).
- **Play Mode:** Sets the play mode:
  - Once:
  - Loop:
  - Swing:
  - Animation:
  - Anim-Dissolve:
- **Reverse:** Reverses the animation (not in **Animation** play mode).
- **Animation Position:** Sets the animation position (e.g. where to start and stop the animation). The position is a counter for the number of images in the folder referenced by the Image setting. This setting is enabled when **Play Mode** is set to **Animation**.
- **Play Speed:** Sets the speed of the animation.
- **Advanced:** Enables the advanced settings (see below).
- **Time Mode**: Enables time specific settings.
- **Stand Alone Image:** Enables the same clip to be played in different speeds.
- **First Image**: Defines the first image of the animation.
- **Nof Image**: Defines the number of images, relative to the First Image, that should be part of the animation.
- **Geo Ref**:
- **Crop**: Crops the image in percent from the left, right, bottom and top side.
- **Scale By**: Enables the scale setting.
  - **Scale:** Scales all images to the closest power of 2. When not selected, automatic texture coordinates will be applied so the image that will fit the texture.
- **Keep Under**: Forces the image size. For example if you have selected 64, the image will be trimmed to the size 64 x 64.
- **Base Path**:
- **Format**:
- **Weather**:
- **Blend Images**:
  - **Dissolve Speed**:
- **Control Texture Map**:
- **Memory Type:** Sets the memory type.

    - **Continuous:** Uses one big chunk of memory to store all images.
    - **Array:** Uses divided chunks of memory to store all images.
    - **Thread:** Loads only requested frame on-the-fly, with [n] frames loading time limit.
- **Mapping**:
- **Play (button):** Plays the animation.
- **Stop (button):** Stops the animation.
- **Reinitialize (button):** Reinitializes all settings.

To Add an Image Clip

1. Add a Container to the Scene Tree.
2. Add the Image Clip plug-in to it.
3. In the Image Clip properties click the  (Browse) button to locate the folder that has the required images.
4. Select the first image in the sequence.
5. Click **OK**.
    - The image sequence now loads into the ImageClip plug-in, and is visible in the scene.
6. Click **Play** to test the image sequence.
    - If you change the image sequence and you need to reload it, click the Reinitialize button.
7. Change the Play Mode to Loop to play the image sequence continuously.
    - The image sequence will start playing automatically.
8. Change the Play Mode to Animation to animate the image sequence frame position to control the sequence within the Stage Editor.
    - The image sequence now stops playing and a new property is revealed that is called Animation Position. This refers to the frame number that is currently shown for the image sequence.
9. Animate the Animation Position value.
    - The Animation Position maximum value is restricted to the number of images.
10. Another possibility is to create a file with a `.vln` extension. This file includes the base path and also the names of the images to load. In this case the images must not have a counter number in their filename. You can handle this file as an ordinary text file.
11. Load the vln-file instead of loading an image file located in a directory.
    Example:

```
BASE_PATH 'C:/clip/images' {  'radar_200504110800.png' 2005_04_11_10:00  'radar_200504110815.png'
 2005_04_11_10:15  'radar_200504110830.png' 2005_04_11_10:30  'radar_200504110845.png' 2005_04_11_10:45
'radar_200504110900.png' 2005_04_11_11:00  'radar_200504110915.png' 2005_04_11_11:15
'radar_200504110930.png' 2005_04_11_11:30  'radar_200504110945.png' 2005_04_11_11:45 }
```

⚠ **Note:** Viz Engine will not interpret between Key Frames, so if you choose to slow down an animation then the motion will not be smooth.

See Also

- Video Clips
- MoViz

- SoftClip

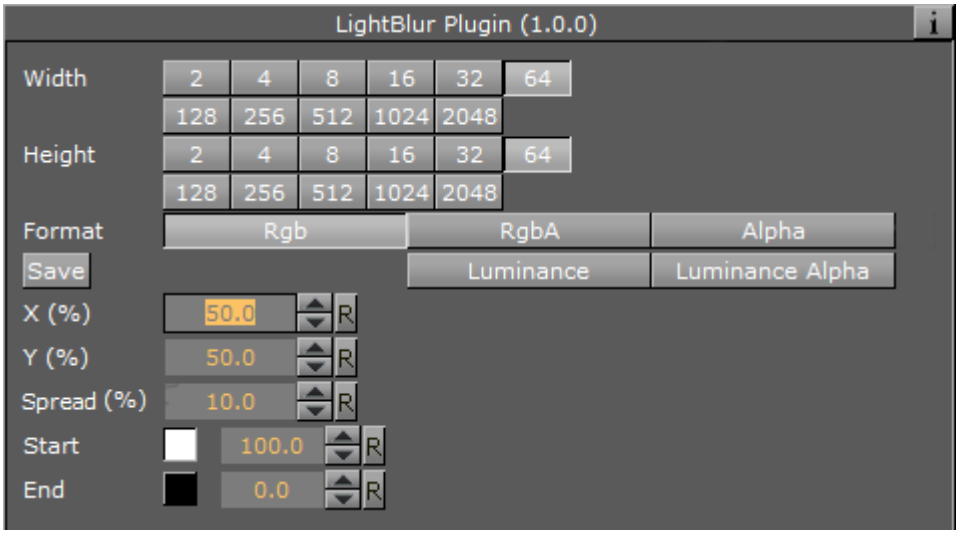
14.18.6 Light Blur

Allows the user to configure and create a light blurred image.

> ⚠ **IMPORTANT!** The light blur functionality is now a built in feature of the Viz Artist/Engine core. For future use it is therefore recommended to use the **Dynamic Texture** Media Asset.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

Light Blur Properties



- **Width and Height:** Sets the width and height of the blurred light. Available options are 2 to 2048.
- **Format:** Sets the image format. Options are:
  - RGB
  - RGBA
  - Alpha
  - Luminance
  - Luminance Alpha
- **Save:** Saves the blur as an image.
- **X and Y (%):** Sets the position of the blur on the X and Y axis (50% places the blur at the origin).

- **Spread (%):** Sets the spread of the blur.
- **Start:** Sets the start color of the blur.
- **End:** Sets the end color of the blur.

14.18.7 MoViz

With the MoViz plug-in media files or streams can be played inside the Viz Engine.

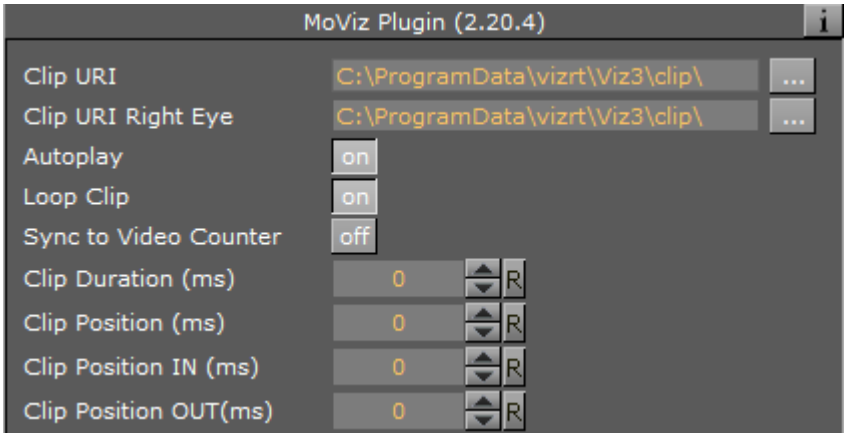
MoViz is not a QuickTime player or an AVI player. It uses the Microsoft DirectShow Filter Graph framework to play the media files, which means it can play both types, but only certain installed Filter Graphs (some of the QuickTime and some AVI).

MoViz is not natively QuickTime, it uses the AVI wrappers and filters for QuickTime and AVIs. This means that not all QuickTime codecs can be used, only the ones that are supported within DirectX wrappers and filters.

> ⚠ **Note:** Throughout this plug-in documentation, media file and media stream is referred to as media and may be used interchangeably unless stated otherwise.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

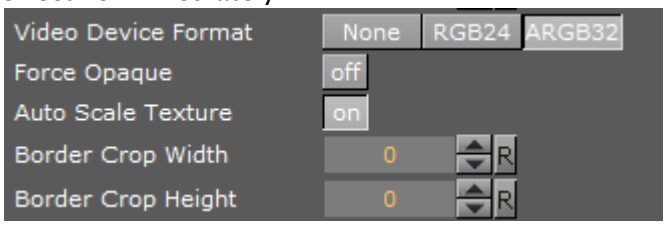
MoViz Properties



- **Clip URI:** Click the **Browse** button () to load clips from the file system (or enter the location directly). It takes the form of a valid URI (e.g. http://... mms://...). If the parameter is changed and the plug-in is already started it will stop the playback before loading a new clip. Changes are effective immediately.

> ⚠ **IMPORTANT!** All video clips from the file system should be stored locally on the D: drive of the machine that is used to play out the graphics. Remember to save the clip locally on the client machines with preview (e.g. Viz Trio) and use the same D: drive so that the clip can be seen in preview.

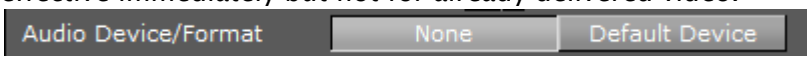
- **Clip URI Stereo:** Path to Clip that is loaded for the right eye camera in a stereo configuration with Stereo Mode = Right Eye. You can load clips from the file system by using the **Browse** button, or by typing the path location directly. It takes the form of a valid URI. (e.g. http://... mms://...). If you change the parameter after starting the plugin, it will stop the playback before loading the new clip. Changes are effective immediately.
- **Autoplay:** Sets the automatic start of the clip after it has loaded, to On or Off. Changes are effective immediately (Default is off).
- **Loop Clip:** Sets looped playback to On or Off. If the clip position is at the clip out position or EOF it will rewind to the clip in position. Changes are effective for the next EOF, clip out position (Default is Off).
- **Sync to Video Counter**: Enables or disables the use of the system retrace counter. If enabled, the clip will be played back with the speed of Viz's internal retrace counter. In a video version, this is the tick count of the SDI output, for a VGA version it is the monitor refresh rate. If disabled, the clip plays back at the speed of it's own internal clock.
- **Clip Duration (ms):** Shows the duration of the clip in milliseconds (read only). This parameter makes only sense with media that supports this kind of information.
- **Clip Position (ms):** Gives the current position of the clip in milliseconds. When in pause mode this parameter may be changed to seek to the gives time. Do not change this parameter when not in pause mode. The position may be changed between Clip Position IN and Clip Position OUT. This parameter makes only sense with media that supports this kind of information. Changes are effective immediately.
- **Clip Position IN (ms):** Gives the first frame where the clip should start in milliseconds. This parameter makes only sense with media that supports this kind of information. Changes are effective immediately.
- **Clip Position OUT (ms):** Gives the last frame where the clip should stop in milliseconds. This parameter makes only sense with media that supports this kind of information. Changes are effective immediately.



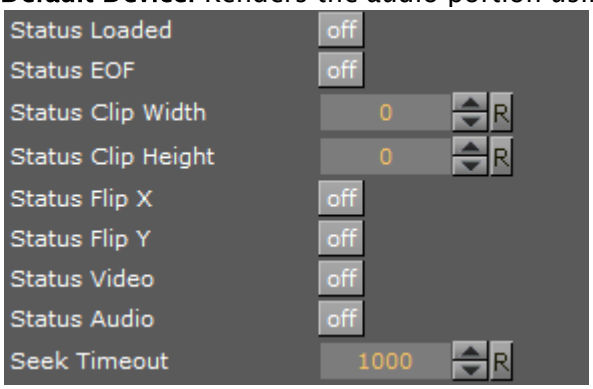
- **Video Device Format:** Selects a video format for rendering. Changes are effective at load time.
  - **None:** Does not render the video portion.
  - **RGB24:** Delivers the video portion to Viz Artist/Engine with the RGB24 format. This format uses 8 bits per Red, Green and Blue component, hence the name RGB24.
  - **ARGB32:** Delivers the video portion to Viz Artist/Engine with the ARGB32 format. This format uses 8 bits per Red, Green and Blue component and in addition 8 bits for Alpha

(8bits * 4 components), hence the name ARGB32 indicating RGB with Alpha. This format is often referred to as RGBA.

- **Force Opaque:** (ARGB32 only) Sets the alpha value to 100% opaque (if required) if the clip has an alpha channel. Changes are effective for each frame delivered.
- **Auto Scale Texture:** Scales captured video according to underlying geometry when enabled.Changes are effective immediately but not for already delivered video.
  - **Border Crop Width:** Crops pixels on the horizontal sides of the video. Changes are effective immediately but not for already delivered video.
  - **Border Crop Height:** Crops pixels on the vertical sides of the video. Changes are effective immediately but not for already delivered video.



- **Audio Device/Format:** Selects which audio device/format to use for rendering. Changes are effective immediately but not for already delivered audio. For performance reasons, `None` is the recommended setting.
  - **None:** Does not render the audio portion.
  - **Default Device:** Renders the audio portion using the default DirectShow renderer.

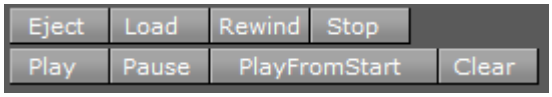


- **Status Loaded:** Shows if the clip has loaded.
- **Status EOF:** Shows if the clip is at EndOfFile. **Status EOF** allows user defined actions upon registering for change requests in a script. Whenever a change in the parameter `StatusEof` is happening and a Viz System Shared Memory (SHM) variable named `MoViz[A,B]_eof` exists, this variable is set to 1 when **Status EOF** is `On`, and to 0 when Status EOF is `Off`. The variable's parameter `A` is the hexadecimal value of the *scene ID* in uppercase A-F range. The variable's parameter `B` is the hexadecimal value of the `Container ID` in uppercase A-F range.

⚠ **Note:** Since Viz Engine 3.8.1, this is supported for MoViz Version 2.20.0 onwards.**Status Clip Width:** Shows the video width

- **Status Clip Height:** Shows the video height.
- **Status Video:** Shows if the clip has a usable Video component.
- **Status Audio:** Shows if the clip has a usable audio component.
- **Seek Timeout**: Specifies the timeout for seeking to a position in the movie file. Higher values can prevent jump back to the expected position when clip playback takes longer to start

then expected. However, the playback time may not be as exact.



- **Eject**: Unloads the clip and destroy the graph. Changes are effective immediately.
- **Load:** Loads the given clip and builds the graph. Changes are effective immediately.
- **Rewind:** Rewinds the clip to the start, when supported. Changes are effective immediately.
- **Stop:** Puts the graph in stop mode. Changes are effective immediately.
- **Play:** Puts the graph in play mode. i.e. start playback. Changes are effective immediately.
- **Pause:** Puts the graph in pause mode. Changes are effective immediately.
- **Play From Start:** Rewinds the clip to the start before playback. Changes are effective immediately.
- **Clear**: Clears the texture to opaque white.

> ⚠ **Note:** Moviz will handle videos with alpha, but the codecs in use must provide a valid ARGB32 stream and Containers you want visible should use the KEY Function, otherwise the alpha channel will be blank, making the whole video totally transparent. See also Advanced Issues with Video Codecs.

See Also

- Video Clips
- Supported Codecs
- Advanced Issues with Video Codecs
- Image Clip
- SoftClip
- Scripting
- Shared Memory (SHM)

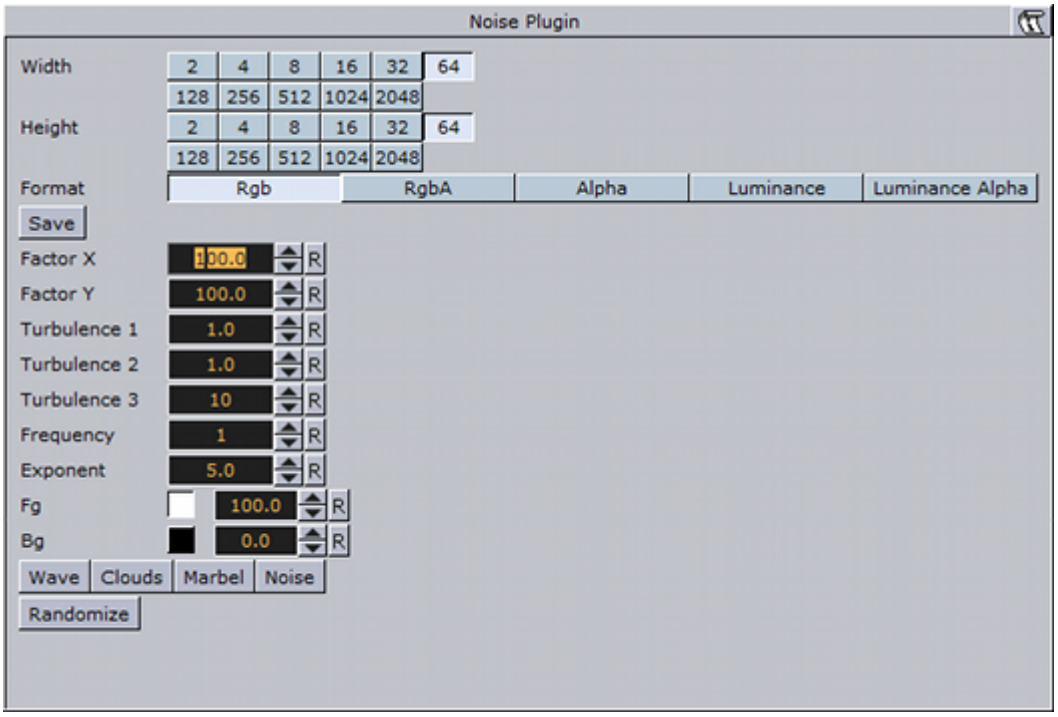
14.18.8 Noise

Allows the user to configure and create a noise image.

> ⚠ **IMPORTANT!** The noise functionality is now a built in feature of the Viz Artist/Engine core. For future use it is therefore recommended to use the **Dynamic Texture** Media Asset.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

Noise Properties



- **Width and Height:** Sets the width and height of the noise. Available options are 2 to 2048.
- **Format:** Sets the image format. Available options are RGB, RGBA, Alpha, Luminance and Luminance Alpha.
- **Save (button):** Saves the noise as an image.
- **Factor X and Y:** Sets the noise factor on the X and Y axis (range is from 0.000 to 1000.000).
- **Turbulence 1, 2 and 3:** Sets the turbulence of the noise.
- **Frequency:** Sets the frequency of the noise.
- **Exponent:** Sets the exponent of the noise.
- **Fg:** Sets the foreground color.
- **Bg:** Sets the background color.
- **Wave, Clouds, Marble and Noise (buttons):** Allows the user to select between four different presets.
- **Randomize (button):** Sets random factor, turbulence and frequency. Randomize works together with the Wave, Clouds, Marble and Noise selections.

14.18.9 SoftClip

The SoftClip is used for playing video clips projected on a texture, and supports AVI files only. This format supports RGB and RGBA (alpha channel).

In its simplest form, the SoftClip plug-in can exist on an empty container to play back a video clip in Viz Artist/Engine. Other plug-ins can be mixed with SoftClip to obtain different results. For example you can apply a SoftClip plug-in to a Sphere and add a Material to affect the overall shape, color and shading of the video clip.

With SoftClip you are able to control the video clip within the stage, unlike with the **Live Video** Media Asset method. The video clip size can also be whatever you decide, however, it is recommended that you keep the dimensions in multiples of 8 (e.g. 128x128 or 256x512) for performance purposes.

Use of the SoftClip plug-in might have an affect on performance and quality, hence, it is recommended to use the Performance Editor when working with the SoftClip plug-in.

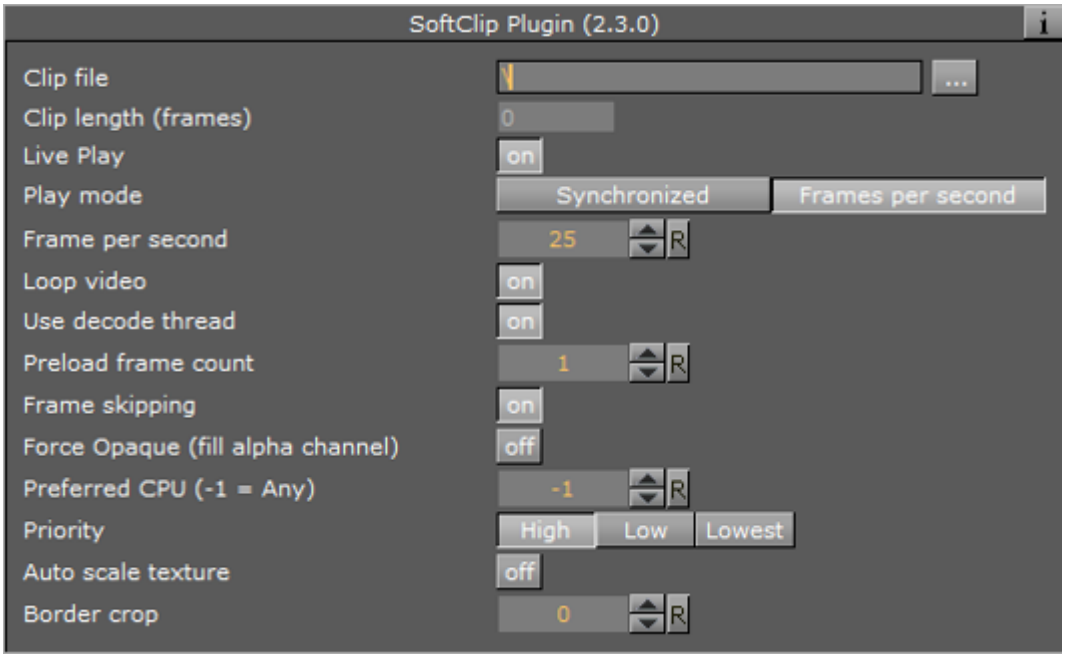
⚠ **IMPORTANT!** Viz Engine keeps all designer, operator and automation loaded scenes and plug-ins in memory. As a consequence, if 30 different templates that load 30 different scenes are loaded, that each has a single SoftClip plug-in on it, Viz Engine will have thirty SoftClip plug-ins with 30 threads activated in memory. If every plug-in has specified that it should have 10 preloaded frames, then Viz Engine will have 300 uncompressed decoded frames preloaded in memory. This can take a substantial amount of memory, and the combination of having that much data loaded and that many threads running is likely to affect performance. This workflow situation has to be taken into consideration when designing and testing the design. Therefore, unloading scenes and cleaning up the Viz Engine renderer, a function in many Vizrt control client programs, should be used.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Texture

This page contains the following topics and procedures:

- SoftClip Properties
- To Use SoftClip

SoftClip Properties



- **Clip file**: Use this function to browse for the clip to be played out.

  > ⚠ **Note:** It is important that all video clips are stored locally on the D: drive of the machine used to play out the graphics. Remember to save the clip locally on the client machines with preview (e.g. Viz Trio) and use the same D: drive so that the clip can be seen in preview.

- **Clip length (frames):** Shows the number of frames for the clip.
- **Live Play:** Makes sure that the clip will be played and synchronized to the frame rate of the Viz Engine. This is set in the **Output Format** section of Viz Configuration (see the **Configuring Viz** section of the Viz Engine Administrator Guide). Disable the Live Play to allow the clip playback to be Key Frame animated.
- **Play mode:**
  - **Synchronized:** Runs the clip synchronized with the scene playback. This is set in the **Output Format** section of Viz Configuration.
  - **Frames per second**: Adjusts for different speeds, set in the next parameter.
- **Frame number:** Enables the frame numbers to be manually animated. This is an alternative to playing the whole clip (available when **Live Play** is set to Off).
- **Frames per second:** Sets the frame rate (available when **Play mode** is **Frames per second**.

  > ⚠ **Note:** There is no frame interpolation done when you change the frame rate from the default specified.

- **Loop Video:** Loops the video automatically when enabled.

- **Use decode thread:** Enables split processing of the clip evenly between the machine's processors. It is recommended to have this **on** (default). On multi-processor machines this will gain performance, but threading makes scheduling less predictable and may cause the Viz Engine frame rate to be less stable on a single CPU machine.

  > ⚠ **Tip:** If **Priority** is set to **High**, for best results set **Use Decode Thread** to **off**.

- **Preload frame count:** Works as a buffer to enable smoother playback of the clip. If using Live Play, the Preload frame count can help reducing spike-loads and avoid dropped frames.
- **Frame skipping**: With frame skipping enabled, Viz Engine will skip frames if it needs, to try and maintain real-time graphics playback. With frame skipping disabled the clip will play out in full, as fast as possible but will not try to maintain real-time playback. (Graphics will stagger on the output if the renderer drops out of real-time.) When using the video card's ring buffer functionality the SoftClip frame skipping should be turned off.
- **Force Opaque:** Fills the alpha channel, thus making the video completely opaque. When running videos where the codec leaves the alpha channel blank, the video rendering may become completely transparent.
- **Preferred CPU**: Enables the User to specify a particular processor to handle video play out. The recommended CPU value is based on the specifications of the running machine:
  - If it is a quad core processor the CPU value can be set to 4, but to avoid errors it is recommended to set the value to 2.
  - If it is a machine with lower specifications then the recommended value is 1.
  - The value -1 enables the machine operating system to select which CPU to use.
- **Priority:** Sets the priority of drawing the video to a custom level in Viz Engine giving it more or less processor priority.
- **Clear before playback**: Clears the image before playback when set to **On**. **Live Play** must be set to **Synchronized** and the **Frame number** to be shown from the clip is smaller than the current frame number +1 (available when **Live Play** is set to Off).
- **Auto scale texture:** Scales the video clip to the size of the underlying rectangle on which the video is placed in the container.
- **Border crop:** Determines how much of the border to crop when using the auto scaling over. Due to bilinear texture interpolation, you might see that color from the underlying texture is bleeding into the video texture area. Use this parameter to crop away the affected border pixels.

To Use SoftClip

1. Add a group container to the Scene Tree.
2. Add the **SoftClip** plug-in.
3. Open the SoftClip editor.
4. Click the **Browse...** button to select and load a clip.
   - Loading a clip will make it visible in the Scene. By default, the SoftClip plug-in has the Live Play and Loop video settings enabled, hence, the clip will instantly start playing (and looping) in the render window.
5. Optional: Enable the **Use decode thread** option.
6. Disable the **Live Play** option to enable the **Frame number** setting.

7. Animate the Frame number setting (e.g. from frame 10 to 30).

> ⚠ **Note:** Viz Engine will not interpret between Key Frames, so if you choose to slow down an animation then the motion will not be smooth.

See Also

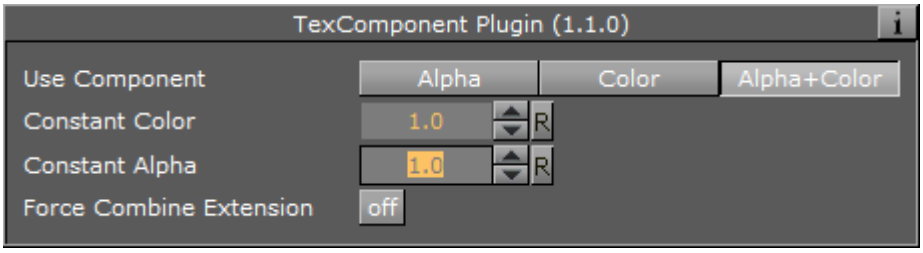
- Video Clips
- Advanced Issues with Video Codecs
- Image Clip
- MoViz

14.18.10 Tex Component

The Tex Component plug-in selects which component of a texture is used for the texture mapping process.

> ⚠ **Note:** This plug-in is a legacy plug-in and the functionality is now incorporated into the Texture Editor.

Tex Component Properties



- **Use Component:**
  - **Alpha**: Uses only the alpha of the texture. The RGB color of the texture will be set to a constant color specified by the **Constant Color** parameter.
  - **Color**: Uses only the RGB color of the texture. The alpha will be set to a constant value specified by the **Constant Alpha** parameter.
  - **Alpha+Color**: Uses both the alpha and the RGB color of the texture (default setting).
- **Constant Color:** Sets the constant RGB color of the texture when **Alpha** is selected.
- **Constant Alpha:** Sets the constant alpha value when **Color** is selected.

14.19 Ticker

The following container plug-ins are located in the Ticker folder:

- Scroller Action

14.19.1 Scroller Action

The Scroller Action plug-in makes it possible to design scrolling tickers that trigger actions when scroller items are played out. Actions can be triggered both before and after reading the next item from the input source, as well as delaying the reading of the next item a certain number of frames.

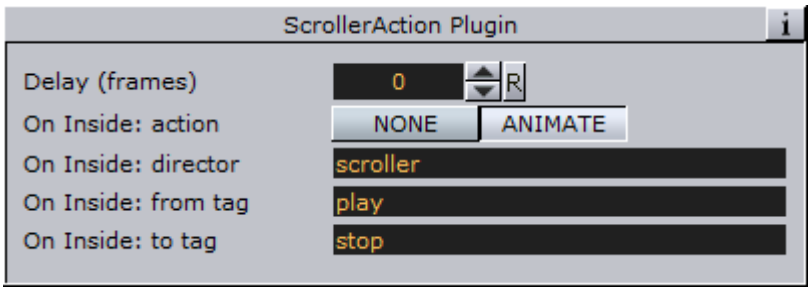
A scrolling carousel scene must have a SoftClip Draw Pixels plug-in attached to a container and be configured to receive items from an external source. In most cases this external source is Viz Ticker's Ticker Service running on the same machine. During scene design, it is useful to use a debugging syntax to specify a fake source that refers to graphics templates located directly in the scene.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> Ticker*

This section contains information on the following topics:

- Scroller Action Properties
- Controlling the Progression
- To Control the Progression

Scroller Action Properties



- **Delay (Frames)**: Adds a delay after the current item is fully inside the scroller, so that insertion of the next item is postponed the given number of frames.
- **On Inside:** Defines if the item should trigger anything after it fully enters the scroller.
  - **Action**:
    - **None:** Does nothing.

        - **Animate:** Allows a director to be triggered and defines the start and stop point of the director animation.
    - **Director**: Defines the director to run when On Inside is triggered.
    - **From Tag**: Defines where to start the director animation.
    - **To Tag**: Defines where to stop the director animation.
- **After Delay:** Defines if the item should trigger anything after the scroller has finished delaying the next item.
    - **Action**:
        - **None:** Does nothing.
        - **Animate:** Allows a director to be triggered and defines the start and stop point of the director animation.
    - **Director**: Defines the director to run when After Delay is triggered.
    - **From Tag:** Defines where to start the director animation.
    - **To Tag**: Defines where to stop the director animation.

    The source syntax (separator) (text) will create a scroller showing an endless stream of items alternating between a separator and a text graphics design. The scroller will search through the scene tree for containers named separator and text, use the graphics template found there, and then copy them to create the item instances that are being scrolled.

Controlling the Progression

A typical example of when the Scroller Action plug-in is necessary, is when text messages (SMS) should scroll into the screen, animate to a halt, wait a few seconds, and then restore the scroll speed whenever new text messages are available.

This paragraph provides guidance on how to create an upwards scroll to show text messages, where the scroller should pause for a while after playing out each message to the screen. The scroller should not immediately follow the text design with the next separator design. Instead, the text design should enter the screen fully, and then the scroller should smoothly slow down to a halt. The scroller should wait a given number of frames before slowly starting the scroller again and pushing the next separator item onto the screen. If an external source is used, the scroller should not start again until there is a new item available from the source.

To Control the Progression

1. Create a director named **scroller.**
    - This director animates the scroller speed.
2. On this director, create a stop point, and call it **stop.**
3. Set the stop scroll speed to **0.**
4. Create another stop point, and call it **play.**
5. Set the play scroll speed to a suitable speed.
   The actual names that are used here are not important, but they must match the names specified in step 8.
6. Add the Scroller Action plug-in to the text template top node (the container holding the Control Object plug-in).
7. In the Scroller Action plug-in editor, define the **Delay** frame count.
   This setting adds a delay after the current item, so that the insertion of the next item is postponed the given number of frames.

The delay counter starts to count from the time the previous item has fully entered the screen, which is a few frames after it would normally try to insert a new item. When delay is set to *0*, the scroller will perform as normal, and not wait until the previous item has fully entered the scroller before inserting a new item.
The *delay* parameter specifies the minimal amount of time that must pass between pushing out the current item, which is based on the template graphics with the *Scroller Action* plug-in, and the next item. The actual amount of time might be greater if there are no new items available after the *delay* frame count has expired. Setting this parameter does not affect the speed of the scroller, it only delays the inserting of the next item, in effect increasing the resulting gap between the previous item and the next.
The default value of the *delay* frame count parameter is 0, which means that the scroller will attempt to insert the next item immediately after the previous item.

8. In the Scroller Action plug-in editor, click the *On Inside: action***Animate** button.
   When *animate* is selected, three additional on *inside* related parameters become available:
   - **On Inside**: director (default value is scroller)
   - **On Inside**: from tag (default value is play)
   - **On Inside**: to tag (default value is stop)

   The parameters should remain at the default values.
   When the current item has fully entered the screen, the scroller will start to animate the *scroller* director from a *play* tag to a *stop* tag. The rules of the *GOTO_TRIO* director command defines this behavior.
9. In the Scroller Action plug-in editor, click the *After Delay: action***Animate** button
   When *animate* is selected, three additional *after delay* related parameters become available:
   - **After Delay: director** (default value is *scroller*)
   - **After Delay: from tag** (default value is *stop*)
   - **After delay: to tag** (default value is *play*)

   The parameters should remain at the default values.
   When the *delay* frame count expires, the scroller will start to animate the *scroller* director from a *stop* tag to a *play* tag. The rules of the *GOTO_TRIO* director command defines this behavior.

See Also

- Transition Logic

14.20 Time

The following container plug-ins are located in the Time folder:

- Analog Watch
- Clock Rotation

14.20.1 Analog Watch

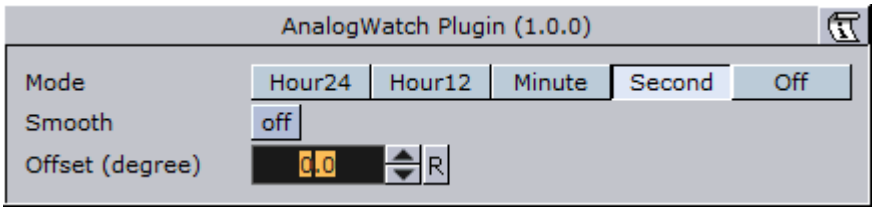
The Analog Watch plug-in lets you create a real-time animated clock of objects.

⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> Time*

This section contains information on the following topics:

- Analog Watch Properties
- To Create an Analog Watch

Analog Watch Properties



- **Mode:** Sets the mode. Available options are Hour24, Hour12, Minute, Second and Off.
- **Smooth:** Ensures the clock hands rotate smoothly. Check this button to enable smooth rotation. Hour hands always rotate smoothly, so this switch is inactive if the plug-in is set to Hour24 or Hour12 mode.
- **Offset:** Changes the rotation offset. 1200 usually is equal to a rotation of 0 degrees.
  - 1 hour in Hour24 mode: 15 degrees.
  - 1 hour in Hour12 mode: 30 degrees.
  - 1 minute in Minute mode: 6 degrees.
  - 1 second in Second mode: 6 degrees.

To Create an Analog Watch

1. Add a **Circle** geometry to the scene tree, add material to it, and name it **Clock**.
2. Open the **Circle editor** and set scaling to **2.0**.
3. Add three **Arrow** geometries as Sub-Containers of the **Clock** container, add material to them, and name the first **Hours, the** second **Minutes**, and the third as **Seconds**.
4. Open the **Arrow editor** for Hours and set the following properties:
   - Style1 to Flat.
   - Width and Arrow Width to 4.0
   - Percent to 30.0.
5. Open the **Arrow editor** for Minutes and set the following properties:

- Style1 to Flat.
- Width and Arrow Width to 3.0
- Percent to 40.0.

6. Open the **Arrow editor** for Seconds and set the following properties:
   - Style1 to Flat.
   - Width and Arrow Width to 2.0
   - Percent to 50.0.
7. Add the Analog Watch plug-in to the Hours, Minutes and Seconds containers.
8. Open the Analog Watch editor for the Hours container and set it to Hour24 or Hour12.
9. Open the Analog Watch editor for the Minutes container and set it to Minute.
10. Open the Analog Watch editor for the Seconds container and set it to Second.

See Also

- Clock Rotation
- System Time

14.20.2 Clock Rotation

Animates any object as a rotating clock.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Clock Rotation Properties

Clock Rotation Properties

- **Clock Unit:** Sets the units for the rotation of the object. Available options are Seconds, Minutes and Hours.
- **Rotation axis:** Sets the axis for the rotation of the object. Available options are X, Y and Z.
- **Motion type:** Sets the motion type for the rotated object. Digital will show a ticking motion, and Chrono a smooth motion.
- **Reverse:** Sets the direction of the rotation. When disabled (off) the rotation is clockwise, and when enabled (on) the rotation is counter-clockwise.

See Also

- Analog Watch
- System Time

14.21 Tools

The following container plug-ins are located in the Tools folder:

- Advanced Counter
- Autofollow
- Autorotate
- Bounding Actions
- Cloner
- Colorize
- Counter
- DVE Follow
- Heartbeat
- Hide On Empty
- Image Link
- Jack
- Level Of Detail (LOD)
- Magnify
- Match It
- Max Size
- Max Size Lines
- Object Zoom
- Omo
- Pablo
- Parliament
- Rotations Order
- Slide Show
- System Time
- Temo
- Text Auto Scale
- TextBG
- Text Link
- Text Parameters
- TransitionLayers
- VCF Parameter

14.21.1 Advanced Counter

The Advanced Counter plug-in allows you to easily create an animated counting sequence. It shows values with decimals and it has the possibility to create a user defined format mask. Furthermore it can have prefixed values.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Advanced Counter Properties

- To Create an Advanced Counter

Advanced Counter Properties

- **Use Integers**: Enables the use of integers only.
- **Value (Dec.)**, **Value (Int.)**: Sets the current value of the counter. Animate this value to create the counting sequence.
- **Leading blanks:** Selects what the counter does with leading blanks. They can either be **Cut** away or put in as **Blank**, asterisk, hash, dot or zeros.
- **Force prefix:** Enables both positive and negative values to have a prefix, not only negative values as is the default (+/-).
- **Delete unnecessary commas**: Removes superfluous commas when a specified format mask has more commas than the number needs. If for instance the mask `$###,###,###,###.##` is defined, and the number `4120.37` is entered, it will be output as **$,,4,120.37**. If by enabling this option, the number will get a correct format: **$4,120.37**.
- **Decimal separator**: Switches the decimal separator between point and comma.
- **Format mask:** Allows format mask definition. Each hash symbolizes an item of the total number. To alter, add or remove hashes. You can also add constant values, for example DM, NOK etc.
- **Initialize**: Starts the counter.

To Create an Advanced Counter

1. Add a group container to the scene tree.
2. Add a font and material to the group container.
3. Add the Advanced Counter plug-in to the group container.
4. Open the Advanced Counter editor and animate the parameters.

14.21.2 Autofollow

The Autofollow plug-in is used to connect one object with another. This object is from then on dependent on the connected plug-in. Depending on the axis of your Object to follow, the bounding-box will grow into the reverse direction.

If you want to follow for example a text object, which writes from left to right, you have to set the axis of your text object to the left side, because the bounding box will grow to the right side. You can select for the following object either one axis (X,Y,Z) or two (XY, XZ, YZ).

Drag the Autofollow plug-in over the object container which should be dependent. It will automatically follow the Previous object in the scene tree. You may also set this to be the First object or select one specific object by adding it to the object placeholder exposed when selecting the Other option.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Autofollow Properties

Autofollow Properties

- **X Previous Center:** Defines the alignment of your related object.
- **Default Distance (Min):** Allows you to set a distance from the object to the related one. This is a relative value that is dependent on the size of the related container.
- **Progress (%):** Allows you yo animate the related object. In this case you can set a default distance value and a maximum distance. What you need to do is just set Key Frames on the progress. At the beginning of the animation you can for example have 100%, and at the end 0%. The object will run from the maximum to the default distance value.
- **Maximum Distance:** Allows you to set the maximum distance from the object to the related one. This is an absolute value, which is independent of the size of the related container.
- **Reference Container:** With these buttons (Previous, First, Other) you can change the related object. The default preference is 'Previous', where the previous container in the scene-tree is selected. If you choose 'Other', you have to drag your wished reference-container onto the empty button, which shows when you select 'Other'.
- **Direction**: Allows you to choose in which axes there should be a dependency on. Options are: X, Y, Z, XY, XZ or YZ.
- **Follow Negative:** If you select this feature, all numerical values are negated. For example when you change 'Default Distance (Min)' to a positive value your object will be positioned in negative direction of the selected axis or axes.
- **Ignore hidden containers:** If the reference container is hidden, search for the next visible container depending on the reference container setting (only if set to Previous or First).
- **Ignore default distance on hide:** Allows you to hide the reference container if you do not want the default distance to be considered. In this case it will be null. If you make the reference container visible, the default distance will show again.
- **X Local Center:** Allows you to select your alignment for the local center in X-direction.

14.21.3 Autorotate

With the Autorotate function, a container can be set to continuously rotate round one of its axes.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Autorotate Properties

Autorotate Properties

- **Axis:** Selects the rotation axis either **X**, **Y** or **Z**.
- **Direction:** Selects the direction of the rotation, either **Right** or **Left**.
- **Velocity**: Sets the speed of the rotation. The unit is degrees per frame.
- **Ease In (frames):** Sets the number of frames the object will use to accelerate smoothly from no motion to the speed set in **Velocity**.
- **Ease Out (frames)**: Sets the number of frames the object will use to retard from the rotation speed to a stop.
- **Starting Angle**: Sets the angle from the objects current position, from which the object is to start from.
- **Ping Pong**: Enables a “ping pong” motion where the object rotates forward and backward between two angles on the axis.
- **Ping Pong Angle 1:** Sets angle 1 for the “Ping Pong Mode”.
- **Ping Pong Angle 2**: Sets angle 2for the “Ping Pong Mode”.
- **Ease in**: Initializes the rotation from the starting angle selected and with the **Ease in** value selected.
- **Ease out:** Stops the rotation. The stop motion is influenced by the **Ease out** parameter.
- **Continue:** Starts the rotation from the point where it was halted. The value of **Ease in** is disregarded.
- **Halt**: Stops the rotation without regard to **Ease out**.

14.21.4 Bounding Actions

The Bounding Actions plug-in enables you to run Viz Artist/Engine actions depending on the size of the bounding box.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Bounding Actions Properties
- To use bounding actions

Bounding Actions Properties

- **Text box**: The command, that will be executed every time the bounding box changes.
- **Scale X, Y and Z**: Applies a scaling to the values %dx %dy %dz.
- **Offset X, Y and Z**: Adds an offset to the values %dx %dy %dz.

Syntax:

- **%dx:** Substitutes the width of the bounding box.
- **%dy:** Substitutes the height of the bounding box.
- **%dz:** Substitutes the depth of the bounding box.
- **%xbool:** 0 if dx <= 0 else 1.
- **%ybool:** 0 if dy <= 0 else 1.
- **%zbool:** 0 if dz <= 0 else 1.
- **%container:** Path of the container the bounding action is applied to.
- **%container[path]:** Relative path of the container the bounding action is applied to.

Example:

```
THIS_SCENE*TREE*#297*TRANSFORMATION*POSITION SET %dx %dy %dz
```

Example:

```
%container[$textnode1$textnode2]*GEOM*TEXT SET %xbool
```

To use bounding actions

1. Drag the plug-in onto a container.
2. Insert a command in the Text box (see examples above).
    - Now every time the bounding box changes (a Sub-Container is moved outside the current bounding box or the bounding box plug-in is used to modify it), this command is executed.

14.21.5 Cloner

Creates a number of clones of a given container (target).

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Cloner Properties
- To clone container

Cloner Properties

- **Clones Count:** Specifies the maximum number of clones to be created.

- **Clones Rows:** Defines, together with Clones Cols, how clones are being laid out. Layout depends on the Shape parameter which can be Matrix (grid), Ellipse and Sphere.
- **Clones Cols:** See Clones Rows description above.
- **Shape:** Sets the different shapes. Available options are Matrix, Ellipse and Sphere.
- **Rename Subtrees:** Clones will be labeled "group_x_y" per default. If this flag is set, clones will be labeled "<name of target>_x_y". Only takes effect upon future changes in the subtree.
- **Rename Excess Containers:** Suppose a matrix of 10x10 clones. If you now change the matrix to be made of 5x5 clones you will be left with 75 orphans, which will be removed if this flag is set. Only takes effect upon future changes in the subtree.
- **Remove All (button):** Removes all clones immediately.
- **Remove Only Excess (button):** Removes excess clones immediately.

Matrix and **Ellipse** shape specific settings:

- **Plane:** Specifies along which axes, newly generated clones are going to be laid out. Default is XY, alternatives are XZ and YZ.
- **Delta X, Y and Z:** Specifies an relative offset in X,Y and Z axis between each clone. Actually defines how closely clones are being packed into the matrix or ellipse.
- **Delta represents:Step** directly affects the translation matrix of the clones 1:1. For example a value of 200 for Delta X means that clone A (located at position 0) will be followed by clone B translated 200 units in X. The container's transformation matrix is being modified accordingly. **Bounding Box** does not affect the clones transformation matrix 1:1, but instead applies the transformation relative to the bottom-left edge of the target's bounding box in matrix mode. Suppose a rectangle of 100x100 is used as target. Delta X of 100 in Step mode will cause all clones to be positioned with an offset of 100 units in X. In Bounding Box mode a value of 100 will produce an effective offset of only 50.
- **R1 and R2:** Specifies values of radius 1 and radius 2, defining the ellipse. Used in ellipse mode.
- **Min and Max Angle:** Specifies an open ellipse from for example 45˚ to 175˚ instead of 0˚ to 360˚. Used in ellipse mode.
- **Offset Angle:** Rotates the individual clones along the axis. This is not affected by setting of Plane. If the ellipse is laid out in XY this parameter will affect the Z-axis. Used in ellipse mode.

Sphere shape specific settings:

- **Radius:** Specifies the radius of the sphere composed by the given number clones.
- **Min and Max Azimuth:** Same as Min and Max angle for the ellipse.
- **Min and Max Pitch:** Specifies the position of the poles of the sphere. The poles can only be shifted towards the center of the sphere. This will actually result in a sphere with its poles being cropped.

To clone container

1. Add a new group to the scene tree, and add the Cloner plug-in to it.
2. Add the object (target) to be cloned as the first child of this group.
    - By modifying parameters Clones Count, Rows and Cols a corresponding number of clones will be created.

- Operating in shape mode Matrix you can alter Delta X, Y and Z parameters to create a layout for the newly generated clones.

14.21.6 Colorize

The colorize plug-in allows you to transform the colors of containers within a container group. If you for instance have 15 Sub-Containers in a group, the colorize plug-in assign numbers to each container by the order they have in the group. You can then define for each Sub-Container which color it should have to start with and which color it should change to.

The plug-in can define up to 10 colors. Each color is able to influence more than one container. The Num Color1-10 parameters allow you to link multiple containers to one color in the colorize plug-in. Each of the Sub-Containers must have its own material for the plug-in to work.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Colorize Properties

Colorize Properties

- **Colorize:** Runs the color transformation. One number corresponds to one container.
- **Animation Start:** Allows you to select the starting point for the color change.
- **Default Color**: Sets the color for containers in the group which are not selected for color transformation. To set it, use the color editor below or drag a material from the Server Panel onto the small square.
- **Num Color1 A to 10 A**: Assigns the starting colors to containers, from which the color change starts. The value sets the number of containers to be linked with the corresponding color in the Color1 A: 10 A parameters. The function starts counting from the first undefined container. This means that, if color 1 A has the value 2, color 2 A has the value 3 and 3 A has the value 1, the two first containers get the properties of color 1 A, container 3: 5 get the properties of color 2 A and container 6 gets the properties for color 3 A.
- **Color1 A to 10 A**: Sets the colors for 1 A: 10 A. You can set each color using two methods:
  - Either select the color by clicking the color icon and then set the color properties at the color editor at the bottom, or
  - Drag a material from the Server Panel onto the color icon of the color you want to set.
- **Num Color1 B to 10 B**: Assigns the ending color to containers, to which the color change ends. The value sets the number of containers to be linked with the corresponding color in the Color 1 B: 10 B parameters. The function starts counting from the first undefined container. This means that, if color 1 B has the value 2, color 2 B has the value 3 and 3 B has the value 1, the two first containers get the properties of color 1 A, container 3: 5 get the properties of color 2 B and container 6 gets the properties for color 3 B.

- **Color1 B to 10 B**: Sets the color for 1 B: 10 B. You can set each color using two methods:
  - Either select the color by clicking the color icon and then set the color properties at the color editor at the bottom, or
  - Drag a material from the Server Panel onto the color icon of the color you want to set.
- **Rebuild**: After having made changes to colors and assigning containers, click rebuild to apply the changes.

14.21.7 Counter

The counter is a simple plug-in that can create a count up or count down at any given range and at any given speed. It shows integral numbers.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Counter Properties
- To Use the Counter

Counter Properties

- **Number:** Sets the current number of the counter. Animate this value to create the wanted count-up or -down.
- **Initialize**: Starts the operation.

To Use the Counter

- Add the Counter plug-in to a container with a font, and click Initialize.

14.21.8 DVE Follow

DVE Follow connects a video input channel or a clip channel of the containing Scene with the bounding box of a Container. When connected the channel, if set to DVE, will follow the bounding box of the Container. The plug-in does not actively set the channel to DVE.

> ⚠ **IMPORTANT!** Be aware of the fact that the DVE animation is done on the video board while the animation of the ruling container is done together with all other graphics on the GPU. The output delays of the graphics and the DVE animation are different and this can lead to notable offsets between graphics and DVE (especially notable with fast animations).

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

DVE Follow Properties

- **Channel Type:** Type of channel to manipulate:
  - Video Channel
  - Clip Channel
- **Video/Clip Channel Number:** The plug-in supports a maximum of eight video channels and 16 clip channels.
- **Layer:** Selects the layer which contains the Scene in which the DVE is located.
- **Video Width:** Sets the width of the containers bounding box.
- **Video Height:** Sets the height of the containers bounding box.
- **Alpha:** Sets the alpha value of the DVE channel.
- **Priority:** Sets the priority value of the DVE channel.
- **Crop:** Sets the left, right, top and bottom crop value of the DVE channel.

14.21.9 Heartbeat

The Heartbeat plug-in creates a heart beat animation. Includes animation of:

- Size, along different axes, axis-pairs or all axes
- Rotation, around three main axes
- Alpha
- Color

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Heartbeat Properties

Heartbeat Properties

- **Animation Length:** Sets the length of the animation. Parameter Name: `animLength`.
- **Swing:** Enables the object to have a swinging action (back and forth). Parameter Name: `swing`.
- **Pause:** Enables the object to pause before repeating the action. Parameter Name: `pause`.
  - **Pause Length:** Sets the length of the pause. Parameter Name: `pauseLength`.

- **Change Size:** Enables the change size parameters for changing the vector(s), initial and final weight of the object's animation. Parameter Name: `changeSize`.
  - **X, Y, Z, XY, XZ, YZ, XYZ:** Sets animation vector(s). Parameter Name: `changeVector`.
  - **Initial Weight:** Sets the initial weight of the object. The higher the value the slower the animation will be at first. Parameter Name: `initialWeight`.
  - **Final Weight:** Sets the final weight of the object. See also Initial Weight. Parameter Name: `finalWeight`.
- **Change Rotation**: Enables the rotation parameters changing the rotation axes, initial and final angles. Parameter Name: `changeRotation`.
  - **X, Y, Z:** Rotates the object on the X, Y or Z axis. Parameter Name: `rotAxes`.
  - **Initial Angle:** Sets the initial angle of the object. Parameter Name: `initialAngle`.
  - **Final Angle:** Sets the final angle of the object. Parameter Name: `finalAngle`.
- **Change Alpha:** Enables the alpha parameters changing the initial and final alpha values. Parameter Name: `changeAlpha`.
  - **Initial Alpha:** Sets the initial alpha value. Parameter Name: `initialAlpha`.
  - **Final Alpha:** Sets the final alpha value. Parameter Name: `finalAlpha`.
- **Change Color:** Enables the color parameters changing the color of the object. Parameter Name: `changeColor`.
  - **Source Color:** Sets the source color. Parameter Name: `sourceColor`.
  - **Target Color:** Sets the target color. Parameter Name: `targetColor`.

14.21.10 Hide On Empty

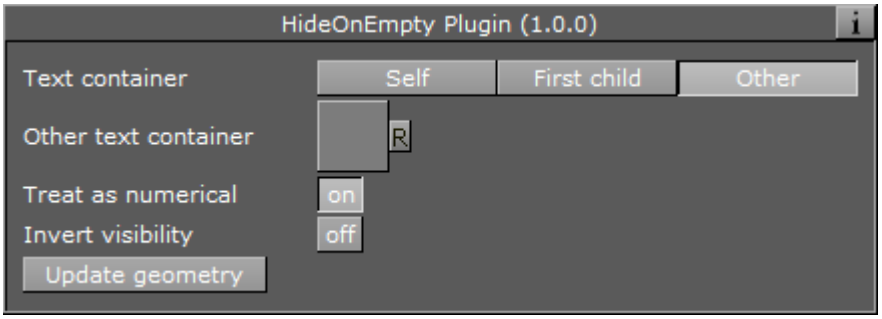
The Hide On Empty plug-in hides its container when a target text geometry matches a given condition. The condition may be an empty text, or text displaying a zero value, such as `0` or `0.0`, depending on the plug-in parameters.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Hide On Empty Properties

Hide On Empty Properties



- **Text container**: Specifies the container of text of which to check the content.
  - **Self**: The text geometry is in the same container as the plug-in is applied to.
  - **First child**: The text geometry is in the first child container of the container the plug-in is applied to.
  - **Other**: Enables the **Other text container** drop area.
- **Other text container**: Drag the container containing the text geometry to set the target. Enable by selecting `Other` as **Text container**.
- **Treat as numerical**: Hides the target container if the specified text is `0` or `0.0`, in addition to empty text.
- **Invert visibility**: Displays the container when the condition is satisfied instead of hiding the target container when the condition matches.
- **Update Geometry**: Updates the visibility state of the container holding the plug-in after changes, removal, or assignment of target text geometry, during the design process.

Note: You cannot use the control plug-in Control Hide on Empty in conjunction with Control Text using the same field identifier. In such a case, use the **Hide On Empty** property in the TextBG plug-in.

14.21.11 Image Link

The Image Link plug-in copies the image/texture from the source container to up to 10 linked containers. The image you load in the source container will be repeated in all the linked containers.

Note: This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Image Link Properties
- To Link an Image to Multiple Containers

Image Link Properties

- **Container 1**: **10**: Drop zones for the containers you want to link.
- **Initialize**: Initializes the plug-in.
- **Do it now:** Performs the operation immediately.

To Link an Image to Multiple Containers

1. Add a group container to the scene tree.
2. Add the Image Link plug-in and the source image to the group container.
3. Add a number of containers to the scene tree with a dummy image on each.
4. Open the Image Link editor.
5. Drag and drop the containers with the dummy images onto the Container 1-n drop zones.
6. Click **Initialize** to apply change.

See Also

- Text Link

14.21.12 Jack

The Jack plug-in allows you to show a digital clock and date. The plug-in is highly customizable allowing you to set your own time and date formats and correction values.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Jack Properties

Jack Properties

Time mode has the following properties:

- **Hour format:** Sets how to show hours (either off, on or with a leading zero if the value is less than 10).
- **Min format:** Sets how to show minutes (see Hour format).
- **Sec format:** Sets how to show seconds (see Hour format).
- **Hour/Min separator:** Sets character separation for hours and minutes.
- **Blink mode:** If set to on the separator between hours and minutes will blink, this means - every odd second it will be turned off.
- **Min/Sec separator:** Sets a separator for minutes and seconds.

> ⚠ **Note:** Although a format is switched off, the separator will still be shown.

- **am/pm or 24 hours:** Switches between the am/pm and 24 hour format.
- **Time/meridian separator:** Sets a separator for the am/pm hour format.
- **am/pm format:** Sets the case of am/pm to upper or lower.
- **Hour correction:** Enables you to correct the hour value of the system time. Useful to show the local time of a location in a different time zone.
- **Min correction:** Sets correction value for minutes (see Hour correction).
- **Sec correction:** Sets correction value for seconds (see Hour correction).

Date mode has the following properties:

- **Year format:** Sets how to show the year format (off, 2 digits or 4 digits, please keep in mind that in the year 10000 this will be 3 digits or 5 digits, due to the algorithm used).
- **Month format:** Sets how the month is shown, either off, on or with a leading zero from January to September.
- **Day format:** Sets how the day is shown (see Month format).
- **First separator:** Sets the separator character for the first two items of the date. If the dot value is selected an additional blank character will be inserted automatically.
- **Second separator:** Sets the separator character for the last two items of the date.
- **Order:** Sets the order of the items of the date.
- **Year correction:** Sets a correction value for the year.
- **Month correction:** Sets a correction value for the month.
- **Day correction:** Sets a correction value for the day.

14.21.13 Level Of Detail (LOD)

Level Of Detail (LOD) is a mechanism for controlling the level of complexity of a 3D object, depending on camera position and object size. As an object becomes smaller on the screen, the objects polygons become smaller. At a certain size those polygons are small enough to be replaced by a lower number of larger polygons, without loosing realism. The goal is to maintain the overall shape of the object so it will not change significantly, when the number of polygons are changing.

Viz Artist has a built-in LOD mechanism for text objects and internal objects, such as Sphere, Cylinder and so on. This mechanism changes the tessellation of the object, based on its size on screen. However, there is no LOD mechanism for imported objects and models. This is where the LOD plug-in can be used. The LOD plug-in enables switching between different representations (levels) of the same object, based on its distance from the camera and on the zoom level of the camera. A switching range is assigned to each representation: to determine the specific distance from camera in which it will be replaced by the next representation.

In addition to that, the zoom of the camera is taken into consideration; objects that are far away from the camera will look closer (and larger) at a narrow zoom position, or smaller at wide zoom. To accommodate for different zoom values, the LOD plug-in assumes that the switching ranges are correlative to a reference zoom value. When the actual zoom value differs from the reference zoom value, the switching ranges are automatically adjusted.

To use the LOD plug-in on an object, you have to import several representations of that object from an external source. The different representations of the object have to be placed under a group node, in the order of detail, this means the most detailed level will be on the top of the group. When the LOD plug-in is placed on the group container it will make only one of its children (levels) visible at a given time, based on the distance between the center of the object and the camera, as well as the current zoom value, and other selectable parameters of the plug-in.

The LOD container plug-in works in conjunction with the Level Of Detail (LOD) Manager scene plug-in that controls all the LOD plug-ins in the scene.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- LOD Properties

LOD Properties

- **Lock**: Disables distance switching. In certain cases, it is desirable to disable the distance switching and lock the LOD plug-in to show one of the levels.
- **Lock On Index:** Shows index of the selected level when **Lock** is **on**.
- **Center**: Defines whether the distance from the camera will be measured from the origin of the object (**Automatic**) or from a specific point in the object, for example one of the corners (**Manual**)
- **Center X**, **Y** and **Z:** Sets the point from which the distance from the camera is measured (in object coordinates), if the **Center** parameter is set to **Manual**.
- **Range Scale**: Sets a scale factor on the switching ranges. This scale factor allows shortening or lengthening the distance at which object representations are switched.
- **Range 0**: Sets the switching ranges. Those are the distances at which object representations are switched. The number of entries is `n+1` where `n` is the number of children in the group on which the LOD plug-in is assigned
- **Initialize**: Re-initializes the LOD plug-in. This is required if the number of children of the LOD group has changed after the LOD plug-in was assigned to the group.

See Also

- Level Of Detail (LOD) Manager

14.21.14 Magnify

The Magnifying plug-in offers the possibility to create a simulation of a looking glass and other similar effects. At an area defined by an overlying object an image can be shown in a magnified way.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Magnify Properties
- To Magnify a Texture

Magnify Properties

- **Scale**: Sets the scaling of the magnifying effect.
- **Effect Type:** Adds an extra effect, either **Blur** or **Pixel**. This only works on onux.
- **Effect Value**: Sets the value for the selected extra effect.
- **Alpha**: Sets the alpha value for the container.
- **Rebuild**: Applies the changes to the texture.

To Magnify a Texture

1. Add a Container to the Scene Tree.
2. Name it **source**.
3. Add the image or texture that is to be magnified to the **source** Container.
4. Add a Sub-Container to the **source** Container
5. Name it **magnify**.
6. Add the same image or texture to the **magnify** Container that was used in the **source** container.
7. Add the Circle geometry to the **magnify** Container.
8. Open the transformation editor for the **magnify** Container.
9. Set Position Z to 1.0.
10. Add the Magnify plug-in to the **magnify** container.
11. Open the **Magnify** editor.
12. Set Scale to 2.0.

14.21.15 Match It

Sets the axis center (i.e. center for rotations) to the camera and moves the object away from the camera by the specified amount of units. By setting an arbitrary camera distance and initializing the plug-in, the container with the Match It plug-in will be positioned in the specified distance and any rotation will be done with respect to the camera position. The container's axis center parameter can also be edited for a more flexible configuration.

⚠ **Note:** Keep in mind that you have to re-initialize the plug-in if the camera is moved.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Match It Properties

Match It Properties

- **Initialize:** Resets the pivot to the camera and places the object 'CamDistance in Z' units away.
- **CamDistance in Z:** The distance from the camera on the z axis.

14.21.16 Max Size

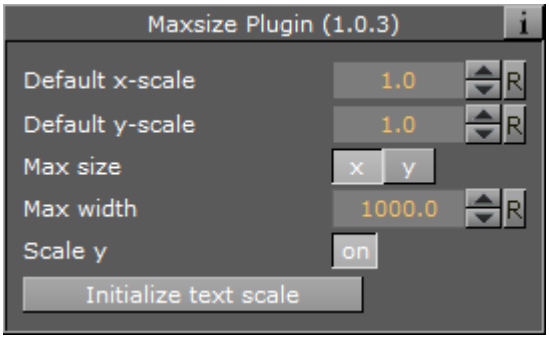
The Max size plug-in is used to set max size parameters for text objects. This allows you to have control over the space the text should use so it does not 'overflow' the designated text area.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Max Size Properties
- To Set Maximum size

Max Size Properties



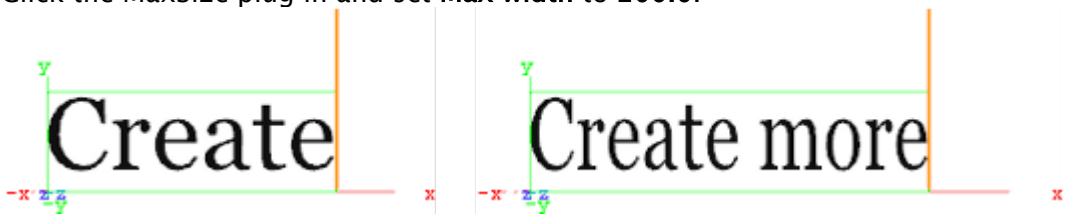
- **Default x-scale:** Sets the default x-scale.
- **Default y-scale:** Sets the default y-scale.
- **Max size:** Sets the direction of the size to be controlled:

- **x**: Controls the width.
- **y**: Controls the height.

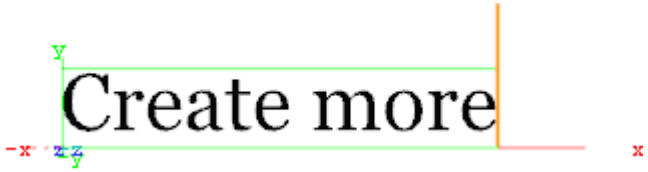
- **Max width/height:** Sets the maximum width or height, depending on whether **x** or **y** is selected for **Max size**.
- **Scale y/x:** Scales the **y** or **x** value accordingly when maximum width or height is reached, respectively.
- **Initialize text scale:** Initializes the plug-in parameters.

To Set Maximum size

1. Add a group container to the scene tree.
2. Add a group container as a Sub-Container of the first (scaled to 1,1,1) and name it **text**.
3. Add a font (e.g. Georgia Regular) to the text container.
4. Add the **MaxSize plug-in** to the text container.
5. Click the MaxSize plug-in and set **Max width** to **200.0**.



6. Click the font and type *Create.*
7. Now type *Create more.*
   - This will automatically scale the font's X value as it limits itself to the Max width.



8. *Optional*: Click the MaxSize plug-in and enable the **Scale Y** parameter to automatically scale the font's Y value.
   When working with scenes that includes vertically aligned text direction, such as East Asian languages where top to bottom columns of text can be desirable, please pay attention to the vertical orientation selected in the Text editor. When the text scale is adjusted with Max size, the selected vertical orientation can affect the position of the text geometry:

   - **Top:** Centers the geometry at the top of the boundary. Changes to the text scale will not change the geometry position.
   - **First Line:** Centers the geometry at the base of the first line. Because of this, changes to the text scale will also move boundary of the geometry.
   - **Center:** Moves the boundary when the text scale changes because the center of the geometry is not the same as the center of the boundary, the boundary will be moved slightly when the text scale is changed.

- **Bottom:** Centers the geometry at the base of the last line of text, but this is not the same as the bottom boundary. This results in the boundary being moved when the text scale is changed. This allows for more flexible choices when designing a scene, and should be selected based on the desired result.

See Also

- Max Size Lines
- Text Editor

14.21.17 Max Size Lines

The Max Size Lines plug-in adds multi-line support for TextFX with right-to-left languages. Each line of text must be in separate containers, and organized as a sub group of the container holding the Max Size Lines plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Max Size Lines Properties

Max Size Lines Properties

- **Default x-scale:** Sets the default x-scale.
- **Default y-scale:** Sets the default y-scale.
- **Max width:** Sets the max width.
- **Scale y:** Scales the y-value accordingly when max width is reached.
- **Initialize text scale:** Initializes the plug-in parameters.

See Also

- Max Size

14.21.18 Object Zoom

Object Zoom lets an object follow the camera and match its orientation. This creates an effect as if the camera would smoothly zoom to the object.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Default

This section contains information on the following topics:

- Object Zoom Properties
- To Zoom an Object Towards the Camera

Object Zoom Properties

- **Counter:** Determines the speed of the zoom operation. Use values from 1-1000.
- **Dest. Object:** Identifies the name of the container that will be affected by this plug-in.
- **Dest. Dist.:** Determines the distance from the camera at which the object will stop.
- **Dest. Size %:** Determines the scaling of the object.
- **Path HOffset:** Determines the horizontal offset in camera space.
- **Path VOffset:** Determines the vertical offset in camera space.
- **Initialize (button):** Starts the zoom.

To Zoom an Object Towards the Camera

1. Add the Object Zoom plug-in to a container.
2. Set the **Dest. Object** parameter to match the containers name.
3. Set the **counter** to at least **1**.
4. Click **Initialize**.
5. Go to **Views** and move the camera to see the container zooming towards the camera.

14.21.19 Omo

The Object moving (Omo) plug-in gives the user the possibility to create a very realistic animation of a complex imported 3D object, where the object not just changes position or rotates, but where it changes form and moves in a realistic way, like a man walking or a fish swimming.

The plug-in can also create animation sequences with other sorts of objects. Anything you can put in a container can be used as an item in an Omo animation process.

The plug-in hides all Sub-Containers and shows them one by one. The one to be shown is controlled by the **Visible Container** parameter. By animating this value an animation sequence of all the containers is made. The plug-in can also be used to show one group at a time.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Omo Properties
- To Animate One Object at a Time

Omo Properties

- **Visible Container**: Allows you to select which container that is to be visible.
- **Keep Visible**: Enables you to keep containers visible after they have been revealed one by one.
- **Initialize:** Initializes the Omo plug-in. All containers in the group are then hidden, except the one selected by the **Visible Container** parameter.

To Animate One Object at a Time

1. Create a group and add the Omo plug-in to it.
2. Create a number of Sub-Containers to the root container.
3. Add a Sphere geometry and material to each Sub-Container.
4. *Optional*: Animate the sphere.
5. Open the Omo editor, click the Initialize button, and animate the Visible Container parameter.

See Also

- Control Omo

14.21.20 Pablo

Pablo is an object import plug-in that enables you to import multiple indexed images/objects from the pool under the local container Pablo is added to.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Pablo Properties

Pablo Properties

- **Type:** Select wether you want to import images (from the image pool) or 3D objects (from the object pool).
- **Name:** Sets the name of the images/objects you want to import, including path and index number format. Index number format is specified with characters. Use multiple characters for index numbers with leading zeroes. The name must match the path and name of an existing image/object in one of the pools.
- **Start index:** Sets the index number of the first object to be imported.
- **End index:** Sets the index number of the last file to be imported.

14.21.21 Parliament

The Parliament plug-in is specially developed for creating visualizations of parliament seats in graphics designed for election programs, but it can of course be used in other ways. The plug-in creates a “parliament like” seating structure using an object of your selection.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Parliament Properties
- To Create a Parliament Shape

Parliament Properties

- **Number of items:** Allows you to set the number of seats to be created in the parliament structure.
- **Number of Rows**: Allows you to decide how many rows the parliament seats should be distributed over.
- **Scale:** Sets the Scaling (locked) of the generated Sub-Containers. Default is 1.0.
- **Rebuild Geometry**: After having made changes to either **Number of items** or **Number of rows**, you must click on this button to apply the changes.
- **Inner Radius**: Sets the inner radius of the parliament.
- **Outer Radius:** Sets the outer radius.
- **Start Angle**: Sets the starting angle which by default is -90˚ from the Y-axis.
- **End Angle**: Sets the ending angle, default 90˚.

To Create a Parliament Shape

1. Add a group to the scene tree, add material, rotate and set parameters (except Position and Scaling).
2. Add the Parliament plug-in to the container, and click the Initialize button.
    - By default the seating is created with 40 seats in 3 rows. The plug-in then by default creates 40 Sub-Containers that are copies of the core-object.
    - All the newly created Sub-Containers have the same properties as the source object, except from Position and Scaling (if set). Position is set by the plug-in to create the parliament structure.
3. Open the transformation editor for the first Sub-Container and set Scaling (locked) to 0.2.
4. Still having the transformation editor open for the first Sub-Container, select all other Sub-Containers and drag and drop the Scaling property onto on of the selected Sub-Containers.
    - This will set the same Scaling parameters for all selected Sub-Containers.

> ⚠ **Note:** If the object you used for creating the seats contains many polygons, the total product of the parliament container can quickly be too heavy to render.

14.21.22 Rotations Order

Rotations Order changes the rotation order of a container in Viz Artist to match the Softimage XSI rotation order. This plug-in is automatically added to some containers when Softimage XSI scenes are imported into Viz Artist.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Rotations Order Properties

Rotations Order Properties

- **Rotation Order**: Sets the order of rotation to either:
  - **XYZ**
  - **ZYX**
  - **ZXY**
  - **YXZ**

14.21.23 Slide Show

The Slide Show plug-in allows you to create a sequence where a group of containers fade in and out one by one in the same manner as a normal slide show.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Slide Show Properties
- To Create a Slide Show

Slide Show Properties

- **Blend Value:** Controls the slide show sequence. Animate this value to create the change from slide to slide.
- **Use Offset**: Enable this option if the slides have the same initial position and you want them to move along some of the axes as the slides are being shown.
- **X**, **Y** and **Z Offset**: Sets the offset values of each slide calculated form the one previously shown.
- **Fade Out**: May be set **Off**, **On** (every slide) or **Last** (last slide only)
- **Start with:** Sets what should be visible if slide show is not started.

To Create a Slide Show

1. Add a group to the scene tree, and name it **slideshow**.
2. Add a number of Sub-Containers that will be used to house the slide(s).
3. Add the Slide Show plug-in to the slideshow container.
4. Position the Sub-Containers in advance using the transformation editors for each container, or open the Slide Show editor and offset them to create a moving sequence.
5. Animate the blend value to create the slide show sequence.
6. Click the Initialize button to initialize the plug-in.
    - Each of the Sub-Containers automatically get an alpha icon.

14.21.24 System Time

The System Time plug-in creates a variety of time and date settings based on the system time. This plug-in adheres to the standard **IEEE Std 1003.1, 2004 Ed.**

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- System Time Properties

System Time Properties

- **format (aAbBcdDHIijlmMpSUwWxXyYzZ/:,.)**: Sets the format of the system time.
- **Language**: Sets the language format of the system time. Options:
    - Default (system specific)
    - English
    - German
    - Norwegian
- **Seconds offset from system clock**: Sets the number of seconds the system clock should be offset (default is 0).

- **Hours offset from system clock**: Sets the number of hours the system clock should be offset (default is 0).
- **System time zone:** Sets the system time zone.
- **Show Time Zone:** Shows the result of the system time zone.
- **Lower Case Text:** Shows all text in lower case.

Time Zone: tzn[+|-] hh[:mm[:ss]] [dzn]

- **tzn:** Time zone name, e.g. PST
- **hh,mm,ss:** Offsets from the local timezone to UTC (not the reverse).
  - For time zones ahead of the set one, the time difference is negative.
  - For time zones behind the set one, the time difference is positive.
- **dzn:** Daylight-saving-time zone, e.g. PDT

Examples: EST5EDT, PST5PDT, GST-1GDT

See Also

- Analog Watch
- Clock Rotation

14.21.25 Temo

The Temo plug-in gives the user the possibility to create an image animation sequence, using a method much like the one used for making a cartoon film. The basic input for the plug-in is a single image consisting of many tiled and equally sized squares set up in a matrix.

Each square of the image is made up to be a snapshot of an animation sequence, just like each picture frame is on a normal film. The image must be made up in advance with the aid of a image editing tool. In the property editor you tell the Temo plug-in how many tiled squares there are in the X- and the Y-axis. The plug-in is then able to show the tiles of the image one by one. By animating this, a “film like” sequence is created. The plug-in has only a small influence on the rendering performance.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Temo Properties

Temo Properties

- **Number of tiles horizontal:** Defines the number of tiles in the X-axis of the source image.
- **Number of tiles vertical:** Defines the number of tiles in the Y-axis of the source image.
- **Show Tile Number:** Selects the tile to be shown. The range of tile numbers is the product of the two above parameters. If there are three in the horizontal and four in the vertical plane, the total is of course 12. The tile with the coordinates X1Y1 is shown first, then X2Y1 and so on. By animating the whole range of this value, a “film animation” can be created. The

quality of the animation depends of course on the number of tiles and on the degree of transformation from one tile to the next throughout the whole sequence.

14.21.26 Text Auto Scale

Text Auto Scale automatically scales text within a **Text Box** area as defined in the Text Editor.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

- **Enabled**: The plug-in is enabled by default when added to a container. Click the button to disable.
- **Default x-scale**: Sets the default scale value for the X-axis.
- **Default y-scale**: Sets the default scale value for the Y-axis.

> ✓ **Tip**: When using the Text Auto Scale plug-in, Text Box scaling should be set to `Off` to avoid conflict in scaling operations.

14.21.27 TextBG

The TextBG plug-in creates a dynamically sized background for text, images or video. For example, you can use it to create a background that expands as text is entered in the Text Editor. The plug-in must be added to a container that also holds a Rectangle, Noggi, Fade Rectangle or Cube geometry plug-in. To use the plug-in with anything other than fonts, set the **Text Parameter** option to **Scale**.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- TextBG Properties
- To Add Background to the Previous or Next Containers

TextBG Properties

- **Size Parameter:** Sets the size of the background to adjust to:
  - **W/H (width/height):** The width and height of the background geometry will adjust to the size of the typed text.
  - **Scale:** The background will scale to the size of the target container.

- **Text Container:** Sets the inputs source for the TextBG plug-in:
  - **PREV (previous):** The background geometry is applied to the container previous to the TextBG container.
  - **PARENT:** The background geometry is applied on the parent container.
  - **NEXT:** The background geometry is applied to the next container from the TextBG container.
  - **OTHER:** Enables the **Other container** field.
- **Other container:** Available when **OTHER** is selected. Drag a container from the same group to the drop area. Click on **R** to remove the Container.
- **Direction:** Sets the direction in the background grows when adjusting itself to the text. Available options are **X**, **Y**, or the **XY**-axes in combination.
- **Hide On Empty**: Hides the background container if there is no text. Set to `off` to enable **Always Keep Margins**.
  - **Always Keep Margins**: Makes the background geometry retain the defined margins as minimum width and height, even when the text container is empty. Enabled by setting Hide On Empty to `off`.
- **High Precision**: Adds one decimal point precision to the **Margin** fields.
- **Right, Left, Bottom and Top Margin:** Sets the margin between the text and the background in centimeters. Accepts values in the range `-1000.0` to `1000.0`.
- **Lock Margins:** Sets the margin to be the same for all.
- **Enable Substring:** Enables the **First** and **Last Character** options. Substring cannot be used with multiple texts.
  - **First Character:** Sets on which character the background should start animating.
  - **Last Character:** Sets on which character the background should stop animating.

To Add Background to the Previous or Next Containers

1. Add a group container to the Scene Tree.
2. Add two sub-containers to the `group` container:
   - Name the top one `Text1`.
   - Name the bottom one `Background`.
3. Add a font to the **Text1** container.
4. Open the Transformation Editor of the Text1 container.
   a. Set **Position Z** to 1.0.
   b. Enter some text. In this example, the text is `More Control`.
5. Copy the `Text1` container and place the copy below the `Background` container in the scene tree. Move the container so that both texts are visible, as seen in the screenshot above.
6. Rename the new container to `Text2`.
7. Add a geometry, for example the Noggi, a material, and the **TextBG** plug-in to the `Background` container.
8. Open the TextBG editor.
9. Toggle the **Text Container** property from **PREV** to **NEXT**. The background geometry moves from `Text1`, or the *previous* container, to `Text2`- the *next* container. If you resize the text in the Text2 container, the background geometry will automatically resize to fit the text.

> ⚠ **Tip:** To apply a background to a different container, select **OTHER** as **Text Container** and drag the desired container to the drop area.

To use TextBG to apply a background to the parent container, which is `group` in this example, select **PARENT** as **Text Container**. This requires that the parent container contains a text, image or video object working with TextBG.

14.21.28 Text Link

The Text Link plug-in copies the text from the source container to up to 10 linked containers. The text you enter in the source container will be repeated in all the linked containers. The linked containers must contain a font.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Text Link Properties
- To link text to multiple containers

Text Link Properties

- **Container 1**: **10:** Drop zones for the containers you want to link.
- **Initialize**: Initializes the plug-in.
- **Do it now**: Preforms the operation immediately.

To link text to multiple containers

1. Add a group container to the scene tree.
2. Add the Text Link plug-in and a font to the group container.
3. Add a number of containers to the scene tree with a font on each.
4. Open the Text Link editor.
5. Drag and drop the containers with the fonts onto the Container 1-n drop zones.
6. Click **Initialize** to apply change.

See Also

- Image Link

14.21.29 Text Parameters

The Text Parameters plug-in offers the designer the possibility to animate text parameters, as

standard text parameters in the Text Editor cannot be animated. The parameters for the Text Parameters can only be used for animation.

The settings are not visible on the text until an animation update has been made.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- Text Parameters Properties

Text Parameters Properties

- **Use Kerning**: Enables or disables the use of Kerning.
- **Tracking**: Sets the space between each letter.
- **Leading**: Sets the space between each line of text.
- **Word Spacing**: Sets the space between each word.
- **Use Soft Shadow**: Enables the use of Soft Shadow. Level 4 is the most blurred, softest shadow. The Drop Shadow option must be enabled for the Soft Shadow to be visible.
- **Use Drop Shadow**: Enables or disables Drop Shadow.
- **Direction**: Sets the direction of the Drop Shadow in degrees.
- **Distance**: Sets the distance of the Drop Shadow.
- **Z Offset**: Sets the offset of the Drop Shadow in the Z-axis.
- **Color:** Sets the color of the shadow.

14.21.30 TransitionLayers

The TransitionLayers plug-in enables a new way to define the two dynamic images of a scene-transition scene, using the **Dynamic Scene** Media Asset (see Create Transition Effects).

TransitionLayers Properties

- Layer1 Container: Container holding the scene to transition *from*. The referenced containers' name is shown to the far right
- Layer2 Container: Container holding the scene to transition *to*. The referenced containers' name is shown to the far right
- Reset buttons: Each layer container has a reset button, which will empty the container reference

> ⚠ **Tip:** By pressing the group button for the source or destination layers, the corresponding container will be selected in the scene tree.

When adding the TransitionLayers plug-in to a container, the plug-in will automatically create the two sub-containers required and add these to the plug-in properties:

1. Parent container holding the TransitionLayers plug-in
2. Sub-container holding the scene to transition *from*, or Layer1 Container
3. Sub-container holding the scene to transition *to*, or Layer2 Container
   In addition, a default two second alpha in animation will be added for the Layer2 Container to the **Default** director in the Stage. This default animation can of course be changed as desired.

⚠ **Tip:** Containers holding the dynamic images for the transition scene can also be dragged onto the desired Layer container in the plug-in properties pane.

14.21.31 VCF Parameter

The VCF (Virtual Camera Flight) Parameter container plug-in works in conjunction with the VCF scene plug-in. They allow you to create a seamless interpolated transition from a VCF to a real camera: and vice versa. This is only a relevant function to set up, if you have purchased the virtual studio expansion components.

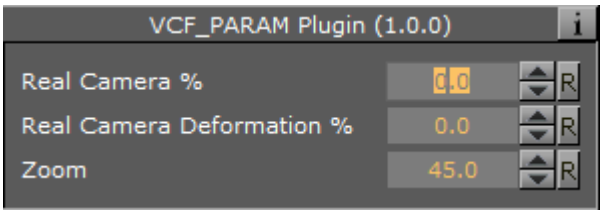
You must have a real camera with data tracking enabled, which is set in remote mode in the camera editor. In case you have several tracked cameras, the virtual camera will interpolate its position to the real camera that is selected on air.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Tools

This section contains information on the following topics:

- VCF Parameter Properties
- To Use the Virtual Camera Flight

VCF Parameter Properties



- **Real Camera %:** Defines the percentage the real camera influences the position of the virtual camera.
- **Real Camera Deformation %:** Defines the percentage the real camera influences the lens distortion.

- **Zoom**: Sets the actual zoom (FOV-Y) value of the virtual camera

To Use the Virtual Camera Flight

1. Add the VCF scene plug-in to the scene setup plug-ins drop zone.
2. Create a new group in your scene and add two new containers under it.
    - These are to be the objects that will define the virtual camera flight.
    - One will define the position, the other the direction.
3. Name the containers according to the names you entered in the VCF scene plug-in (e.g. T_POS and T_ROT).
4. Click **Initialize** to finish.
5. Add the VCF Parameter container plug-in to the container that holds the position object.
6. Animate your virtual camera flight using the two objects to define position, and direction. Do **not** switch the whole container invisible or else the animation will not run. You may switch the objects to be invisible, at any time.
7. Animate the **Real Camera %** and **Real Camera Deformation %** parameters in what frame you want to have the real camera at 0% visibility and when at 100% visibility (the longer the animation is, the smoother the interpolation will be: 100 F should be good).
    - If you start with a virtual camera the first Key Frame (0) should have 0% of the real camera, and the last should be 100% of the real camera.
    - The virtual camera will generate a full key (masking the FG), if you choose virtual studio in the scene setup: background.
8. Make sure your FG object/talent are not seen while switching to the real camera. They should show in the frame after the switch.

> ⚠ **Note:** It is not possible to animate the roll of the camera.

See Also

- VCF scene plug-in
- Create Animations

14.22 Transformation

The following container plug-ins are located in the Transformation folder:

- Justifier
- VertexBone and VertexSkin Plugin

14.22.1 Justifier

The Justifier plug-in enables you to animate the justification of an object using the object's height, width or depth as its coordinate system rather than using the Viz Artist/Engine coordinate system.

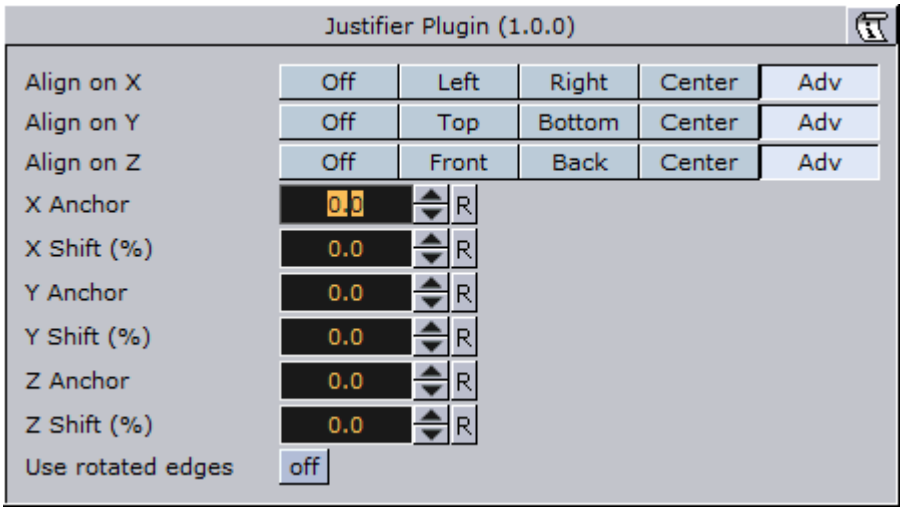
This plug-in acts on the local space of the container it is attached to. So a parent container can be used to rotate the container for use of arbitrary (world space) axes.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Transformation

This section contains information on the following topics:

- Justifier Properties
- To Justify an Object

Justifier Properties



- **Align on X, Y and Z:** Aligns the position. Left, Right and Center enables the Anchor option(s). Adv (advanced) enables the Anchor and Shift option(s).
  - **X, Y and Z Anchor:** Sets the offset for the alignment.
  - **X, Y and Z Shift (%):** Sets the shift of the position values to specify alignment between left/right, top/bottom and front/back (e.g. an Anchor at 50.0, Shift at -10%, gives an Anchor at 45.0.). Range is -100 to 100.
- **Use rotated edges:** If this is set to true and the container has been rotated, a new axis aligned bounding box will be used to computed the alignment. (I.e. a non rotated bounding box completely containing the rotated container).

To Justify an Object

1. Create a container that can only move up and down.
2. Add the Justify plug-in on the container.
3. Set Align on X to Center.
4. Set Align on Z to Center.
    - Now the container can only be moved along its y axis.
5. Use the X Anchor and Z Anchor to move it along the respective axis.

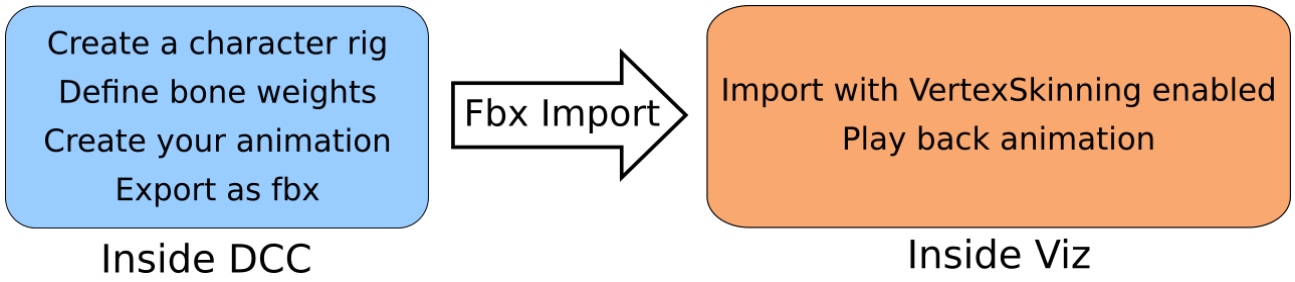
14.22.2 VertexBone and VertexSkin Plugin

Introduction

VertexSkinning describes the technique of using rigs (consisting of auxiliary objects called bones) to animate meshes. Instead of animating each vertex (shape animation or morphing) bones are animated and deform the mesh accordingly. This saves time during animation and memory.

The process of defining which vertices are deformed by a given bone is called weight painting. A vertex can be influenced by more than one bone to guaranty smoother animation around joints. As the mesh stretches around a joint like skin, this functionality is commonly called vertex skinning.

Quick Steps



Step-by-step

Preparation

- Load a character (or any other model) in the Digital Content Creator (DCC) of choice (Maya, Cinema4D, Blender, 3dsMax etc).
- Create a skeleton (armature). Normally, this will be a hierarchy of bones so when the spine rotates the whole upper body rotates with it. This should look something like the skeleton in Illustration 1. These bones will later become containers in the imported Viz scene.
- Apply the weight painting. Most DCC programs can this automatically with pretty goods results for most meshes (e.g. Maya). Other programs offer only a good starting point that needs further refinement. Illustration 1 shows a selected bone (outlined in cyan) and a mesh with weight paints (blue: weight = 0.0, red: weight = 1.0).
- Animate the character using the bones created earlier.

- Export the scene from the DCC. Remember to check that bones and animations will be exported.
- Switch to VizArtist.

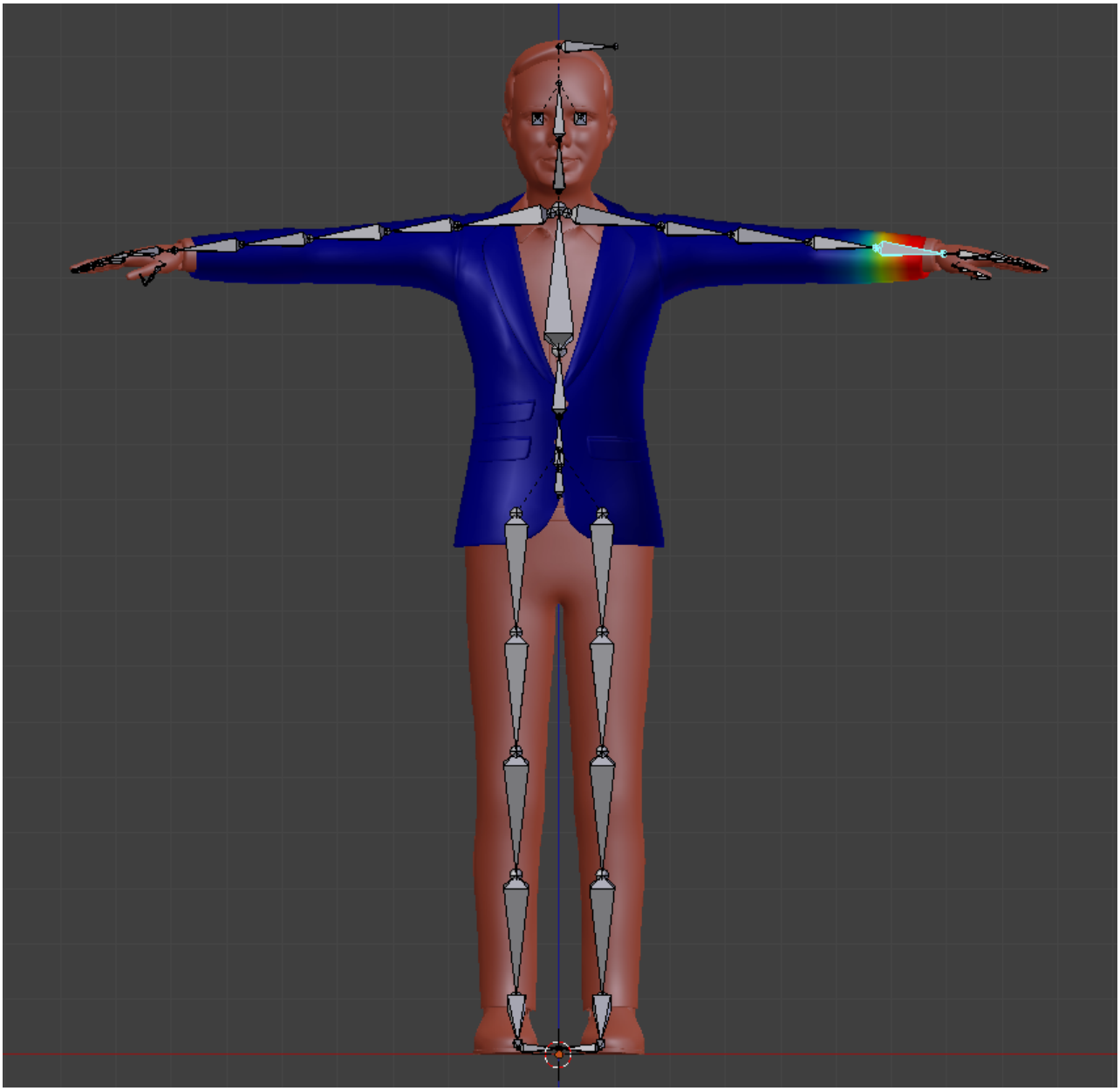


Illustration 1: Weight Painting

VizArtist

- Import the .fbx file of your scene in Viz Artist.
- Enable the import skinned animations option (as shown in Illustration 2).
- Play the scene using the default director after the scene is loaded.

> ⚠ **Important:** Do not change the hierarchy of the containers with VertexBone plugins! Containers can be added, but the parent-child relationship of bones should not be altered.

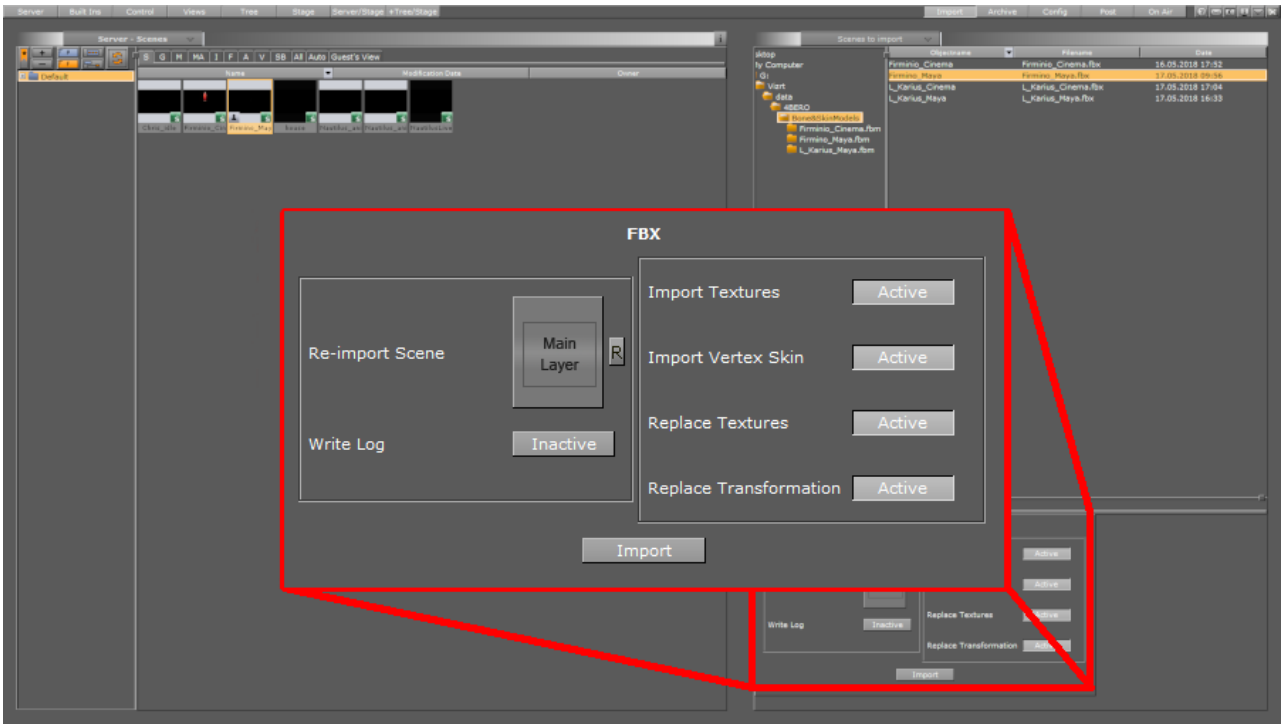


Illustration 2: Viz import screen

Capabilities of VizArtist VertexSkinning

VizArtist can:

- Import skinned meshes via .fbx files.
- Playback animations.

As the skeleton is represented via Containers marked by a VertexBone plugin new animations could be created in VizEngine but for now there are is no functionality to properly support this (like inverse kinematic or movement constraints on bones).

VizArtist cannot:

- Perform inverse kinematics.
- Perform rigging or skinning.
- Handle multiple animations (currently only the first animation, called the Default Take in .fbx terms).
- Allow VertexSkin and VertexBone plugins to be placed by hand. There are hidden parameters imported from the .fbx file that cannot be edited or generated inside VizArtist.
- Provide compatability with external shader packages (e.g. RTT shaders) with Vertex Skinning in Viz.

- Support cubemaps in VertexSkinning.

Role of plugins

VertexSkin

VertexSkin is the main plugin that marks a mesh that uses Vertex Skinning.

> ⚠ **Note:** Do not remove this plugin as this would need a clean, new import (not re-import) to get the scene running again.

If you want some additional skinning debug information, you can activate:

- **Show bones:** Verifies the location of each bone visually.
- **Show the weight of a bone:** Visualizes the influence of the selected bone container on the target mesh. Only useable with no other shader present.

Motion Capture Live Tracking

To apply live motion capturing data to a Viz geometry, the following steps are required:

- Setup Viz Tracking Hub for Motion Analysis Motion Capturing (See the **Motion Analysis Integration** page in the **Appendices** section of the Tracking Hub User Guide).

  > ⚠ Requires Tracking Hub 1.1.2. or newer

- Load motion capture data in Motion Analysis Cortex control application.
- Run Viz Artist with activated Tracking Hub usage.
- Load the imported mesh and navigate to the first VertexSkin plugin instance in your scene.
- Activate *Shared Mem Tracking* with the key name set in Cortex. You should see all available tracking bones in the VertexSkin plugin's textbox below. This indicates that the information flows correctly from Cortex to Tracking Hub to the Viz Engine.
- Due to accuracy issues of the motion capture system, it could make sense to activate *Position Root Bone only*. This translates only the root bone and all related child bones are rotated.

VertexBone

VertexBone is used to tell Viz which containers act as bones.

> ⚠ **Note:** Do not remove this plugin as this would need a clean, new import (not re-import) to get the scene running again.

Common Pitfalls

- Watch out that containers hosting the meshes and the root container of the bone containers (which should be on the same level) have the same scaling (ideally 1.0/1.0/1.0).

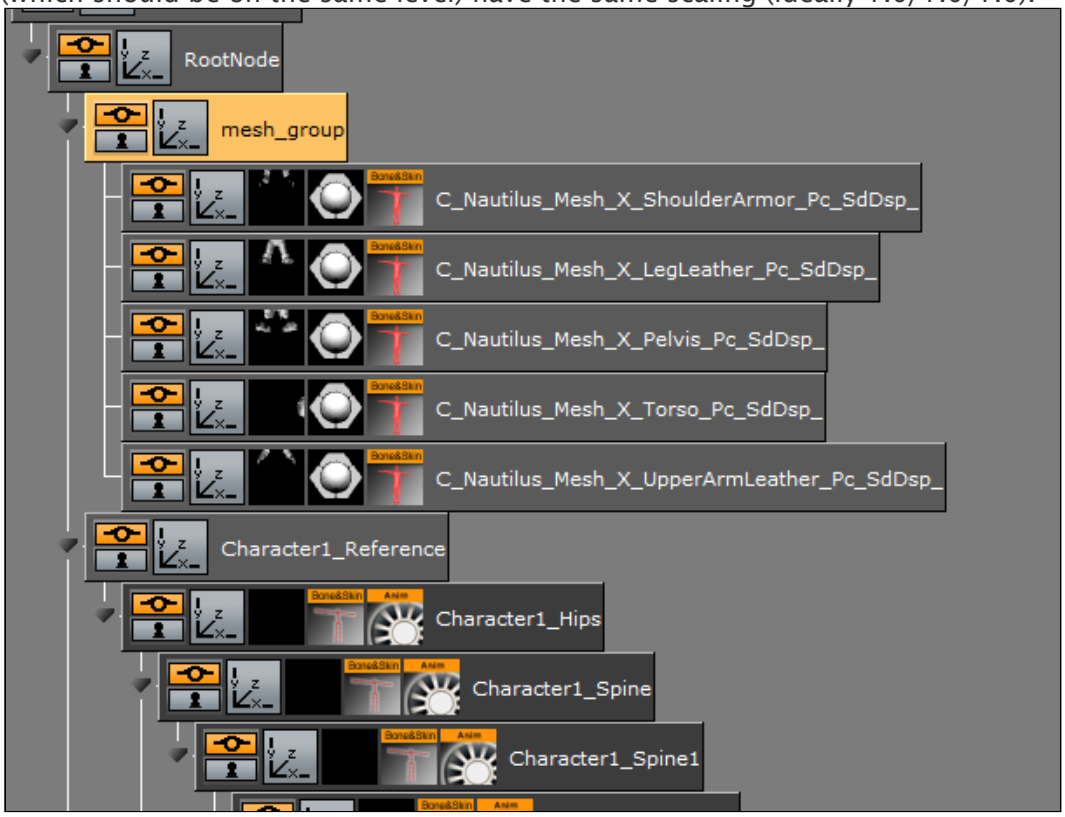


Illustration 3: Skeleton and meshes

- Transformations on the mesh parent container, and the mesh container are ignored. You can disable this via the VertexSkin plugin (uncheck Ignore transformation on mesh container) although this is **not advised**. Transformation of the skinned mesh should only be done via the skeleton.
- mesh_group is no longer the parent of the mesh containers (in Illustration 3) but Character1_Reference is. Therefore, mesh_group should be devoid of all transformations if VertexSkinning is to be used.
- You cannot add the vertex skinning plugins to containers and generate a skinned mesh inside Viz. The plugins hold private data, that is extracted from the .fbx file during import.
- To use the show weight of a bone, no shader plugin can be present on the mesh (deactivating the plugin is sufficient).
- VertexSkinning will only work properly if VBOs are used for drawing.

External References

Here you can find a small list of introduction videos for character rigging within multiple DCCs. As rigging is not a precise science with one correct way to do everything, it might be useful to watch as many videos on the topic (even the ones using software you might not be familiar with).

Maya:	Quick Rigging and Skinning a character in Maya Rigging and Animating a DOTA 2 Courier Playlist
Cinema4D:	Character Animation In Cinema 4D
3Ds MAX:	Rigging a Character in 3ds Max Playlist
Blender:	Character Rigging - Blender Fundamentals

Glossary

- **Vertex Skinning:** The process of using an armature to deform a mesh instead of using key frames for each vertex (Morphing).
- **Rigging:** The act of defining a skeleton that will be used to deform a mesh. This might include setting constraints on the rotation of bones or defining IK.
- **Bone:** A node (container in Viz) that is part of a kinematic chain used for mesh deformation.
- **Weight Painting:** The process of defining the amount by which each vertex will be affected by a given bone. This can be done in a DCC via directly painting on a mesh.
- **DCC:** Digital Content Creation (program). Software used to create digital media like 3D models, textures or the like.
- **Armature:** A set of kinematic chains of nodes used to simulate a skeleton for mesh deformation.
- **Bind Pose:** The deformation a model has when it is bound to its skeleton.
- **IK:** Inverse kinematics are used to compute how a set of bones should be rotated so that the leaf ends up at a certain position. IKs are defined in chains (e.g. from foot to hip).

14.23 Visual Data Tools

The following container plug-ins are located in the Visual Data Tools folder:

- Area Stack
- Bar Stack
- Data Fit
- Data Import
- Data Label
- Data Storage
- Line Stack

See Also

- Visual Data Tools Geometry plug-ins, which offer an overview of the Visual Data Tools.
- Area Chart
- Bar Chart
- Line Chart
- Pie Chart
- Scatter Chart
- Stock Chart

- Tutorial on www.vizrt.com under the Training section.

14.23.1 Area Stack

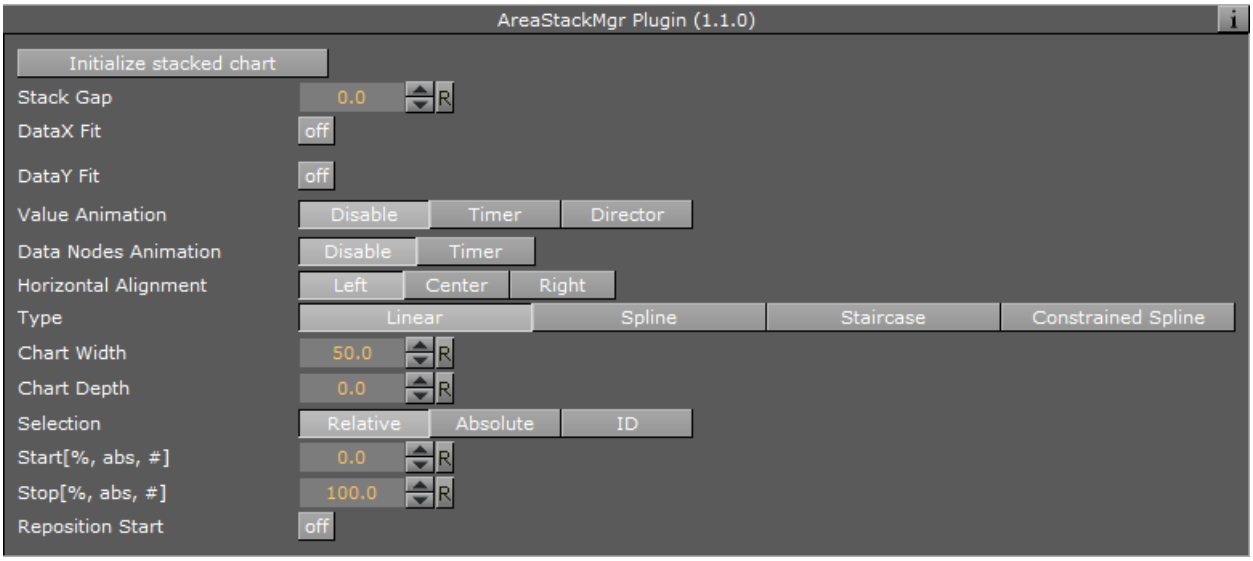
Create a stacked chart consisting of several Area Charts.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> VisualDataTools*

This section contains information on the following topics:

- Area Stack Properties
- To Create a Scene with Area Stack

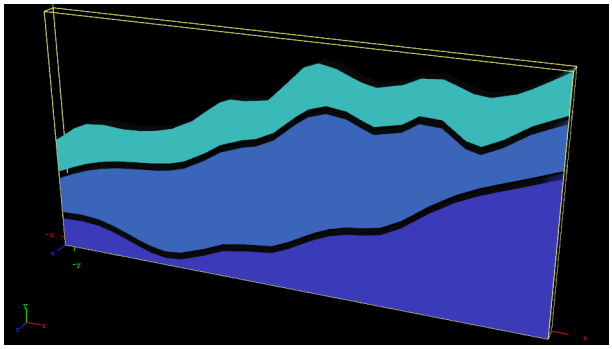
Area Stack Properties



- **Initialize stacked chart:** Refreshes the stacked chart, based on the Area Chart plug-ins in the sub-containers.
  The stacked chart detects when a chart is added/removed from the stack, and will update the chart accordingly. However, some actions, such as re-ordering the charts within a stack, are not updated automatically. In these cases, press *Initialize stacked chart* to update the stack.
- **Stack Gap:** Add space between each chart

⚠ **Note:** The remaining properties in the Line Stack plug-in are the same as used in the Area Chart plug-in. The Area Chart features Specify X Values, DataY Compare, and Bevel, are disabled in stacked charts.

To Create a Scene with Area Stack



1. Create a new container.
2. Add an Area Stack plug-in into this container.
3. Add two or more Area Chart plug-ins to become children of this container.
4. Add Data Y into each chart. They should have the same number of nodes.
5. In the settings of the Area Stack plug-in, press *Initialize Stacked Chart* to refresh the chart.
6. Use the Area Stack plug-in to set the chart parameters that are common for all charts in the stack.

14.23.2 Bar Stack

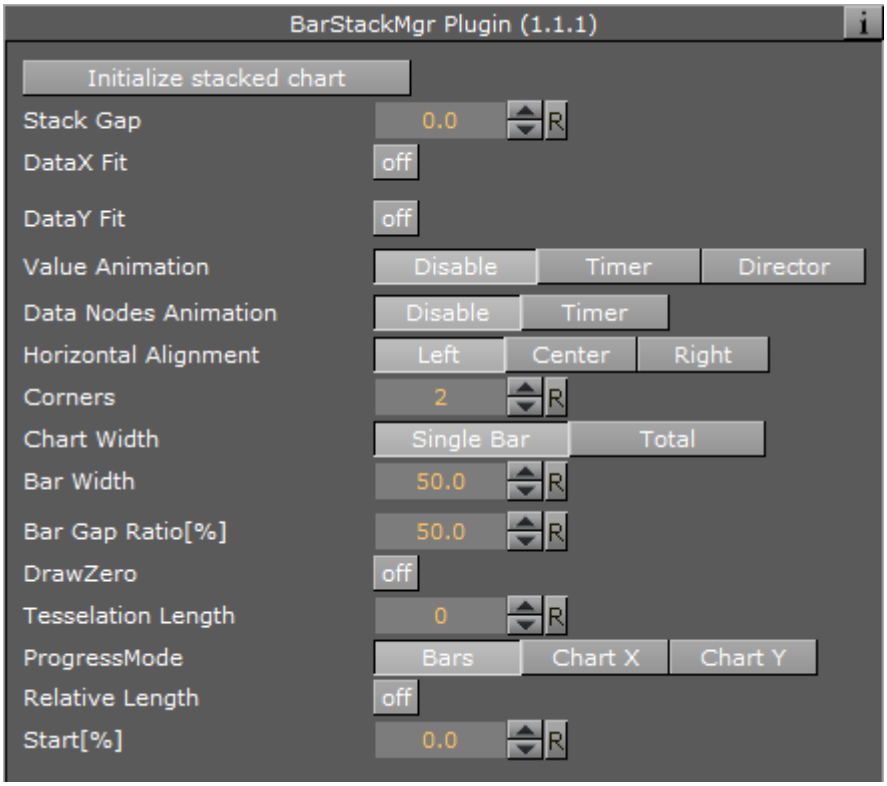
Create a stacked chart consisting of several Bar Charts.

⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> VisualDataTools

This section contains information on the following topics:

- Bar Stack Properties
- To Create a Scene with Bar Stack

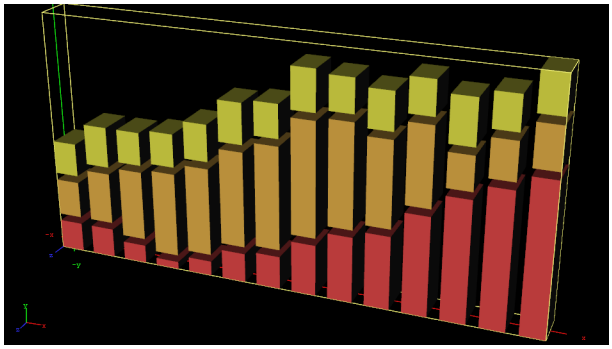
Bar Stack Properties



- **Initialize stacked chart:** Refreshes the stacked chart, based on the Bar Chart plug-ins in the sub-containers.
  The stacked chart detects when a chart is added/removed from the stack, and will update the chart accordingly. However, some actions, such as re-ordering the charts within a stack, are not updated automatically. In these cases, press *Initialize stacked chart* to update the stack.
- **Stack Gap:** Add space between each chart

⚠ **Note:** The remaining properties in the Line Stack plug-in are the same as used in the Bar Chart plug-in. The Bar Chart features Specify X Values, DataY Compare, and Bevel, are disabled in stacked charts.

To Create a Scene with Bar Stack



1. Create a new container.
2. Add a Bar Stack plug-in into this container.
3. Add two or more Bar Chart plug-ins to become children of this container.
4. Add Data Y into each chart. They should have the same number of nodes.
5. In the settings of the Bar Stack plug-in, press *Initialize Stacked Chart* to refresh the chart.
6. Use the Bar Stack plug-in to set the chart parameters that are common for all charts in the stack.

14.23.3 Data Fit

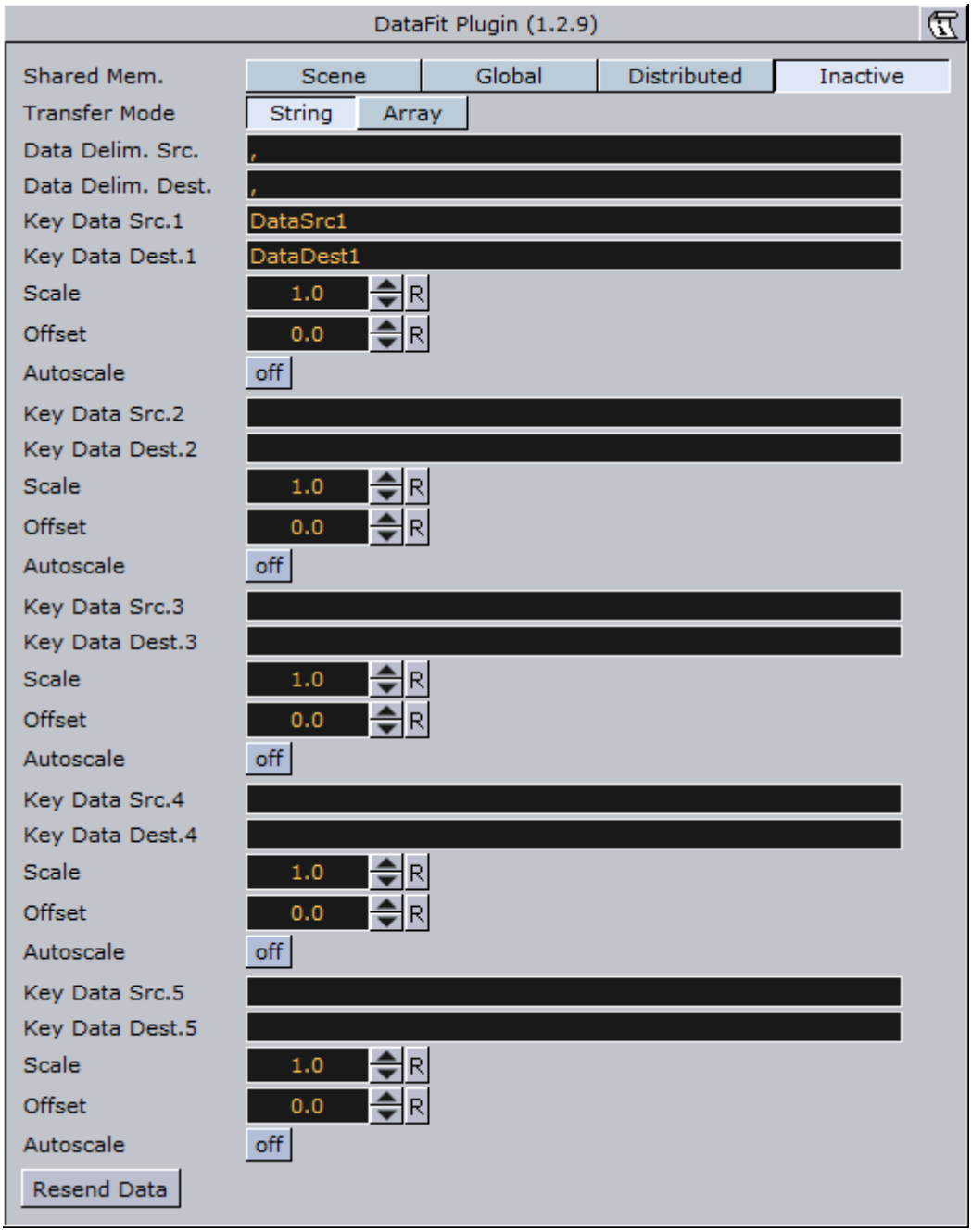
Data Fit listens for incoming data and modifies and redistributes it to another Shared Memory key.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> VisualDataTools

This section contains information on the following topics:

- Data Fit Properties
- To Create a Scene with Data Fit

Data Fit Properties



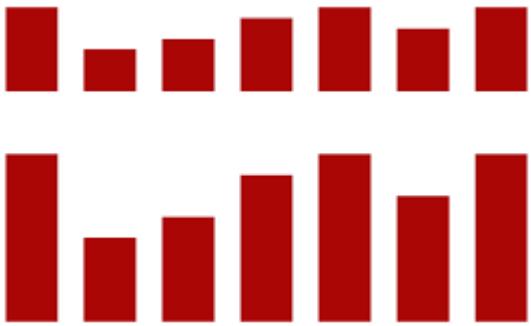
- **Shared Mem:** Changes between Scene-, Global- and Distributed-Shared Memory. Use Inactive memory to not forward any values via Shared Memory.
- **Transfer Mode:** Sets string- or array-based data transfer.
- **Data Delim. Src.:** Defines the value separator sign(s) for all data sources.
- **Data Delim. Dest.:** Defines the value separator sign(s) for all data destinations.
- **Key Data Src.1-5:** Shared Memory key name for input 1-5.
- **Key Data Dest.1-5:** Shared Memory key name for output 1-5.

- **Scale:** Scales each incoming value by that factor.
- **Offset:** Adds an Offset to each incoming value.
- **Autoscale:** Activates automatic scaling mode. Now all data is scaled to a certain range defined by the following 2 parameters:
  - **Start:** Determines autoscale starting range.
  - **Stop:** Determines autoscale stopping range.

To Create a Scene with Data Fit

This procedure will feed two charts with data. One with the original and the other with the modified data from Data Fit.

1. Create a new container.
2. Drag and drop a Data Storage and a Data Fit plug-in into this container.
3. Drag and drop two Bar Chart plug-ins to become children of this container.
4. Open the plug-in editor of the first Bar Chart and set Shared Mem. to Scene and KeyData to “MyData”.
5. Do the same for the second Bar Chart but use “MyDataFit” as the key name.
6. Go to the DataFit plug-in and set its Shared Mem. type also to Scene
7. Set Key Data Src.1 to “MyData” and Key Data Dest.1 to “MyDataFit” > set Scale to 2.0.
   - The input Shared Mem key is specified with Key Data Src. and the output with Key Data Dest.
8. Now edit Data Storage for the input values: set Shared Mem. to Scene, Key Data1 to MyData and type in some example values for Data1: 80,40,50,70,80,60,80.
   - As soon as you start typing you see both charts building up with the different values from Data Fit.



14.23.4 Data Import

This plug-in enables Microsoft Excel (.xls and .xlsx) and Microsoft Access (.mdb) file import via ADO, and distributes it to a text field or a shared memory map. The plug-in supports reading both numbers and text.

To access Excel or Access data using this plug-in, you must first install the Microsoft Access Database Engine 2010 Redistributable.

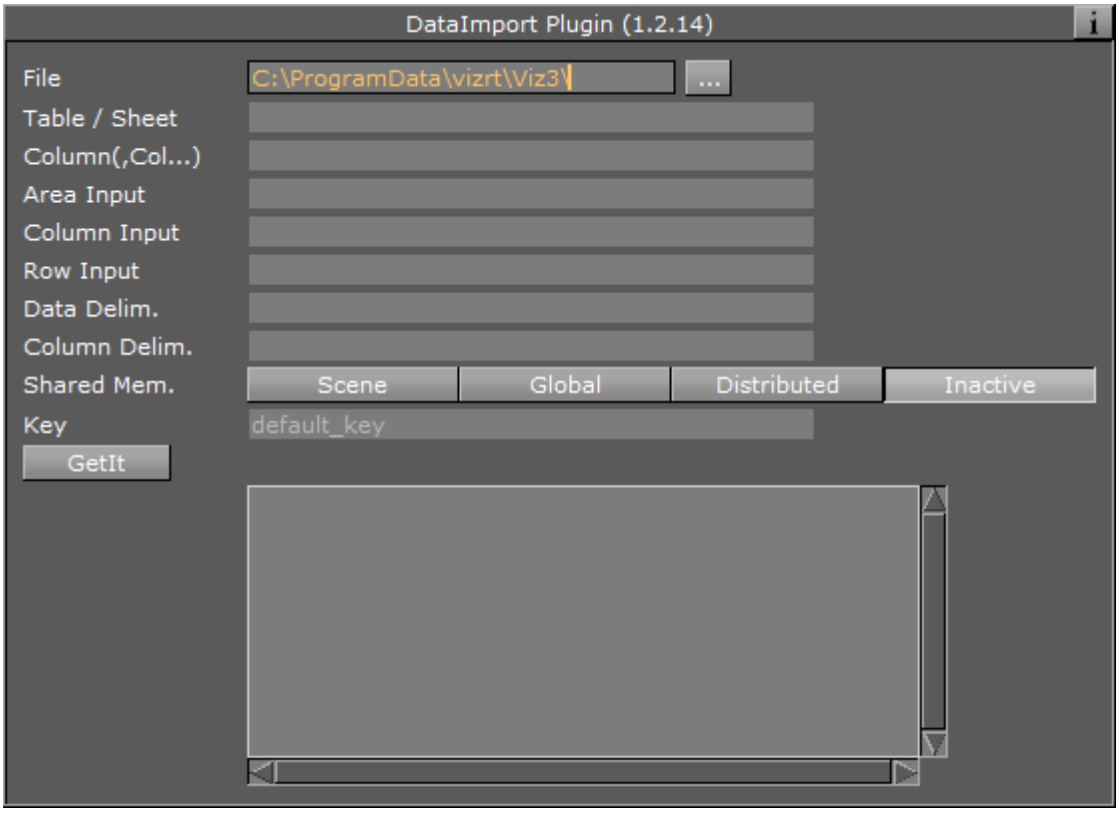
> ⚠ **IMPORTANT! Viz Artist 64-Bit**: If you install Microsoft Office 32-bit on a computer running Windows 64-bit that has Viz Artist/Engine 64-bit installed, remove Microsoft Office completely before installing Microsoft Access Database Engine 2010 Redistributable. If you require Microsoft Office, make sure to install the 64-bit version.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> VisualDataTools*

This section contains information on the following topics:

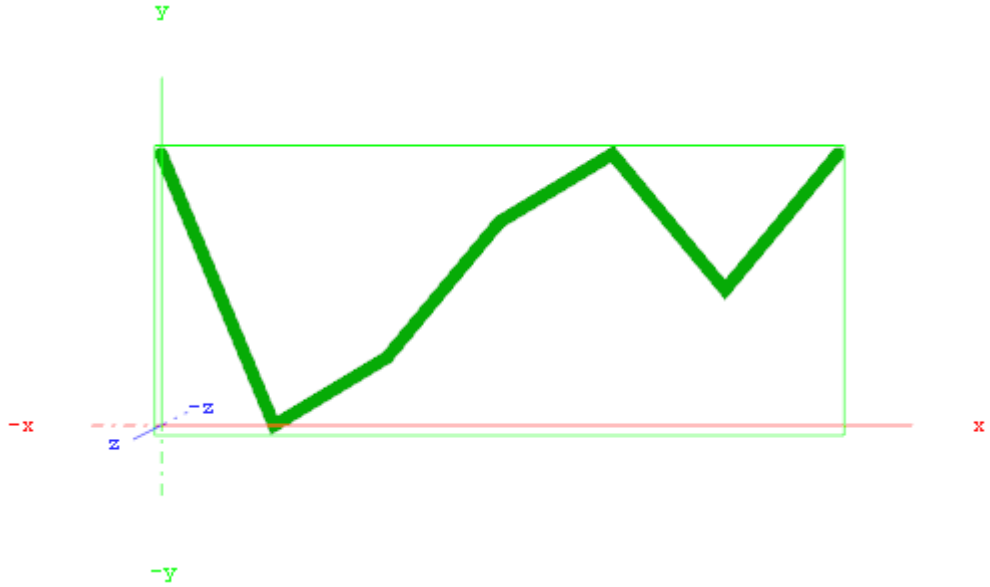
- Data Import Properties
- To Create a Scene with Data Import

Data Import Properties



- **File:** Chooses the required document.
- **Table / Sheet:** Defines the name of the sheet or table.
- **Column(,Col...):** Sets one or more column names to use (comma separated).
- **Area Input:** Uses area information.
- **Column Input:** Uses column information.
- **Row Input:** Uses row information.
- **Data Delim.:** Includes delimiter after each row.
- **Column Delim.**: Includes delimiter for column separator sign(s).
- **Shared Mem.:** Sets Scene-, Global- or Distributed-Shared Memory for data transport. Use Inactive memory to not forward any values via Shared Memory.
- **Key:** Determines Shared Memory key name.
- **GetIt:** Reads the document, sends the required data and shows it in the text box.

To Create a Scene with Data Import



This procedure will import data from an Microsoft Excel sheet and feed the Line Chart plug-in with data through Shared Memory.

1. Start Microsoft Excel.
2. Enter **ExcelDataY** into cell A1.
3. Add some sample values in the cells below (A2-A8: `80,40,50,70,80,60,80`).
4. Rename this first sheet to **MyTable** (can be done with a double click on the sheet name at the bottom).
5. **Save and close** the Excel document.

   > **IMPORTANT!** You must close the document before importing it to Viz Artist.

6. Start Viz Artist and **create a new scene.**

7. Add a group container to the scene tree.
8. Drag and drop the Line Chart and **Data Import** plug-in to the group container.
9. Add a material to the same group container.
10. Open the Transformation Editor and set **Position X** to `-200.0` and **Position Y** to `-100.0`.
11. Set **Shared Mem.** to **Scene** with **MyDataY** as Key name for both plug-ins.
12. Set **Data Delim.** to **#** for both plug-ins.
13. For the *Line Chart* plug-in set **ChartWidth** to `500.0`.
14. Activate **DataY Fit**, **DataY Auto Scale** and **DataY Detect Limits.**
15. Adjust **DataY Stop** to `200.0`.
16. Activate **Const. Thickness** to give the chart a constant line width.
17. For the Data Import plug-in, set the stored Excel file as the **File** parameter.
18. Enter **MyTable** for the Table/Sheet parameter and **ExcelDataY** for Column (,Col...).
19. Click the **GetIt** button and the chart shows.

14.23.5 Data Label

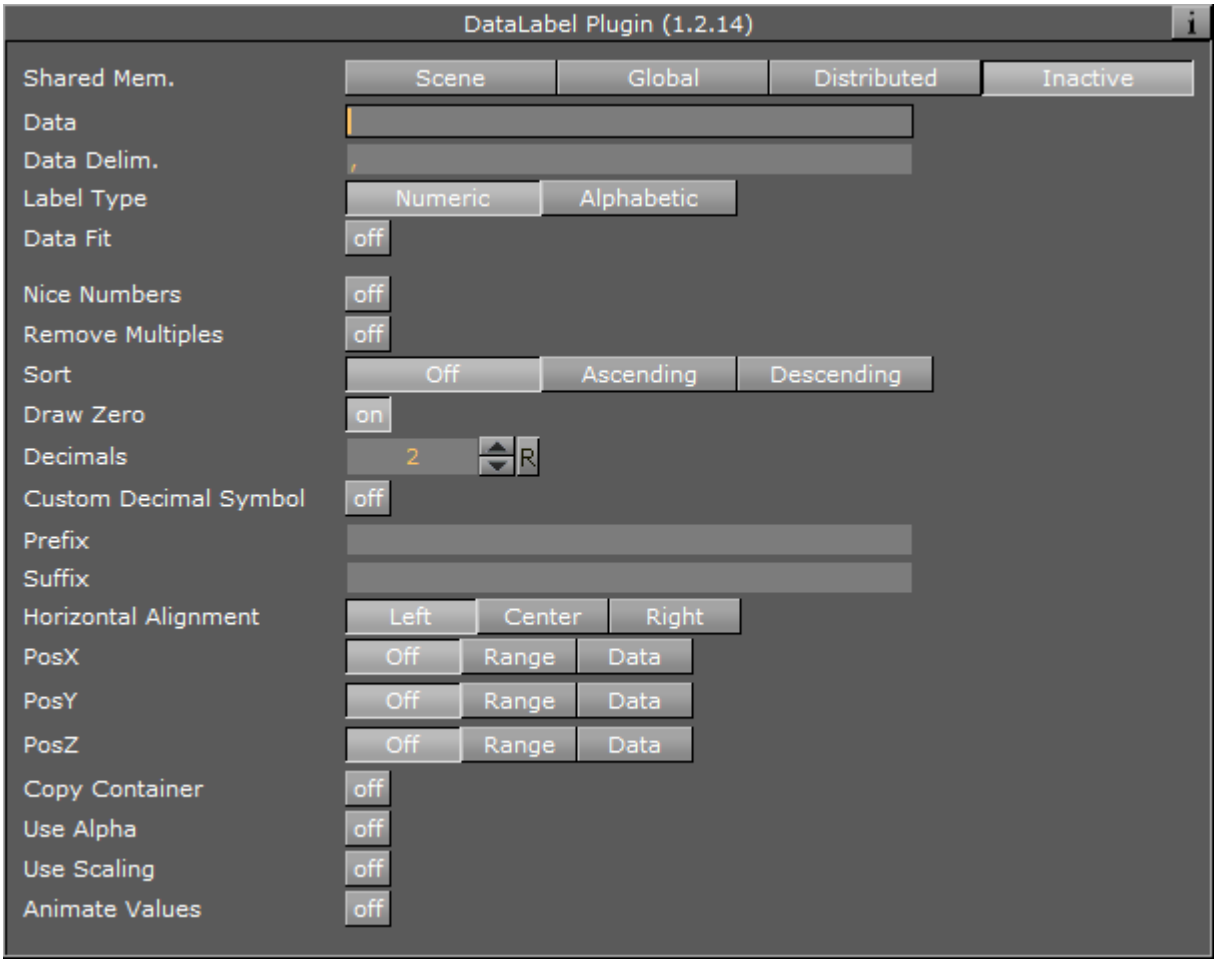
This plug-in enables graph labeling for all Financial plug-ins. All containers beneath the current container get labeled with the values from your Shared Memory variables.

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> VisualDataTools*

This section contains information on the following topics:

- Data Label Properties
- To Create a Scene with Data Label

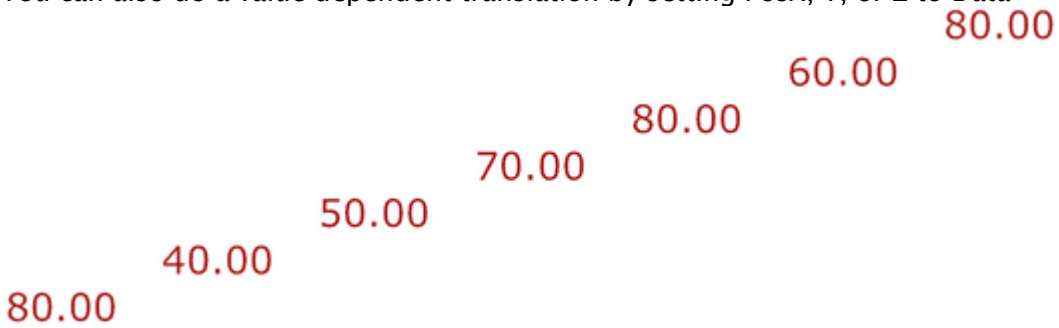
Data Label Properties



- **Shared Mem.:** Changes between **Scene**, **Global** and **Distributed** Shared Memory. Use **Inactive** memory to not forward any values via Shared Memory.
- **Key Data:** Determines Shared Memory key name.
- **Data:** Input Parameter for data.
- **Transfer Mode:** Sets string- or array-based data transfer.
- **Data Delim.**: Defines the value separator sign(s).
- **Label Type:** Sets the label type to numeric values (default) or alphabetic labels (to show weekdays, months, etc.).
- **Data Fit:** Enables data normalization.
    - **Data Threshold**: Adds a definable offset to the detected limit.
- **Data Scale:** Scales input by the selected factor.
- **Data Offset:** Adds an offset to the incoming data.
- **Data Auto Scale:** Enables automatic data normalization.
- **Data Detect Limits:** Detects minimum and maximum of all values and scales them to adjusted Start and Stop.
- **Data Start:** Lower Auto Scale edge.
- **Data Stop:** Upper Auto Scale edge.
- **Nice Numbers:** Rounds to the best fitting next number.
- **Remove Multiples:** Removes multiple values.
- **Sort:** Sorts data items in ascending or descending order.
- **Draw Zero:** Shows or hides zero values.
- **Decimals:** Adjusts decimal digits of all labels.
- **Custom Decimal Symbol:** Enables a custom decimal symbol to be set.
    - **Decimal Symbol:** Specifies the decimal symbol.
- **Prefix**: Adds a label prefix.
- **Unit:** Defines a custom unit for your data values.
- **PosX, Y, Z**: Positions container along the adjusted distance or uses numeric data value input to translate the label containers.
- **OffsetX, Y, Z:** Translates the label container on the current axis.
- **DistanceX, Y, Z:** Sets whole positioning distance for the containers on the current axis.
- **Copy Container:** Adds new containers if there are fewer containers than values.
- **Use Alpha**: Uses input data for alpha scaling.
- **Use Scaling:** Uses input data for alpha scaling.
- **Animate Values:** Lets the current number run from 0 to the defined end value.
- **Relative Length:** If activated, each label will have its own 100% (e.g. 7 labels equal 700%).
- **Const. Speed:** Sets the same animation duration for each label.
- **Total Length%:** Sets the accumulated value of all labels in percent.
- **Alpha max.:** Limits maximum alpha value.
- **Copy Container:** Limits maximum scaling value.

To Create a Scene with Data Label

1. Add an empty container to the scene.
2. Start with an empty scene and drag and drop a DataLabel and a DataStorage plug-in on a container in the scene-tree.
3. Add a Text child container
4. Go to the DataLabel plug-in, set Shared Mem. to Scene and KeyData to “MyData”.
5. In DataStorage also set Shared Mem. to Scene, Key Data1 to “MyData” and Data1 to e.g.: 80,40,50,70,80,60,80
6. Go back to the DataLabel plug-in and activate Copy Container to create multiple instances of the text containers automatically.
   - You see all labels located on the same position now. In the previous examples always the chart plug-in was responsible for the label positioning. In this example we want to move that responsibility to the DataLabel plug-in itself.
7. Switch PosX to Range, DistanceX to 500.0 and OffsetX to -250.0.
   - We want to position the labels over a certain X, Y or Z range in space.
8. Set PosY to Range, OffsetY to -75.0 and DistanceY to 150.0
9. You can also do a value dependent translation by setting PosX, Y, or Z to Data



10. Set Sort to Ascending to sort the labels.

- You see that there are labels with the same values. We want to get rid of them and activate Remove Multiples for that. You can also try the Nice Numbers option now which searches for the next logarithmic style labels. But we continue with deactivated Nice Numbers parameter now.

11. Activate Animate Values and turn down Total Length[%] to 0.0. Create a Key Frame, set Total Length[%] back to 100.0 and create another Key Frame.
    - This will create an animation where the label values should count up to their final value.
    - Start the animation and see the labels counting and distributing over the specified XY range.

14.23.6 Data Storage

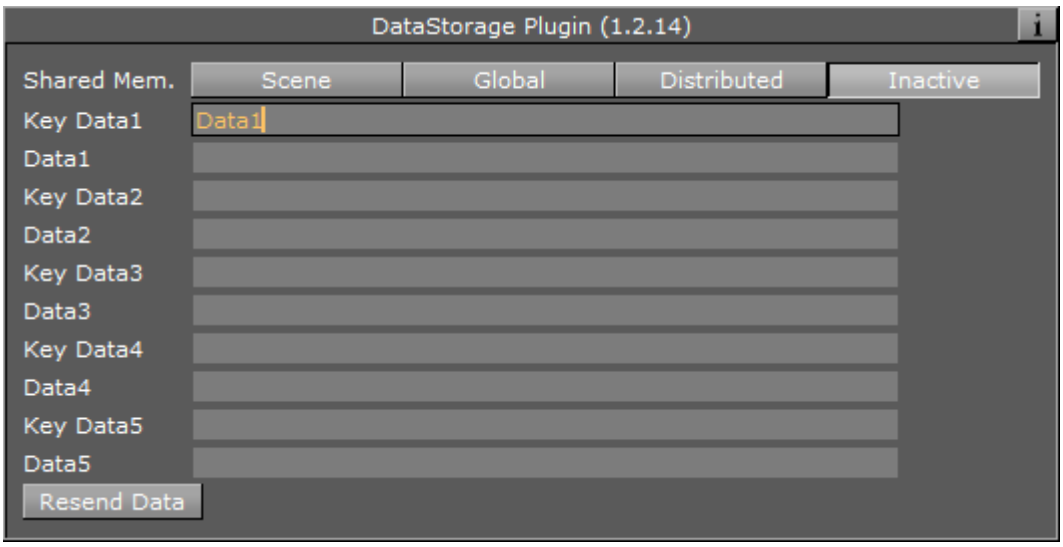
The Data Storage plug-in stores data in the current Scene and sends it to the other plug-ins on startup. This plug-in helps artists during Scene design if there is no data source available, and to hold data within, for example, the Scene (see).

> ⚠ **Note:** This plug-in is located in: *Built Ins -> Container plug-ins -> VisualDataTools*

This section contains information on the following topics:

- Data Storage Properties
- To Store and Send Data
- To Enable Data Input from External Sources

Data Storage Properties



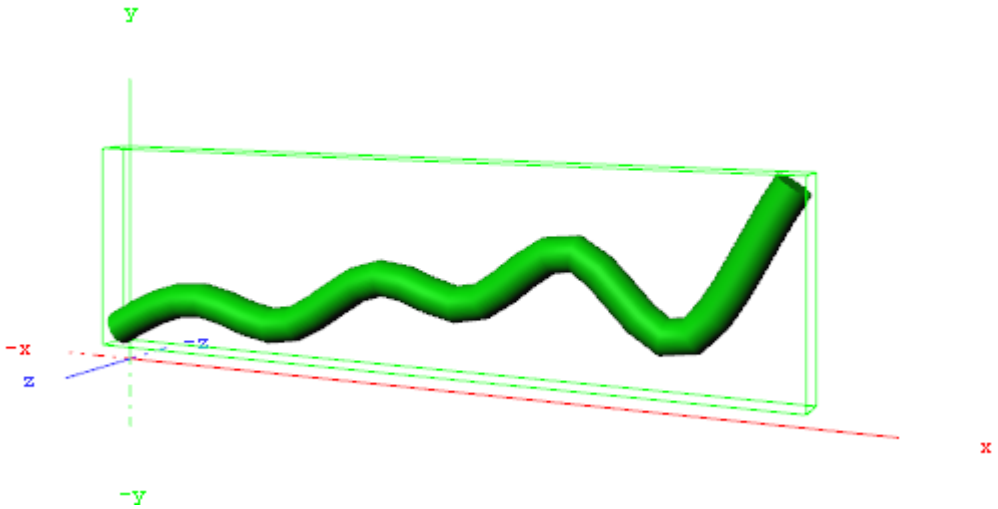
- **Shared Mem.:** Changes between **Scene**, **Global** and **Distributed** Shared Memory. Use **Inactive** memory to not forward any values via Shared Memory.
- **Key Data 1-5**: Determines Shared Memory key name.
- **Data 1-5:** Contains (separated) data.
- **Resend Data:** Resends the actual values.

To Store and Send Data

1. Create a chart scene, e.g. using the Line Chart plug-in and set **Shared Mem.** to **Scene**
2. Add the Data Storage plug-in to the scene and set **Shared Mem.** to **Scene**
3. Open the DataStorage editor and set the following parameters:
   - **Shared Mem.** to **Scene**
   - **Key Data1** to `Data1`
   - **Data1** to `10, 20, 15, 30, 25, 40, 20`

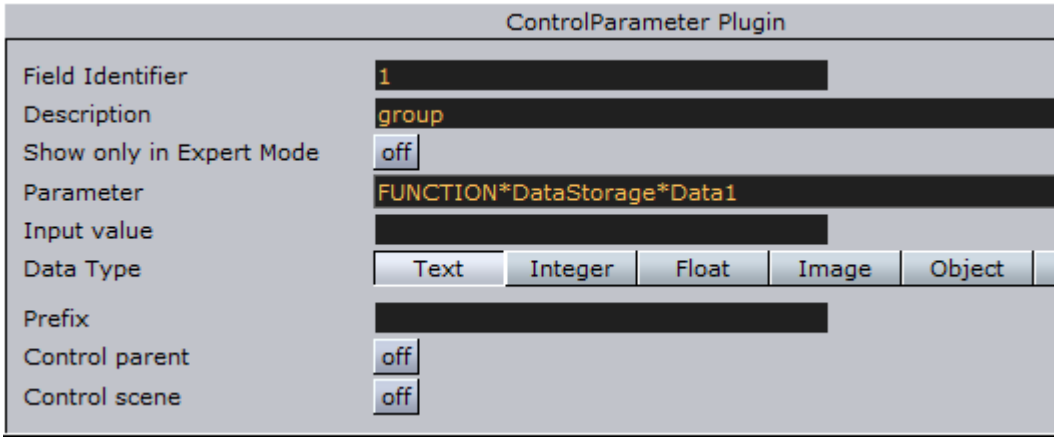
4. Open the Chart plug-in editor and set the **Key DataY** value (e.g. `Data1`) using the Key Data1 value you set for the DataStorage plug-in

> ⚠ **Note:** The chart plug-ins have different property names (e.g. Data and Key Data). Also, some have X, Y or Z appended at the end, defining the axis.



5. Open the DataStorage editor and click the **Resend Data** button.

To Enable Data Input from External Sources



The following example builds on the previous example.

1. Add the Control Parameter to the scene tree
   - The Control Object plug-in is automatically added
2. Open the Control Parameter editor
3. Enter `FUNCTION*DataStorage*Data1` into the **Parameter** field
4. Set **Data Type** to **Text**
   - This will allow you to pass a string of numbers to the Control Parameter plug-in that will be stored in the shared memory (see).
   - It will also allow your template designer or operator to further extend/enhance the input methods through e.g. scripting.

14.23.7 Line Stack

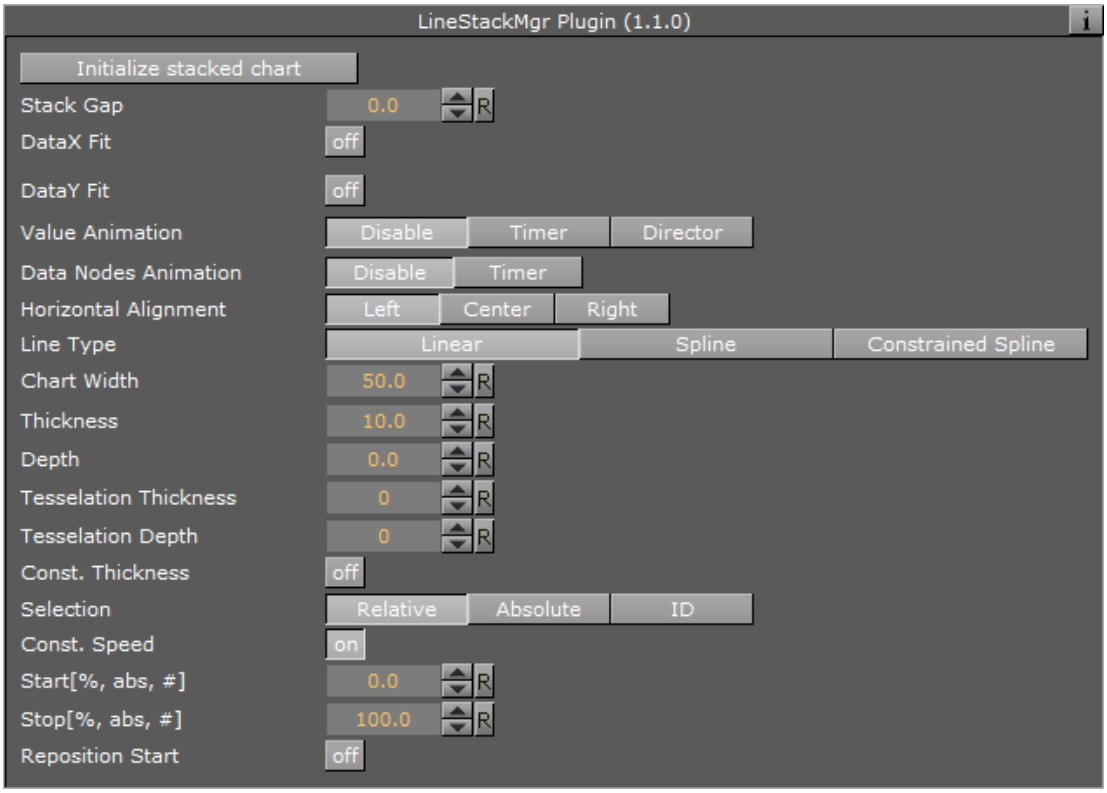
Create a stacked chart consisting of several Line Charts.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> VisualDataTools

This section contains information on the following topics:

- Line Stack Properties
- To Create a Scene with Line Stack

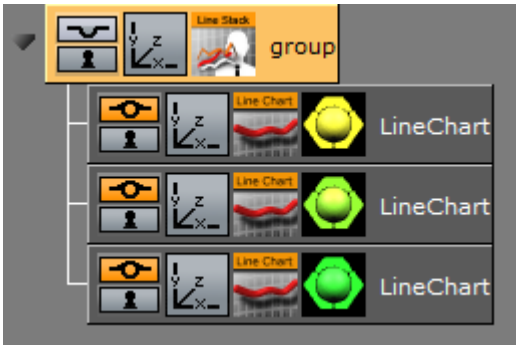
Line Stack Properties

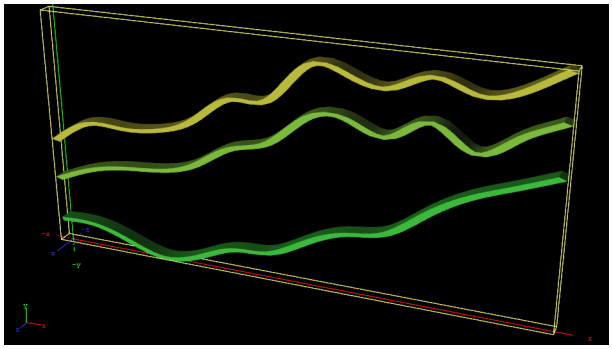


- **Initialize stacked chart:** Refreshes the stacked chart, based on the Line Chart plug-ins in the sub-containers.
  The stacked chart detects when a chart is added/removed from the stack, and will update the chart accordingly. However, some actions, such as re-ordering the charts within a stack, are not updated automatically. In these cases, press *Initialize stacked chart* to update the stack.
- **Stack Gap:** Add space between each chart

⚠ **Note:** The remaining properties in the Line Stack plug-in are the same as used in the Line Chart plug-in. The Line Chart features Specify X Values, DataY Compare, and Bevel, are disabled in stacked charts.

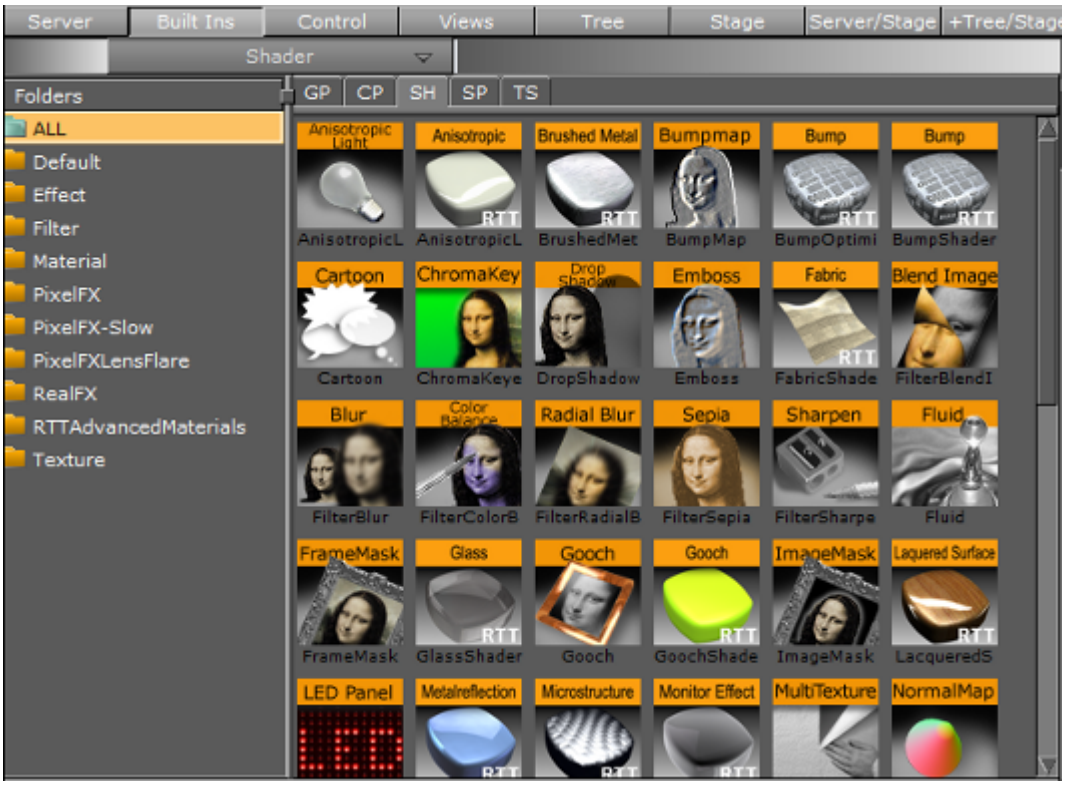
To Create a Scene with Line Stack





1. Create a new container.
2. Add a Line Stack plug-in into this container.
3. Add two or more Line Chart plug-ins to become children of this container.
4. Add Data Y into each chart. They should have the same number of nodes.
5. In the settings of the Line Stack plug-in, press *Initialize Stacked Chart* to refresh the chart.
6. Use the Line Stack plug-in to set the chart parameters that are common for all charts in the stack.

15 Shader Plug-Ins



The default path for the Shader plug-ins is: `<viz install folder>\plug-in`.

> ⚠ **Tip:** Most shaders also work on video clips.

15.1 Default

This folder is empty and reserved for custom Shader plug-ins.

15.2 Effects

The following Shader plug-ins are located in the Effects folder:

- Chroma Keyer
- Fluid
- Frame Mask
- Image Mask
- Soft Mask
- Water Shader

15.2.1 Chroma Keyer

This Shader clear regions in a video defined by a color. The color-space of this Shader is YUV, which is defined as following:

The luminance (Y) component determines the brightness of the color, while the U and V components determine the color itself (the chroma). Luminance (Y) ranges from 0.0 (dark) to 1.0 (bright). U and V ranges from -0.5 to 0.5.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Effects

This section contains information on the following topics:

- Chroma Keyer Properties

Chroma Keyer Properties

- **Video:** Sets the input video channel for keying. Be sure that for the set video channel the "Keying Mode" has been set to M-Zone (see Keying Mode).
- **Blend:** Sets the alpha for the key signal.
- **Fit:** Alters size to fit Screen or Object.
- **Invert:** Flips the key signal.
- **Sampling Filter:**
  - **Nearest:** Color information is taken from one pixel.
  - **1D-Filter:** Color information is taken after applying a 1D-linear filter.
  - **2D-Filter:** Color information is taken after applying a 2D-linear filter.
- **Luminance:** Sets the luminance.
  Basic parameters for first method:

- **HueAngle:** Selects the color to key in YUV space.
- **OpeningAngle:** Tolerance for the color in YUV space.
- **KeyGain:** Gradient for borders to neighboring colors (1=soft, 10=sharp).
- **SatMin:** Lower saturation limit for keyed colors.
- **SatMinGrad:** Gradient for lower saturation limit (0=sharp, 1=soft).
- **SatMax:** Upper saturation limit for keyed colors.
- **SatMaxGrad:** Gradient for upper saturation limit (0=sharp, 1=soft).
  Basic parameters for second method:

- **U-Color:** U-value for color keying.
- **V-Color:** V-value for color keying.

- **UV-Diameter:** All colors between the reference UV and this diameter will be removed.
- **UVGradient:** Gradient for fade out at diameter border (0=sharp, 1=soft).
  Removing keying from Highlights and Shadows:

- **LumMin:** Sets lower luminance limit for keyed colors.
- **LumMinGrad:** Sets gradient for lower luminance limit (0=sharp, 1=soft).
- **LumMax**: Sets upper luminance limit for keyed colors.
- **LumMaxGrad:** Sets gradient for upper luminance limit (0=sharp, 1=soft).
  Sampling reference point:

- **UVRangeCheck:** This option can be used to get an information of the sampled color.
- **URangeMin:** Defines the minimum UColor that should be accepted.
- **URangeMax**: Defines the minimum UColor that should be accepted.
- **VRangeMin:** Defines the minimum UColor that should be accepted.
- **VRangeMax:** Defines the minimum UColor that should be accepted.
- **SampleTrustiness:** Percentage of samples in the limited UV-Color space.

- **Pos. X:** X-coordinate for sampling.
- **Pos. Y:** Y-coordinate for sampling.

> ⚠ **Note:** The origin of the coordinate system is in the upper left corner.

- **Width:** Sets number of pixels in x-direction.
- **Height:** Sets number of pixels in y-direction.
- **Sample at Pos:** If enabled Pos. X and Pos. Y will be replaced by XY-coordinates of the object center. If disabled the XY-coordinates will be set by values in the interface.
- **Get Sample:** Sample another time the reference point.

See Also

- Keying Mode

15.2.2 Fluid

With Fluid you can get a geometry mesh to be elastic during animation.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Effects

This section contains information on the following topics:

- Fluid Properties

Fluid Properties

- **Inertia:** Sets the inertia of the motion.
- **Strength:** Sets the strength of the motion.

15.2.3 Frame Mask

This plug-in adds a framed mask around an object (e.g. image or geometry). In the properties the aspect ratio can also be set.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Effects

This section contains information on the following topics:

- Frame Mask Properties

Frame Mask Properties

- **Left, Right, Top and Bottom Edge:** Sets the tracked borders (see also **Tracking**).
- **Left, Right, Top and Bottom Gradient:** Sets the gradient from the edge of the frame to the edge of the object.
- **Aspect Ratio**: Sets the aspect ratio (**16:9**, **4:3** or **Custom**), if required (default is **Off**).
- **Custom**: Uses to set a custom aspect ratio (e.g. 1.85:1 or 2.39:1).
- **Tracking**: Uses a Container to track the location of the actual frame mask instead of animating all tracked borders manually. Drag a Container to the drop zone (1):

15.2.4 Image Mask

The Image Mask plug-in applies a mask to an image using other images (e.g. an alpha image).

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Effects

This section contains information on the following topics:

- Image Mask Properties

Image Mask Properties

- **Image:** Placeholder for image.
- **Blend:** Blends the image with the object.
- **Position X, Y %:** Sets the position of the mask in percentage for the screen or object.
- **Scale X, Y %:** Sets the size of the mask in percentage for the screen or object.
- **Fit:** Selects the size of the image mask to the screen or the object:
  - **Screen**: Sizes the image mask to the screen. If the object is moved it will move while the image mask stays fixed.
  - **Object**: Sizes the image mask to the object. If the object is moved the image mask will move with it.
- **Wrap:**
  - **Repeat**: Applies the image mask multiple times on the screen or object.
  - **Clamp**: Applies the image mask once on the screen or object.
- **Mask Aspect**: Sets the aspect of the image mask to the screen or the object.
- **Invert**: Inverts the image.
- **Alpha only:** Enables the alpha channel for the image (for example, an RGBA image).
- **Soft Mask:** Applies a soft mask (available when **Fit** is set to **Screen**).

15.2.5 Soft Mask

The Soft Mask plug-in applies a soft mask to an object (e.g. image or geometry).

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Effects

This section contains information on the following topics:

- Soft Mask Properties

Soft Mask Properties

- **Progress**: Moves the mask perpendicular to the mask direction. Default is 0.0 and is based on the configured center position.
- **Rotation**: Rotates the mask around the center.
- **Center X and Y**: Defines the X and Y position of the center the mask will have.
- **Gradient Width**: Sets the size of the gradient.
- **Invert**: Inverts the mask.
- **Mirror Gradient**: Mirrors the mask on both axis.
- **Blank**: Offsets the area that should not be affected by the mask.

- **Tracking**: Allows the progress to be controlled by a tracked Container to build the progress animation easier. Drag a Container to the drop zone.

15.2.6 Water Shader

This shader simulates a water surface.

> **Note:** This plug-in is located in: Built Ins -> Shader -> Effects

This section contains information on the following topics:

- Water Shader Properties

Water Shader Properties

- **Bump Height**: Sets the height of the bump map.
- **Bump Speed**: Moves the bump map of the surface. Useful to simulate a current.
- **Bump Scale**: Scales the bump map.
- **Wave Frequency**: Sets the frequency of the wave.
- **Wave Amplitude**: Sets the amplitude of the wave.
- **Wave Speed:** Defines the big waves of the water. You need geometry with many vertices to get smooth waves.
- **Environment Map**: Defines the texture which is reflected in the water. If no texture is applied the shader will use a default texture. The size of the environment map should be kept low (512x512 max), because it can consume a lot of texture memory.
- **Deep Water**: Sets the color of deep water.
- **Shallow Water**: Sets the color of shallow water.
- **Reflection**: Sets the color of the reflections.

  > **Note:** The blending of deep and shallow water depends on the angle of the camera and how it looks at the water.

- **Water Amount:** Controls how the deep and shallow color affects the look of the water.
- **Reflection Amount, Bias, Power and Multiplier:** Controls the reflection.
- **Blend Texture:** Allows you to blend the texture of the container into the water.

15.3 Filter

The following shader plug-ins are located in the Filter folder:

- Blend Image

- Blur
- Color Balance
- Radial Blur
- Sepia
- Sharpen

15.3.1 Blend Image

Use this plug-in to blend two images in different ways.

Note: This plug-in is located in: Built Ins -> Shader -> Filter

This section contains information on the following topics:

- Blend Image Properties

Blend Image Properties

- **Image**: Image placeholder. Reset removes the image.
- **Mode**: Available blend modes are:
  - **Blend**: Ordinary picture blending.
  - **Darken**: Uses the base or blend color, whichever is darker, as the destination color.
  - **Lighten**: Uses the base or blend color, whichever is lighter, as the destination color.
  - **Multiply**: Multiplies the base color with the blend color. The result color will be always a darker color.
  - **Screen**: Multiplies the inverse of the base and blend color. The result color will be always a lighter color.
  - **Color Burn**: Darkens the base color to reflect the blend color by increasing the contrast of the base and blend color.
  - **Color Dodge**: Brightens the base color to reflect the blend color by decreasing the contrast of the base and blend color.
  - **Overlay**: Screens or multiplies the colors depending of the base color.
  - **Soft Light**: Darkens or lightens the colors, depending on the blend color.
  - **Hard Light**: Multiplies or screens the colors, depending on the blend color.
  - **Add**: Brightens the base color to reflect the blend color by increasing the brightness.
  - **Subtract**: Subtracts the base and blend colors.
  - **Inverse Subtract**: Same as subtract but inverses the base and blend colors.
  - **Difference**: Subtracts either the base color from the blend color or the blend color from the base color depending on which color has the greater brightness value.
  - **Inverse Difference**: Same as difference but inverses the base and blend color.

    - **Exclusion**: Similar to the **Difference** mode, but lower in contrast.
- **Blend**: Blends the image with the object.
- **Position X and Y %**: Positions the image on the object.
- **Scale X and Y %**: Scales the image.
- **Wrap**: Wraps the object with the image. Available options are:
    - **Repeat**: Repeats the image.
    - **Clamp**: Clamps the image (no repeat).

15.3.2 Blur

Use this plug-in to blur an object with an image texture and/or a material on it. Values can be animated. You also have a blur plug-in directly on the image editor but that will only work on one image, using the shader all images within a group can be blurred.

The samples above depict the same image with blur quality set to Low, Normal and High with a range of 100.0. The last is the original (reference) with no blur filter.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Filter

This section contains information on the following topics:

- Blur Properties

Blur Properties

- **Quality:** Sets the quality of the blur filter. Low will be more pixelated, whereas high will be smooth. Available options are Low, Normal and High.
- **Range:** Sets the range of the blur. Range is from 0.0 to 100.0.

See Also

- Radial Blur

15.3.3 Color Balance

With Color balance you can decide brightness, contrast, saturation and hue.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Filter

This section contains information on the following topics:

- Color Balance Properties

Color Balance Properties

- **Brightness:** Sets the brightness of the object. Range is 0.0 to 1000.0.
- **Contrast:** Sets the contrast of the object. Range is -1000.0 to 1000.0.
- **Saturation:** Sets the saturation of the object. Range is -1000.0 to 1000.0.
- **Hue:** Sets the hue of the object. Range is 0.0 to 360.0.

15.3.4 Radial Blur

Use this plug-in if you want to blur the object in different ways.

The samples above depict the same image with radial blur quality set to Low, Normal, High and Very High with Center X and Y set to -25.0 and 25.0, Inner Range and Scale set to 25.0 and 50.0, respectively.

> ⚠ **Note:** Radial blur does not clamp the object.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Filter

This section contains information on the following topics:

- Radial Blur Properties

Radial Blur Properties

- **Quality:** Sets the quality of the blur filter. Low will be more pixelated, whereas very high will be smooth. Available options are Low, Normal, High and Very High.
- **Center X and Y:** Sets the position of the object's radial blur.
- **Inner Range:** Sets the inner range of the blur.
- **Scale:** Sets the scale of the object's blur.

See Also

- Blur

15.3.5 Sepia

Sepia can be used to generate a sepia effect using two colors, and to adjust the desaturation and tone of the color for example to blend the image with the color scheme of the overall scene. Sepia is similar to what is known as duo tone in most photo editing suites.

The samples above depict the same image with tone levels set to 100.0, 50.0 and 0.0. All other values are set to default. The last image is the original (reference) without the sepia plug-in.

⚠ **Note:** The color range is from 0.0 - 100.0 divided by 255 (0.392, 0.784 etc.).

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Filter

This section contains information on the following topics:

- Sepia Properties

Sepia Properties

- **Desaturation:** Sets the saturation level. Range is 0.000 (color) to 100.000 (black & white).
- **Tone:** Sets the color tone. Range is 0.000 to 100.000.
- **Light Color:** Specifies the light color and the strength of it. Range is 0.000 to 100.000.
- **Dark Color:** Specifies the dark color and the strength of it. Range is 0.000 to 100.000.

15.3.6 Sharpen

Use this plug-in for sharpening the given images on objects.

The samples above depict the same image with Quality set to High, and Scale values at 0.0, 100.0, 200.0 and 300.0, respectively. Range is set to 25.0 (default). The first image is the original (reference) with Scale set to 0.0.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Filter

This section contains information on the following topics:

- Sharpen Properties

Sharpen Properties

- **Quality:** Sets the quality of the sharpen filter. Low will be more pixelated, whereas high will be smooth. Available options are Low, Normal and High.
- **Range:** Increase this value to prepare more pixels to be included in the operation.
- **Scale:** Increase this value to sharpen.

15.4 Material

The following shader plug-ins are located in the Material folder:

- Anisotropic Light
- Bump Map
- Cartoon
- Gooch
- Normal Map
- Simple Bump Map

15.4.1 Anisotropic Light

This shader uses a lookup texture to compute the distribution of the specular component. All the shader properties are defined by the material of the container. Only Light number 1 is used and the light is always treated as a local light source.

In the figure below the left image is with anisotropic light and the right image is with a regular material.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Material

This section contains information on the following topics:

- Anisotropic Light Properties

Anisotropic Light Properties

This plug-in does not have any properties or parameters.

15.4.2 Bump Map

Use this plug-in to simulate complex surface structures. Instead of using many triangles to render a 3D object with many structures on its surface, use bump mapping to get a similar result.

Use plug-ins for Adobe Photoshop, Gimp, built-in functions of your modeling package or NVIDIA's tool called Melody.

> ⚠ **Note:** Make sure the geometry supports tangent space vectors. Otherwise use the Simple Bump Map shader.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Material

This section contains information on the following topics:

- Bump Map Properties

Bump Map Properties

- **NormalMap**: Stores a direction of normals directly in the RGB data. Move an normal map via drag and drop onto this field.
- **HeightMap**: Move an height map via drag and drop onto this field. The Height Map file is only visible and used when parallax mapping is enabled.
- **Light Source**: Selects the light source in the range 1-8. Please consider that only one light source is possible. By default the first light source is used.
- **Position X and Y**: Defines the position of the normal map in X-direction and Y-direction.
- **Scale X and Y**: Scales the normal map, respectively changes the width and height size of the image.
- **Wrap**: Repeats or clamps the bump map for the image.
- **Toggle X and Y**: Changes the direction of the normals stored in the normal map image.
- **Scale Height**: Changes the intensity of the bump map effect. Scale Height is visible when parallax mapping is enabled.
- **Parallax Mapping**: Provides an additional map to the texture for more apparent depth and the object is more realism.

See Also

- Simple Bump Map

15.4.3 Cartoon

This shader renders objects in a cartoon like manner. Shading is done in steps and you can define number of steps, the brightest color and the darkest color. Edge width allows you to blend between the steps. Material and texture of the container are ignored. Only Light number 1 is used and the light is always treated as local light source.

Note: This plug-in is located in: Built Ins -> Shader -> Material

This section contains information on the following topics:

- Cartoon Properties

Cartoon Properties

- **Shades**: Defines number of shade levels.
- **Edge Width**: Creates a smooth effect. The transitions are more hard or more soft.
- **Silhouette Width**: Sets the width of the contours around the rendered objects.
- **Silhouette Color**: Sets the color of the contours around the rendered objects. Additionally you can select the alpha value if you want transparent areas. 0 (Invisible) - (Visible) 100%.
- **Surface High**: Sets the color in the broad range. Additionally, you can select the alpha value if you want transparent areas. 0 (Invisible) - (Visible) 100%.
- **Surface Low**: Sets the color in the dark range. Additionally, you can select the alpha value if you want transparent areas. 0 (Invisible) - (Visible) 100%.

15.4.4 Gooch

Gooch is a per pixel light with a reflective high light. Material of the container is ignored, only the shader parameters are used to define the surface color. Texture mapping works as usual, but the texture can be blended with the surface color using the Texture Alpha parameter. Only Light number 1 is used and the light is always treated as local light source.

In the figure below the left image is with Gooch and the right image is with a regular material.

Note: This plug-in is located in: Built Ins -> Shader -> Material

This section contains information on the following topics:

- Gooch Properties

Gooch Properties

- **Surface**: Describes the intensity of color like the ambient color of a material.
- **Cool**: Defines the cool color. Both colors cool and warm will be merged. The result is a color gradient.
- **Warm**: Defines the warm color.
- **Highlight**: Sets the diameter of the highlight.
- **Highlight Alpha**: Sets the alpha or transparent value of the highlight.
- **Texture Alpha**: Describes the visible value of the texture image in percent. 0 (Invisible) - (Visible) 100%.

15.4.5 Normal Map

The Normal Map Shader adds a color to each face according to the direction of the face's normal. RGB values change between 100-200 while the normal XYZ rotates.

If the Normal is pointing along the Y-axis, this will give the RGB value (100,200,100). A normal along the X-axis give the RGB value (200,100,100). 45° between x and Y-axis will give RGB value (150,150,100).

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Material

This section contains information on the following topics:

- Normal Map Properties

Normal Map Properties

This plug-in does not have any properties or parameters.

15.4.6 Simple Bump Map

With this plug-in you have the opportunity to simulate complex surface structures. Instead of many triangles to render a 3D object with many structures on its surface you can use bump mapping to get a similar result.

You can only use a gray scaled image as normal map. If you want a more accurate result and better effects please use the Bump Map plug-in instead which is using an RGB normal map image.

To generate a Simple Bump Map you can use the plug-ins for Photoshop, Gimp, the built-in functions of your modeling package or NVIDIA's tool Melody. The Height Map (used for parallax mapping) should be a grayscaled image (white -> highest, black -> lowest bumps).

Note: All calculations are done in object space.

Note: This plug-in is located in: Built Ins -> Shader -> Material

This section contains information on the following topics:

- Simple Bump Map Properties

Simple Bump Map Properties

- **NormalMap**: Stores a direction of normals directly in the RGB data. Move an normal map via drag and drop onto this field.
- **HeightMap**: Move an height map via drag and drop onto this field. The Height Map file is only visible and used when parallax mapping is enabled.
- **Light Source**: Selects the light source in the range 1-8. Please consider that only one light source is possible. By default the first light source is used.
- **Position X and Y**: Defines the position of the normal map in X-direction and Y-direction.
- **Scale X and Y**: Scales the normal map, respectively changes the width and height size of the image.
- **Wrap**: Repeats or clamps the bump map for the image.
- **Toggle X and Y**: Changes the direction of the normals stored in the normal map image.
- **Scale Height**: Changes the intensity of the bump map effect. Scale Height is visible when parallax mapping is enabled.
- **Parallax Mapping**: Provides an additional map to the texture for more apparent depth and the object is more realism.

See Also

- Bump Map

15.5 PixelFX

The Pixel Shader plug-ins are part of the PixelFX plug-ins set.

Viz Artist does not support more than one general shader on a Container, or in a nested fashion. However, in some cases, PixelFX, by its nature, does support multiple shaders per Container.

Some have high cautionary indications. Some are very light. Some are stackable, and others are not.

The following Shader plug-ins are located in the *PixelFX* folder:

15.5.1

- PixelFXLenseFlare
- pxBCubic
- pxCCBase
- pxEqualize
- pxGradient
- pxInvert
- pxLensDistort
- pxMotionBlur
- pxNoise
- pxPixelate
- pxPosterize
- pxRecolor
- pxRipple
- pxSparkle
- pxTurbDissolve and pxTurbWipe
- pxTurbulence
- pxTwirl
- pxWaves

15.5.2 Common Properties

15.5.3 Most of the plug-in icons have mini icons that show certain properties of the particular plug-in:

- At the bottom left, there is a performance bar, with three blocks (like a cellphone battery indicator). One block means the plug-in will not have a drastic performance effect, and three means the plug-in is taxing and should therefore be used with caution.
- At the bottom right the orange stack shows if the plug-in is stackable, meaning it can be used in conjunction with color correction plug-ins.
- Some plug-in icons themselves are divided in half with one side showing the visual effect with the plug-in and the other without it.

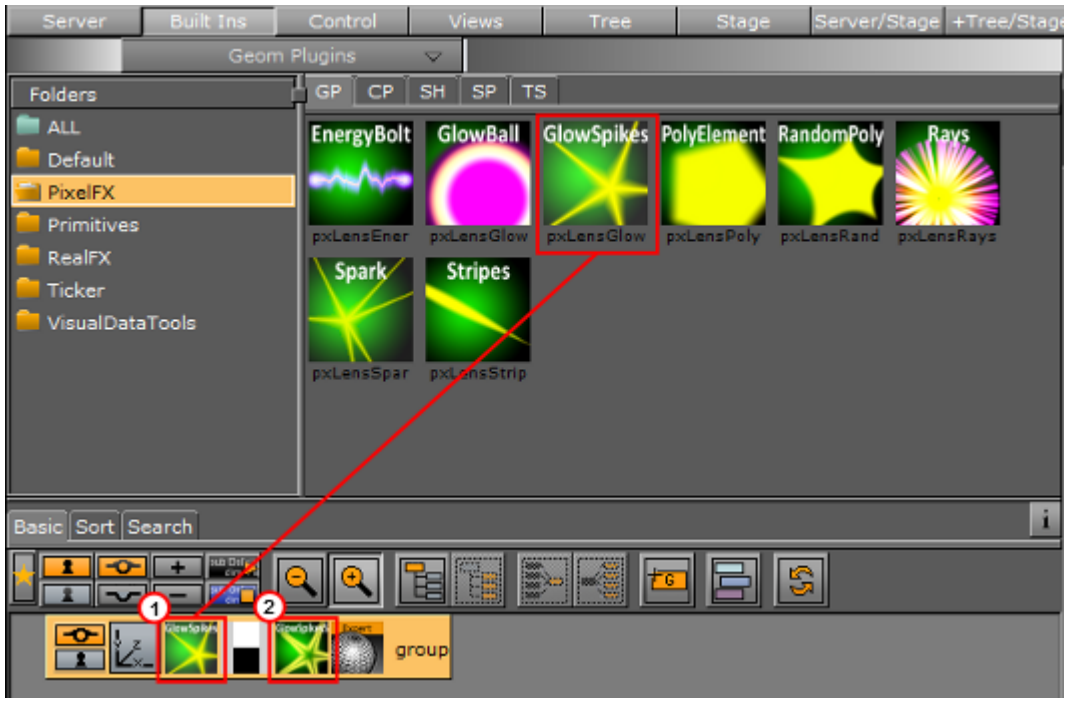
15.5.4 PixelFXLenseFlare

The Lense Flare plug-ins, which are part of the PixelFX plug-ins set, simulate the effect of streaks and spots of light caused in real cameras when light enters and bounces inside a camera lens. This plug-in set contains geometries that simulate different shapes, which can be used for lens flare or simple flare effects.

The Flare draws itself as a combination of geometry shader and pixel shader.

When a Flare Container is created (1) a non-editable pixel shader (2) is automatically added to the created Flair Container.

The non editable shaders are needed to do more (detailed, on a per pixel basis) graphical effects on the GPU (manipulating colors, textures, lighting, etc.), which cannot be done with Geometries only.



These plug-ins are located at: Built Ins -> Shader -> PixelFXLenseFlare

See Also

- PixelFX plug-ins in Geometry plug-ins

15.5.5 pxBCubic

When dealing with images that contain line art or lines of high contrast, you may want to protect yourself from the less than the desirable effect of pixelation when an image is over scaled. The BCubic plug-in introduces three bi-cubic algorithms. Each is useful in different contexts of its respective type. It is very easy to find the best adjustment to your specific scenario.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.6 pxCCBase

The pxCCBase (Color Correction) applies a cumulative set of color corrections using the pxStack plug-in.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.7 pxEqualize

pxEqualize allows you to adjust the black, mid-range and white color levels in an image.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.8 pxGradient

The pxGradient plug-in has a similar functionality to the gradient function in Photoshop. It can be applied on a container that holds a geometry, an image, or even an empty container. You can define as many gradient stops as you wish, as well as their origin points, angles, and extent. You can also define the gradient on the alpha channel (256 levels) and introduce turbulence noise that will comply with the defined gradient.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

At the top of the plug-in parameters, you see the controls that let you define the direction, radius and origin of the gradient. Below that you see a two radio button sets. The first lets you set the gradient type, followed by three Repeat types

Below that, you see the color slider, the color values and alpha ramp settings.

⚠ **Note:** Under the title Color Ramp and Alpha Ramp, you have an active/inactive option, should you want to have an RGB ramp and no alpha ramp, or vice versa.

Below the RGB and Alpha area of the editor, you find the option of adding Turbulence to the gradient. The turbulence has three simple parameters. Amount determines how aggressive the turbulence looks. Wavelength determines how big or small the turbulence increments are. Progress

(similar to that in TextFX plug-ins) allows you to breathe life into the turbulence, providing interesting an effect when animated.

At the very bottom, it is important to make sure that you turn off the Show GUI button, whose default is On; if shown on air, the effect will be more embarrassing than dramatic.

Best Practices

It is important to understand that the Gradient plug-in is actually drawing pixels. Therefore if this plug-in is applied to an empty group or if it is applied to a geometry object that does not contain a texture, the gradient plug-in will generate the texture into which it will draw its pixels. So if you have a cube and add the Gradient to it, it behaves exactly as a texture. However it can also work in conjunction with a texture. If you then take an image and drop it on the cube, you give a color wash to the image that works in a gradient-like fashion. The gradient-generated texture can also be edited via the texture editor as any other texture would be.

The biggest benefit of using the Gradient is conservation of texture memory. Instead of storing very many different alpha and color ramps in your image pool, you can simply achieve the overwhelming majority of these effects by well thought out and judicious use of the Gradient plug-in. It has a minimal performance penalty and it can be animated. If you use ramps often in your designs, the Gradient is one of your handiest tools.

Tips and Tricks

Often you put reflection maps on a translucent geometry to create a glass-like reflection (think of a light box or an aquarium). For the reflection texture, one often uses an image of static noise or ramps of sorts. Since the texture is mapped on the object using a reflection mapping mode, it is very difficult to create the right intensity of noise in your texture so that the resulting reflection looks good. The Gradient turbulence functionality allows you to set the intensity of the noise of the texture “on the fly”, minimizing the round-trips to Photoshop.

15.5.9 pxInvert

Often you create an image with an alpha channel in Photoshop, import it to Viz Artist/Engine and only then realize that your alpha channel is inverted. Or you find an image in the Viz Artist/Engine pool that you want to use, but the way you want to use it is where the RGB is actually the alpha channel.

pxInvert allows you to perform these operations without adding yet another image to your image pool, or overloading your scenes with a few versions of the same image, where the difference is simple flips of the channels.

This plug-in allows you to invert each of the color channels RGB and Alpha individually, and to use the RGB as alpha and the alpha as RGB.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.10 pxLensDistort

pxLensDistort creates a “fish eye” lens distortion effect on an image. It can also be used to create the effect of an old television with a curved screen.

The amount of the effect can be configured using the following parameters:

- Amount
- 2nd Order
- Scale
  The center around which the distortion goes can be configured by using the X Center and Y Center parameters.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.11 pxMotionBlur

pxMotionBlur applies motion blur to the texture. Supports pxColorWorks and pxStack plug-ins.

Adjust the following parameters as required to achieve the required effect:

- X Center
- Y Center
- Rotate
- Zoom
- X Shift
- Y Shift
- Fix Edges (on/off)
- Samples (enter a value)

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.12 pxNoise

pxNoise is a white noise generator. It deploys the same turbulence as seen in other PixelFX plug-ins. Similarly to the pxGradient, it generates its own texture if applied to a container without an image, or blends with the existing texture if the container has an image applied.

Adjust the following parameters as required to achieve the required effect:

- **Size**
- **Amplitude**: Intensity/contrast of grains

- **Color**
- **Speed**: Turbulence noise
- **Stretch**: Stretch the grain to achieve a rain look of noise
- **Stretch Angle**
- **Movement**: Transforms the generation of noise
- **Transform as Texture**- **On**: Conform to texture coordinates. **Off**: Ignore texture coordinates and behave as stencil mapping.
- **Apply to: RGB, RGBA, Alpha**: The channels in which the noise will be generated.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.13 pxPixelate

Similar to the pixelate function in Photoshop, the pxPixelate effect appears as if you reduce the resolution of your image. You may work in a proportional (locked) or disproportional mode.

To achieve a nice animated transition, tweak the Smoothness parameter so that the change between the different pixelation levels will be smoother or harsher.

Adjust the following parameters as required to achieve the required effect:

- Lock X/Y (on/off)
- X Size
- Y Size
- Smoothness
  Supports pxColorWorks and pxStack plug-ins.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.14 pxPosterize

pxPosterize recolors an image using only a limited number of colors for a 'posterize' effect. You can adjust the number of levels and set the transition to one of **Off**, **Fade** or **Glow**.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.15 pxRecolor

pxRecolor 'false-colors' an image by using either Red, Green, Blue or Luma as the input to a color ramp.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.16 pxRipple

pxRipple creates a pond ripple effect. Supports pxColorWorks and pxStack plug-ins.

Adjust the following parameters as required to achieve the required effect:

- X Center
- Y Center
- Radius
- Amount
- Wavelength
- Progress

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.17 pxSparkle

The pxSparkle plug-in generates a sparkle pattern as a ray emitting from a point. The plug-in editor provides options to control the density of the rays, the x,y position from where they are emitted, the brightness (Amount) of rays, as well as other parameters, which will be covered specifically in context. Before delving into these parameters however, it is important to understand the pxSparkle, like other PixelFX shaders (pxGradient, pxNoise) generates its own pixels. As well, it has the ability to be applied on top of an image and can blend with the image's pixels.

When the pxSparkle plug-in is applied to a container without an image, it creates sparkle results. However you can also apply it on a container with an image. In this case, you need to decide whether the sparkle pattern should blend on top of the image or whether it cuts the image. If the Cut Image option is on, the host image is seen only in areas that are brighter than zero. The brighter the sparkle, the more apparent the host image will be.

The pxSparkle plug-in can be used as a flare to achieve its effect in a radius (the size of the hot spot). The inner Amount (the brightness of the hot spot) and the radius will crop or fade the sparkle radially. You can also so the Speed parameter to set the sparkle in motion.

- Transform as texture: This option forces the rendering of the sparkle to ignore the viewmatics - the sparkle will always be facing the eye-point. This option is handy when applying the sparkle on top of a hierarchy.
- Aspect: This option allows you to set the proportion of the sparkle independently of the texture transformation. Often you stretch textures to get what you want. However when applying the pxSparkle plug-in to a distorted texture, you may want to maintain its roundness. Therefore you can use the Aspect parameter to compensate for the texture distortion.
- Angle: This lets you rotate the sparkle without manipulating the texture coordinates. This option is handy when you have a static image on top of which you want to apply a rotating sparkle.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.18 pxTurbDissolve and pxTurbWipe

Both pxTurbDissolve and pxTurbWipe utilize the native multi-texturing support of Viz Artist.

pxTurbWipe includes the parameters of Angle and Softness.

To apply TubDissolve and/or TurbWipe

1. Insert an image into the scene tree.
2. Go to the image's texture editor and make sure that the texture Unit is set to 1 and the Inheritable option is active.
   This will be your Image A in an A-B transition paradigm.
3. Place a child image underneath the first one.
   This will be Image B.
4. In Image B's texture editor, set the Unit to 2 and make sure that the Inheritable option is inactive.
5. Apply the Dissolve plug-in to the Image A (the parent image).
6. In the plug-in editor, first adjust the Transition, followed by the other parameters, as per your liking.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

See Also

- pxGradient

15.5.19 pxTurbulence

pxTurbulence creates a distortion effect using a Perlin turbulence function.

Supports pxColorWorks and pxStack plug-ins. Adjust the wavelength, amplitude and speed to achieve the required effect.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.20 pxTwirl

pxTwirl creates a distortion effect that twists an image around a central point.
Supports pxColorWorks plug-ins and pxStack plug-ins.

Adjust the following parameters as required to achieve the required effect:

- X Center
- Y Center
- Radius
- Amount
- Edge Softness

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.5.21 pxWaves

pxWaves creates a distortion effect that makes an image look wavy. Supports pxColorWorks plug-ins and pxStack.

Adjust the following parameters as required to achieve the required effect:

- Type: Sine, Noise, Triangle, Square
- Angle
- Amount
- Wavelength
- Progress
- Seed

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> PixelFX

15.6 RealFX

The RealFX plug-in set enables you to create particle effects in Viz Artist.

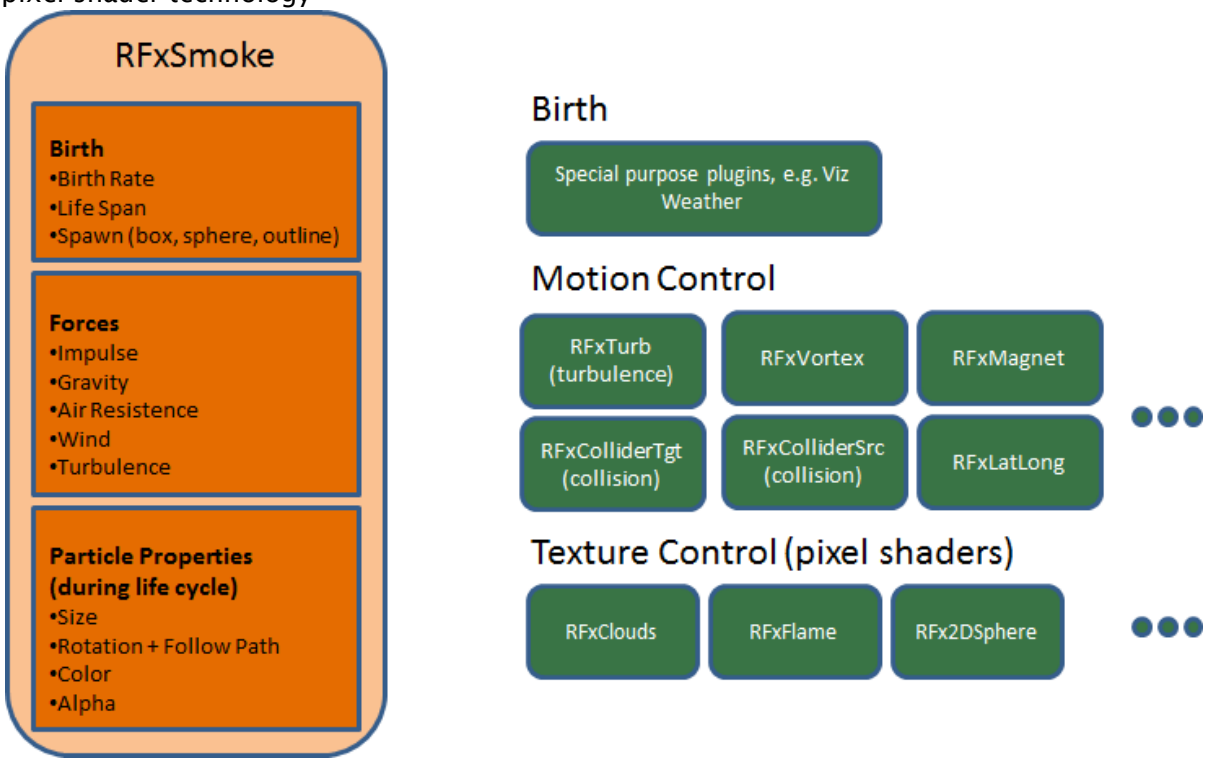
Particle systems are a computer graphics technique to simulate certain physics-based effects, which are otherwise very hard to reproduce with conventional rendering techniques. Examples of such effects which are commonly replicated using particle systems include fire, explosions, smoke, weather effects, sparks, falling leaves, dust, meteor tails, or abstract visual effects like glowing trails, magic spells, etc.

The particle effects in Viz Artist/Engine run in real-time, meaning that there are a few inherent constraints that must be taken into account when considering best practices for employing this plug-in set. For example, there is a trade-off between the number of particles and performance optimization; more generally there needs to be a considered balance between performance and visual quality.

RFxSmoke is the baseline plug-in within the RealFX plug-in set. The remaining plug-ins in this set are applied on top of RFxSmoke in any given container. RFxSmoke includes built-in functionality and the ability to host the additional functionality contained in the other plug-ins in this set. Part of the built-in functionality, e.g. turbulence, is kept for compatibility with previous version of Viz Artist.

There are three categories of additional plug-ins:

- **Birth plug-ins:** refer to where the particles are spawned
- **Motion control plug-ins:** govern the position, direction, velocity, size and color of each particle
- **Texture control plug-ins:** affect the texture mapping and the “look” of each particle by using pixel shader technology



This section contains information on the following topics:

- RFx2DSphere
- RFxClouds

- RFxFlame

15.6.1 RFx2DSphere

RFx2DSphere creates a 2D sphere-like look on each particle. The sphere is affected by light sources.

Adjust the following parameters as required to achieve the required effect:

- Color Mode (Material, Color, Color-Material)
- Radius
- Blend

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RealFX

15.6.2 RFxClouds

Located under the Shader tab, RFxClouds create a dynamic plume-like texture on each particle, and changes randomly for each particle over time It can be used on clouds and smoke.

Adjust the following parameters as required to achieve the required effect:

- Brightness
- Turb Scale X
- Turb Scale Y
- Turb Speed

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RealFX

15.6.3 RFxFlame

RFxClouds create a dynamic flame-like texture on each particle, and changes randomly for each particle over time.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RealFX

See Also

- RealFX plug-ins in Geometry plug-ins
- pxColorWorks in Container plug-ins
- RFxSmoke

15.7 RTT Advanced Materials

RTT Advanced Materials plug-ins allow you to create a number of high-quality materials using state-of-the art shader technology. Using the plug-ins is very easy. Simply drag the shader plug-in of your choice onto selected geometry, assign a material and set up the additional parameters as shown by the shader in its own editor. Some plug-ins remain inactive until you have applied all necessary textures, depending on the shader plug-in. This may include a basic texture on your container and/or additional textures to be dropped in the plug-in interface. You can modify all parameters of the applied material on your object, such as Ambient Color, Diffuse, Specular and Emission, Shininess and Alpha. All these parameters will also affect the shader.

All RTT Advanced Materials plug-ins (except the Bump Optimized Shader) allow manipulating the texture mapping of each texture map individually. This means you can use a different position, rotation and scaling for each texture. As for the mapping of additional textures inside the Shader, the same mapping method as being assigned to the texture on your container is used. It is also possible to change the alignment of the environment textures to achieve the correct reflections on the respective object surface.

If you do not want to use individual texture transformations, you may switch the option off. Each additional texture (except the environment texture) offers the button “Enable Individual Transform”. With the button turned “off “, the texture mapping coordinates of the base texture will be used.

RTT Shaders support up to eight light sources. The supported light types are infinite and local lights; also the light color of these lights is supported.

If you import the Viz Artist/Engine archive **RTT_ShaderPresets_V1** you will obtain about 60 sample scenes which are meant to give you an idea how to work with the Shaders. These presets are free to use, but in general, they are thought as a guideline for how to work with textures and the Shaders. Correctly working with the textures will dramatically improve the look of your Shader. The Shaders use color maps, bump maps (in the normal map format), specular maps and cubic environment maps in HDR and LDR formats.

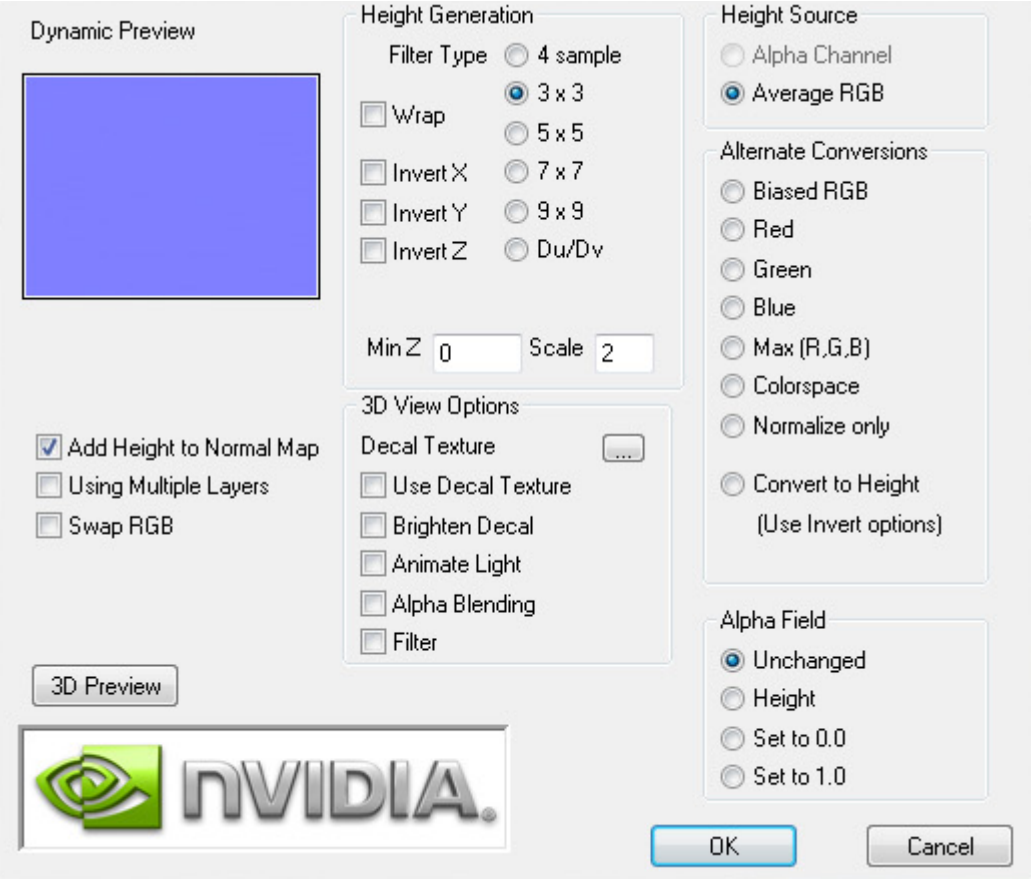
Creating color maps is straightforward. They are regular RGB textures that you can use either on the container or inside the Shader, depending on the Shader you are using.

Creating specular maps is also straightforward. They are grayscale textures where you define, through a gradient from white to black, where light is reflected or not.

Creating normal maps for the bump inside your Shader is not complicated either; just go to NVIDIA’s corporate web page and download the relevant NVIDIA Photoshop plug-ins, or use this link NVIDIA Texture Tools for Adobe Photoshop.

After installing the filter in question, you have a new section in the filter category in Adobe Photoshop. By applying this filter, you will be shown a window with various settings (Please read

the corresponding NVIDIA User's Guide); yet there is not much work involved. Just have a look at the settings in the NVIDIA Photoshop plug-in Image below and start exploring them on your own.



Creating Cubic Environment maps is no secret either. Cubic Environment Maps can be rendered either with a preferred Render Engine (Mental Ray, V-Ray, etc.), if it should be a 3D generated environment map, or they can be created with HDR Images, which are then transformed with HDR Shop.

⚠ **Note:** HDR Shop has a menu for panoramic transformations, here you can create a cubic environment image by transforming it from latitude/longitude to an cubic environment by just a few mouse clicks (see instructional video).

⚠ **Note:** Creating Cubic Environment Maps (HDR or LDR) is a well known technique for 3D experts. This Viz Artist User Guide does not explain the creation of these Environment Maps in detail, as there are numerous tutorials on the web for this texture creation procedure.

If you do not have a license for the RTT Advanced Materials plug-ins, you may contact either one of your local Vizrt Support people or you can send an e-mail to license@vizrt.com to obtain either a time-limited demo license or a full license. Without a license, you will be able to load the preset scenes but you cannot modify them; only a few parameters will be adjustable.

This section also contains information on the following topics:

- Cube Map
- HDR
- Normal Map
- Fresnel
- RTT

15.7.1 Cube Map

Cube map textures are typically used for approximated environmental reflection and refraction. They consist of a set of six two-dimensional textures that form a textured cube centered at the origin. Unlike 2D sphere maps, they grant higher details for the whole environment. The RTT Advanced Materials plug-ins accept only cube maps as reflection textures and require them to be stored in a vertical cross layout. Cube maps can soon be created easily by using Vizrt's "Dynamic Texture" plug-in. To achieve the best results with the Shaders we strongly recommend to create your cube maps in HDR format.

⚠ **Note:** The vertical cross cubemap size is required to be one of the following: 192 by 256, 384 by 512, 768 by 1024, 1536 by 2048 or 3072 by 4096 pixels, where a higher resolution indicates higher image quality. The general rule of thumb is that the contained six images need to have power-of-2 resolution.

15.7.2 HDR

HDR (High Dynamic Range) images can store a much wider range of brightness values than common (i.e. low dynamic range) images. Therefore, it is possible to create more brilliant highlights and reflections when using HDR images. HDR images can be shot by using special spherical cameras, or created with the aid of HDR image processing tools. Furthermore ready-to-use HDR cube maps are available as DVD collections. All plug-ins with environmental reflections support HDR cube maps. `.hdr` format images are supported.

15.7.3 Normal Map

Normal maps are textures used to calculate bump mapping. They encode surface details – the normals of a surface – as RGB color values and therefore show usually in some shades of blue. They can be created easily from (grayscale) height maps with plug-ins for Adobe Photoshop or Gimp.

15.7.4 Fresnel

The Fresnel effect describes how much light is reflected on a surface and how much light is refracted through this surface. At shallow angles the reflection is strong while there is almost no refraction.

15.7.5 RTT

All RTT Advanced Materials are developed by Realtime Technology AG. Realtime Technology AG is a worldwide leading supplier of 3D real-time visualization technologies and services for industrial applications in the automotive, aircraft and consumer goods industries.

⚠ **Note:** All RTT shaders can be uninstalled from the Viz Artist/Engine program menu.

15.7.6 Anisotropic Light Shader

An anisotropic highlight is calculated to simulate advanced surfaces. The highlight can be controlled in using two parameters. In addition, a color texture can be used to add more details. This shader is particularly useful when no surrounding reflection is required.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Anisotropic Light Shader Properties
- Best Practices

Anisotropic Light Shader Properties

- **Roughness X and Y**: Controls the width and height of the anisotropic highlight.
- **Color Map 2**: Offers the possibility to assign a second diffuse color texture that is blended with the first color texture (according to the chosen technique)
- **Enable Individualized Transform**: Activates the option to define an independent transformation (Position, Rotation and Scale values next to the Image Parameter) for Normal- and/or Bumpmaps instead of using the regular texture coordinates.
- **Technique**: Shows a list of available techniques.
  - **Standard**: Is a technique that creates an anisotropic highlight (to be used with geometry that has no texture coordinates and normals).
  - **Standard_Texture**: Is a technique that mixes a diffuse texture color into the material color.
  - **Standard_MultiTexture_Add**: Is a technique to additively blend Color Map 2 with the texture.
  - **Standard_MultiTexture_Blend**: Is a technique to blend Color Map 2 with the texture according to their alpha values.

- **Standard_MultiTexture_Subtract**: Is a technique to subtractively blend Color Map 2 with the texture.
- **Standard_MultiTexture_Modulate**: Is a technique to blend Color Map 2 by multiplying it by the texture.
- **Binormal**: Is a technique to create an anisotropic highlight using binormals and tangents of the geometry (to be used with geometry having texture coordinates and normals).
- **Binormal_Texture**: Is a technique to mix a diffuse texture color into the material color.
- **Binormal_MultiTexture_Add**: Is a technique additively blend Color Map 2 with the texture.
- **Binormal_MultiTexture_Blend**: Is a technique to blend Color Map 2 with the texture according to their alpha values.
- **Binormal_MultiTexture_Subtract**: Is a technique to subtractively blend Color Map 2 with the texture.
- **Binormal_MultiTexture_Modulate**: Is a technique to blend Color Map 2 by multiplying it by the texture.

Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material. When applying a texture technique it is compulsory to assign a texture as a color/basic texture. In case of multi texturing, an additional second texture must be assigned in the corresponding Shader rollout. Therefore, do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess.

15.7.7 Brushed Metal Shader

By using bump mapping and reflections from a cube map, the impression of more sophisticated metal surfaces, such as brushed metals, is created. The surface is lit by an anisotropic highlight. To describe the surface structure, a normal map is used.

This section contains information on the following topics:

- Brushed Metal Shader Properties
- Best Practices

Brushed Metal Shader Properties

- **Roughness X and Y**: Controls width and height of the anisotropic highlight.
- **Bump Height**: Affects the height of the created bumps and wrinkles on the surface.
- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.

- **Reflection Multiplier / Exposure**: Affects the exposure of the environment map. This is especially helpful when working with HDR images.
- **Gamma of EnvMaps**: Controls the gamma correction of the environment map. This is once again particularly helpful when working with HDR images.
- **Bump Map / Normal Map**: Is the texture that defines the surface structure with encoded normals.
- **Enable Individualized Transform**: Activates the option to define an independent transformation (Position, Rotation and Scale values next to the Image Parameter) for Normal- and/or Bumpmaps instead of using the regular texture coordinates.
- **Environment Map**: Is a vertical cross cube map that describes the environmental surrounding used to calculate reflections.
- **Technique**: Shows a list of available techniques.
  - **Standard Bump**: Is a technique that should be used with geometry that has no standard UV texture coordinates and normals.
  - **Binormal Bump**: Is a technique that should be used with geometry having texture coordinates and normals.

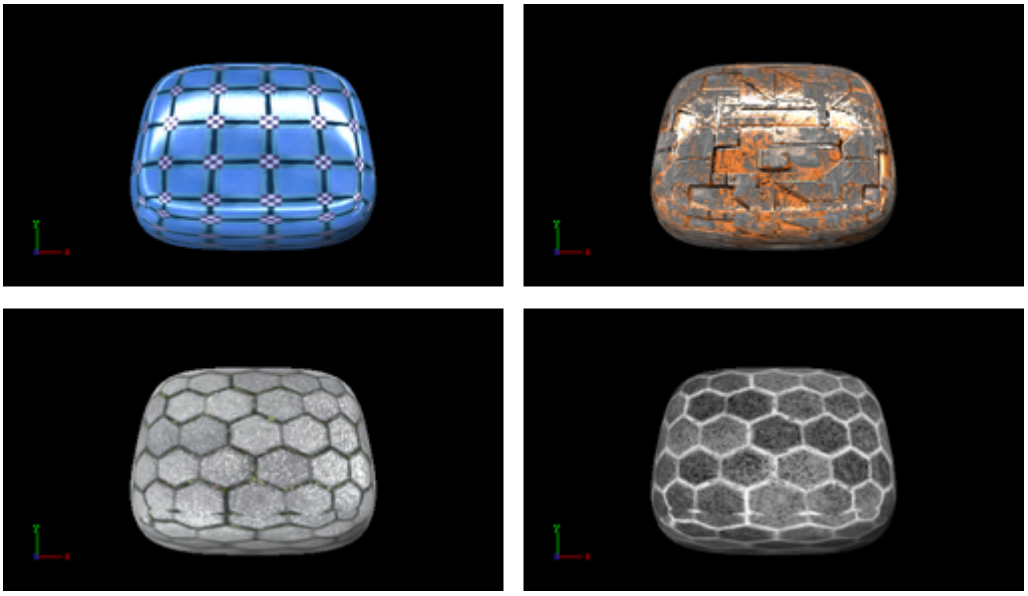
Best Practices

The Brushed Metal Shader should be preferred to Anisotropic Light Shader whenever you want to create highly detailed metal surfaces. However, the shader is more complex and requires more rendering time. Therefore, it may be unsuitable, depending on the complexity of the other parts of the scene and also on the graphics hardware available.

Apart from the parameters in the plug-in container, it is necessary to assign a texture as a color/basic texture and a material. Do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess, to achieve the required look for your surface.

15.7.8 Bump Optimized Shader

By using bump mapping, the impression of detailed surface structures is created while the structures are defined by normal maps. In addition to a highlight, the reflection of the surrounding is calculated (from a cube map). It is possible to add blurriness to the reflection to create rougher surfaces. A parameter is available to modulate whether the reflections follow the surface structures or behave like a clear coat layer on the top of it. The specular map describes where on the surface highlight and reflections are shown and where the surface shows only diffuse lighting.



This plug-in offers the same functionality as the Bump Shader plug-in, but as an optimization, it does not allow separate texture mapping for bump and specular maps. It consequently requires less rendering time and should be used whenever additional parameters are not required, i.e. you have matching sets of textures that can be used with the shader or you do not need to animate the textures. This shader should be preferred especially in complex scenes.

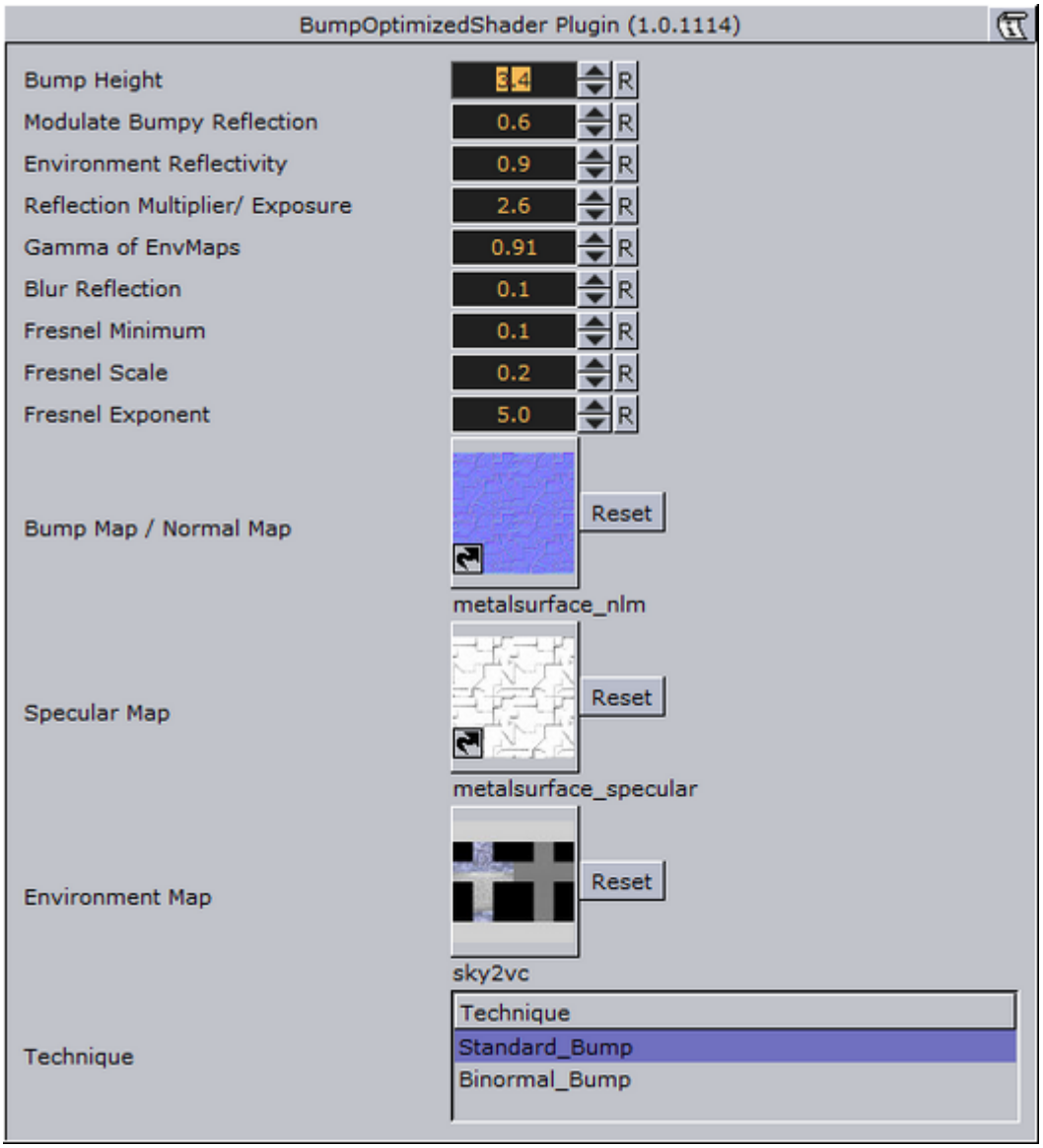
> ⚠ **Note:** All RTT shaders can be uninstalled from the Viz Artist/Engine program menu.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Bump Optimized Shader Properties

Bump Optimized Shader Properties



- **Bump Height**: Manipulates the height of the created bump on the surface.
- **Modulate Bumpy Reflection**: Controls whether the environment reflection is smooth or influenced by surface details. To calculate reflections on a smooth clear coat layer, value 0.0 is set by default. On the other hand, value 1.0 is used to incorporate surface details.
- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.
- **Reflection Multiplier/Exposure**: Affects the exposure of the environment map. This is especially helpful when working with HDR images.
- **Gamma of EnvMaps**: Controls the gamma correction of the environment map. This is once again particularly helpful when working with HDR images.

- **Blur Reflection**: States how much the environment reflection is blurred to create the impression of a rougher surface.
- **Fresnel Minimum, Fresnel Scale and Fresnel Exponent**: Affect the Fresnel effect upon the reflection.
- **Bump Map/Normal Map**: Is the texture that defines the surface structure/bump with encoded normals.
- **Specular Map**: Is the texture that defines where the surface is lit by specular highlight and reflection.
- **Environment Map**: Is a cube map that describes the environmental surrounding used to calculate reflections.
- **Technique**: Shows a list of available techniques.
  - **Standard Bump:** Is a technique to be used with geometry that has no standard UV texture coordinates and normals.
  - **Binormal Bump:** Is a technique to be used with geometry having texture coordinates and normals.

15.7.9 Bump Shader

By using bump mapping, the impression of detailed surface structures is created while the structures are defined by normal maps. In addition to a highlight, the reflection of the surrounding is calculated (from a cube map). It is possible to add blurriness to the reflection to create rougher surfaces. A parameter is available to modulate whether the reflections follow the surface structures or behave like a clear coat layer on the top of it. The specular map describes where on the surface highlight and reflections are shown and where the surface shows only diffuse lighting.

Note: This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Bump Shader Properties
- Best Practices

Bump Shader Properties

- **Bump Height**: Manipulates the height of the created bump on the surface.
- **Modulate Bumpy Reflection**: Controls whether the environment reflection is smooth or influenced by surface details. To calculate reflections on a smooth clear coat layer, value 0.0 is set by default. On the other hand, value 1.0 is used to incorporate surface details.
- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.

- **Reflection Multiplier/Exposure**: Affects the exposure of the environment map. This is especially helpful when working with HDR images.
- **Gamma of EnvMaps**: Controls the gamma correction of the environment map. This is once again particularly helpful when working with HDR images.
- **Blur Reflection**: States how much the environment reflection is blurred to create the impression of a rougher surface.
- **Fresnel Minimum, Fresnel Scale and Fresnel Exponent**: Affect the Fresnel effect upon the reflection.
- **Bump Map/Normal Map**: Is the texture that defines the surface structure/bump with encoded normals.
- **Specular Map**: Is the texture that defines where the surface is lit by specular highlight and reflection.
- **Environment Map**: Is a cube map that describes the environmental surrounding used to calculate reflections.
- **Technique**: Shows a list of available techniques.
  - **Standard Bump:** Is a technique to be used with geometry that has no standard UV texture coordinates and normals.
  - **Binormal Bump:** Is a technique to be used with geometry having texture coordinates and normals.

Best Practices

If you use this shader in complex scenes, you should favor the Bump Optimized Shader. The difference between the Bump Shader and the Bump Optimized Shader is that you can modify the texture coordinates inside the Bump Shader, which, however, needs more rendering performance. If you do not have to adjust the texture coordinates individually, or animate them separately, you should use the BumpOptimizedShader to save rendering performance.

Apart from the parameters in the plug-in container, it is necessary to assign a texture as a color texture and a material.

15.7.10 Fabric Shader

On a dull surface, the impression of detailed surface structure is created by using bump mapping. The structure is defined by a normal map. A highlight is placed along the edges of the object to increase the impression of fabric surfaces.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Fabric Shader Properties

Fabric Shader Properties

- **Velvety (Limb) Parameter**: Controls the strength of the velvety effect at grazing angles of view.
- **Bump Height**: Manipulates the height of the created bumps and wrinkles on the surface.
- **Bump Map/Normal Map**: Is the texture that defines the surface structure with encoded normals.
- **Technique**: Shows a list of available techniques.
  - **Standard_Bump**: Is a technique has to be used with geometry that has no standard UV texture coordinates and normals.
  - **Binormal_Bump**: Is a technique should be used with geometry having texture coordinates and normals.

15.7.11 Glass Shader

Apart from the parameters in the plug-in container, it is necessary to assign a texture as a color/ basic texture and a material. Do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess, to achieve the required look for your surface.

Both refraction and reflection are taken from the surrounding (from a cube map) and are blended to create the dielectric effect of glass. In addition, lighting by a highlight and a color texture are used to describe the material in detail. The opacity of the object can be controlled to modulate between the solidly lit surface and the transparent impression of glass.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Glass Shader Properties
- Glass Shader Best Practices

Glass Shader Properties

- **Refraction Index**: Controls the refraction index of the material, and accordingly the strength of the refraction.
- **Fresnel Minimum, Fresnel Scale and Fresnel Exponent**: Affects the Fresnel effect upon the reflection.
- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.
- **Bump Height**: Manipulates the height of the created bumps on the surface.

- **Reflection Multiplier/Exposure**: Affects the exposure of the environment map.
- **Gamma of EnvMaps**: Controls the gamma correction of the environment map. This is particularly helpful when working with HDR images.
- **Blur Reflection**: States to which extent the environment reflection and refraction are blurred to create the impression of a rougher surface.
- **Opacity**: Influences the mixing of the solid material color and the dielectric glass effect (reflection and refraction).
- **Bump Map/Normal Map**: Is the texture that defines the surface structure with encoded normals.
- **Environment Map**: Is a cube map that describes the environmental surrounding used to calculate reflections and refractions.
- **Technique**: Shows a list of available techniques.
    - **Glass**: Is a technique to be used for plain glass.
    - **Glass_Texture**: Is a technique to mix a diffuse texture color into the material color.
    - **Glass_BumpStandard**: Is a technique to create bump mapping with the given normal map on the surface of the geometry that has no standard UV texture coordinates and normals.
    - **Glass_BumpStandard_Texture**: Is a technique to use a diffuse color texture and bump mapping on the surface of the geometry that has no texture coordinates and normals.
    - **Glass_BumpBinormal**: Is a technique to create bump mapping for geometry with texture coordinates and normals.
    - **Glass_BumpBinormal_Texture**: Is a technique to use a diffuse color texture and bump mapping on the surface of geometry with texture coordinates and normals.

Glass Shader Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material and a texture as a color/basic texture. Do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess, to achieve the required look for your surface.

15.7.12 Gooch Shader

Gooch shading is used to calculate the lighting of the surface. Accordingly, a warm color is shown in lit areas while a cold color is used in unlit areas. A highlight is further added. Besides, a color texture may be used to add more details. This shader is particularly useful when no surrounding reflection is required.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Gooch Shader Properties
- Gooch Shader Best Practices

Gooch Shader Properties

- **Cold Color**: Assigns the color used for unlit surfaces.
- **Warm Color**: Assigns the color used for lit surfaces.
- **Back Splash**: Controls a factor to increase the perception of curvature on grazing angles. The resulting color is weighted according to the back splash factor.
- **Texture Alpha**: Defines to which amount the diffuse color texture is blended into the resulting color.
- **Highlight Alpha**: Defines to which extent the calculated highlight is blended into the resulting color.
- **Color Map 2**: Offers the possibility to assign a second diffuse color texture that is blended with the first color texture (according to the chosen technique).
- **Technique**: Shows a list of available techniques.
  - **GoochShading**: Is a technique to use Gooch shading.
  - **GoochShading_Texture**: Is a technique to mix a diffuse texture color into the material color.
  - **GoochShading_MultiTexture_Add**: Is a technique to additively blend Color Map 2 with the texture.
  - **GoochShading_MultiTexture_Blend**: Is a technique to blend Color Map 2 with the texture according to their alpha values.
  - **GoochShading_MultiTexture_Subtract**: Is a technique to subtractively blend Color Map 2 with the texture.
  - **GoochShading_MultiTexture_Modulate**: Is a technique to blend Color Map 2 by multiplying it by the texture.

Gooch Shader Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material. Therefore, do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess. When applying a texture technique it is compulsory to assign an texture as a color/ basic texture. In case of multi texturing an additional, second texture must be assigned in the corresponding Shader rollout.

15.7.13 Lacquered Surfaces Shader

The RTT Lacquered Surface shader simulates smooth lacquered surfaces lit by a highlight. On the clear coat layer, the surrounding is reflected (from a cube map). The reflection blurriness can be controlled to create the appearance of a rougher surface.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Lacquered Surfaces Shader Properties
- Lacquered Surfaces Shader Best Practices

Lacquered Surfaces Shader Properties

- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.
- **Reflection Multiplier/Exposure**: Affects the exposure of the environment map. This is especially helpful when working with HDR images.
- **Gamma of EnvMaps**: Controls the gamma correction of the environment map. This is once again particularly helpful when working with HDR images.
- **Blur Reflection**: States how much the environment reflection is blurred to create the impression of a rougher surface.
- **Fresnel Minimum, Fresnel Scale and Fresnel Exponent**: Affects the Fresnel effect upon the reflection.
- **Environment Map**: Is a cube map that describes the environmental surrounding used to calculate reflections.
- **Technique**: Shows a list of available techniques.
  - **PixelShading**: Is a technique to be used for finer details on the surface.
  - **VertexShading**: Is a technique to be used for objects in the distance or for highly tessellated objects.

Lacquered Surfaces Shader Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material and a texture as a color/basic texture. Do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess, to achieve the required look for your surface.

15.7.14 Metal Reflection Shader

The object is lit by a highlight and by the reflection of the surrounding (from a cube map). Both highlight and reflection take into account the material color of the object to generate a realistic metal effect.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Metal Reflection Shader Properties

- Metal Reflection Shader Best Practices

Metal Reflection Shader Properties

- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.
- **Reflection Multiplier/Exposure**: Affects the exposure of the environment map. This is especially helpful when working with HDR images.
- **Gamma of EnvMaps:** Controls the gamma correction of the environment map. This is once again particularly helpful when working with HDR images.
- **Environment Map**: Is a cube map that describes the environmental surrounding used to calculate reflections.

Metal Reflection Shader Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material and a texture as a color/basic texture. Do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess, to achieve the required look for your surface. To achieve a photo realistic, metallic look, it is recommended to assign any color but white as a specular color.

15.7.15 Microstructure Shader

A detailed surface structure is simulated by bump mapping and lit with the aid of a highlight. Additionally, a clear coat layer reflects the surrounding (from a cube map). A normal map is used to define the surface structure.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Microstructure Shader Properties
- Microstructure Shader Best Practices

Microstructure Shader Properties

- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.
- **Reflection Multiplier/Exposure**: Affects the exposure of the environment map. This is especially helpful when working with HDR images.

- **Gamma of EnvMaps:** Controls the gamma correction of the environment map. This is once again particularly helpful when working with HDR images.
- **Bump Map / Normal Map:** Is the texture that defines the surface structure with encoded normals.
- **Environment Map:** Is a cube map that describes the environmental surrounding used to calculate reflections.
- **Technique**: Shows a list of available techniques.
  - **Standard_Bump**: Is a technique to be used with geometry that has no standard UV texture coordinates and normals.
  - **Binormal_Bump**: Is a technique to be used with geometry having texture coordinates and normals.

Microstructure Shader Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material and a texture as a color/basic texture. Do not forget to adjust also the material parameters like ambient, diffuse, specular color and shininess, to achieve the required look for your surface

15.7.16 Monitor Shader

The RTT Monitor shader simulates the appearance of a flat screen display where saturation decreases at grazing angles of view. Therefore, the shown color texture blends into the diffuse color of the material. In addition, a reflection from the surrounding (from a cube map) is added, increasing in strength towards more acute angles of view.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Monitor Shader Properties
- Monitor Shader Best Practices

Monitor Shader Properties

- **Saturation/Reflection/Color Gradient:** Defines to which amount the material's diffuse color is mixed into the resulting color at grazing angles.
- **Environment Reflectivity**: Manipulates the influence of the reflection color from the environment map on the resulting color.
- **Reflection Multiplier/Exposure**: Affects the exposure of the environment map. This is especially helpful when working with HDR images.

- **Gamma of EnvMaps:** Controls the gamma correction of the environment map. This is once again particularly helpful when working with HDR images.
- **Environment Map**: is a cube map that describes the environmental surrounding used to calculate reflections.
- **Technique**: Shows a list of available techniques.
  - **Saturation_and_Reflection_Gradient**: Is a technique to influence saturation and reflection appearance alike.
  - **Reflection_Gradient**: Is a technique to influence only the reflection appearance.

Monitor Shader Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material and a texture as a color/basic texture. For this shader, a background image should be selected as the basic texture to achieve a highly realistic monitor effect. Do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess, to achieve the required look for your surface.

15.7.17 Velvet Shader

A highlight along the edges of the surface is created to generate velvety impression. The highlight is controlled in using two parameters that describe the transition towards diffuse lighting. Furthermore, a surface color can be added to brighten the barely lit areas. A color texture may be used to add more surface details. This shader is particularly useful when no surrounding reflection is required.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> RTTAdvancedMaterials

This section contains information on the following topics:

- Velvet Shader Properties
- Velvet Shader Best Practices

Velvet Shader Properties

- **Surface-Color**: Assigns the color of the surface (underneath the velvety hairs).
- **Specular-Color**: Assigns the specular color of the material.
- **Edge Rolloff**: Defines how much of the surface color is to be seen in barely lit areas.
- **Specular low-cut and specular high-cut**: Defines the sharpness of the velvety highlight.
- **Color Map 2**: Offers the possibility to assign a second diffuse color texture that is blended with the first color texture.
- **Technique**: Shows a list of available techniques.
  - **Velvet**: Is a technique to use velvety shading,

- **Velvet_Texture**: Is a technique to mix a diffuse texture color into the material color.
- **Velvet_MultiTexture_Add**: Is a technique to additively blend Color Map 2 with the texture.
- **Velvet_MultiTexture_Blend**: Is a technique to blend Color Map 2 with the texture according to their alpha values.
- **Velvet_MultiTexture_Subtract**: Is a technique to subtractively blend Color Map 2 with the texture.
- **Velvet_MultiTexture_Modulate**: Is a technique to blend Color Map 2 by multiplying it by the texture.

Velvet Shader Best Practices

Apart from the parameters in the plug-in container, it is necessary to assign a material. Therefore, do not forget to adjust material parameters in addition, such as ambient, diffuse, specular color and shininess. When applying a texture technique it is compulsory to assign an texture as a color/ basic texture. In case of multi texturing an additional, second texture must be assigned in the corresponding Shader rollout.

15.8 Texture

The following shader plug-ins are located in the Texture folder:

- Drop Shadow
- Emboss
- MultiTexture
- Substance

15.8.1 Drop Shadow

Drop Shadow is a shader plug-in that generates 2D shadow of a texture.

⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Texture

This section contains information on the following topics:

- Drop Shadow Properties

Drop Shadow Properties

- **Distance**: Sets the distance of the shadow.
- **Direction**: Sets the direction of the shadow in degrees.

- **Threshold**: Sets the shadow threshold.
- **Soft Shadow**: Enables the Soft Distance parameter for applying a soft shadow.
  - **Soft Distance**: Sets the distance of the soft shadow.
- **Color**: Sets the color of the shadow.

15.8.2 Emboss

Emboss is a simple shader plug-in that allows you to raise highlighted surfaces and lower shadows of textures and images creating an embossed look.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Texture

This section contains information on the following topics:

- Emboss Properties

Emboss Properties

- **Direction**: Sets the direction of the emboss. Options are:
  - None
  - South-East
  - South-West
  - North-East
  - North-West
- **Emboss**: Sets the Emboss level, in percentage.

15.8.3 MultiTexture

The MultiTexture shader plug-in blends (linear) two textures together.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Texture

This section contains information on the following topics:

- To Create a Simple Blended Image Scene
- Blended Images Example

To Create a Simple Blended Image Scene

1. Create a new Container.

2. Add an image to the new Container.
3. In the Image Editor set the image to:
    - Unit: Level one
    - Inheritable
4. Create a Sub Container.
5. Add an image to the Sub Container.

6. In the Image Editor set the image to:
    - Unit: Level two
7. Add the MultiTexture plug-in to the Sub Container.
8. Click on the MultiTexture plug-in.
9. Adjust the image blend as required.

Blended Images Example

15.8.4 Substance

This section details the Substance Shader plug-in, which renders *Allegorithmic PBR* (Physically Based Rendering) substances in Viz Engine.

The plug-in takes into account the position, color and enabled/disabled state of the first eight Lights in the Scene. It also supports materials with alpha properties.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Shader -> Texture

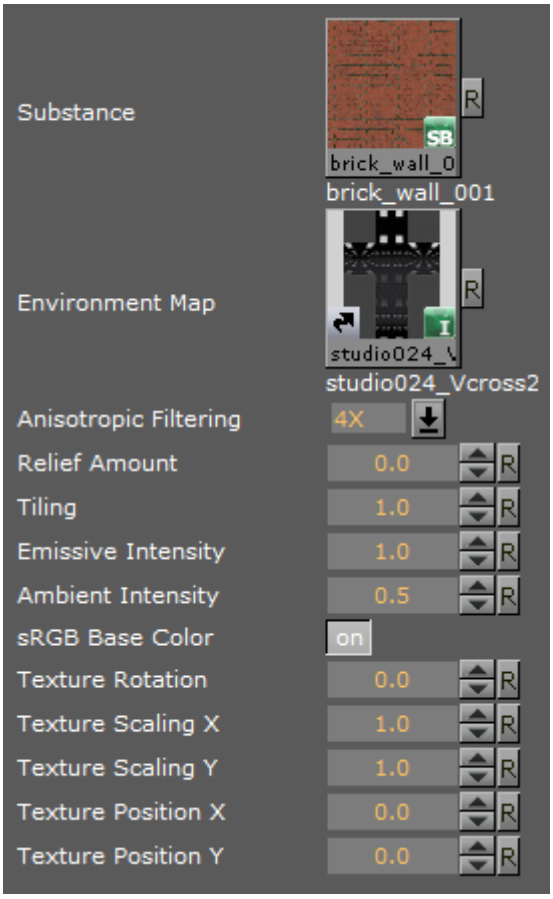
This section contains information on the following topics:

- Substance Properties
- To Create a Substance Shader Effect

Substance Properties

> ⚠ **Note:** The Substance properties panel parameters are different for each selected Substance.

The following options are available in the Basic Tab of the Substance Shader:



The substance settings depend on the settings the creator of the material has exposed:

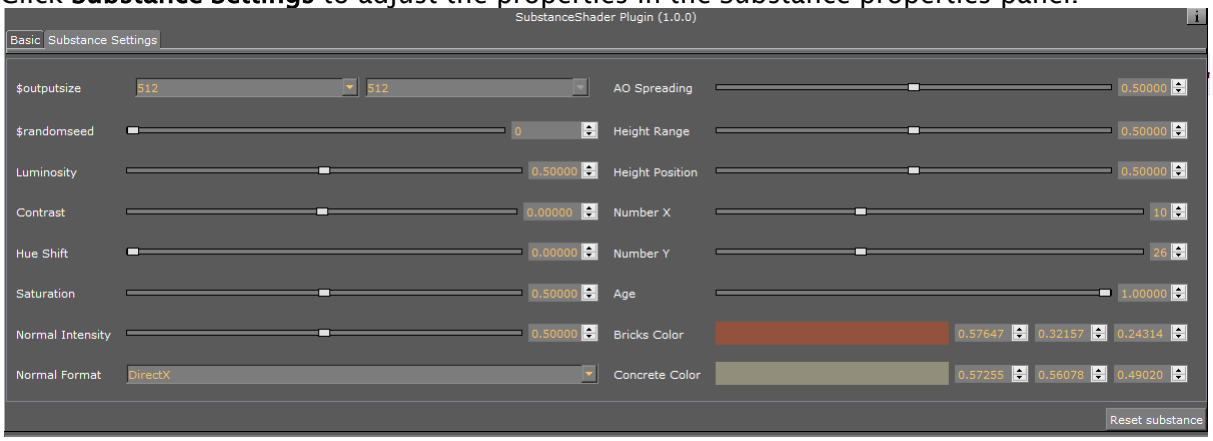
- **Substance**: Uses a PBR Substances item (.sbsar file). Drag to the drop zone.
- **Environment Map**: Uses a latitude/longitude panorama environment map image. Drag to the drop zone.
- **Anisotropic Filtering**: Sets the image quality. The default value is `4X`.
- **Relief Amount**: Sets the relief amount that the shader applies.
- **Tiling**: Sets the tiling.
- **Emissive Intensity**: Sets the intensity of the emissive properties (if the substance has any).
- **Ambient Intensity**: Sets the ambient light taken into account by the shader.
- **sRGB Base Color**: Converts the base color map to sRGB (default: on).
- **Texture Rotation**: Sets the rotation of the texture on the geometry.
- **Texture Scaling X/Y**: Sets the scaling of the texture on the geometry.
- **Texture Position X/Y**: Sets the position of the texture on the geometry.

To Create a Substance Shader Effect

1. Create a new Container.
2. Add a **Cube** geometry.
3. Add the **Substance Shader**.
4. Drag a **Substance** to the **Substance** drop zone.

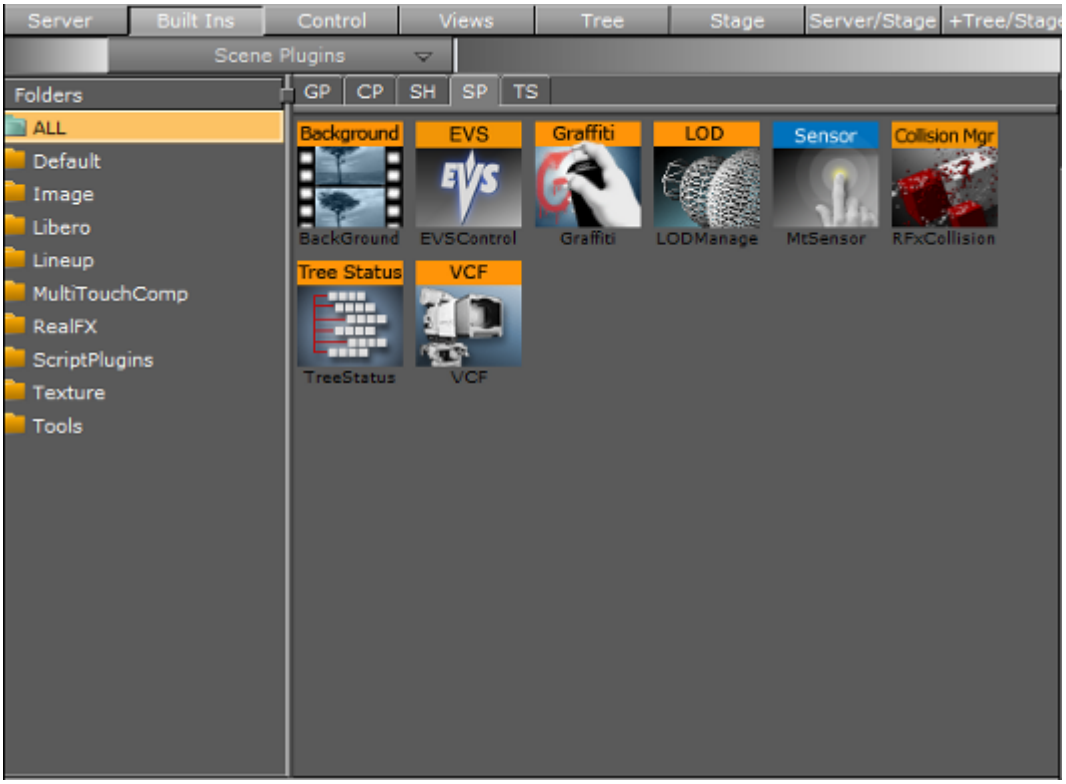
5. Drag a latitude/longitude panorama image to the **Environment Map** drop zone.

6. Use the Substance Shader properties to modify the result, if required.
7. Click **Substance Settings** to adjust the properties in the Substance properties panel.



16 Scene Plug-Ins

Scene plug-ins are global functions that have influence on the whole Scene.They are mainly used when external programs are interfaced with Viz Artist.



The default path for the Scene plug-ins is `<viz install folder>\plug-in`

16.1 Default

The following scene plug-ins are located in the Default folder:

- VCF

16.1.1 VCF

The VCF scene plug-in works in conjunction with the VCF Parameter plug-in that allows you to create a seamless interpolated transition from a virtual camera flight (VCF) to a real camera - and conversely. This is only a relevant plug-in to set up if you have purchased the virtual studio expansion components.

You must have a real camera with data tracking enabled which is set in remote mode in the camera editor. In case you have several tracked cameras, the virtual camera will interpolate its position to the real camera that is selected on air.

⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> Default

This section contains information on the following topics:

- Properties
- To Animate a Virtual Camera Flight

Properties

- **Tracking Object (POS)**: Defines the position or eye-point of the virtual camera flight. Use the name of the container in which center the camera should be placed.
- **Tracking Object (ROT)**: Defines the direction of the virtual camera flight. Use the name of the container the camera will be looking at.
- **Initialize (button)**: Places the camera in position and sets the direction.

To Animate a Virtual Camera Flight

1. Start by adding the VCF scene plug-in under the Video Clip Playback Considerations plug-in tab.
2. Create a new group in your scene and add two new containers under it.
   - These are to be the objects that will define the virtual camera flight. One will define the position, the other the direction.
1. Name the containers according to the names you entered in the VCF scene plug-in (for example T_POS and T_ROT).
2. Click Initialize to finish.
3. Add the VCF Parameter plug-in onto the container that holds the position object.
4. Animate your virtual camera flight using the two objects to define position and direction.
   - Do **not** switch the whole container invisible because the animation will not run.
   - You may switch the objects to be invisible at any time.
   - It is not possible to animate the roll of the camera.

16.2 Image

The following scene plug-ins are located in the Image folder:

- Background Clip

16.2.1 Background Clip

This plug-in is designed to play back a sequence of still images (tga, tiff, etc.) in the background of your scene rather than playing an AVI-file.

Since it plays the sequence from memory, the scene has a higher loading time and the drawback is that it consumes system memory which can influence the system stability. So keep track of how much memory each sequence needs so you do not run out of memory on the render engine. Scene plug-ins are added under the Scene Settings plug-in tab.

Since all images are loaded into memory a large number of images or a large image size would require large amounts of memory. Memory can be calculated as follows:

```
number of images * image width * image height * 3 (or 4 for alpha).
```

⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> Image

This section contains information on the following topics:

- Background Clip Properties
- To Add a Background Image Clip

Background Clip Properties

- **Image**: Allows you to enter the name of the image sequence or browse for it. Select the first image you would like to use in your clip. Do not use clip names with numbers (except the counter).
- **Play Mode**: Can be set to
  - **Once:** Plays the clip once.
  - **Loop:** Repeats the clip over and over again.
  - **Swing:** Plays the clip forwards and backwards in a loop.
- **Reverse**: Plays the clip backward when enabled.
- **Play speed**: Lets you alter the speed in percent.
- **Advanced**: Opens advanced parameters if set.
- **Stand alone image**: Needed when you want to play the same clip in different speeds.
- **First image**: Select the number of the image which you want to start with.
- **Nof images**: It stands for “number of images”. Please type in here the number of images you want play as a clip, otherwise all images will be played back.
- **Crop**: Crops the image in percent from the left, right, bottom and top side.
- **Scale**: Scales all images to the closest power of 2. When not selected an automatic textures coordinates will be applied. So the image will fit the texture.
- **Keep under**: Forces the image size. For example if you have selected 64, the image will be trimmed to the size 64 x 64.
- **Format**: Loads the file in original condition if **Image** format is selected. Please select another format if you want load a black and white image (“Alpha”) or full color image (“RGB”, “RGBA”: with alpha value).
- **Play, Stop and Reinitialize**: Enables you to play, stop and reinitialize the clip.

To Add a Background Image Clip

1. Create a directory and populate it with image files, for instance `abc001.png`, `abc002.png`, `abc003.png`, etc.

- The images will be played as a sequence based on the filename including numbers.
- After you have created the images, load the first image of the sequence and the clip is now visible in the Scene Editor.

1. Another possibility is to create a file with a "*.vln" extension. This file includes the base path and also the names of the images to load. In this case the images must not have a counter number in their filename. You can handle this file as an ordinary text file.
2. Load the vln-file instead of loading an image file located in a directory. Example:

```
BASE_PATH 'C:/clip/images' {
'radar_200504110800.png' 2005_04_11_10:00
'radar_200504110815.png' 2005_04_11_10:15
'radar_200504110830.png' 2005_04_11_10:30
'radar_200504110845.png' 2005_04_11_10:45
'radar_200504110900.png' 2005_04_11_11:00
'radar_200504110915.png' 2005_04_11_11:15
'radar_200504110930.png' 2005_04_11_11:30
'radar_200504110945.png' 2005_04_11_11:45
}
```

See Also

- Video Clips
- Image Clip

16.3 Libero

The following Scene plug-in is located in the Libero folder:

- EVSControl plug-in

The plug-ins in this folder are inactive by default. To view the folder and use the plug-ins, set all plug-ins to active in **plug-ins** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

16.3.1 EVSControl plug-in

The EVSControl plug-in allows an EVS video server to be controlled from the Viz Engine (like a tape deck over RS422). This gives the ability to load and control playback of EVS video server clips.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> Libero

This section contains information on the following topics:

- EVSControl Properties
- Clip Loaded Successfully
- To Go to a Specific Time Code
- To Get the Current Time Code
- Example Scenes

EVSControl Properties

Various EVSControl plug-in settings can be modified in Viz Artist from Scene plug-in pane:

⚠ **Note:** Do not change OUT parameters manually or using a script.

- **Debug:** Enables/disables extended debug output on the console.
- **Serial Port (COMx):** Sets the serial port to be used for controlling the EVS.
- **Initialize:** Initializes the Controller. Without initialization, the plug-in does not work.
- **EVS Clip ID:** The ID of the EVS clip to be loaded. The value must match the format “PBCK/MM” where:
  - P = Page number (1-9)
  - B = Bank number (1-9)
  - C = Clip number (1-9)
  - K = Camera letter (A-F)
  - MM = EVS Machine number (`00`, `01`, etc.), where `00` is the local EVS.
- **Load Clip:** Loads the clip shown in the “EVS Clip ID” field.
- **Clip loaded:** Shows when the Load Clip was clicked to confirm the clip was loaded.
- **Clip IN:** Sets the IN time code of the successfully loaded clip (Format hh:mm:ss:ff).
- **Clip OUT:** Sets the OUT time code of the successfully loaded clip (Format hh:mm:ss:ff).
- **Clip Duration:** Sets the duration of the successfully loaded clip in milliseconds.
- **Stop:** Stops play back.
- **Play:** Starts play back.
- **TC Hour:** Sets the time code hour value for the “Goto TC” action.
- **TC Minute:** Sets the time code minute value for the “Goto TC” action.
- **TC Second:** Sets the time code second value for the “Goto TC” action.
- **TC Field:** Sets the time code frame value for the “Goto TC” action.
- **Goto TC:** Moves to the time code shown by the above fields.
- **Get TC:** Gets the current timecode.
- **Current TC:** Retrieves the value of the “Get TC” action.

Clip Loaded Successfully

If a clip has been loaded successfully, the EVSControl plug-in editor will show:

- the ‘Clip loaded (out)’ flag is set.

- the ‘Clip IN (out)’ shows the Clip IN timecode.
- the ‘Clip OUT (out)’ shows the Clip OUT timecode.
- the ‘Clip Duration (out)’ shows the Clip length in milliseconds.
- EVS goes to IN timecode of clip.

To Go to a Specific Time Code

1. Enter the hour, minute, second, and frame in the fields:

- TC Hour
- TC Minute
- TC Second
- TC Field

1. Click on the **Goto TC** button

To Get the Current Time Code

1. Click on the **Get TC** button
2. The current Time Code is shown in the Current TC field.

Example Scenes

The plug-in comes with two example scenes:

- One shows a slider to scrub a clip
- The other shows scanning for clips and creating clip buttons for playback control

---

16.4 Lineup

The following scene plug-ins are located in the Lineup folder:

- Tree Status

16.4.1 Tree Status

The Tree Status plug-in is needed by the Lineup template in Viz Pilot to collect information about the scene hierarchy.

The Tree Props container plug-in is required for use of the Tree Status plug-in. Move the Tree Props plug-in onto the group holding the transformation which is to be controlled by Viz Pilot. Viz Pilot will then be able to build its own internal tree properties list.

⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> Lineup

This section also contains information on the following topics:

- Tree Status Properties

Tree Status Properties

- **Result:** Contains the entire information about the complete scene tree.
- **S2VResult:** Holds the information X-, Y-, Z-translation and the Y-rotation given in the card file virtual Studio. S2V stands for *scene to virtual*.
- **Command:** Allows a Viz command to take place.
- **Initialize (button):** Initializes the *Result* field.
- **InitializeS2V (button):** Initializes the *S2VResult* field.

16.5 MultiTouch Plug-Ins

The following Scene plug-ins are located in the MultiTouchComp folder:

- MtSensor plug-in

All the MultiTouch plug-ins Container plug-ins require the MtSensor plug-in to be available. If a MultiTouch plug-ins Container plug-in is added to a Scene it will automatically add the MtSensor plug-in to the Scene if it is not there.

The plug-ins in this folder are inactive by default. To view the folder and use the plug-ins, set all plug-ins to active in **Plug-ins** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).

16.5.1 MtSensor plug-in

The MtSensor plug-in manages global telestration and plug-in settings.

The MtSensor plug-in is the scene plug-in that must be available for any MultiTouch feature to work. It is automatically added to the Scene if any of the Container or Geometry plug-ins that require it are inserted in the Scene. It can also be dragged from the Server Panel under *Built Ins -> SP -> MultiTouchComp*, to the plug-in tab under Scene Settings.

The MtSensor plug-in also provides a screen-wide telestration facility. It can be controller through plug-in parameters and buttons that are available for scripting.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> MultiTouchComp

This section contains information on the following topics:

- MtSensor Configuration
- Events

MtSensor Configuration

- **Active:** Enables/disables telestration input from user.
- **Minimum Telestration Width:** Assigns minimum brush width based on pressure applied to sensor.
- **Maximum Telestration Width:** Assigns maximum brush width based on pressure applied to sensor.
- **Color:** Assigns a color for subsequent telestration drawing.
- **Id:** Provides additional context in the handler script, specify a string that will identify any notifications dispatched by this plug-in. This is often included as an argument for the event so a common script may handle events from a number of plug-ins.
- **Set Shared Memory:** Enables shared memory to be updated for the plug-in notifications if set to **On**.
- **Shared Memory Prefix:** Sets a 'prefix name' to be prepended to the shared memory variables maintained by the plug-ins notifications. For plug-ins that maintain multiple fields each field name has the prefix prepended to it followed by a dot, so as to mimic member access to an object. For example, if the prefix is 'Obj', the fields 'field1' and 'field2' would be identified with the strings 'Obj.field1' and 'Obj.field2'. The shared memory field 'Obj' is also maintained and is simply an integer that is modified every time any of its 'subfields' is updated.
- **Shared Memory Type:** Selects the shared memory area to update. Can be either **Global**, **Scene**, or **Distributed**.
- **SetDataPool:** Shows 'you wish' plug-in notifications to set a DataPool variable.
- **DataPoolVariable:** Shows the name of the DataPool variable one wishes to have set.
- **Init:** Clears current telestration and make Telestration active.
- **Clear:** Clears the contents of the attached telestration.

Events

The MtSensor plug-in does not generate any events for the scripting engine. The scene-wide telestration feature emits the same events that the MtTelestrator plug-in emits.

16.6 Script Plug-Ins

The Script plug-ins is a folder to save created Scene Script plug-ins.

A script can be saved as its own script plug-in, and used in future Scenes (see Create Script-based plug-ins). To save a Script plug-in, drag a compiled script into the Script plug-ins folder.

> ⚠ **Note:** Script plug-ins are saved to *<viz data folder>\Scriptplug-ins*.

See Also

- Script Editor

16.7 Texture

The following scene plug-ins are located in the Texture folder:

- Graffiti

16.7.1 Graffiti

The Graffiti scene plug-in allows telestration on top of scenes in the screen space. Telestration is done by drawing with a brush shape using a mouse or a 6DOF interface. The plug-in can also recognize some rendered shapes and replace the hand-drawn items with the recognized shapes (e.g. circle, ellipse, cross, arrow).

The graffiti scene plug-in is not intended for use on video output. To use graffiti textures on video output, use the GraffitiTex container plug-in.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> Texture

This section contains information on the following topics:

- Graffiti Properties
- To Create a Scene Level Graffiti Effect
- 6DOF (6 Degrees of Freedom)

Graffiti Properties

- **Active**: Enables/disables drawing.
- **Pointer Source**: Supports input from the mouse, or from another plug-in via a dispatcher interface. This button selects whether the plug-in will listen to the mouse or the dispatcher for input.
- **Mask Container**: Points to either a container or a group of containers. These containers mask the area over which the brush is rendered.
- **Alpha**: Adds transparency to the brush.
- **Key**: Selects whether the brush is rendered in the key signal
- **Brush Type**: Selects color or eraser brush.
- **Brush Image**: Determines the shape of the brush using an image (optional). If empty, a round brush will be used.
- **Brush Width**: Determines width of the brush in pixels. Visible only if the color brush is selected.
- **Eraser Brush Width**: Determines width of the eraser in pixels. Visible only if the eraser brush is selected.
- **Brush Color**: Selects the color of the brush.
- **Recognize Shapes:** Enables/disables shape recognition mode.

- **Recognize Ellipse**: Specifies whether shape recognition will try to recognize ellipse shape.
- **Recognize Circle**: Specifies whether shape recognition will try to recognize circle shape.
- **Recognize Cross**: Specifies whether shape recognition will try to recognize cross shape.
- **Shape Delay**: Sets number of frames to wait from mouse up before trying to recognize shapes.
- **Draw Arrow**: Specifies whether non-recognizable shapes will be converted to an arrow.
- **Arrow Length**: Determines length of arrow head.
- **Arrow Width**: Determines width of arrow head.
- **Circle Rad**: Sets radius of the circle replacing a recognized circle. If zero, the radius of the recognized circle will be used.
- **Cross Width**: Sets width of the cross replacing the recognized cross shape. If zero, the width of the recognized cross shape will be used.
- **Max. Undos**: Sets maximum number of undo operations.
- **Clear (button)**: Clears the canvas.
- **Undo (button)**: Undoes an operation.

To Create a Scene Level Graffiti Effect

- Add the plug-in to the Plug-in Panel in **Scene Settings**, set the plug-in properties, set Viz Artist in On Air Mode and start drawing.

6DOF (6 Degrees of Freedom)

6DOF is a tool for artists, script writers and plug-in developers to position and orientate objects in the 3D space (3 translation and 3 rotation axis => 6DOF).

6DOF works directly with the built in Scene-Grids (see Grid Tool-bar) and uses them to calculate the intersection of the projected user input (mouse or touchscreen position) with the current grid plane. As soon as a touch or click input is detected, it triggers the corresponding 6DOF callbacks (OnButtonDown6DOF, OnMove6DOF, etc...) and passes the calculated position and rotation.

For example: This could be used to move players on a football field interactively without doing any maths at all.

Additionally, 6DOF has a distribution mechanism which can send the user input to multiple engines, at the same time. This is configured in the **Global Input** of the Viz Configuration (see the Viz Engine Administrator Guide)

See Also

- Plug-in Panel
- GraffitiTex (container plug-in)

16.8 Tools

The following scene plug-ins are located in the Tools folder:

- Level Of Detail (LOD) Manager

- Scene Synchronized Properties

16.8.1 Level Of Detail (LOD) Manager

The LOD Manager scene plug-in works in conjunction with the Level Of Detail (LOD) container plug-in which is rendering objects with different details depending on the camera range; however, an integration with the LOD Manager is not required as the LOD plug-in is capable of initializing the parameter values by itself. The LOD Manager plug-in is used to change the visual view of the complete scene during rendering depending on the camera and scale settings.

> ⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> Tools

This section contains information on the following topics and procedures:

- Level Of Detail (LOD) Manager Properties
- To Use the LOD Manager plug-in

Level Of Detail (LOD) Manager Properties

- **Reference Zoom**: Sets the reference zoom that is used to correlate switching ranges based on the actual camera zoom value.
- **Range Scale**: Sets the global range scale which is used to scale all LOD ranges on all LOD plug-ins in the scene.
- **Reference Width**: Not used.
- **Reference Height**: Not used.

To Use the LOD Manager plug-in

1. Start by adding the LOD Manager scene plug-in under the Scene Settings' plug-in tab.
2. Add a group container to your scene tree and add at least two new sub containers of the group container.
3. Add a sphere geometry to the sub containers as it has many triangles.
4. Add a material to each sub container.
5. Open the Sphere editor for the all sub containers and set different Tesselation values for each to change the number of triangles and therefore the level of detail of an object.

   > ⚠ **Note:** All spheres should, for the sake of example, be placed in the same position.

6. Add the Level Of Detail (LOD) plug-in to the group container.
7. Open the LOD editor and change the Range parameter values provided by the LOD plug-in. For example set Range 0 to 500, Range 1 to 1000 and so on.

- These settings affects the instant of time when an object 1, object 2 will be rendered, which you have placed in step 3.
- The high resolution object is rendered at first and if the camera to object distance is increasing the next object will be rendered because now a low-resolution object has near the same visual quality despite fewer triangles. Consider that the visual quality depends on your settings and object graduation.

8. Finally, change the LOD Manager parameters Reference Zoom and Range Scale to adjust the scene view additionally.

16.8.2 Scene Synchronized Properties

⚠ **Note:** This plug-in is located in: Built Ins -> Scene plug-ins -> Tools

The Scene Synchronized Properties plug-in can be used to synchronize all property changes within the same scene loaded on a cluster of Viz Engines. Synchronization needs to be configured for each Viz Engine taking part in the clustered setup. This plug-in works on a per scene basis. A container plug-in is available, allowing synchronization to take place on a per container basis.

This feature works for all container properties and plug-in parameters, and requires that the Engines render the same scene, referencing the same UUIDs. This allows designers to work with different versions of the same scene, without any need to update the changed IDs in external applications. If a parameter change is transferred to another engine then the container UUID of the last loaded scene will be used for the update. If new UUIDs are required, a new scene will have to be created, either from scratch, or by merging the complete scene into a container which is then split into a new scene.

To Synchronize Scene Property Changes on Multiple Viz Engines

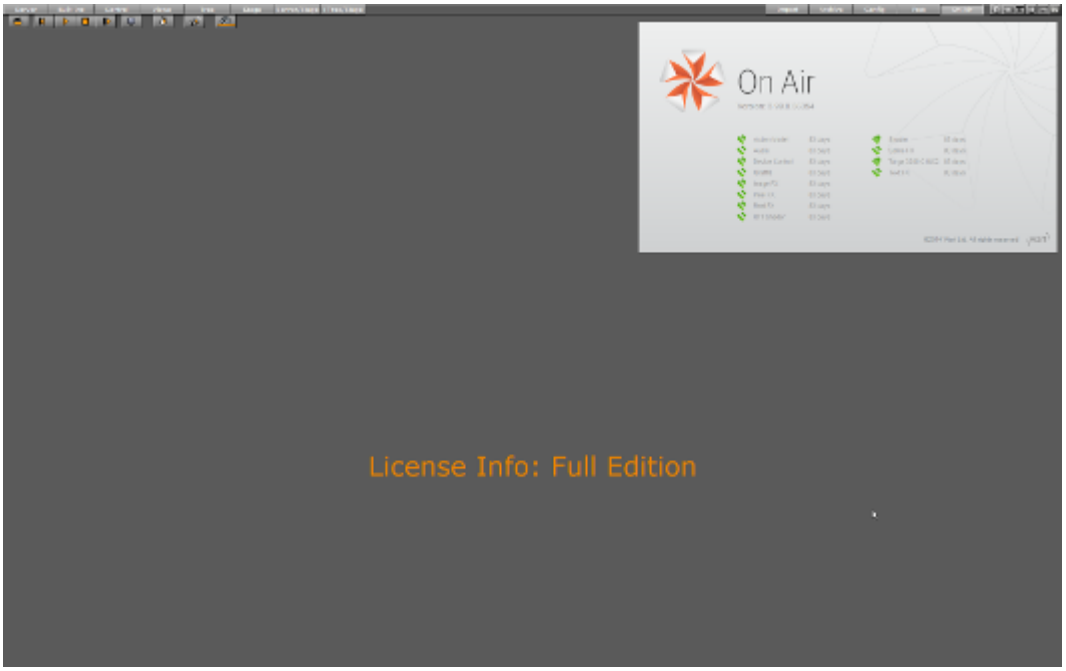
- Configure the involved Engines for synchronization. Please refer to the **Global Input** page in the **Configuring Viz** section of the Viz Engine Administrator Guide for further information.
- Create the scene that should be synchronized.
- Add the **Scene Synchronize Properties** plug-in to the Scene Setting's Plug-in Panel. No scripting is required for synchronization.

See Also

- Synchronized Properties Container plug-in
- For more information about how to synchronize multiple Viz Engines, see the **Global Input** page in the **Configuring Viz** section of the Viz Engine Administrator Guide.

17 On Air Mode

The On Air interface can vary, depending on the software and hardware configuration used. In Viz Artist, designers can click the On Air button on the main menu to switch Viz Artist from a modeling tool to a render engine. The application will then wait for control commands. However, scene animations can also be rendered by the use of Control Buttons (top-left corner).



The top left of the On Air screen shows a set of Control Buttons, and the Director Control Panel and Performance Editor buttons. All License Information is shown at the top right of the screen.

Depending on the software and hardware settings, additional buttons and information is available. For example in design mode, the scene will be shown in an output window (lower right).

All script events that are added to a Scene or to single Containers will be executed in On Air mode. To learn more about scripting see Scripting and Program Examples.

⚠ **Tip:** To prevent unexpected data loss when switching to **On Air** mode, Viz Artist automatically creates a backup of the active scene with the extension `_onair_backup`. If for example the scene is called *ABC*, the created backup will be called *ABC_onair_backup.*

This section contains information on the following topics:

- Control Buttons
- Director Control Panel
- Viz Artist Performance
- On Air Information
- License Information
- Lens File Editor

See Also

- On Air Shortcuts

17.1 Control Buttons

This section contains information on the Controls Buttons, which include the Play and the On Air buttons.

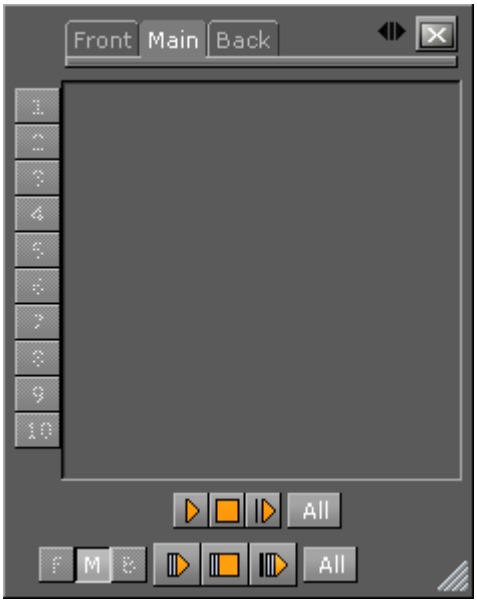
-  **Clapper Board**: Shows or hides the Director Control Panel window.
-  **Back**: Jumps to the beginning of the animation in the scene.
-  **Play**: Starts the animation of the scene.
-  **Stop**: Stops the animation of the scene.
-  **Continue**: Continues the animation after it stopped at a stop point.
-  **Render Preview**: Shows or hides the VGA Preview window (only available on machines with video cards). The keyboard shortcut for activating or deactivating this window is Shift+Backspace. The state of the VGA Preview window depends on how the **Render preview default setting** in the **Video Output** section of Viz Config is configured:
  - Inactive: Does not display by default.
  - Active: Displays by default.
  - Fullscreen: Sets the On Air window to screen size.

  > ⚠ **Note:** Preview is always enabled on Viz Artist machines without a video card.

-  **Performance Editor**: Shows or hides the Performance Editor.
-  **System Information**: Shows or hides the On Air Information window.
-  **Lens File Editor**: Shows or hides the Lens File Editor. Use to adjust the lens files for virtual studios. Adjust the field of view, lens distortion, nodal point and center shift. It is mainly used for lens calibration.

17.2 Director Control Panel

The Director Control Panel window can be used to select and animate one, multiple or all directors in the front, main or back layer. In addition it can be used to set slots and to animate a combination of director(s).



Click the  button, when in On Air mode, to open the Director Control Panel window.

See Director Control Panel for a detailed description of the Director Control Panel.

17.3 Viz Artist Performance

The performance of Viz Artist can be analyzed with either:

- The Performance Editor: This closely monitors a range of parameters to analyze the real-time performance of Viz Artist/Engine
- The Performance Analyzer: This monitors key performance and camera parameters as a head-up display in the renderer window. It also initiates the log of statistic, command and event information to the log file(s).
  This section contains information on the following topics:

- Performance Analyzer
  - Open the Performance Analyzer
  - Performance Analyzer Properties
- Performance Editor
  - Open the Performance Editor
  - Performance Editor Properties

17.3.1 Performance Analyzer

The performance analyzer enables key performance and camera information to be shown in the renderer view as a head-up display. Additionally the performance analyzer can be used to write statistics, command and event information to the log files.

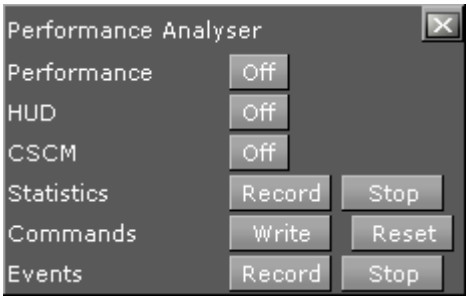
⚠ **Note:** Log files are located at *<viz data folder>*.

This section contains information on the following topics and procedures:

Open the Performance Analyzer

- Press and hold `<Ctrl>`, then right-click on the **X** (close) button in Viz Artist.

Performance Analyzer Properties

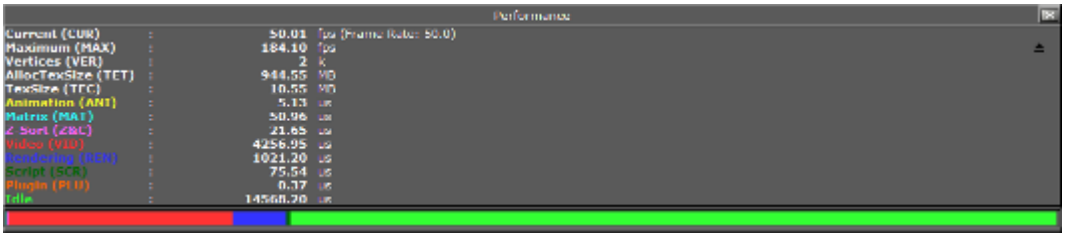


- **Performance**: Shows or hides the Performance Bar (Current (CUR) and Maximum (MAX) parameters) in the Scene Editor and Render Preview  .
- **HUD**: Enables the Head-up Display (HUD). The HUD shows the following parameters in the renderer view:
  - **Camera 1-n**: Shows the currently selected camera.
  - **Position**: Shows the camera's X, Y and Z position.
  - **Pan/Tilt/Twist**: Shows the camera's pan, tilt and twist parameters.
  - **FovX/FovY**: Shows the camera's field of view (fov) for the horizontal (X) and vertical (Y) plane.
  - **Center Shift**: Shows the X and Y position of the camera's center shift.
- **CSCM**: Shows the center shift as a cross hair in the renderer.
- **Statistics**: Writes statistics to the log file.
- **Commands**: Writes a command to the log file.
- **Events**: Writes events to the log file.

⚠ **Note:** These write to log file buttons are useful if there are ongoing problems with Viz Artist. Record and send the log file to Vizrt (see Customer Support Request).

17.3.2 Performance Editor

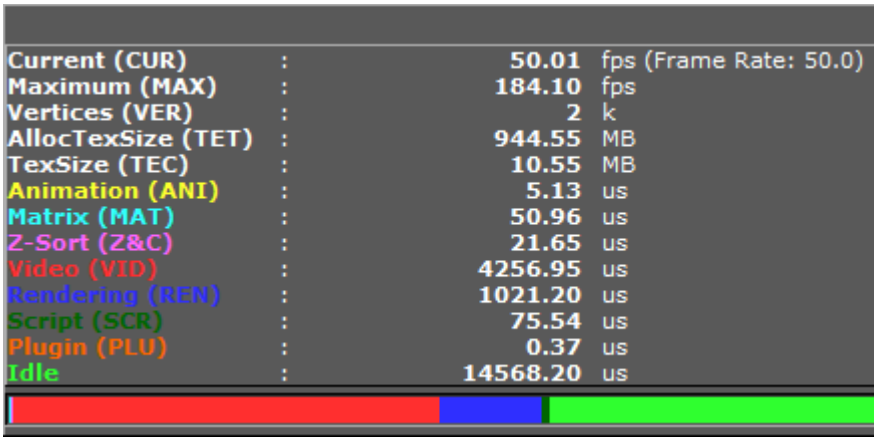
The performance Editor gives an idea of the current scene rendering performance (frames per second).



Open the Performance Editor

1. Click  .
2. Click  (Expand button) to show or hide the performance parameters.

Performance Editor Properties



Current (CUR)	:	50.01	fps (Frame Rate: 50.0)
Maximum (MAX)	:	184.10	fps
Vertices (VER)	:	2	k
AllocTexSize (TET)	:	944.55	MB
TexSize (TEC)	:	10.55	MB
Animation (ANI)	:	5.13	us
Matrix (MAT)	:	50.96	us
Z-Sort (Z&C)	:	21.65	us
Video (VID)	:	4256.95	us
Rendering (REN)	:	1021.20	us
Script (SCR)	:	75.54	us
Plugin (PLU)	:	0.37	us
Idle	:	14568.20	us

- **Current (CUR)**: Shows how many frames per second the scene will render at in On Air mode. The number should be above 50 (PAL) or 60 (NTSC), according to the rate that has been specified in **Output Format** (see the **Configuring Viz** section of the Viz Engine Administrator Guide).
- **Maximum (MAX)**: Shows how many frames per second the scene can render at without waiting for vertical retrace. The higher the maximum value, the more performance is left. If the maximum value is reduced to below 50 or 60, the scene is not rendering in real-time.
- **Vertices (VER)**: Shows the number of vectors in the scene.
- **AllocTexSize (TET)**: Shows the total allocated size of texture memory.
- **TexSize (TEC)**: Shows the size of the currently used texture memory.
- **Animation (ANI)**: Shows how many microseconds all active directors and animation channels take. This indicator is linked to the yellow bar.
- **Matrix (MAT)**: Transforms each container in the scene into world coordinate space. This indicator is linked to the cyan bar.
- **Z-Sort (Z&C)**: Refers to Z-sort and Culling, and sorts all containers for correct transparency drawing and determines if containers are visible in the current camera view. This indicator is linked to the pink bar.

- **Video (VID)**: Shows how many microseconds video input (live video texture) and video output take. De-interlaced video inputs take longer time than progressive and interlaced. The only way to improve this value is to use a faster system. This indicator is linked to the red bar.
- **Rendering (REN)**: Shows how many microseconds it takes to render all objects on the screen. A faster graphics card will improve this value. This indicator is linked to the blue bar.
- **Script (SCR)**: Shows consumed time in microseconds from all active scripts. This indicator is linked to the dark green bar.
- **plug-in (PLU)**: Shows time, in microseconds, all active plug-ins spend in each render cycle. This indicator is linked to the orange bar.
- **Idle**: Shows available resources, in microseconds, the renderer has available. This indicator is linked to the light green bar.

17.4 On Air Information

The On Air Information shows details about the currently running Viz Engine, and shows all connected clients with the IP address, host name and Viz Port. To show or hide the On Air Information panel, click  .

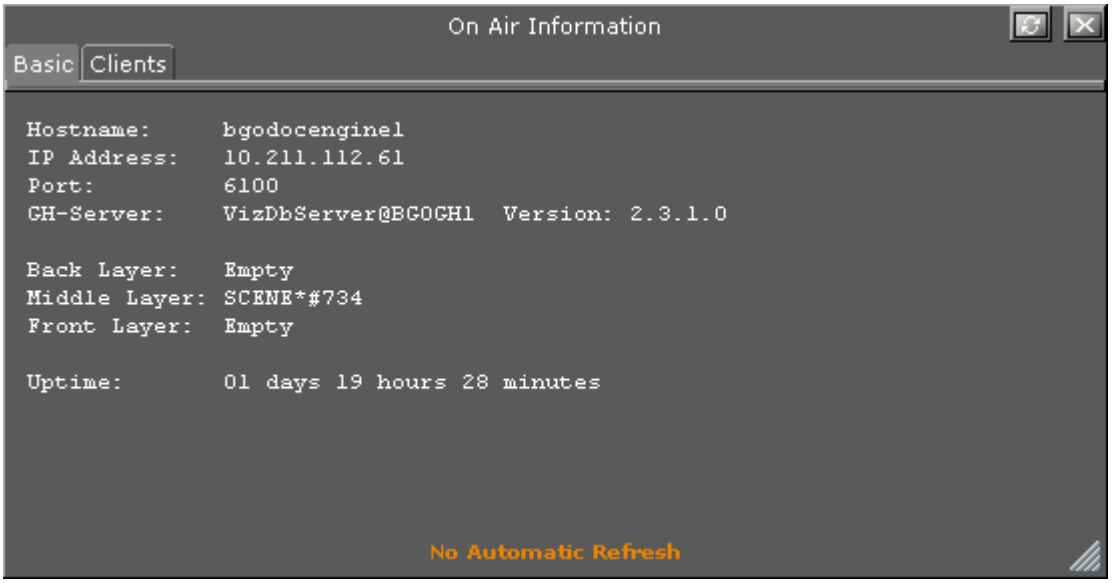
> ⚠ **Note:** Polling for On Air information can decrease the performance. For information on how to adjust the On Air Update Interval see User Interface in the Viz Engine Administrator Guide.

This section contains information on the following topics and procedures:

- Basic Tab
- Clients Tab

17.4.1 Basic Tab

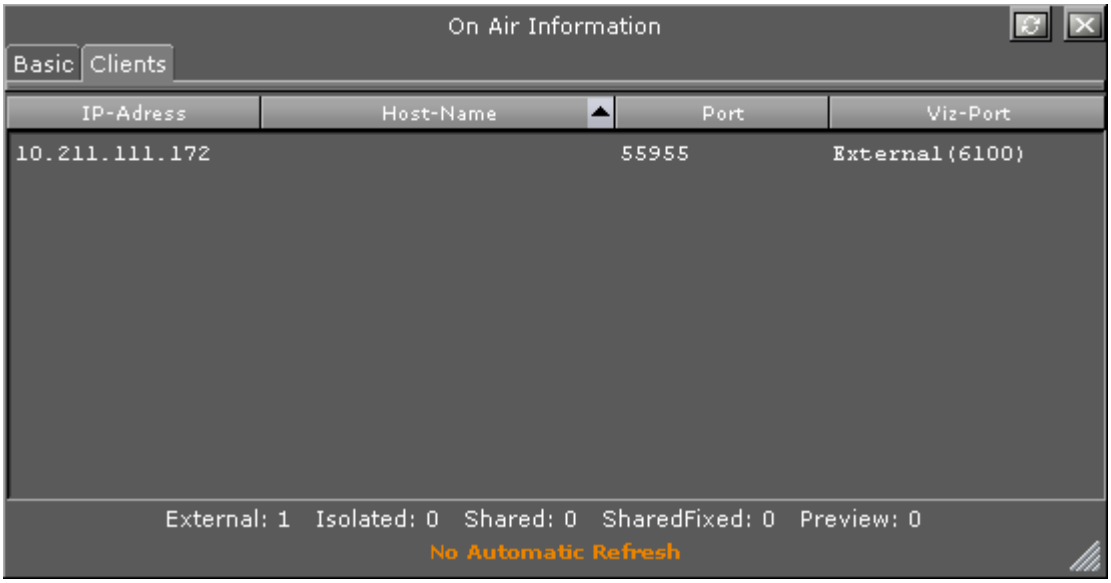
The **Basic** tab displays some basic information, such as name and IP address of the current host, database connection, and the current scene.



- **Refresh button**: Refreshes the status information.
- **Hostname**: Shows the name External Control Applications can use to communicate with Viz Artist.
- **IP Address**: Shows the IP address External Control Applications can use to communicate with Viz Artist.
- **Port**: Shows the port Viz Artist is using. Default port is 6100 (see the Viz Engine Administrator Guide).
- **GH-Server**: Shows the Graphic Hub server Viz Artist is connected to.
- **Back Layer**: Shows the name of the scene that is defined to run in the background layer.
- **Middle Layer**: Shows the name of the scene that is defined to run in the middle layer.
- **Front Layer**: Shows the name of the scene that is defined to run in the foreground layer.
- **Uptime**: Shows the time elapsed since Viz Artist/Engine was started.

17.4.2 Clients Tab

In the **Clients** Tab, all connected clients are shown, with the IP address, host name and Viz Port.



17.5 License Information

The license information panel shows the licensing information such as licensed features and how many days the license has left before it must be renewed.

17.6 Lens File Editor

The Lens File Editor is used to create and change lens files used by the Tracking Hub for virtual sets. Use the Lens File Editor to calibrate lenses by adjusting the field of view, lens distortion and nodal points for the cameras in use for a virtual studio set.

This section contains information on the following topics:

- Lens File Editor Workspace
- Lens File Editor Contextual Menu
- To Create a New Lens

17.6.1 Lens File Editor Workspace

The Lens File Editor workspace consists of two areas; the visualization area (1) and control area (2). The size of the Lens File Editor window can be adjusted by dragging the arrow symbol in the lower right corner of the window frame.

The control area is organized in several sections, for ease of use. Information on the current modes and selected properties is displayed in the far left side of the control area (A) at all times. To the right of this section, the different modes (B) can be set:

- **Cursor Visible/off (B):** Displays a cross-hair with guiding lines in the visualization area when in **Cursor mode** and **Cursor** is set to `Visible`.
- **Cursor mode/Tracking mode (B):** Uses guiding cross-hairs if set to `Cursor mode`, or use tracking position data received from the tracking system when in `Tracking mode`.
- **ProbMode On/Off (B):** Updates the **FoV Y** value automatically according to changes to the **FoV X** value if ProbMode is set to `On` when in free aspect ratio mode.

The next section (C) is used to select which parameter will be displayed in the visualization area (1):

- **K1/K2 (C):** Displays the current adjustments to the distortion values **K1** and **K2**, respectively. The K1 and K2 values are used to correct for lens distortion.
- **FoVX/FoVY (C):** Shows the Field of View controls, used to define the field of view for the selected camera, for the **X** and **Y** planes, respectively.
- **Cx/Cy (C):** Shows the center shifts when zooming out due to imperfect mount of lens on camera body. Corrections to the center shift are derived from the values given here for the **X** and **Y** planes, respectively.
- **Nodal (C):** Displays the configured nodal point for the current lens setup. The Nodal point changes when changing zoom and focus, and can change with as much as up to 20 centimeters for most broadcast lenses.

The middle section (D) is used to change the resolution and zoom level in the visualization area, as well as saving and loading lens files:

- **S+/S- (D):** Increases or decreases the value resolution of the visualization area (1).
- **Z+/Z- (D):** Zooms in to or out of the visualization area (1).
- **Save/Load File (D):** Saves the current setup to a new lens file or load a previous setup.

The next three sections (E, F and G) display the numeric representations of the current field of view, lens distortion, nodal point and linear scaling. The values are automatically updating when adjustments are made to the splines within the visualization area. Values can also be manipulated directly by clicking and holding the value to be adjusted, then dragging the mouse left or right to adjust the value up or down, respectively:

- **FovY/FovX:** Displays the current Field of View values for the X and Y planes.

- **K1/K2:** Displays the current lens distortion values.
- **Nodal:** Displays the current nodal point value.
- **Sc Y W:** Displays the linear scaling **Wide** value for the **Y** axis, for field of view adjustments.
- **Sc X W:** Displays the linear scaling **Wide** value for the **X** axis, for field of view adjustments.
- **Sc Y N:** Displays the linear scaling **Near** value for the **Y** axis, for field of view adjustments.
- **Sc X N:** Displays the linear scaling **Near** value for the **X** axis, for field of view adjustments.

The far right section (H) is used to select the aspect ratio for the current lens:

- **NONE**: Free aspect ratio.
- **16x9 y**: Sets aspect ratio to 16:9
- **16x 9 x**: Sets aspect ratio to 16:9
- **4x3**: Sets the aspect ratio to 4:3

17.6.2 Lens File Editor Contextual Menu

Click and hold the right mouse button anywhere in the visualization area of the Lens File Editor to bring up the contextual menu. Select an option by moving the mouse cursor and release the button.

- **Select Camera:** Up to 16 cameras can be configured in the same lens file. Select which camera to modify settings for.
- **Parameter:** Select which parameter to display in the visualization area:
  - **K1, K2:** Displays splines for the lens distortion parameters.
  - **FovX, FovY:** Displays splines for the field of view parameters.
  - **CenterX, CenterY:** Displays splines for the center shift parameters.
  - **Nodal:** Displays splines for the nodal point parameter.
- **Display Mode:** Select the display mode for the visualization area:
  - **Triangulation:** Shows the default view.
  - **Grid:** Displays a grid overlay.
  - **Cut Focus:** Displays lines to check the curve quality.
  - **Cut Zoom:** Displays lines to check the curve quality.

    ⚠ **Note:** The visualization area will not be automatically updated if making changes to parameters while **Display Mode** is in **Cut Focus** or **Cut Zoom** mode

- **Merge From**: Copies settings from another previously configured camera (1-16).
- **Create:** Creates a new raw lens with 3x3, 5x5, 7x7 or 9x9 reference points. Vizrt recommends starting with a 3x3 configuration, then using the Insert option to add more reference points first after successful 3x3 calibration.
- **Insert:** Inserts more reference points. Available insertions are:
  - 5x5
  - 7x7
  - 9x9

17.6.3 To Create a New Lens

1. In Viz Artist, click the **On Air** button.
2. Click the **Show Lens File Calibration Editor** button.
3. Right click in the visualization area and select **Create**, then **3x3**.
4. Select a parameter to bring up the spline editor for adjustments.

18 Transition Logic

Transition Logic is first and foremost a method of designing a graphics package that allows the designer to maintain the look and feel of the graphics while also allowing anyone else, such as journalists or producers, to add graphics items to a rundown without the need for any kind of technical knowledge of how the system works.

Transition Logic allows the independent control of any number of graphics "layers", providing a code-free and design-based method for building graphics that will gracefully animate in and out, and transition from one to another automatically. This is accomplished by using a *Master Scene* which coordinates the animation of independently controlled objects which make up the whole. The master scene commonly contains the background items of the graphics package. Such items can be looping backgrounds or the design items of the lower third, over the shoulders, and full-screen graphics. The *variable* or changing content, such as the text in a lower third, is stored separately in *Object Scenes*.

When a lower third is played on air, the object scene for the lower third is triggered. This tells the engine to load the master scene, place the object scene inside the master, and animate the appropriate timeline(s). Thankfully, all of that is handled automatically, which is what makes Transition Logic so valuable.

Transition Logic scenes are designed with a built in logic that allows for morphing background graphics, handles multiple layers of graphics, and manages the transition from the graphic to graphic of each layer. The built in logic also minimizes the operator's need to know how scenes affect each other, as the considerations of these various options are handled in the design phase. This ultimately results in a simplified work-flow and a look that is consistent and glitch-free.

To understand what Transition Logic is, one needs to understand a few basic concepts:

- A Transition Logic scene is not a single scene, but a set of Viz graphics scenes that consist of a *masterscene* that may have multiple "layers" of graphics that can be on air at the same time and independently controlled
- Each "layer" in the master scene may have multiple referring *object scenes*. However, only one object scene per "layer" can be active at any given time.
- Transition Logic "layers" are conceptual, not spacial.

> ⚠ **Note:** With Transition Logic scene design, *take in* and *take out* commands are still used as with standalone scene design. Where standalone scene design demands that only a single scene can be on air at a time, however, Transition Logic allows for more than one scene to be on air simultaneously. This means using Transition Logic will allow a graphic covering the lower third of the screen, and another covering the left and/or the right side of the screen for over the shoulder graphics, to be on air at the same time.

Transition Logic can be played out by most of Vizrt's control applications such as Viz Trio, Viz Pilot and Viz Multichannel.

This section contains information on the following topics:

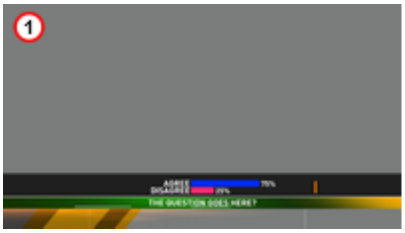
- Standalone versus Transition Logic Scene Design
- Toggle-Layer
- State Transition Animation
- Cross Animation

- Geometry Animation
- Master Scene
- Object Scene
- Tutorial

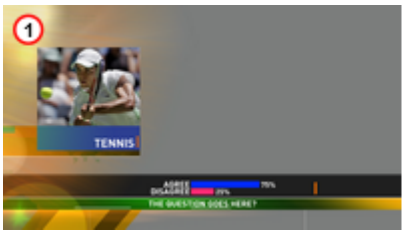
18.1 Standalone Versus Transition Logic Scene Design

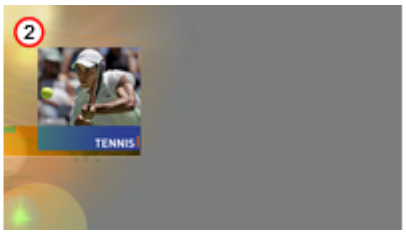
Unlike standalone scenes, a Transition Logic scene can have multiple defined "layers" that control the transitioning of multiple object scenes per layer, for example transitioning between variants of lower and top thirds. The following images shows the difference in results when executing commands towards Transition Logic scenes compared to standalone scenes. The images to the left (1) shows a Transition Logic scene with two "layers" and two object scenes, an Over the Shoulder and a Lower Third. The images to the right (2) shows two standalone scenes that are identical to the Transition Logic Object scenes.

18.1.1 Take Lower Third

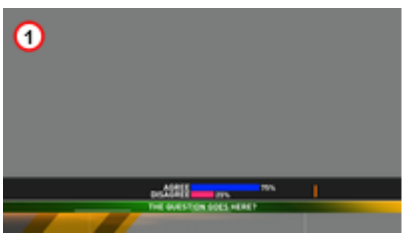


18.1.2 Take OTS

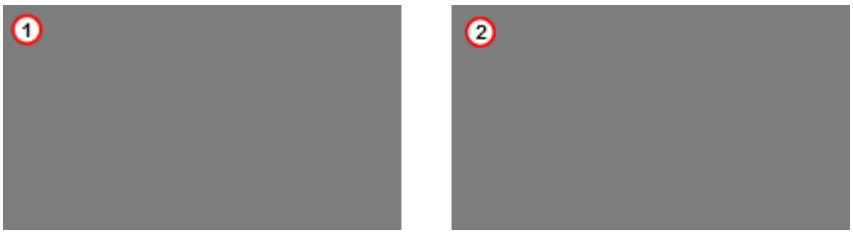




18.1.3 Take Out OTS



18.1.4 Take Out Lower Third

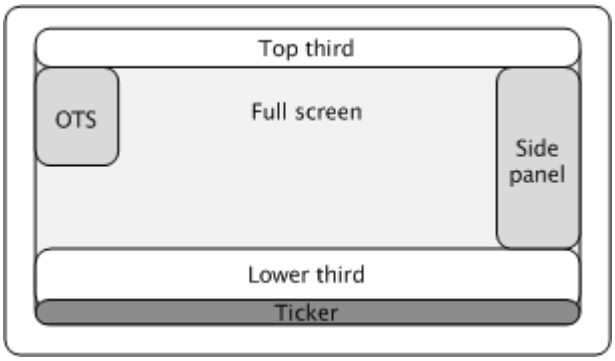


The first action, *take* lower third, results in a lower third graphics being placed on-screen. The second action, *take* over the shoulder (OTS), places the OTS on-screen; however, with standalone scene design this will also cause the lower third to be taken out, that is, a hard cut.

The third action, *take out* OTS, will take out the OTS; however, with Transition Logic scene design the lower third will remain on-screen until it is taken out (last action).

18.2 Toggle-Layer

The **Toggle** plug-in is used to establish a Toggle-layer in the master scene and acts as a placeholder for the contents of one or more graphical items. This toggle is used to establish the location of a placeholder for the insertion point of the object scene into the master scene. In other words, a Toggle-layer is actually a combination of this toggle placeholder and a linked **Director** on the **Stage**. The toggle container, together with a director of the same name, is what defines a layer.



Implementing multiple Toggle-layers creates the ability to place graphical items into several logical placeholders simultaneously, allowing multiple "layers" of graphics to be shown on-screen at the same time.

> ⚠ **Note:** Layer, in Transition Logic terms, does not lend itself to how the term layer is understood with most 2D applications. In Viz Artist, Toggle-layers, or more commonly just layers, refer to any segment of a graphics package which requires independent control.

A toggle-layer needs to be created for anything in the graphics package which requires independent control. Layers may or may not be allowed to be on-screen simultaneously. All toggle-layers reside in a common scene known as the master scene. This master scene therefore requires at least as many toggle-layers as there are independent items to be shown on screen at the same time, for example lower-third, over-the-shoulder, current time & temperature, etc. It may even be as complete as containing the additional toggle-layers needed to display the entire graphics

package. Additional graphics often include full-screens, a breaking news banner, and a locater, to name a few.

In the example above, it would be possible to place the top-third, lower third, and a news ticker in the same toggle-layer. However, only one of those items could have been on-screen at any given time. If the news ticker and the lower-third need to be on-screen at the same time they have to be in different toggle-layers. Creating placeholders for logical layers of graphics is done using the built in Toggle plug-in, which lends its name to the term toggle-layer. Incidentally, since it usually would not be useful to have both the top-third and lower-third on screen at the same time, as their use would normally be one or the other, it could be beneficial to place both the top-third and lower-third in the same toggle-layer, depending on the design.

Residing in different logical layers also means that their animations may affect each other. For example, having the lower third and news ticker on air at the same time may for instance push the lower third further up the screen to make room for the news ticker.

See Also

- Master Scene
- Toggle

18.3 State Transition Animation

As previously mentioned, the make-up of a toggle-layer in a Transition Logic scene is a relationship between the **Toggle** plug-in and a **Director** in the stage. Through the use of the Director's timelines, it is possible to animate between different variations in the design, called states. In order to make use of these states, each object scene refers to a specific toggle-layer and desired state.

18.3.1 Toggle layer states

Animating between different toggle layer states is called state transition animation. The example above shows a lower third's backplate being animated from a narrow version to a wide version (two states).

18.3.2 Defining States

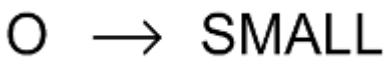
A toggle layer is required to have a default state named O. The O state is often referred to as the Out state as it is the state the toggle layer will be taken to when the operator issues a Take Out command. This means that in most cases, this is also the state where the toggle layer's referring foreground scene is no longer visible on-screen.

In addition to the O state, at least one additional state is required to run animations when taking items to air. Any state that is not the O state is often referred to as an *in state*, or *visible* state, because it most often refers to the state when a referring object scene is shown on-screen.

> ⚠ **Tip:** The master scene should be saved with the play head at the beginning of the timeline, which is equal to the O state. This will make sure that the scene is properly initialized in the O state.

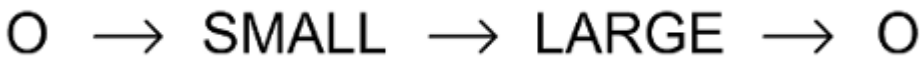
Use the following conventions when naming states:

- O states must be named in upper case only. This is a reserved state that is tied to the take-out command.
- It is recommended to name states according to the referring control scene's appearance or other distinguishing characteristic. For example, for a toggle-layer responsible for content displayed in the lower third of the screen that varies in size from large to small will typically have two states named `LARGE` and `SMALL`. A graphic placed over the shoulder of a presenter, where the graphics should switch between the left and right side of the screen, would simply be named `LEFT` and `RIGHT`, or `OTS_LEFT` and `OTS_RIGHT`, with `OTS` being short for Over The Shoulder.
- To maintain consistency between designers, it is recommended to write all state names using upper case characters only. This will also make it simpler to implement the state change in later steps.

  When taking an item which requires the SMALL state, the master scene will animate from the O state to the SMALL state, once that item is taken on air. When a take-out command is issued, the master scene will then animate in reverse to the O state.

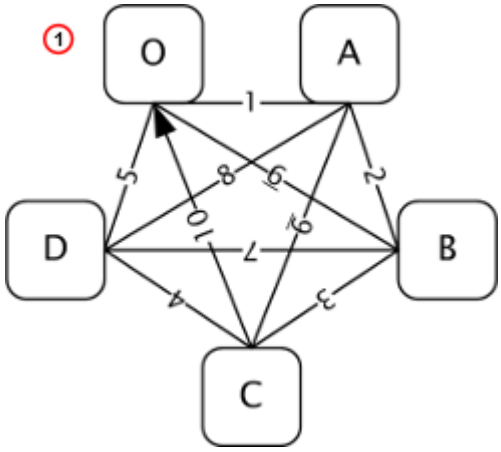
Transitioning from a SMALL to an O state is achieved by using the animation from the O to the SMALL state. In such a case the animation will be played in reverse. If this is not required, you simply add another O state after the SMALL state, and create a new animation.



If the layer should transition from the small sized version to the large version, you simply add the state to the toggle-layer director, and create a transition animation for the layer going from `SMALL` to `LARGE`. This also requires that the `LARGE` state has an `O` state such that the graphics can transition and animate out from the `LARGE` state.

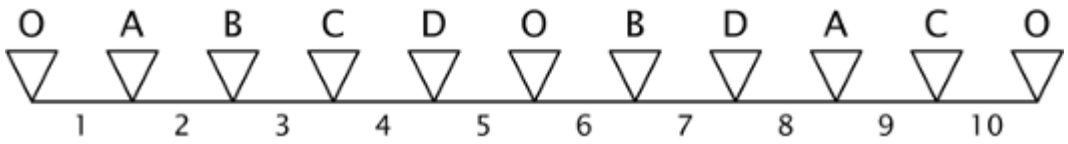
18.3.3 Identifying States

There are no hard limits to the number of states. However, increasing the number of states also increases the number of animations, and consequently the complexity of the scene. To decide the minimum number of animations required it is possible to do a Euler Walk (1).



The Euler Walk, although not always possible, is defined as the optimal route taken to visit each point exactly once. However, the theory can be applied to Transition Logic to identify the minimum required states needed to connect each state to the others exactly once.

Straightening the Euler walk out onto the layer's director will show the required transition animations:



⚠ **Note:** When performing transition animations from one state to the other Viz Artist/Engine will search from left to right on the layer's director for the correct combination of states. If the operator wishes to go from state A to B, Viz Artist/Engine will traverse the layer's director for state A and look to the right and see if it finds state B. If a match is found, Viz Artist/Engine will perform the transition animation.

Looking at the above example director, it becomes clear that not all combinations can be found by traversing from left to right. If a change from state A to D is not found, Viz Engine will first traverse from left to right, then start to backtrack and traverse from right to left looking for a D from A animation.

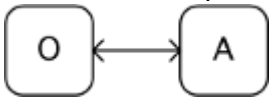
When the minimum required animations have been discovered, it is time to add extra animations required by the overall design. For example, it might be desirable to add a transition from state B to O, to animate the layer out from state B in a different way than reversing the in animation. As an example, think of a lower third graphics animating in from the left, stopping, then animating out to the bottom. Adding new out states quickly increases the complexity of a layer. As all states are required to be connected to each other, it would not be advisable to simply make an O state in place of the B to C animation.

Detailed below are the minimum required number of transition animations for a graphics layer with a given number of states:

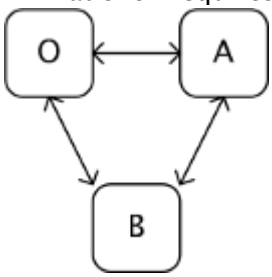
- **1 State**:

- Animations: Not possible.

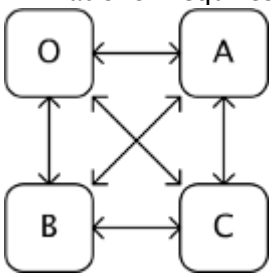
- **2 States**:
  - Animations: Requires a minimum of 1 animation



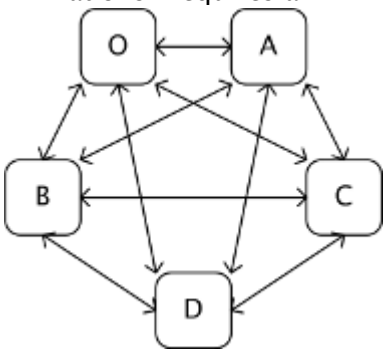
- 3 States:
  - Animations: Requires a minimum of 3 animations



- **4 States**:
  - Animations: Requires a minimum of 6 animations

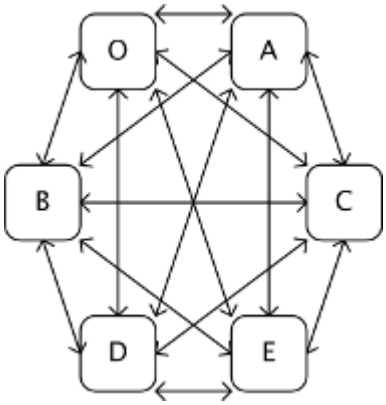


- **5 States**:
  - Animations: Requires a minimum of 10 animations



- **6 States**:

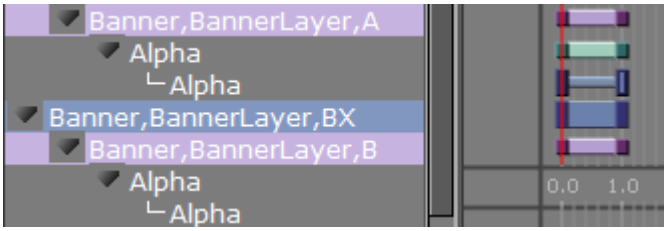
- Animations: Requires a minimum of 15 animations



18.4 Cross Animation

Cross animations are used in Transition Logic scene design to animate one object in and another out. The example below shows a lower third that has an alpha cross fade fading from a two lines of text to a one line of text.

Cross animation directors are added by the Toggle plug-in. The directors are always tagged with the letters AX and BX. AX and BX holds the cross animation and effect used to animate from one object to another. The default cross animation is a 1 second cross fade of the alpha value.



Animation and effect length can be changed by adjusting the director's animation and effect(s) on the stage. In addition it is also possible to remove or add effects (e.g. material, position and scaling).

See Also

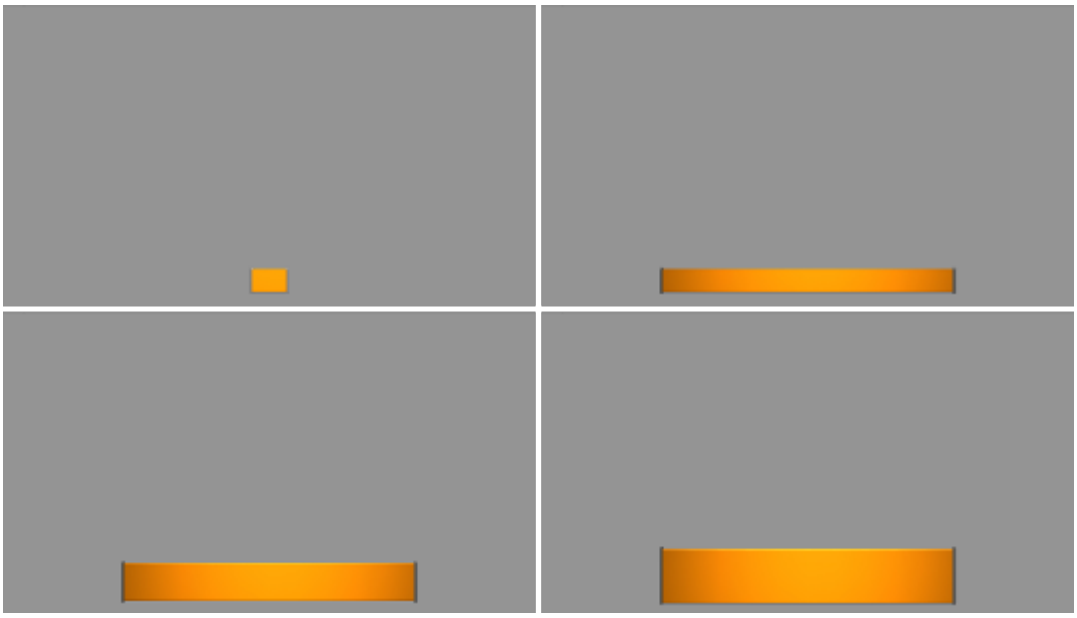
- Toggle

18.5 Geometry Animation

Geometry animations are used in master scenes. The animation itself is identical to an animation which is created for a standalone scene. Geometry animations are subject to the master scene,

which dictates when the animation of the object scene starts through the AO and BO directors. The geometries in the master scene are typically used as backplate graphics. Text or images would then be added on top of the backplate by the object scene.

The example below shows a lower third master scene, with animated geometries that form a simple lower third backplate:



18.6 Master Scene

The master scene serves several important functions. Most importantly it is the controlling part of a Transition Logic scene, and is essentially what makes standalone scene design different from Transition Logic scene design.

A Transition Logic master scene is:

- Constantly on air, and is used for controlling state transition animations on backplate graphics in combination with the cross animations of object scenes, such as toggling from one line to two lines of text.
- Able to replace a layer's referring object scene without taking other layers off air.
- Able to have graphical items such as backplates, which separates the background for object scenes.
  Having the master scene constantly on air means that when controlling the toggle layers, it can smoothly animate the backplate graphics from one size to another, providing an artifact-free backplate that will fit the changing requirements of the object scenes. With a geometry holding one to two lines of text, the text will as a bare minimum have a built-in cross-fade animation. Additionally, as the background scene is already on air, the transition animations on the background items will have no artifacts, or jittering, caused by graphics being loaded.

See Also

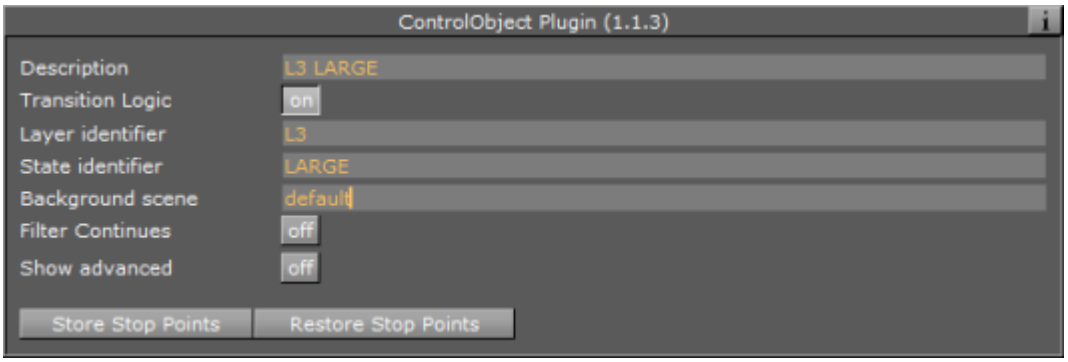
- State Transition Animation
- Cross Animation

- Object Scene

18.7 Object Scene

Transition Logic object scenes are similar to traditional standalone scenes that contain graphical items such as geometrical shapes and textures (for example images or video clips), but are different in that they are capable of affecting the state of the master scene's toggle layer, and typically only contain the variable data of a graphic such as the changeable text, while the design items are located in the master scene.

For object scenes to have an effect on the master scene's toggle layer, it has to refer to the master scene, the layer and the layer's state. This is obtained by adding the Control Object plug-in to the object scene's scene tree at root level.



When Control Object is added to an object scene, it must be Transition Logic enabled. This enables the object scene to set layer, state, and master scene parameters. When an object scene is taken on air, these parameters are sent as commands to Viz requesting the master scene to change the toggle layer's current state to the new state the object scene is referring to. If the object scene's Control Object parameters are incorrect, Viz will not be able to trigger the correct state transition animation, and the object scene will not be able to animate in.

Control Object also keeps track of other control plug-ins' properties that, when exposed, allow the operator to change their parameters (e.g. text size, color, geometry, texture, video clip and so on).

⚠ **IMPORTANT!** The ControlObject fields are case sensitive.

See Also

- Toggle-Layer
- Geometry Animation
- Master Scene

- Control plug-ins

18.8 Tutorial

This section contains information on the following topics:

- Create a Project
- Create a Master Scene
  - To Define the Toggle Layers
  - To Create a Toggle Layer
  - To Add Key
- Create an Object Scene
  - To Create Object Scenes
  - To Create a Background Image for the Object Scene
  - To Create a Preview Point

18.8.1 Create a Project

It is good practice to create a project with a folder for each new design. This will make sure that a project with more than one design does not get mixed up, and potentially overwrite other files.

18.8.2 Create a Master Scene

When creating a new Transition Logic scene design, it is common practice to first define the Toggle-Layers. After the toggle layers have been defined, one would typically move on to create the Master scene, and add the defined toggle layers.

⚠ **Note:** Creating the Master scene first is recommended, as it is needed to complete the object scenes.

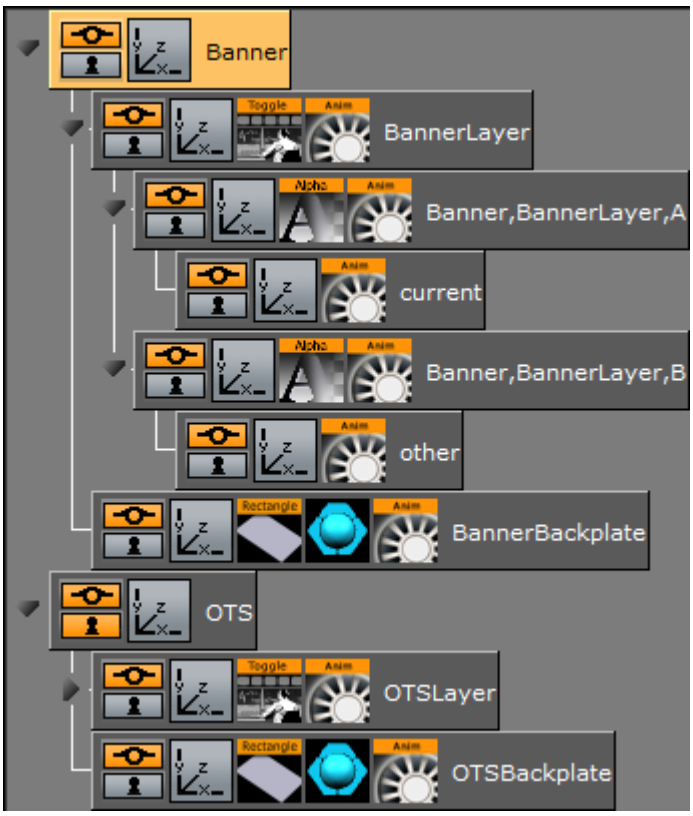
To Define the Toggle Layers

1. Create a list of all graphical items to be shown. For example:
   - A large two lines lower third.
   - A small one line lower third.
   - A small one line top third.
   - A left side over the shoulder (OTS).
   - A right side OTS.
   - A logo bug.
   - A “coming up next” bug.
   - A news ticker.
   - A clock.
2. Create an operational scenario where you decide how items will affect each other, and which items can be on-screen at the same time. For example:

- Only one lower or top third can be on-screen at a time (toggle layer 1).
- Only one OTS can be on-screen at a time (toggle layer 2).
- Only one logo or “coming up next” bug can be on-screen at a time (toggle layer 3).
- The news ticker and clock should be possible to take in and out independently of all layers (toggle layers 4 and 5).

> ⚠ **Note:** All toggle layers can potentially be on-screen at the same time, but only one object scene per layer can be shown at the same time.

To Create a Toggle Layer



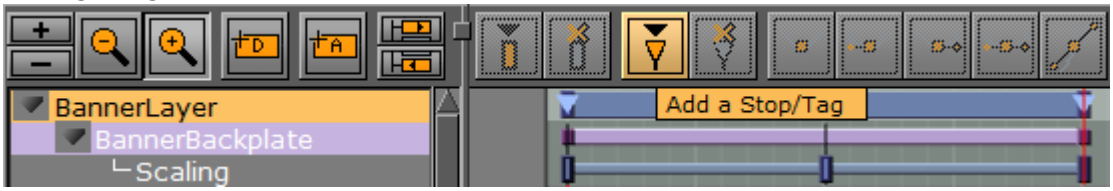
1. Create a new scene, and name it `default`. This will be the Master scene.

   > ⚠ **IMPORTANT!** This scene **must not** be named `background`, as this will trigger legacy mode in other software such as the Media Sequencer.

2. Add a group container to the scene tree, and name it `Banner`.
   - This will be the placeholder for the banner’s toggle layer and backplate graphics for the banner object scenes.
3. Add a **sub-container** to the Banner container, and name it `BannerLayer`.
   - This will act as the placeholder for the Toggle plug-in.
4. Add the **Toggle plug-in** to the BannerLayer container.

- This will generate the default object placeholders for the toggle layer' s current and other state.

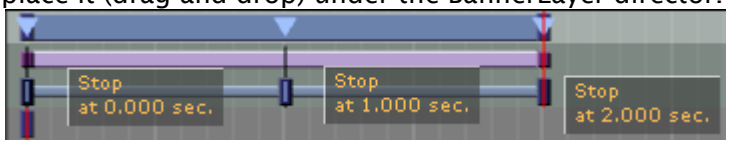
5. Add a **sub-container** to the Banner container, and name it `BannerBackplate`.
   - This will be the placeholder for the banner's backplate(s).
6. Add a **rectangle** and **material** to the BannerBackplate container, and adjust the Transformation Editor properties such that it creates the look and feel of a simple lower third graphics.
   - Axis Center: Set X to `left` (L) and Y to `bottom` (B). The X and Y values are set to -50.
   - Position: X: `-330`, Y: `-160`, Z: `0`
   - Scaling (single): X: `1.0`, Y: `0.4`, Z: `1.0`



7. Open the **Stage** by clicking the Stage button in the main menu.
8. Add a **new director**, and name it `BannerLayer`. This will be the placeholder for the backplate animations.
9. **Animate** the banner backplate:
   - Scaling X: `0` at 0 fields
   - Scaling X: `6.6` at 50 fields
   - Scaling Y: `0` at 100 fields

> ⚠ **Tip:** In this example, the animations are for a PAL resolution setup. If running an NTSC setup, the fields should be set to `0`, `60` and `120`, respectively. When creating an animation, clicking the **Add keyframe** button automatically adds 50 or 60 fields, depending on the current setup.

10. Open the **Stage** and check that the animation resides under the BannerLayer director. If not, place it (drag and drop) under the BannerLayer director.



11. Select the Banner director, set the director's Time Marker to `0.000` seconds, and add a **Stop/Tag**.
12. Select the tag and name it `O` (uppercase O, not zero).
13. Add a new tag for the animation's in point and name it `LOWERTHIRD`, and a second out point named `O`. These tags represent the basic states the graphics can be in.
14. Open the Toggle plug-in editor, and click the **Default Key Frames** button.
    - Adding Key Frames adds a 1 second default alpha cross fade effect to objects placed on that layer.

> ⚠ **Note:** Adding default Key Frames is not possible until the layer's director has at least two states defined where one has to be the O state.

15. Now it is time to add a layer for the Over the Shoulder graphics (OTS). Repeat steps 2-14 for a new layer called OTS with the following transformation properties and animations:

- Axis Center: Set X to `left` (L) and Y to `top` (T). This sets the X value to -50 and the Y value to 50.
- Position: X: `-330`, Y: `100`, Z: `0`
- Scaling (single): X: `1.0`, Y: `0.05`, Z: `1.0`
- Scaling X: `0.0`
- Scaling X: `1.0` at 50 frames
- Scaling Y: `1.4` at 100 frames
- Scaling X: `0.05` at 150 frames
- Scaling X: `0.0` at 200 frames

16. **Save** the scene with all directors in their O state.

> ⚠ **IMPORTANT!** Make sure to set all directors to an O state before saving the scene. This is because backplate graphics saved in a visible state will be shown on-screen the first time any toggle layer is taken on air.

To Add Key

- Open the Scene Settings Editor and select the **Auto Key** option under **Render Sequence** in the Global Settings Panel.

> ⚠ **Tip:** Click the **Key** button in the Scene Editor to toggle the key.

18.8.3 Create an Object Scene

Before creating object scenes you should start by creating a master scene. After the master scene has been created, the next step is to create the object scene(s) that reside on the different toggle layers defined in the master scene.

An important part of object scene design is to add the Control Object plug-in. The Control Object plug-in will enable the object scene to “tell” Viz Engine which layer it resides on and the state the master scene should be in when taken on air by the operator.

The object scene can expose other parameters to the operator by adding other control plug-ins, such as Control Text and Control Image. Additionally, the object scene can be used to add preview tags, so that the scene can be previewed by newsroom users working with the Viz Pilot Newsroom client.

To Create Object Scenes

1. Create a new scene, and name it `1000`.
2. Add container to the scene tree, and name it `lowerthird`.
3. Add **Control Object** plug-in to the lowerthird container.
4. Add a **sub-container** to the lowerthird container, and name it `text`.
5. Add a **font** to the text container.
6. Add a **Control Text** plug-in to the text container.
7. Open the **Control Object**'s editor, and set the following parameters:
   - Description: `Headline`
   - Transition Logic: `On`
   - Layer Identifier: `BannerLayer`
   - State Identifier: `LOWERTHIRD`
   - Background scene: `default`
8. Open the text container's transformation editor, and set the following properties:
   - Position: X: `-320`, Y: `-150`, Z: `1`
   - Scaling (`locked`): `0.5`
   - Scaling (`locked`): `0.0` at 0 frames
   - Scaling (`locked`): `0.5` at 50 frames
9. **Save** the `lowerthird` scene.

10. **Repeat** steps 2-11 for the OTS object scene with the following parameters, transformation properties and animations.
    - Scene name: `2000`
    - Description: `Text`
    - Transition Logic: `On`
    - Layer Identifier: `OTSLayer`
    - State Identifier: `OTS_LEFT`
    - Background scene: default
    - Position: X: `-330`, Y: `120`, Z: `1`
    - Scaling (`locked`): `0.3`
    - Scaling X: `0.0` at 50 frames
    - Scaling X: `0.3` at 100 frames

    > ⚠ **Tip:** Open the **Control Text** Editor and disable the Single line parameter.

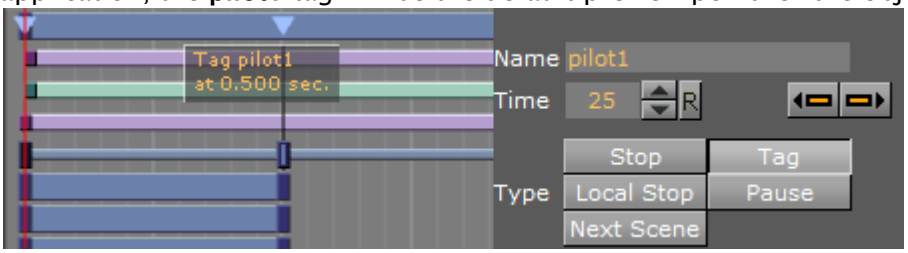
11. **Save** the OTS scene.

To Create a Background Image for the Object Scene

1. Open the Master scene.
2. Open the Stage view and set the layer's state where it matches the state of the Object scene, for example *LOWERTHIRD*.
3. In the Scene Editor, click the **Snap** button and save the snapshot. Make sure to give it an identifiable name, for example `banner_lowerthird`.
    - The snapshots can be used as "placeholders" for the object scenes.

To Create a Preview Point

1. Open the Stage, and select the **Default** director.
2. **Position** the **Time Marker** from where you would like the scene to be previewed.
3. Click the **Add a Stop/Tag** button.
4. In the appearing editor enter Name as `pilot1` (case sensitive), and set **Type** to `Tag`.
    - When previewing using a remote (external) Viz Artist/Engine preview, either from a newsroom system, or when connected to a Media Sequencer with a control application, the **pilot1** tag will be the default preview point for the object scene.



19 Scripting

To learn and use the computer language **Viz 3 Script**, first the basic principles of programming must be understood.

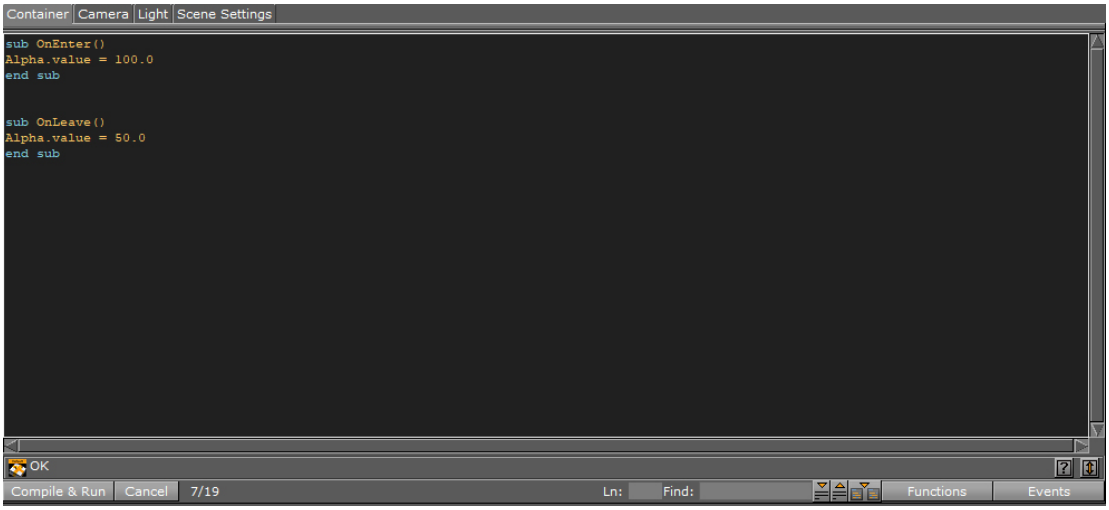
This chapter serves as a quick introduction on how to create and run Viz scripts; how to create reusable plug-ins from scripts; and lastly provides a few example scripts that will help develop an understanding of the Viz script functionality. For more in-depth information, refer to the built-in API and Scripting documentation located in the Viz installation folder.

This section contains information on the following topics:

- Script Editor
- Create and Run Scripts
- Create Script-based plug-ins
- Control 3D Stereoscopic Clip Playback
- Program Examples
- Event Pool

19.1 Script Editor

This section contains details how to open, create, search and show scripts in the Script Editor.



Viz Artist can be scripted on two levels, using Scene Scripts or Container Scripts. The events available for Scene and Container scripts differ, and can be accessed by clicking the Events drop-down button in the script editor.

Before adding scripts and code, the designer needs to think about where the logic should reside. If there are functions acting like an interface to the scene, it is a good idea to add this logic as a scene script. Also, if functions or variables are used throughout several other scripts within the scene, the scene script is the proper location for those. On the other hand, if the script code is container specific or planned for a container prototype-instance(s) approach, the script code should be hosted in a container script.

This section contains the following topics:

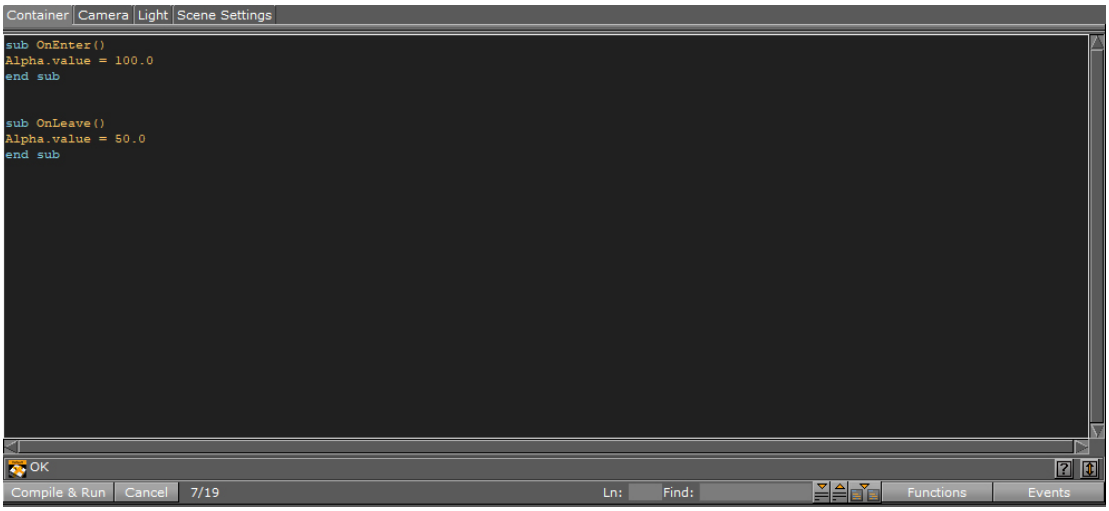
- Script Editor Properties

- Script Editor Search
- Script Editor Procedures

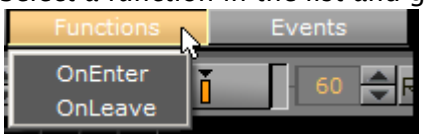
19.1.1 Script Editor Properties

The **Scene Script Editor** and the **Container Script Editor** have the same properties and functionality, but a script created in one can not be placed in the others folder.

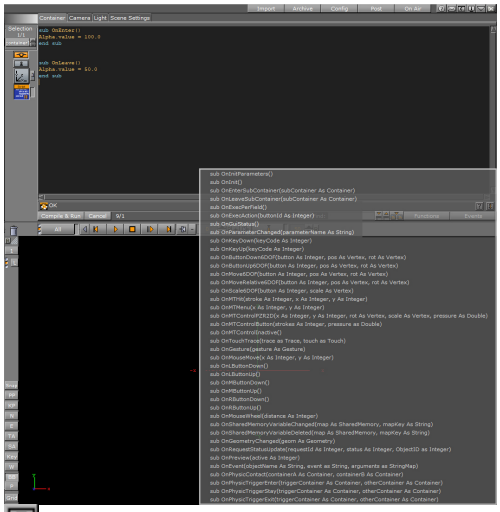
This section gives details on Script Editor properties and context menu.



- OK Shows that all parts of the script entered are correct. If there is an error, it will show where the error is.
- Compile & Run Checks the code entered and translates it into byte code, so the script can be run. If there is an error, a message is shown in the text above.
- Cancel Ignores the last source code changes and compiles previous version.
- Ln: Goes to the line number entered in the field.
- Find: Enters free text search. The up icon ( ) searches above the cursor location. The down icon ( ) searches below the cursor location.
- Searches down.
- Searches up.
- Searches in all scripts used in this scene.
- Functions Shows a list of used functions in the script (both built-in and self defined ones). Select a function in the list and go there in the script editor.



- Events Shows a list of built-in events. Select one to add it to the script:

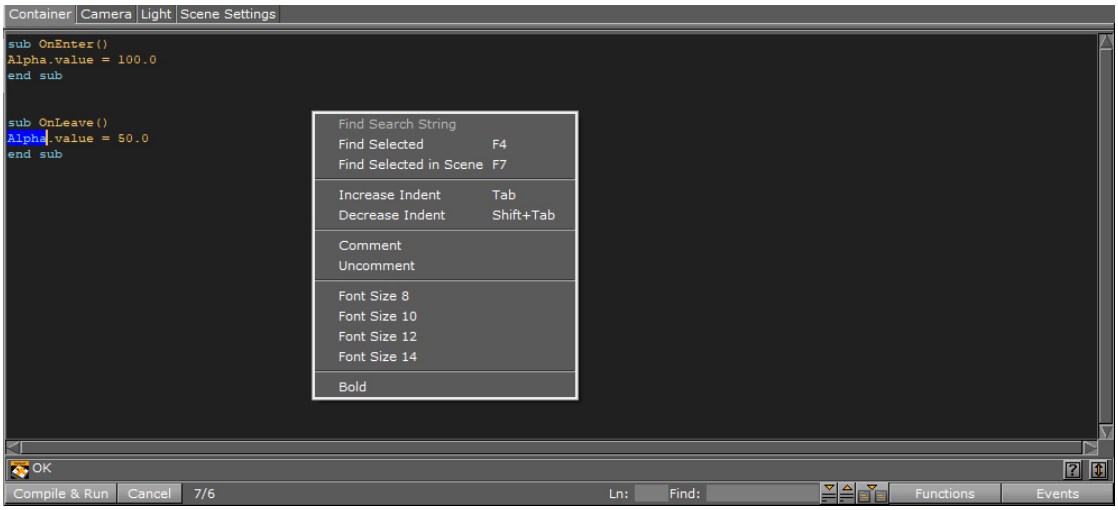


- Opens the script documentation.
- Opens or closes a script editor floating window.

> **Tip:** The floating window can be scaled to any size.

Script Editor Context Menu

Right-click in the Script Editor to open the Script Editor context menu:

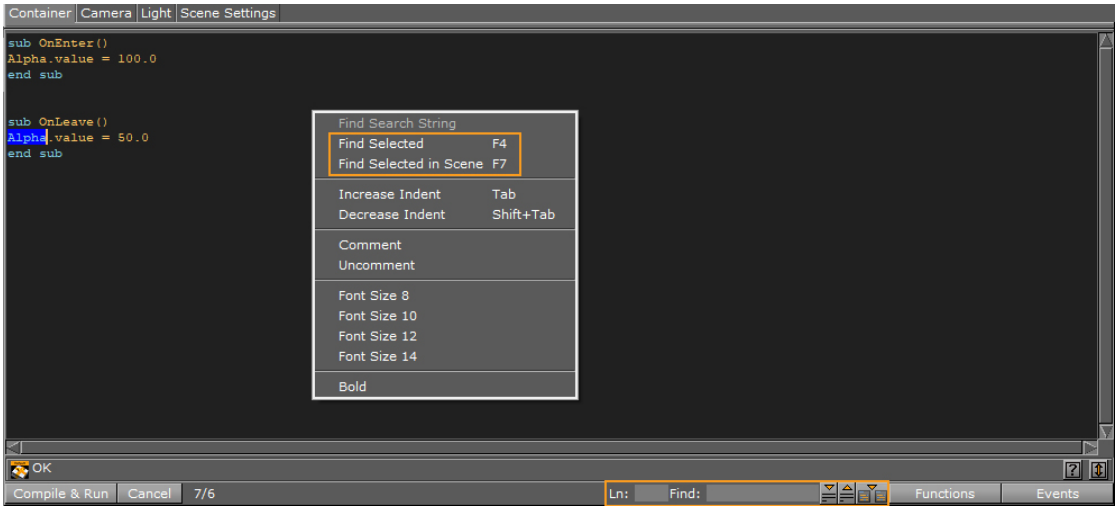


- **Find Search String**: Searches for text entered in the **Find** input box.
- **Find Selected (F4)**: Searches for highlighted text.
- **Find Selected in Scene (F7)**: Searches for highlighted text in all scripts in the scene.
- **Increase Indent (Tab)**: Increases the indent of the selected text.
- **Decrease Indent (Shift+Tab):** Decreases the indent of the selected text.
- **Comment**: Comments the selected text.
- **Uncomment:** Uncomments the selected text.

- **Font Size:** Sets the font size.
- **Bold:** Sets the font to bold.

19.1.2 Script Editor Search

This function allows you to search for highlighted elements in the actual container script, but it can also search in scripts on all containers in your scene.



> **Tip:** You can also search in scripts from the Scene Tree Menu.

19.1.3 Script Editor Procedures

This section details common script Editor procedures:

- To Create a Container Script plug-in
- To Show Container Script plug-ins in the Script Editor
- To Create a Scene Script plug-in
- To Show Scene Script plug-ins in the Script Editor

To Create a Container Script plug-in

1. Add the Script plug-in to the target Container.

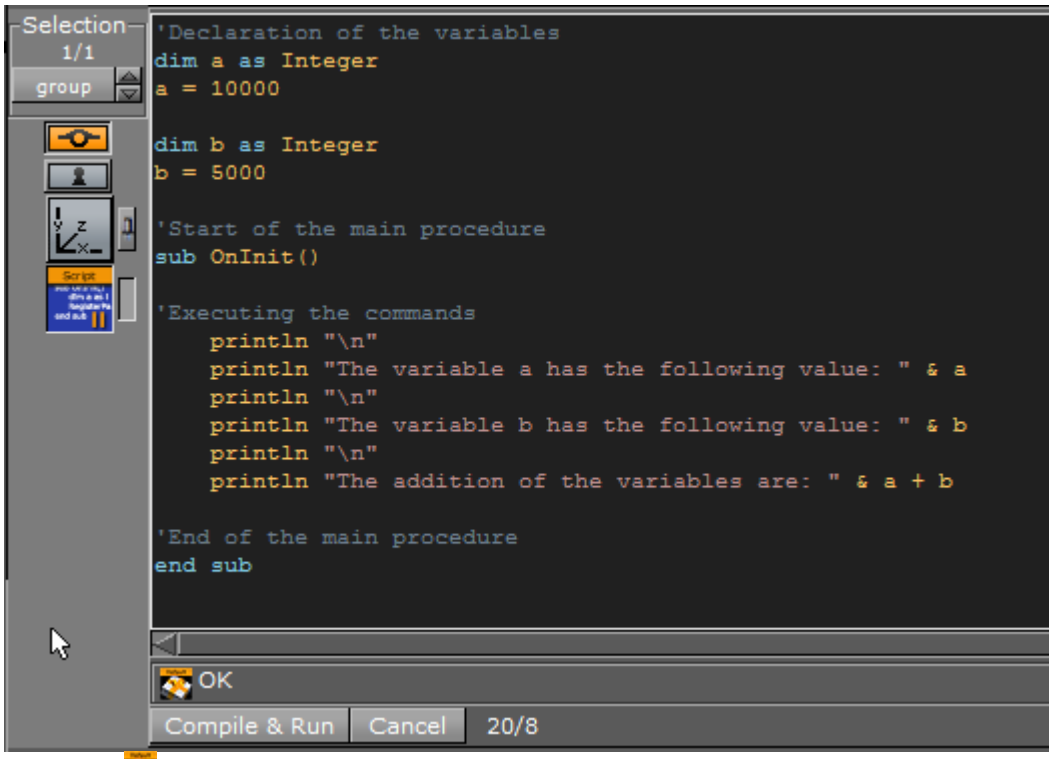
   > **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

2. Click on the Script plug-in icon.

3. In the Script Editor type in the new script (see also Script Editor Properties).

> **Note:** Also, copy and paste a script from another source, if required.

4. Click Compile & Run .



5. Drag the icon to *Container plug-ins -> Scriptplug-ins* folder.

> **Note:** The created plug-in is saved in *<viz data folder>/Scriptplug-ins* as *<plug-in_name>.vsl*.

To Show Container Script plug-ins in the Script Editor

1. Add the Script plug-in to the target Container.

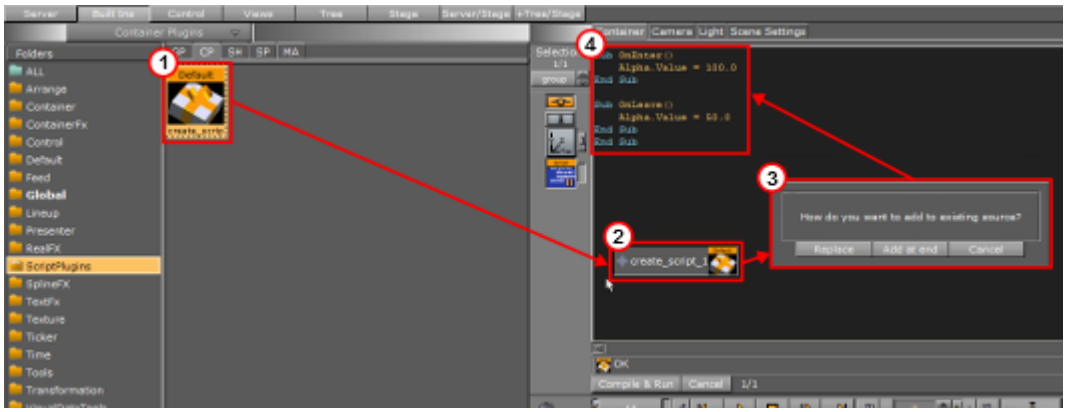
> **Note:** This plug-in is located in: Built Ins -> Container plug-ins -> Global

2. Drag and drop a created Script plug-in (1 (see Script plug-ins)) icon to the Script Editor (2).

> **Note:** These plug-ins are located in: Built Ins -> Container plug-ins -> Scriptplug-ins

3. Click **Replace** or **Add to End** (3).

> **Example:** Replaced text (4).



To Create a Scene Script plug-in

1. Click *Scene Settings -> Script*.
2. In the Script Editor type in the new script (see also Script Editor Properties).

> **Note:** Also, copy and paste a script from another source, if required.

3. Click Compile & Run .
4. Drag the icon to *Scene plug-ins -> Scriptplug-ins* folder.

> **Note:** The created plug-in is saved in *<viz data folder>/Scriptplug-ins* as *<plug-in_name>.vsl*.

To Show Scene Script plug-ins in the Script Editor

1. Click *Scene Settings -> Script*.
2. Drag and drop a created Scene Script plug-in (see Script plug-ins) icon to the Script Editor.

> **Note:** These plug-ins are located in: Built Ins -> Scene plug-ins -> Scriptplug-ins

3. Click **Replace** or **Add to End**.

> **Note:** You can also do an advanced search from the Scene Tree Menu to search for terms across all scripts used in the scene.

19.2 Create And Run Scripts

This section contains two script examples:

- Example I: This example shows how to write a simple script that controls the alpha value of an object
- Example II: This example shows how to write a simple script that scales the x-scaling value of a container, based on x-scaling value of another container.

19.2.1 Example I

Start with a simple event-driven script.

1. Add a Sphere to a Container.
2. Add a Material and some light.
3. Create a script object.
   a. Drag the built-in Script plug-in from the Container plug-ins' Global folder onto a Container.
   b. Click on the Script plug-in icon in a Container
   c. Copy and paste the code, below, into the Script Editor.

```
Sub OnEnter()
     Alpha.Value = 100.0
End Sub
Sub OnLeave()
     Alpha.Value = 50.0
End Sub
```

   This script consists of two event procedures, **OnEnter** and **OnLeave**, which are recognized by the system and invoked whenever the mouse cursor enters or leaves the area occupied by the container in the output window. Take a closer look at what happens inside **OnEnter**:

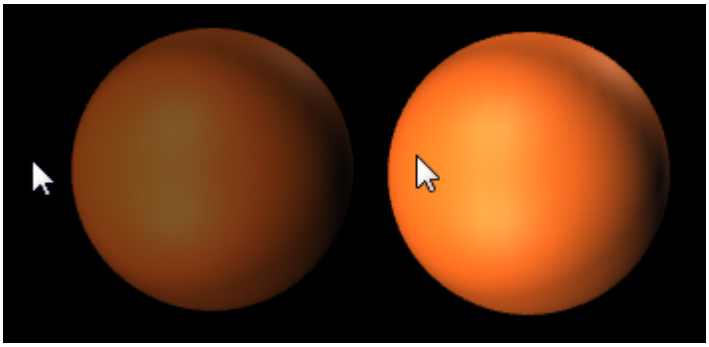
```
Alpha.Value = 100.0
```

   This code sets the alpha value of the container to 100 (the maximum value). If there is no Alpha object, no action occurs, so make sure that the container has an alpha object by adding the Alpha plug-in. **Alpha** refers to the container's alpha object, **Value** is a member (in this case: the only member) of the alpha object, representing its value. Changes made to **Alpha.Value** are immediately made visible in the output window. Similarly, the code in the body of **OnLeave** sets the alpha value back to 50.
4. Add the Alpha plug-in to the container.
5. Execute the script, click 

.
6. Click **Edit** to return to editing mode if required.
   To interact with the scene click the 

button (Scene Editor Buttons) in the Scene Editor.

⚠ **Note:** The E button offers a good way for designers to test touch based interaction points added to a scene. The script is then the logic which triggers what happens next (i.e. events).



19.2.2 Example II

This example defines the event procedure **OnExecPerField**, which is called for each field. The functions and amount of code used in this procedure needs to be decided upon with great consideration. Include only the code that requires execution on a per-field basis, to keep the resulting performance impact as low as possible.

```
Sub OnExecPerField()
     Dim cube As Container
     cube = Scene.FindContainer(“Circle”)
     If scaling.x > 1.5 Then
         cube.Scaling.x = 2.0
     Else
         cube.Scaling.x = 1.0
     End If
End Sub
```

In the first line, a variable named **cube**, of type Container, is defined:

```
Dim cube As Container
```

Variables of type Container can be initialized to point to a container in the scene tree. In the next line, we call the built-in function FindContainer (which is a member of the scene class, accessible via the container's Scene property) to initialize cube to point to a container named cube. If there is no such container, cube is set to Null, the consequence being that any further operations on cube will have no effect.

```
cube = Scene.FindContainer("Cube")
```

Now, we check if the x-scaling of the current container (that is, the container holding the script) is greater than 1.5, and, depending on the result, we set the x-scaling of the container named cube to 2.0 or 1.0:

```
If Scaling.x > 1.5 Then
     cube.Scaling.x = 2.0
Else
     cube.Scaling.x = 1.0
End If
```

⚠ **Note:** The expression `Scaling.x` refers to the current container while `cube.Scaling.x` refers to the container referenced by the variable cube.

19.3 Create Script-Based Plug-Ins

Scene plug-ins and Container plug-ins can be created, saved and used again in other Scenes or Containers. To do this create a plug-in of the created script.

Created Script-based plug-ins are saved as '.vsl' files at *<viz data folder>\Scriptplug-ins*, and are available in these folders:

- Container plug-ins -> Script plug-ins
- Scene plug-ins -> Script plug-ins

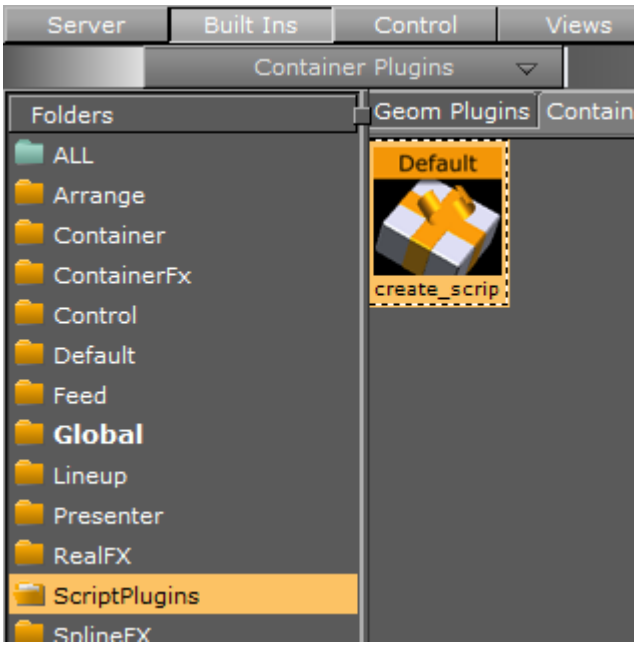
Script-based plug-ins can be used the same as Scene or Container plug-ins, and allow the additional following operations:

- Can be deleted: Drag the script to the trash can.
- The source code of a script-based plug-in can be copied into a script editor. Drag the plug-in from Script plug-ins folder to the Script Editor.

This section contains the following procedures:

- To Create a Script-based plug-in
- To Edit a Script-based plug-in

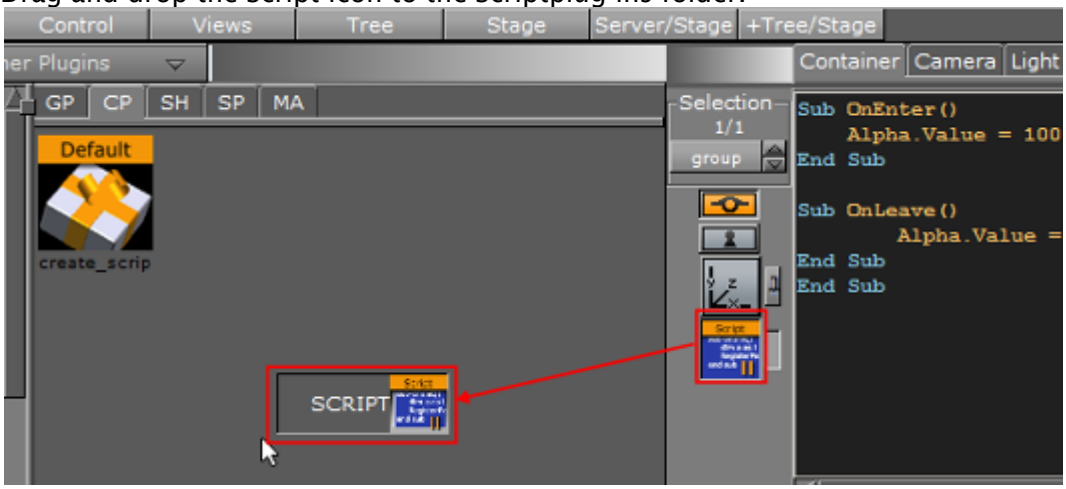
19.3.1 To Create a Script-based plug-in



1. Create a script (see To Create a Container Script plug-in).
2. Click Compile & Run .

> ⚠ **Note:** If the script is bad a Script plug-in can not be created. Correct any bad script first.

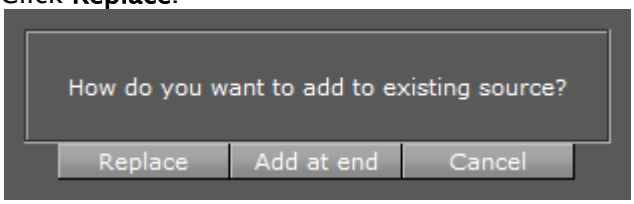
3. Go to **Built Ins** -> **Container plug-ins -> Scriptplug-ins**.
4. Drag and drop the Script icon to the Scriptplug-ins folder.



5. Enter a name for the Script plug-in.
6. Click **OK**.

19.3.2 To Edit a Script-based plug-in

1. Add a Container to the Scene Tree.
2. Add a Script plug-in.
3. Open the Script Editor*(Scene Settings -> Script*).
4. Drag and drop the required script plug-in from the **Scriptplug-ins** folder to the Script Editor.
5. Click **Replace**.



6. Edit the script as required.
7. Click the **Compile & Run** to save.

See Also

- Script plug-in
- Script Editor

19.4 Control 3D Stereoscopic Clip Playback

To playback 3D Stereo Clips, Viz Engine must be started with the -3D parameter. This sets Viz Engine in a mode where a side by side clip will be split and played out on channel A and B on the Matrox board.

> ⚠ **Note:** This is for clips only.

The playback on each output channel can be controlled with several shared memory variables. Each channel has its own set of variables indicated by the A and B in the names. The meaning of each of them is the same for both output channels.

- **HScale / VScale**: Scales the output image relative to the image anchor (see XAnchor and YAnchor) in the respective direction.
- **HSort / VSoft**: Applies softness to the scaled region of the source image only, and is defined as the percentage of pixels to be softened from the border to the center.
- **XTrans / YTrans**: Translates the output image in the respective direction.
- **XAnchor / YAnchor**: Defines the position of the anchor on the input image. The axis center equals the center of the input image.
- **Sharpness**: Sets the amount of additional sharpness to apply to the output image.
  To control the behavior of clip playback, the following shared memory variables exist:

```
vizrt.com.StereoClip.changed
vizrt.com.StereoClip.A.HScale
vizrt.com.StereoClip.A.VScale
vizrt.com.StereoClip.A.HSoft
```

```
vizrt.com.StereoClip.A.VSoft
vizrt.com.StereoClip.A.XTrans
vizrt.com.StereoClip.A.YTrans
vizrt.com.StereoClip.A.XAnchor
vizrt.com.StereoClip.A.YAnchor
vizrt.com.StereoClip.A.Sharp
vizrt.com.StereoClip.B.HScale
vizrt.com.StereoClip.B.VScale
vizrt.com.StereoClip.B.HSoft
vizrt.com.StereoClip.B.VSoft
vizrt.com.StereoClip.B.XTrans
vizrt.com.StereoClip.B.YTrans
vizrt.com.StereoClip.B.XAnchor
vizrt.com.StereoClip.B.YAnchor
vizrt.com.StereoClip.B.Sharp
```

See Also

- For more information about setting stereo modes, see the **Camera** page in the **Configuring Viz** section of the Viz Engine Administrator Guide.
- Calculate the Days of a Week and Month script example

19.5 Program Examples

To explain the described terms and definitions, this section shows some source code examples and explains the important steps and commands.

Most of the examples detailed here are written in an easy and understandable form for programing beginners. Some are basic, while others are more advanced and specific for their use case. More efficient code structures and improvements are possible.

This section contains information on the following topics:

- Initializing Scripts
- Register Parameters
- Mouse Events I (Set)
- Mouse Events II (Start, Stop and Reverse)
- Mouse Events III (Grow and Shrink)
- Simple Gesture Recognition
- Tetris
- Calculate the Days of a Week and Month
- Update a Scene Using a Text File

19.5.1 Initializing Scripts

This example declares and defines two variables with a value. In the main procedure `OnInit()` those two variables and their addition are output to the console using Println.

Callback procedures like `OnInit()` are used to initialize scripts (see).

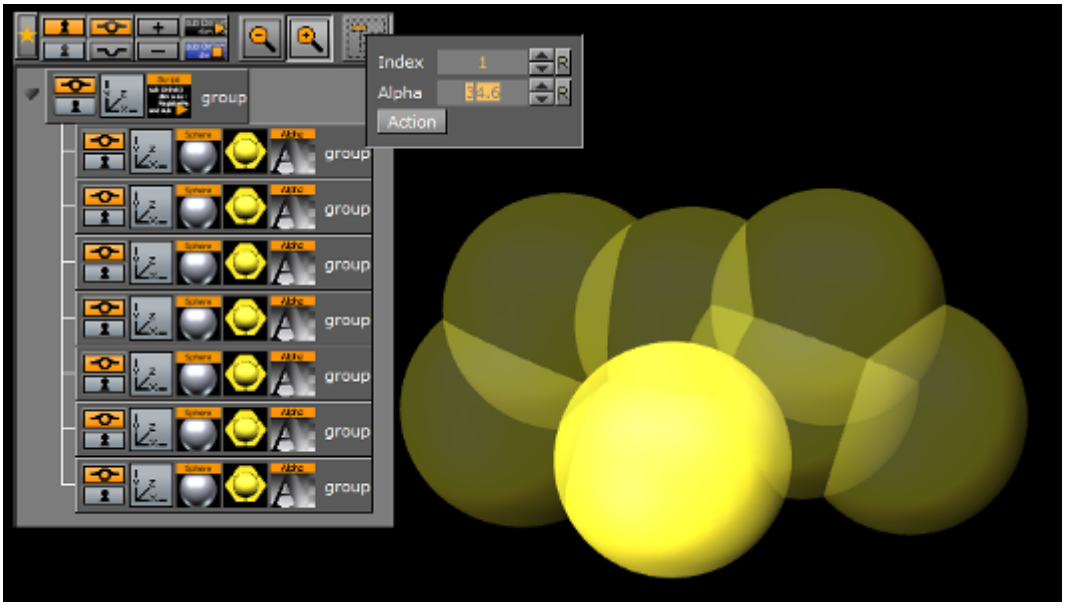
```
'Declaration of the variables
dim a as Integer
a = 10000
dim b as Integer
b = 5000
'Start of the main procedure sub OnInit()
'Executing the commands
println "\n"
        println "The variable a has the following value: " & a
println "\n"
     println "The variable b has the following value: " & b
println "\n"
     println "The sum of the variables is: " & a + b
'End of the main procedure
end sub
```

Result output to the Viz Engine console:

```
The variable a has the following value: 10000

The variable b has the following value: 5000

The addition of the variables are: 15000
```

19.5.2 Register Parameters

This script example creates a custom editor for filling in index and alpha values. The index value decides which container that will not receive the alpha value whereas the alpha value is set for all other containers.




```
sub OnInitParameters()
```

```
     RegisterParameterInt("index", "Index", 1, 1, 7)
     RegisterParameterDouble("alpha", "Alpha", 50.0, 0.0, 100.0)
     RegisterPushButton("action", "Action", 1)
end sub
sub OnExecAction(buttonId as Integer)
     dim c as Container
     dim i, index as Integer
     dim a as Double
     index = GetParameterInt("index")
     a = GetParameterDouble("alpha")
     i = 1
     c = ChildContainer
     do while c <> null
         if i = index then
             c.alpha.value = 100.0
         else
             c.alpha.value = a
         end if
     i = i + 1
     c = c.NextContainer
     loop
end sub
```

19.5.3 Mouse Events I (Set)

This script example shows how to use a 'mouse event'" to set an object's alpha value with OnEnter and OnLeave.

```
sub OnEnter()
     alpha.value = 100.0
end sub
sub OnLeave()
     alpha.value = 50.0
end sub
```

19.5.4 Mouse Events II (Start, Stop and Reverse)

This script, when added to a container, allows an animation to be started with the click of a mouse button over an object (e.g. rectangle). Press again and it will stop. Press again and it will continue from the point it stopped.

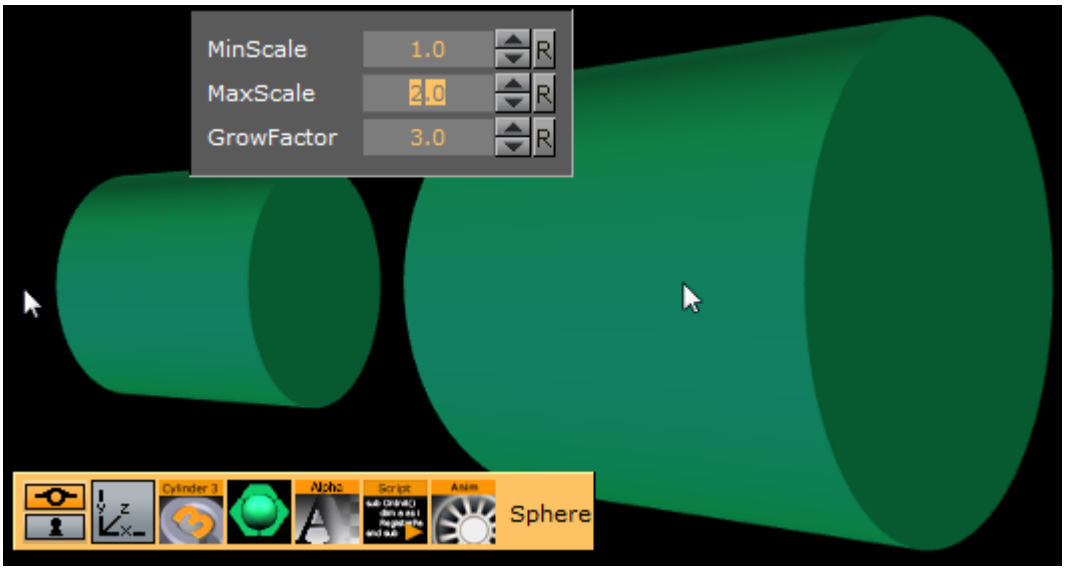
```
dim my_dir = stage.FindDirector("Default")
sub OnLButtonDown()
     if my_dir.IsAnimationRunning() then
          my_dir.StopAnimation()
     else
         my_dir.ContinueAnimation()
     end if
end sub
```

With the script detailed below, start an animation and let it run until a stop point. Click again and it will run in reverse:

```
dim my_dir = stage.FindDirector("Defult")
sub OnLButtonDown()
     menu.ContinueAnimation()
     if my_dir.reverse == true then
         my_dir.reverse = false
     else
         my_dir.reverse = true
     end if
end sub
```

19.5.5 Mouse Events III (Grow and Shrink)

With this script added to a Container an object will grow when the mouse is hovering over the object and conversely. In this case the center of the object is set to the lower right corner to let it grow out from that center.




```
dim growing as Boolean = false
sub OnInitParameters()
     RegisterParameterDouble("min", "MinScale", 1.0, 0.0, 10.0)
     RegisterParameterDouble("max", "MaxScale", 1.5, 0.0, 10.0)
     RegisterParameterDouble("fac", "GrowFactor", 1.1, 1.0, 10.0)
end sub
sub OnEnter()
     growing = true
end sub
sub OnLeave()
     growing = false
end sub
```

```
sub OnExecPerField()
     dim min = GetParameterDouble("min")
     dim max = GetParameterDouble("max")
     dim fac = GetParameterDouble("fac")
     if growing then
         scaling.xyz *= fac
         if scaling.x > max then scaling.xyz = max
     else
         scaling.xyz /= fac
         if scaling.x < min then scaling.xyz = min
     end if
end sub
```

19.5.6 Simple Gesture Recognition

This example script can be used to detect a drawn square on the screen:

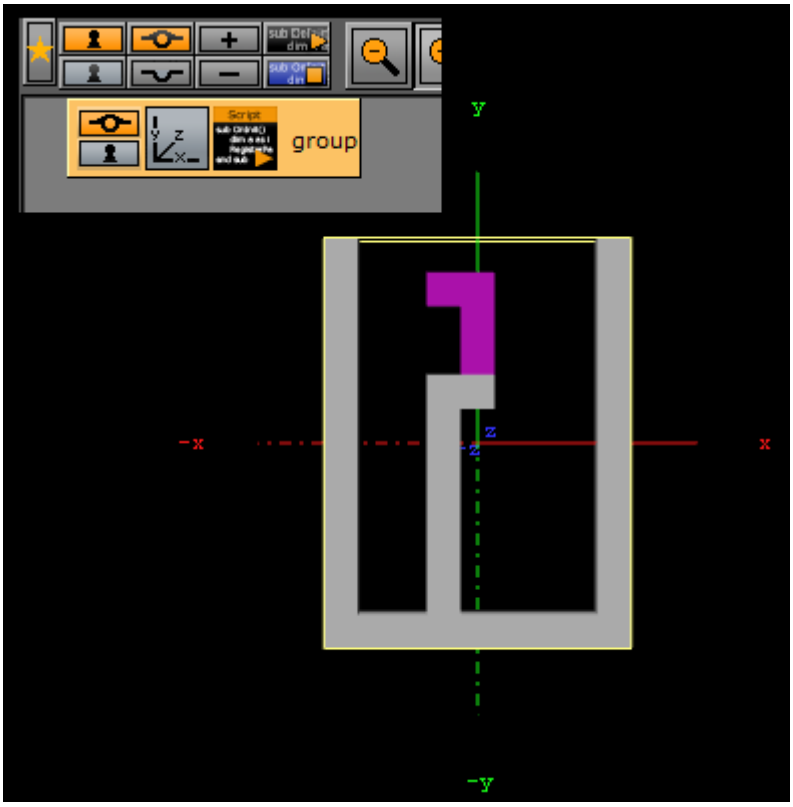
```
dim gRecognizer as gesturerecognizer
sub OnInit()
        gRecognizer.handlepatterns = true
        gRecognizer.addhandler(scriptplugininstance)
end sub

sub OnGesture(gesture As Gesture)
        if gesture.patternid == PATTERN_SQUARE then
                println("This is a square")
        end if
end sub

sub OnTouchTrace(trace as Trace, touch as Touch)
        gRecognizer.addtrace(trace)
end sub
```

⚠ **Note:** Some gesture patterns need to be checked for their alignment as well, for example *up*, *down*, *left*, and so on. For this you can activate the `CheckPatternOrientation` property in the gesture recognizer object.

For more information about the possible properties and procedures, please refer to .

19.5.7 Tetris

This script example creates the classical game of Tetris.




```
dim HEIGHT, WIDTH, TOTAL, NUM_BRICKS as Integer
dim currentBrick, currentRot, counter as Integer
dim xpos, ypos as Integer ' position of the falling brick

dim squares as Array[integer]
dim squareContainers as Array[Container]
dim bricks as Array[String]

HEIGHT = 15
WIDTH = 9
TOTAL = HEIGHT * WIDTH

squares.size = TOTAL

sub OnInit()
        dim i, j as Integer
    ' 0100
    ' 0100
    ' 0100
    ' 0100

    bricks.push("0100010001000100")
    bricks.push("0000111100000000")
    bricks.push("0100010001000100")
    bricks.push("0000111100000000")
```

```
' 0000
' 0000
' 0110
' 0110
bricks.push("0000000001100110")
bricks.push("0000000001100110")
bricks.push("0000000001100110")
bricks.push("0000000001100110")

' 0000
' 0100
' 0110
    ' 0010
bricks.push("0000010001100010")
bricks.push("0000000001101100")
bricks.push("0000010001100010")
bricks.push("0000000001101100")

' 0000
' 0010
' 0110
' 0100

bricks.push("0000001001100100")
bricks.push("0000000011000110")
bricks.push("0000001001100100")
bricks.push("0000000011000110")

' 0000
' 0110
' 0010
' 0010

bricks.push("0000011000100010")
bricks.push("0000001011100000")
bricks.push("0100010001100000")
bricks.push("0000011101000000")

' 0000
' 0110
' 0100
' 0100

bricks.push("0000011001000100")
bricks.push("0000111000100000")
bricks.push("0010001001100000")
bricks.push("0000010001110000")

' 0000
' 0000
' 1110
' 0100

bricks.push("0000000011100100")
```

```
    bricks.push("0000010011000100")
    bricks.push("0000010011100000")
    bricks.push("0000010001100100")

        NUM_BRICKS = bricks.size / 4

        deleteGeometry
    update
    deleteChildren

    dim blocksGroup as Container
    blocksGroup = addContainer(TL_DOWN)

    open = false

    for i=0 to HEIGHT-1
                for j=0 to WIDTH-1
                        dim b as Container
                                if i >= 3 then
                                        b = blocksGroup.addContainer(TL_DOWN)
                                        b.createMaterial
                                        b.name = "f_" & j & "_" & i
                                        b.createGeometry("BUILT_IN*GEOM*Cube")
                                        b.scaling.xyz = 0.1
                                        b.position.x = (j-WIDTH/2.0+0.5) * 10.0
                    b.position.y = - (i-HEIGHT/2.0-1.0) * 10.0
                                end if
                                squareContainers.push(b)
                next
            next
        initGame
end sub

sub display()
        dim i, j, index, fb as Integer
        dim fc as Container

    for i=3 to HEIGHT-1
        for j=0 to WIDTH-1
            index = i * WIDTH + j
            fb = squares[index]
            fc = squareContainers[index]
                        if fb = 0 then
                                fc.active = false
                        else
                                fc.active = true
                                if fb = 1 then
                                        fc.material.diffuse = CColor(60.0, 60.0, 60.0)
                                elseif fb = 3 then
                                        fc.material.diffuse = CColor(30.0, 30.0, 30.0)
                                else
                                        fc.material.diffuse = CColor(40.0, 0.0, 80.0)
                                end if
                        end if
```

```
                next
        next
end sub

sub initGame()
        dim i as Integer
        for i=0 to TOTAL-1
                squares[i] = 0
        next
        for i=0 to HEIGHT-1
                squares[WIDTH*i] = 3
                squares[WIDTH*i + WIDTH-1] = 3
        next
        for i=TOTAL-WIDTH to TOTAL-1
                squares[i] = 3
        next
        newBrick
        display
end sub

sub newBrick()
        xpos = WIDTH / 2 - 2
        ypos = -1
        counter = 0
        currentBrick = random(NUM_BRICKS)
        currentRot = 0 end sub sub clear2()

        ' clear the squares occupied by the falling brick
        dim i as Integer
                for i=0 to TOTAL-1
                        if squares[i] = 2 then
                                squares[i] = 0
                        end if
        next
end sub

function drawBrick(x as Integer, y as Integer, rot as Integer, fill as Integer) as Boolean
        dim b as string
        dim index, i, xb, yb, x2, y2 as Integer

        clear2
        b = bricks[currentBrick * 4 + rot]
        for i=0 to 15
                xb = i mod 4
                yb = i / 4
                if b.getChar(i) = "1" then
                        x2 = x + xb
                        y2 = y + yb
                        if y2 >= 0 then
                                index = y2 * WIDTH + x2
                                if x2 < 0 or x2 >= WIDTH or y2 >= HEIGHT or squares[index] <> 0 then
                                        drawBrick = false ' collision
                                        exit function
                                elseif y2 >= 3 or fill <> 1 then
```

```
                                        squares[index] = fill
                                end if
                        end if
                end if
        next

        drawBrick = true ' no collision
end function

function isLineFilled(line as Integer) as Boolean
        dim i, index as Integer

        index = WIDTH * line
        for i=0 to WIDTH-1
                if squares[index+i] = 0 then
                        isLineFilled = false
                        exit function
                end if
        next
        isLineFilled = true
end function

sub eraseFullLines()
        dim line, l2, r as Integer
        for line = HEIGHT - 2 to 0 step -1
                if isLineFilled(line) then
                        ' move lines one line down
                        for l2 = line to 0 step -1
                                for r=0 to WIDTH-1
                                        squares[l2*WIDTH+r] = squares[(l2-1)*WIDTH+r]
                                next
                        next

                        ' clear top line
            for r=1 to WIDTH-2
                                squares[r] = 0
                        next
                        line++ ' test the same line again
                end if
        next
end sub

sub OnKeyDown(keyCode as Integer)
        dim x, y, r as Integer
                x = xpos
                y = ypos
                r = currentRot

                if keyCode = KEY_RIGHT then
                        x++
                elseif keyCode = KEY_LEFT then
                        x--
                elseif keyCode = KEY_DOWN then
                        y++
```

```
                elseif keyCode = KEY_S then
                        initGame
                        exit sub
                elseif keyCode = KEY_CONTROL then
                r = (r + 1) mod 4
                end if

                if r <> currentRot then
                        if drawBrick(x, y, r, 2) then
                                currentRot = r
                        elseif drawBrick(x-1, y, r, 2) then
                                xpos = x - 1
                                currentRot = r
                        elseif drawBrick(x+1, y, r, 2) then
                                xpos = x + 1
                                currentRot = r
                        else
                                drawBrick(xpos, ypos, currentRot, 2)
                        end if
                elseif x <> xpos or y <> ypos then
                        if drawBrick(x, y, r, 2) then
                                xpos = x
                                ypos = y
                        else
                                drawBrick(xpos, ypos, currentRot, 2)
                        end if
                end if
        display
end sub

sub OnExecPerField()
        counter++
        if counter > 30 then
                counter = 0
                if drawBrick(xpos, ypos + 1, currentRot, 2) then
                        ypos++
                else
                        ' we hit the bottom
                        drawBrick xpos, ypos, currentRot, 1
                        eraseFullLines
                        newBrick
                end if
        display
        end if
end sub
```

19.5.8 Calculate the Days of a Week and Month

This script example calculates the day of the week and the number of days of the month:

```
DIM smap AS Array[String]
```

```
smap.size = 7
smap[0] = "Sun"
smap[1] = "Mon"
smap[2] = "Tue"
smap[3] = "Wed"
smap[4] = "Thu"
smap[5] = "Fri"
smap[6] = "Sat"

function CalcDayOfWeek(iYear AS Integer, iMonth AS Integer, iDay AS Integer) AS String

DIM dt AS DateTime
        dt.year = iYear
        dt.month = iMonth
        dt.dayofmonth = iDay
        dt.normalize()

        CalcDayOfWeek = smap[dt.dayofweek]
end function

function CalcNumOfDaysOfMonth(iYear AS Integer, iMonth AS Integer) AS Integer
DIM dt AS DateTime
        dt.year = iYear
        dt.month = iMonth+1
        dt.dayofmonth = 0
        dt.normalize()
        CalcNumOfDaysOfMonth = dt.dayofmonth
end function

sub OnInit()
DIM dt AS DateTime
        dt.year = 2012
        dt.month = 2
        dt.dayofmonth = 23

        println("Date: " & CalcDayOfWeek(dt.year, dt.month, dt.dayofmonth) & ", " & dt.year & "" & dt.month
& "" & dt.dayofmonth)
        println("Days: " & CalcNumOfDaysOfMonth(dt.year, dt.month))
end sub
```

19.5.9 Update a Scene Using a Text File

This script example updates a Scene from a text file.

```
dim timer,limit as integer
timer = 0
sub OnInitParameters()
        RegisterInfoText("Text File Grabber 1.0" & chr(10) & chr(10)
                & "Grabs text file and puts optional prefix/suffix around the contents of the text file." &
chr(10)
                & "Filename supports paths with drive letters and UNC paths." & chr(10) & chr(10)
```

```
                & "Text in \"Default Text\" box is used if file specified in File box cannot be opend or
accessed." & chr(10) & chr(10)
                & "Listens to scene map variable \"text_update\" -- text files will be re-read when this
variable changes.")
        RegisterFileSelector("file", "File", "c:\test", "1.txt", "*.*")
        RegisterParameterString("default","Default text","",40,50,"")
        RegisterParameterString("prefix","Prefix (optional)","",40,50,"")
        RegisterParameterBool("prefixIfNull","Include prefix if file not found",FALSE)
        RegisterParameterString("suffix","Suffix (optional)","",40,50,"")
        RegisterParameterBool("suffixIfNull","Include suffix if file not found",FALSE)
        RegisterParameterInt("updateFields","Auto-Update interval (fields)",300,60,999999999999)
        RegisterParameterBool("debug","Debug output to console",FALSE)
end sub

sub OnParameterChanged(parameterName As String)
        \qre-populate text when a parameter changes
        if parameterName <> "updateFields" then UpdateText
        if parameterName = "updateFields" then
                limit = GetParameterInt("updateFields")
        end if
end sub

sub OnExecPerField()
        timer = timer + 1
        if timer >= limit then
                if GetParameterBool("debug") then println "Timer limit reached"
                        UpdateText
            timer = 0
        end if
end sub

sub OnInit()
        Scene.Map.RegisterChangedCallback("text_update")
        limit = GetParameterInt("updateFields")
        UpdateText
end sub

sub OnSharedMemoryVariableChanged(map As SharedMemory, mapKey As String)
        if mapKey = "text_update" then
                UpdateText
        end if
end sub

sub UpdateText()
        geometry.text = GetTextFile()
end sub

function GetTextFile() as string
        dim content, path as string
        path = GetParameterString("file")
        if GetParameterBool("debug") then println "File path: " & path
        if System.LoadTextFile( path, content ) then
                if GetParameterBool("debug") then println "Successfully loaded file"
                content.trim
```

```
                if len(GetParameterString("prefix")) > 0 then content = GetParameterString("prefix") & " "
 & content
                if len(GetParameterString("suffix")) > 0 then content = content & " " &
GetParameterString("suffix")
        else
                if GetParameterBool("debug") then println "Could not load file; using default text"
                content = GetParameterString("default")
                if GetParameterBool("prefixIfNull") then content = GetParameterString("prefix") & " " &
content
                if GetParameterBool("suffixIfNull") then content = content & " " &
GetParameterString("suffix")
        end if
        GetTextFile = content
        if GetParameterBool("debug") then println "Returning '" & GetTextFile & "'"
end function
```

19.6 Event Pool

A new event notification system allows different objects to emit and/or listen for events. The event pool tracks all registered objects and possible events from the command line. To enable debugging of the new event system, send the following command:

```
// activate / deactivate debugging (prints message on console for each fired event):
send EVENT_POOL*DEBUGGING SET 1
```

Other commands allow the event pool to fetch a list of registered objects which emit events:

Generic VizEngine commands

```
// get list of all objects which have registered some events (which may be fired at some time)
send EVENT_POOL*REGISTERED_OBJECTS GET

// get list of all registered events of a specific object
send EVENT_POOL*REGISTERED_EVENTS GET $objectname
// example:
send EVENT_POOL*REGISTERED_EVENTS GET #727*STAGE

// get list of all registered objects+events and number of registered callbacks:
send EVENT_POOL LIST
```

19.6.1 Built-In Events

For now, the following objects have built-in events:

- **Stage** (onStarted, onStopped, onDirectorStarted, onDirectorStopped)
- **Superchannels** (onArmed, onDisarmed, onTransitionStarted, onTransitionStopped) and its subchannels (onAssigned, onLoaded, onCleared)

- **Gfx channels** (onPreSetObject, onPostSetObject)
- **Editors** (onPreSetObject, onPostSetObject)
- **Scripts** (defined by the script itself)

Additionally, Superchannels and Scripts can be configured to listen on events.

19.6.2 Technical Notes

Generally, an event can have an arbitrary name (e.g., "onLoaded") and can be sent by any object. Scripts and objects can also listen for specific events and are notified whenever that event has been triggered.

Additional notes:

- Potential events must be registered by sending objects before they are emitted (because listeners can only register to **known** events).
- The name of an object can be chosen by the object itself on creation (e.g., "SUPERCHANNEL1" or "script1") but it is always preceded by the object id of the scene. Example: "#199*SUPERCHANNEL1".

  > ⚠ **Note:** Aliases like MAIN_SCENE are not allowed.

- A KeyValueMap (Stringmap for scripts) can be attached to an emitted event.

19.6.3 Sample Scripts

Scripts can be used to emit events and to listen for specific events. See the sample scripts below.

Emit events

Sample script to emit events

```
sub OnInit()
Dim events as Array[String]
events.push("OnTriggered")
Eventpool.registerEvents("script1", events)

' use commands EVENT_POOL*REGISTERED_OBJECTS GET and
' EVENT_POOL*REGISTERED_EVENTS GET sceneID*script1 to
' verify that events are registered when script is active
' and that they are automatically unregistered when script
' is stopped/modified

' use EVENT_POOL*DEBUGGING SET 1 to
' see which events have been fired
end sub

sub OnInitParameters()
RegisterPushButton("test-button", "press here", 1)
```

```
end sub

sub OnExecAction(buttonId as Integer)
if (buttonId = 1) then
Dim params As StringMap

' now emit our event
Eventpool.emitEvent("OnTriggered", params)

' this will fail, because we have not registered the event:
'Eventpool.emitEvent("OnBlabla", params)

end if
end sub
```

Listen for events

Sample script to react on events

```
sub OnInit()
dim sceneid as String
sceneid = cstr(scene.vizid)
' Register as listener for 3 events of superchannel1 / subchannel A
eventpool.registerAsListener("#" & sceneid & "*SUPERCHANNEL1/A", "onLoaded")
eventpool.registerAsListener("#" & sceneid & "*SUPERCHANNEL1/A", "onAssigned")
eventpool.registerAsListener("#" & sceneid & "*SUPERCHANNEL1/A", "onCleared")
end sub

sub OnEvent(objectName As String, event as String, arguments as StringMap)
'println("GOT EVENT!")
'println(objectName)
'println(event)
'println(arguments)

if (event=="onCleared") then
        material.color.red = 255
        material.color.green = 0
elseif (event=="onAssigned") then
        material.color.red = 100
        material.color.green = 100
elseif (event == "onLoaded") then
        material.color.red = 0
        material.color.green = 255
end if
end sub
```

20 Shared Memory – SHM

The information in this section relates to the `VizCommunication.Map` Data Sharing mechanism. A local `VizCommunication.Map` in each Viz Engine (as part of a cluster), collects and stores data. This data can be internal data, like a scene script pushing data to the map, or data from external control applications through TCP or UDP.

This section contains information on the following topics:

- Data Sharing
- External Data Input
- Internal Data - Interactive Scene
- Synchronization
- Snapshot

See Also

- Visual Data Tools

20.1 Data Sharing

This section describes the data sharing mechanism provided by Shared Memory Maps (SHM). A SHM is a map which holds user defined variables indexed by a string. It is exposed to the script interface via the Shared Memory data type. The variables are stored as Variant objects and thus can hold objects of any other type.

There are three types of SHM and the difference between them is their distribution locations:

- **Scene.Map**: Referenced in plug-ins as Scene. This is the map local to the current Scene only. Every Scene has one map that can be used to exchange data among the scripts in the Scene.
- **System.Map**: Referenced in plug-ins as Global. The system-wide map allows for data sharing among the scenes currently loaded into memory (e.g. data sharing between scenes in different layers).
- **VizCommunication.Map**: Referenced in plug-ins as Distributed. A distributed map that enables data sharing among the Viz Engines connected to the same Graphic Hub. The Viz Engines must be logged in with users from the same Graphic Hub group.

This section contains information on the following topics:

- VizCommunication.Map
- Map Access Through a TCP or UDP Interface
- Map Access through the Command Interface
- Map Access from Within plug-ins
- Script Examples
  - Example I
  - Example II
- Set and retrieve data from the SHM through scripting
  - To Set a Value
  - To Retrieve a Value
  - To Split the Values into an Array

- To Assign a Value to a Text Container
- To Automatically Update Graphics
- Receiving Notification on Map Changes

20.1.1 VizCommunication.Map

The map exposed by the VizCommunication type is used to share data among the client computers connected to a Graphic Hub. Whenever a script on one client adds a new value or changes an existing value in the map, the change is propagated to the other clients via database messaging to update the local copy of each client's map.
This does not necessarily mean that all local client maps are identical. When a client starts up after other clients have made changes to the VizCommunication.Map, it is not informed of those changes. See Synchronization in Shared Memory (SHM).

20.1.2 Map Access Through a TCP or UDP Interface

Elements in the distributed memory map (VizCommunication.Map) can be set (but not retrieved) through a TCP or UDP interface, exposed by Viz Artist. This option is particularly useful when writing external control applications.
For more information about map access through a TCP or UDP Interface, see External Data Input in Shared Memory (SHM).

20.1.3 Map Access through the Command Interface

There is limited support for getting and setting map elements through a command interface. The maps are accessible through these locations:

- `<scene>*MAP` for the scene specific map
- `GLOBAL*MAP` for system-wide map
- `VIZ_COMMUNICATION*MAP` for the distributed map

A full list of supported commands is at: `<viz install folder>\Documentation\CommandInterface\index.html`

From the list of commands, the commands `CLEAR`, `DELETE_ELEMENT` and `PURGE_ELEMENT` will only work when sent through the command interface of Viz Artist.
The command `CLEAR` must be run on each Engine where the MAP is to be reset (`VIZ_COMMUNICATION*MAP CLEAR`).

20.1.4 Map Access from Within plug-ins

Functions for getting and setting map elements are provided by the plug-in interface. For more information, refer to the plug-in API documentation in your Viz installation folder.

20.1.5 Script Examples

This section contains two script examples for Shared Memory (SHM):

Example I

This is an example that shows how a Scene map can be used to pass data from one script to another (within the same Scene).
This script defines the OnEnterSubContainer, which is called whenever the mouse-cursor enters any of its sub-containers. In it, the name of the subcontainer is stored in the Scene map under the key subcontainer_name.

```
‘ Script 1:
Sub OnEnterSubContainer(subContainer as Container)
Scene.Map["subcontainer_name"] = subContainer.Name
End Sub
```

Example II

This example shows how external data can be sent to all Viz Engines connected to a Graphic Hub, and how they can utilize the same data. The example also shows how to group together related external data into an array inside the Viz Engine. This is not the only way of managing variables. They could just as well be sent individually, which will then not demand extra variable splitting in a script.
Vizrt do not recommend this as a method for the import of large amounts of data. For more information see External Data Input.

To create and set a value shared by all Viz Engines, that are connected to the same Graphic Hub, send the following command through the Viz Engine Console:

```
send VIZ_COMMUNICATION*MAP SET_STRING_ELEMENT "my_variable" 1.2;10.8;20.3;15.9
CONSOLE: answer <>
send VIZ_COMMUNICATION*MAP GET_STRING_ELEMENT "my_variable"
CONSOLE: answer <1.2;10.8;20.3;15.9>
```

This example shows a command sent to the Viz Engine, that assigns a value to a variable named my_variable. This image shows a script that uses this variable, and assigns one of its values to a text container named **MyText:**

```
dim MyVariable as string
dim MyArray as array [string]
MyVariable = VizCommunication.map["my_variable"]
MyVariable.split(";",MyArray)

Scene.FindContainer("MyText").geometry.text = MyArray[1]
```

From outside a Viz Engine the shared memory can be accessed by a connection to the Viz Engine listener port 6100. For more information on ways to access shared memory, see Map Access through the Command Interface.

20.1.6 Set and retrieve data from the SHM through scripting

To Set a Value

Through a Command Console:

```
send VIZ_COMMUNICATION*MAP SET_STRING_ELEMENT "my_variable" 1.2;10.8;20.3;15.9
```

or through Script:

```
VizCommunication.map["my_variable"] = 1.2;10.8;20.3;15.9
```

To Retrieve a Value

Dim MyVariable as string

```
MyVariable = (string)VizCommunication.map["my_variable"]
```

Here, string is used to format the data from the SHM to a text value.

To Split the Values into an Array

You can then split the data inside MyVariable into a table called MyArray:

```
Dim MyArray as array [string]
MyVariable.split(";",MyArray)
```

The information in MyVariable has now been split into four:

```
‘ MyArray[0] = 1.2
‘ MyArray[1] = 10.8
‘ MyArray[2] = 20.3
‘ MyArray[3] = 15.9
```

To Assign a Value to a Text Container

One of these values can then be assigned to a text container:

```
Scene.FindContainer("MyText").geometry.text = MyArray[1]
```

To Automatically Update Graphics

When a SHM variable is updated you will have to register the value for updates in the scene:

```
Scene.Map.RegisterChangedCallback("my_variable")
```

Changes to `my_variable` will execute a subroutine called `OnSharedMemoryVariableChanged`. This subroutine can do whatever you decide with the data. For further details, refer to the callback procedures described in the plug-in API documentation.

Receiving Notification on Map Changes

Sometimes you want to be notified when a value in an SHM changes, so you do not have to poll for changes within `OnExecPerField`. This is done by calling a map's `RegisterChangedCallback` procedure, passing it the name of a key that you want to monitor. Now, whenever the variable identified by that key is changed, the `OnSharedMemoryVariableChanged` callback (if present) is invoked. Similarly, the `OnSharedMemoryVariableDeleted` callback is invoked, when the variable is deleted. Example:

```
Sub OnInit()
	Scene.Map.RegisterChangedCallback("subcontainer_name")
End Sub


Sub OnSharedMemoryVariableChanged(map as SharedMemory, mapKey as String)
	If mapKey = "subcontainer_name" Then
		Geometry.Text = (String)map["subcontainer_name"]
	End If
End Sub
```

`OnSharedMemoryVariableChanged` is passed to the map as well as the key whose variable has been changed.

Passing an empty string ("") to `RegisterChangedCallback` will make sure that `OnSharedMemoryVariableChanged` and `OnSharedMemoryVariableDeleted` is invoked whenever any variable in the map is modified or deleted:

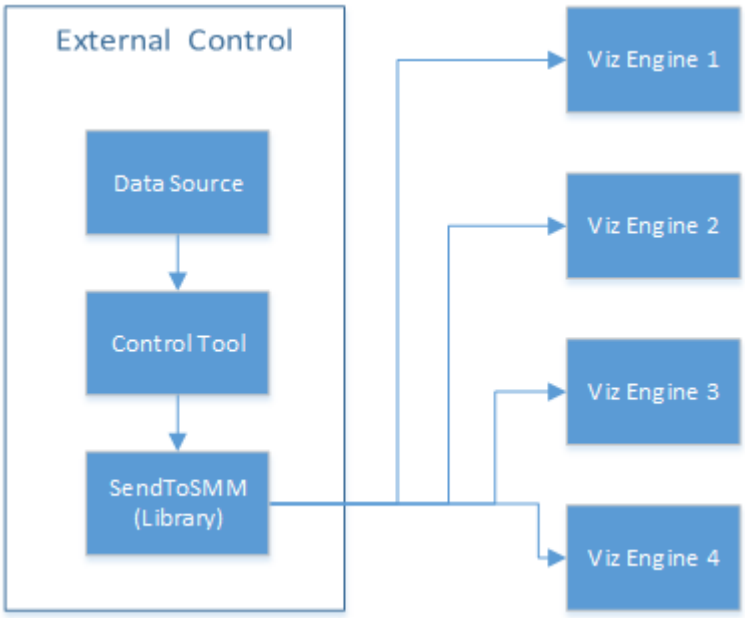
```
Sub OnInit()
Scene.Map.RegisterChangedCallback("")
End Sub
Sub OnSharedMemoryVariableChanged(map as SharedMemory, mapKey as String)
...
End Sub
```

20.2 External Data Input

Data to **Shared Memory** should be fed through the dedicated UDP or TCP IP ports for the SMM. Vizrt provides a set of development components called **SendToSMM** to make developing tools for this purpose easier. The component files are included with the standard Viz installation and located at `<viz install folder>\Tools\SendToSMM`.

Data can also be sent to shared memory through the Command Interface. Data sent through the Command Interface may be seen as a good option because the data would need to be sent to one Viz Engine only, and this Viz Engine engine would then distribute the data to the other Viz Engines. But Data sent to shared memory through the Command Interface has problems:

- Data sent through the Command Interface will block the render queue of the receiving engine causing potential frame drops. Since the data needs to be sent through a command significant more bytes are transferred over the Network.
- This Viz Engine is also a single point of failure
- The data will arrive at this one Viz Engine sooner than on all other Viz Engines
- The notification method of the Graphic Hub is used to distribute the data and can cause additional load for the Graphic Hub
  The preferred method to send data is to use the 'SendToSMM' library (or an equivalent) to send the data to the individual Viz Engines.



The communication protocol for the import of Shared Memory data depends on the type and final output of the data. There are set protocols to use with large amounts of data, in which all of the data must reach its destination graphic, and also where large amounts of data must be received, quickly, but some loss of data is acceptable.

For use-cases, see:

- TCP Protocol
- UDP Protocol

- Plug-in API
- Command Interface

⚠ **Note:** It is also possible to import data through Multicast. This method is not recommended as it can pollute the network.

20.2.1 TCP Protocol

The SHM TCP communication protocol guarantees a reliable delivery of packages. It is a much more efficient than the Command Interface, but not as fast as the UDP Protocol.

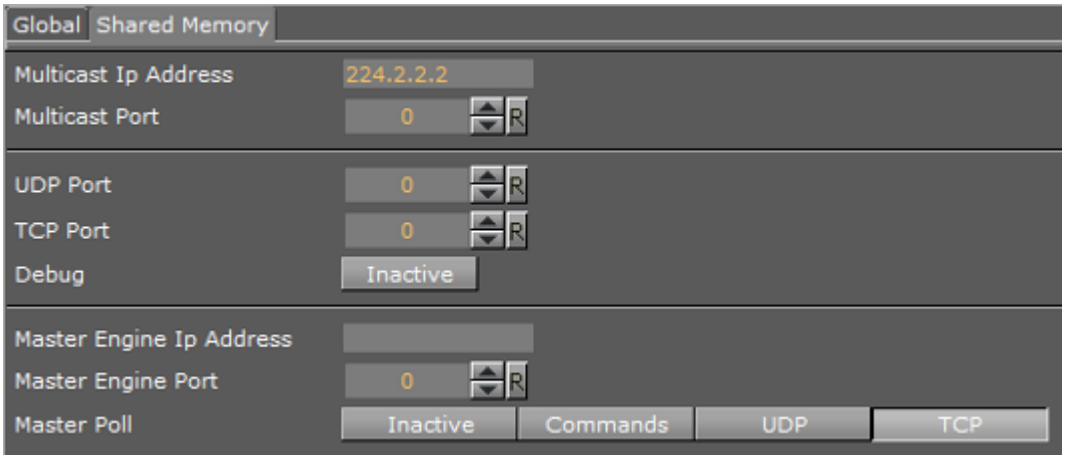
Use cases for a TCP connection could be finance stocks and currencies, or election result information, where the requirement is to deal with large amounts of information, and all of this data must reach its destination graphic. A single piece of lost data can have economic consequences, falsify charts, show mathematically wrong results, etc.

A TCP connection to a Viz Engine can be held open for a long time (this is recommended), and should not be opened and closed between sending variables.

⚠ **Note:** The default maximum number of TCP connections is limited to 255. Within this number of connections a User defined limit of maximum connections can be set (see **To Limit the Number of TCP Connections** in the **Configuring Viz** section of the Viz Engine Administrator Guide).

⚠ **IMPORTANT!** The external program which provides the data, must connect and send the data to each Viz Engine individually. Vizrt provides a C# library for this purpose, called **SendToSMM**, as part of the Viz installation.

To Use TCP for Shared Memory



1. Open Viz Configuration.

2. Click on **Communication**.
3. Click on the **Shared Memory Properties** tab.
4. In the Shared Memory panel set these parameters:
    - **TCP Port:** Vizrt does not recommend a specific port. Always make sure that the selected port is not in use by any other program on the same subnet.
5. Click **Save**.
   When accessing the *VizCommunication* Memory Map, the syntax for sending the key-value pairs is `<key>|<value>`, followed by a null-termination character:

- `key|value\0`

  > ⚠ **Note:** `Key` and `value` are divided by the pipe character, also known as vertical bar: **|**

  Multiple key-value pairs can be sent at once. This requires that each pair is null-terminated:

- `key1|value1\0key2|value2\0`
  When accessing a *Scene* or *System* shared memory map, the syntax requires the inclusion of further information.

- Syntax for accessing the VizCommunication shared memory map:
    - `SharedMemoryMap_SetValue|<key>|<value>\0`
    - `SharedMemoryMap_SetValueDistribute|<key>|<value>\0`
    - `SharedMemoryMap_Createitem|<key>\0`
    - `SharedMemoryMap_Deleteitem|<key>\0`
- Syntax for accessing the Scene shared memory map:
    - `SMM#<scene object id>_SetValue|<key>|<value>\0`
    - `SMM#<scene object id>_Createitem|<key>\0`
    - `SMM#<scene object id>_Deleteitem|<key>\0`
- Syntax for accessing the System shared memory map:
    - `SMMSystem_SetValue|<key>|<value>\0`
    - `SMMSystem_Createitem|<key>\0`
    - `SMMSystem_Deleteitem|<key>\0`

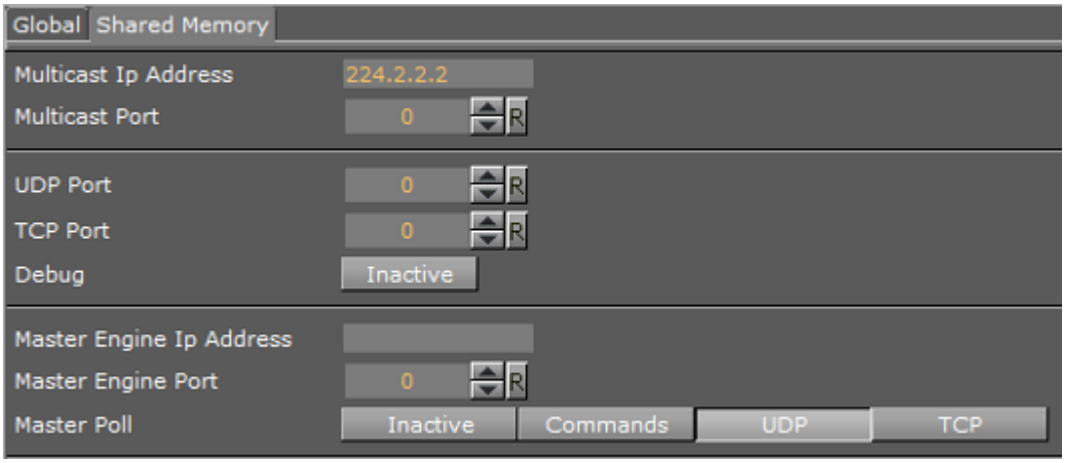
> ⚠ **Note:** A C++ library to write to shared memory maps and an example is included with the local viz installation: `<viz installation location>\Tools\VizControlLib`

20.2.2 UDP Protocol

The SHM UDP communication protocol should be used for the delivery of volatile data. It is quicker than the TCP Protocol, but less reliable, and is much more efficient than the Command Interface.

A use case for UDP would be Motor Sports, where data like speed, velocity, etc., is required. This is where there is a requirement to deal with large amounts of data, but not all of this data must reach its destination. A single piece of data lost will not affect the constant data update.

To Use UDP for SHM



1. Open Viz Configuration.
2. Click on **Communication**.
3. Click on the **Shared Memory Properties** tab.
4. In the Shared Memory panel set these parameters:
   - **UDP Port:** Vizrt does not recommend a specific port. Always make sure that the selected port is not in use by any other program on the same subnet.
5. Click **Save**.
   The syntax for sending key-value pairs is the same as for TCP and UDP Synchronization.

20.2.3 Plug-in API

An option to manipulate data in SHM is by a plug-in interface.

A use case would be where a TCP or UDP connection cannot be used, or is not to be used. It is possible to write a plug-in to import data (e.g. from an XML file, another database, etc.) and push it to SHM.

Another use case would be an interactive Scene (see Internal Data (Interactive Scene)).

> ⚠ **Note:** The plug-in API is documented in the plug-in API documentation and comes with the Viz installer (go to, *Start -> All Programs -> vizrt -> Viz 3.x -> Documentation -> plug-in SDK Documentation -> Classes -> Class List -> Shared_Memory*).

20.2.4 Command Interface

For small and single value changes, the Command Interface of Viz Artist can be used, for example, to update the headline in a Scene.

⚠ **IMPORTANT!** A command operation can block the renderer. If there are too many commands within a short time-frame, or if commands containing a large amount of data is being sent, this can result in not rendering real-time anymore.

⚠ **Note:** Vizrt does not recommend this as a method for data import.

Any external program should consider the performance of the single or all connected Viz Engines. If there is a burst of thousands of SHM variables this can have implications on the Viz Engine rendering performance (Current (CUR) and Maximum (MAX)).

A full list of commands is available at: `<viz install folder>\Documentation\CommandInterface\index.html`

⚠ **Note:** From the list of commands, the commands `CLEAR`, `DELETE_ELEMENT` and `PURGE_ELEMENT` will only work when sent through the command interface of Viz Artist.

⚠ **IMPORTANT!** The command `CLEAR` must be run on each Engine where the MAP is to be reset (`VIZ_COMMUNICATION*MAP CLEAR`).

Whenever a new entry is made in the map (a new key-value pair) or values are changed, then the change is propagated to the other Viz Engines through a database messaging service to update the local copy of each Viz Engine's map (this only works when sent over the general communication port of Viz Engine (default port 6100)).

Command Examples

`VIZ_COMMUNICATION*MAP` can be used to access the map.

`SET_DOUBLE_ELEMENT` and `GET_DOUBLE_ELEMENT`

⚠ **Example:** `VIZ_COMMUNICATION*MAP SET_DOUBLE_ELEMENT "my_double" 1.2`

⚠ **Example:** `VIZ_COMMUNICATION*MAP GET_DOUBLE_ELEMENT "my_double"`

20.3 Internal Data - Interactive Scene

When data is modified on one Viz Engine through a script or through a plug-in, the data change will get reflected on the other Viz Engines automatically.

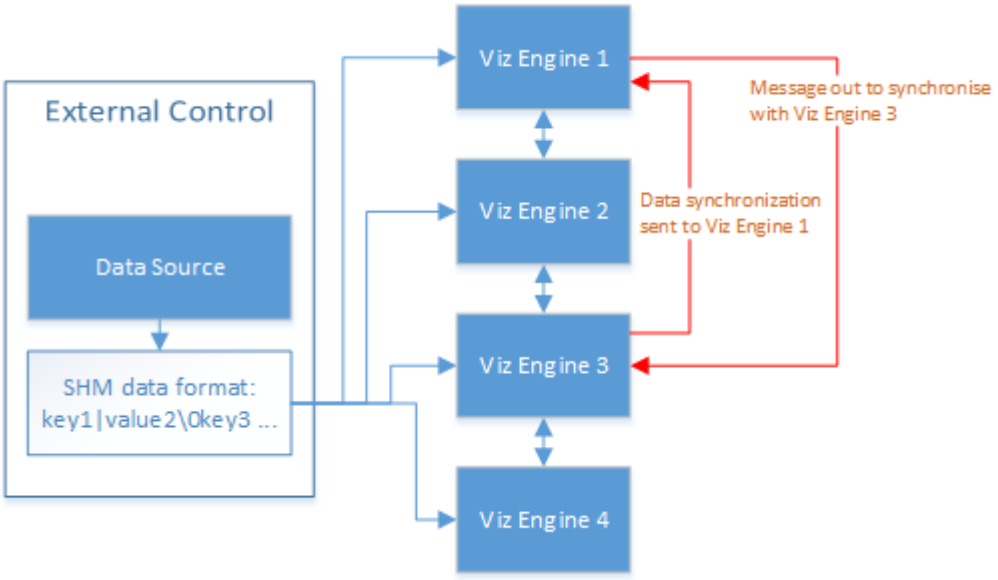
A use case could be a touch screen scene which modifies data, which is also used for HD-SDI Viz Engines or Viz Engines driving a Video Wall.

This synchronization uses the Graphic Hub as a relay. Therefore it is important that all Viz Engines, which are to receive the data, are connected to the same Graphic Hub, which use the same user or at least the same group.

20.4 Synchronization

If a Viz Engine is restarted or added to the cluster of Viz Engines for playback, the `VizCommunication.Map` data will not available on this Viz Engine. The local map on a new or restarted Viz Engine can be synchronized with the local map on another running Viz Engine in the same cluster.

> ⚠ **Example:** If Viz Engine 1 restarts it will look to, for example, Viz Engine 3 to update it's local VizCommunication.Map.



A startup Viz Engine can be synchronized through:

- TCP
- UDP
- An External Control Application
- Commands

This section contains information on the following topics:

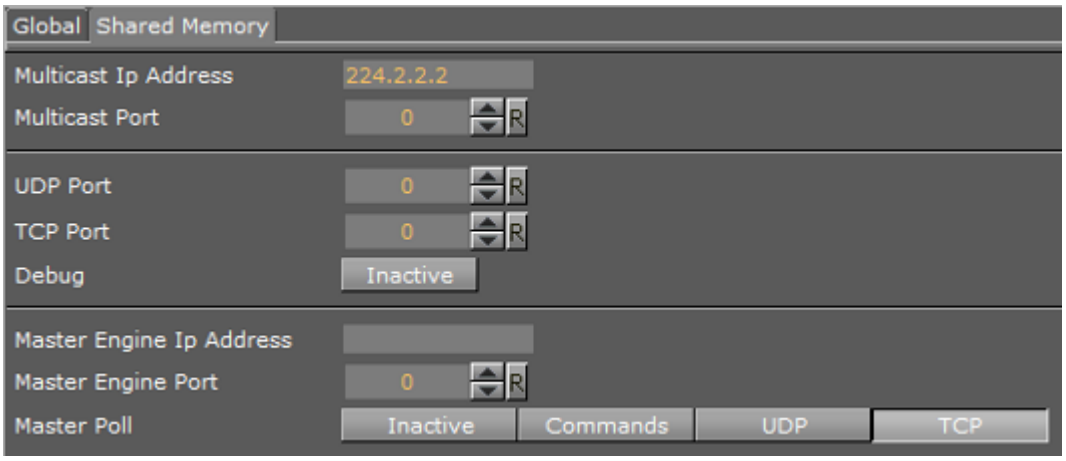
- TCP and UDP Synchronization
- External Control Synchronization
- Command Synchronization

20.4.1 TCP and UDP Synchronization

Use the procedure detailed below to synchronize a restarted or added Viz Engines, in a cluster, with the TCP or UDP protocol:

- **TCP**: The recommended communication protocol to synchronize maps on startup because it is reliable and efficient (see TCP Protocol).
- **UDP**: For fast communication, but has drawbacks. For instance, packets can get lost (see UDP Protocol).

To Synchronize a Viz Engine with TCP or UDP



1. Open Viz Configuration.
2. Click on **Communication**.
3. Click on the **Shared Memory Properties** tab.
4. Enter the port number for either **UDP** or **TCP Port**.
5. In **Master Engine Ip Address**, type the IP address of a running Viz Engine to synchronize with. This be the IP address of a running Viz Engine in the same cluster system.
6. In **Master Engine Port**, type the port number to be used (default 6100).This port sends the single startup synchronization command to start synchronizing. It should be the same as the general communication port of the Viz Engine set in **Master Engine Ip Address**.
7. In **Master Poll**, select either **UDP** or **TCP**, corresponding to the set port.
8. Click **Save**.

⚠ **Note:** Vizrt does not recommend a specific port number. Always make sure that the selected port number is not in use by any other program on the same subnet.

20.4.2 External Control Synchronization

Synchronization can also be done from an External Control Application. The following command has to be sent to the Viz Engine which has the memory map populated:

- **From a Command Interface**

```
VIZ_COMMUNICATION SYNCHRONIZE_TO <engine port>
```

where the Viz Engine is the engine which receives the data through the Command Interface. The port is usually 6100 (standard command interface port).

> ⚠ **Note:** Vizrt do not recommend this as a method for synchronization.

- **Through TCP Communication**

```
VIZ_COMMUNICATION SYNCHRONIZE_SMMTCP <engine port>
```

where the Viz Engine is the engine which receives the data, and the port is the one that was specified for incoming TCP key|value pairs on the Viz Engine which is to receive the data (see TCP Protocol).

- **Through UDP Communication**

```
VIZ_COMMUNICATION SYNCHRONIZE_UDP <engine port>
```

where the Viz Engine is the engine which receives the data, and the port is the one that was specified for incoming UDP key|value pairs on the Viz Engine which is to receive the data (see UDP Protocol).

20.4.3 Command Synchronization

Another communication option is by Commands. Here each map entry is synchronized through commands.

It is reliable, but very slow and blocks the engine for quite an amount of time, if the map is large.

> ⚠ **Note:** Vizrt do not recommend this as a method for synchronization.

To Synchronize a Startup Viz Engine with Commands

1. Open Viz Configuration.
2. Click on **Communication**.
3. Click on the **Shared Memory Properties** tab.
4. In the Shared Memory panel set these parameters:
   - **Master Engine Port:** Enter the communication port of the Master engine (default is 6100).
   - **Master Poll:** Click on **Commands**.
5. Click **Save**.

20.5 Snapshot

You can take a data 'snapshot' (save the whole content of the map) at any time. One Viz Engine can save the whole content of the map by calling the map's `SaveToDb` procedure, and another client can read it by calling `LoadFromDb`.

The downside of this approach is that these functions block the renderer and can cause poor performance of the Graphic Hub database, if the map is stored repeatedly.

⚠ **IMPORTANT!** The resulting SHM map objects are replicated as well. So saving the map each field or every minute can result in serious problems for the Graphic Hub database (replication failing, etc.).

⚠ **Note:** For more information see the script function documentation. Go to, *Start -> All Programs -> vizrt -> Viz 3.x -> Documentation -> Script Documentation -> Data Types and procedures -> Shared Memory*, or *<viz install folder> -> ScriptDoc -> DataTypeSharedMemory*.

See Also

- External Data Input
- Synchronization

21 Third Party Applications And Files

This section details the Third Party applications that can be integrated with Viz Artist.

This section contains the following topics:

- Adobe After Effects
- CINEMA 4D
- FBX Files
- TriCaster
- Ncam AR for Unreal Engine 4

21.1 Adobe After Effects

Viz Artist can import Adobe After Effects project files directly, as it is capable of reading After Effects Project files (`.aep`) and store its compositions as Scenes.

This section describes how to work with Adobe After Effects and Viz Artist. For the system requirements and integration setup, see System Requirements. The section contains the following topics:

- System Requirements
- Importing Adobe After Effect Files
- Import Limitations

21.1.1 System Requirements

The following are the minimum requirements to import After Effects scenes to Viz Artist:

> ⚠ **IMPORTANT!** You must install both Viz Artist and Adobe After Effects on the same machine to use the After Effects importer.

- Viz Artist 64-bit Video version.
- Graphic Hub 2.4 or higher.
- Matrox X.mio3 IP, X.mio3, X.mio2 Plus, or X.mio2 video board, or the Matrox X.open USB dongle.
- Adobe After Effects. Refer to the following version compatibility table:

Viz Artist	Adobe After Effects
3.9.0, 3.10, 3.11	CC 2017
3.8.3	CC 2015.3
3.8.2, 3.8.1	CC 2015

Viz Artist	Adobe After Effects
3.8.0	CC2014
3.7.2, 3.7.1	CC
3.7.0	CS6

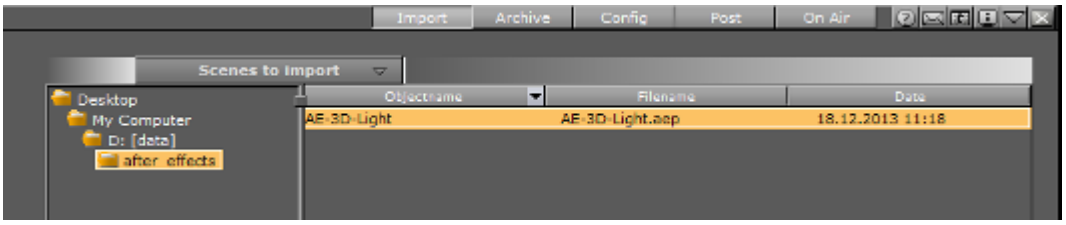
The Viz Artist installer copies the plug-in file, `AEExport.aex`, to the Adobe After Effects folder. If multiple versions of After Effects are found, the installer copies the plug-in to the highest Adobe After Effects version number. If the installer cannot find an existing installation of Adobe After Effects, it does not copy the plug-in to the system during installation.

Because of this, you must install Adobe After Effects before installing Viz Artist for the After Effect Import plug-in to work. If you have installed the software in the wrong order, run the Viz Artist installer again and select the **Repair** option. This will install the file, and resolve this issue.

21.1.2 Importing Adobe After Effect Files

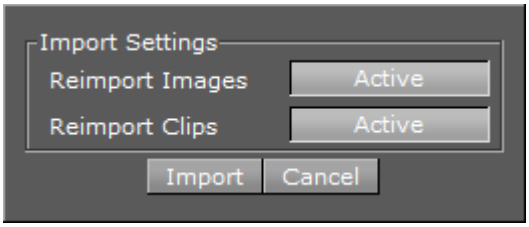
When you import an Adobe After Effects project to Viz Artist, it automatically creates containers for text and image items, and then adds the Control Text and Control Image plug-ins for each text or image container.

To import an Adobe After Effects project, click Import in the Main Menu, and import the project as a scene.



The import process creates the sub-folder `import` within the target folder, and then imports images, fonts, and sub-scenes, to this sub-folder. It does not import fonts that already exist on the Graphic Hub, but references the already existing fonts in instead.

When re-importing a scene, you must select if Viz Artist will also re-import images and clips:



- Import and overwrite existing files (**Active**) or,

- Do not import files which already exist (**Inactive**)
  Set the respective **Import Settings** to `Inactive` if there are no changes to the image or clip files. The speeds up the import process, as it will bypass the Adobe After Effects renderer.

Items that Viz Artist cannot interpret, or that cannot be rendered in real-time (such as special effects), are rendered as a clip by the After Effects renderer. Because of this, Open **Viz Configuration** and make sure that the **Output Format** setting in Viz Artist matches the **Composition** setting in Adobe After Effects. Also, activate the number of clip channels you require in the **Video Input** section, and enable the **w/Alpha** option. Otherwise, Viz cannot play out these clips.

Viz Artist automatically imports these clips to the Graphic Hub as a **Video Clip** item, and composes them in the imported Scene. Like any other file in the Graphic Hub, the Scene references these clips by UUID.

Frame Accurate Output

The imported clip is the Master Clip for the animation's director. To secure frame accurate output of the animation and clip playback, set the following settings in the **Communication -> Global** section of Viz Configuration:

- **Frame Accurate Viz Comm.**: `On`
- **Command Field Dominance**: `Odd Retrace Counter`
  In the **Video Input -> Clip Input** section, set:
- **Texture Delay**: `0`
  Finally, you need to edit the configuration file manually to set the following flags:
- **Matrox0.VideoOut1.FrameBufferDelay** =`0`
- **Matrox0.VideoIn1.VideoDelayDVE** =`0`
- **ClipIn1.VideoDelayDVE** =`0`

21.1.3 Import Limitations

Import of Adobe After Effects compositions is limited to the following Adobe After Effects items:

- Cameras
- Texts
- Solids
- Masks (no animated points and no feathering)
- Shapes (no animated points and no feathering)
- Images
  Items included in nested compositions are imported/exported in the same way as items that belong to a single (root) composition.

⚠ **IMPORTANT!** Two licenses are required for the After Effects import: One license from Adobe for After Effects CS6, and the AE Import license issued by Vizrt.

These items are imported with their animations (Key Frames are not baked, they are converted directly to Viz Artist Key Frames).

All items and/or effects from Adobe After Effects, which cannot be converted to Viz Artist objects, are rendered as **video clips** and imported into the Graphic Hub (see Video Clips). The import tries to combine multiple objects, which cannot be imported, and render them into one video (MPEG2 iFrame HD + Alpha). Currently, the importer is configured to use *no more* than *2 Clip-* and *16 GFX* channels for one scene (including sub-scenes). If more channels are necessary to import the composition, the importer renders layers which are actually supported as native items, but create an importable/not-importable alternation, as clip.

For example, if an imported After Effects scene would need more than 2 clip/16 GFX channels in Artist, then the importer collapses those layers step by step until the resulting import does **not** need more than 2/16 channels. However, this *may* lead to the situation that layers/items which are actually supported as native items in Viz, are still imported as pre-rendered clips. You may want to consider this when working in After Effects before importing into Viz.

⚠ **Note:** This can lead to several video clips if there are many alternations of importable and not importable layers.

Adobe After Effects offers a 2D/3D layer option. In Viz Artist there is one camera per Scene. This means that each time 2D and 3D layers are mixed in the composition the import needs to create a new sub-scene (-> GFX Channel) to set a camera accordingly and still maintain the correct layer z-sorting.

21.2 CINEMA 4D

This section details how to work with CINEMA 4D and Viz Artist.

The CINEMA 4D and Viz Artist integration imports complete geometries, lights, and animations, from CINEMA 4D to Viz Artist. Viz Artist imports scenes from CINEMA 4D without the need for a local installation of CINEMA 4D on the Viz Artist workstation. 3D objects are imported either as their direct primitive object counterpart in Viz Artist, or as meshes.

Viz Artist integrates a CINEMA 4D LiveLink tool that lets you make changes to a scene in CINEMA 4D. When Viz Artist is in On Air Mode, it updates live based on your changes. The CINEMA 4D LiveLink tool is able to update the position, rotation and scaling properties of all objects in a Scene.

⚠ **IMPORTANT!** Vizrt issues the license required for the import of CINEMA 4D Scenes.

This section contains the following topics:

- System Requirements and Configuration
- CINEMA 4D Scenes for Import
- CINEMA 4D LiveLink
- CINEMA 4D Integration Limitations

21.2.1 System Requirements and Configuration

The following are the minimum requirements to import CINEMA 4D Scenes to Viz Artist:

- Viz Artist 64-bit version
- **License**: Maxon CINEMA 4D license
- **To Import**:
  - **Viz Artist**: Version 3.7 or higher (64-bit only)
  - **License**: CINEMA 4D import 1 license
    The following are the minimum requirements to import After Effects scenes to Viz Artist:
- Viz Artist 64-bit Video version.
- Maxon CINEMA 4D. Refer to the following version compatibility table:

Viz Artist	CINEMA 4D
3.9.0, 3.8.3, 3.8.2, 3.8.1	Version R17
3.8.0, 3.7.2, 3.7.1, 3.7.0	Version R15 ⚠ **IMPORTANT!** If the CINEMA 4D version is R15 <= SP2, do the procedure detailed in `<viz install folder>\r15_sp2_fix` after installation.

CINEMA 4D Settings

To make sure that CINEMA 4D saves all the information Viz Artist requires when importing from .c4d files, enable Save Polygons for Melange in CINEMA 4D's preferences.

To enable Save Polygons for Melange

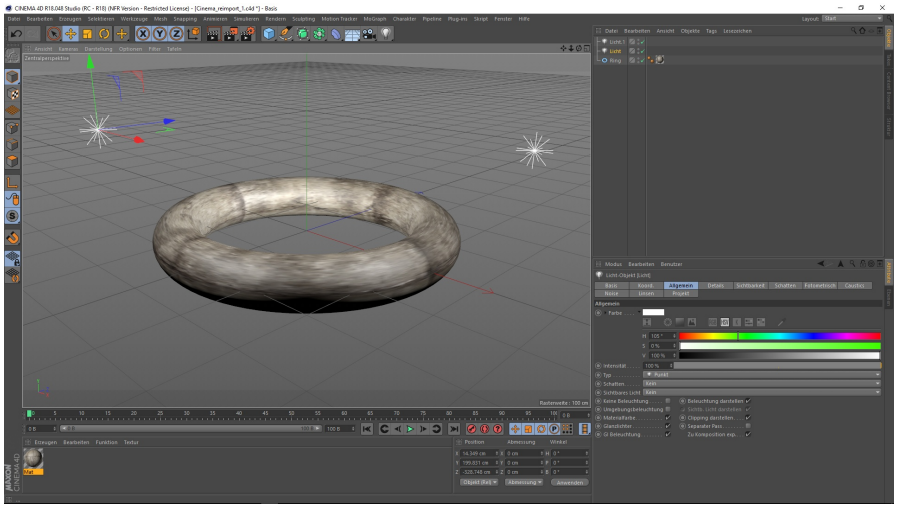
1. Open CINEMA 4D.
2. Choose **Edit** -> **Preferences** -> **Files**.
3. Check the **Save Polygons for Melange** checkbox.
   If required, check the **Save Animation for Melange** checkbox. This setting generates Key Frames for animations that CINEMA 4D's MoGraph toolset produces, by baking the key frames into the timeline. This can have a performance impact when playing out the scene in Viz Artist.

21.2.2 CINEMA 4D Scenes for Import

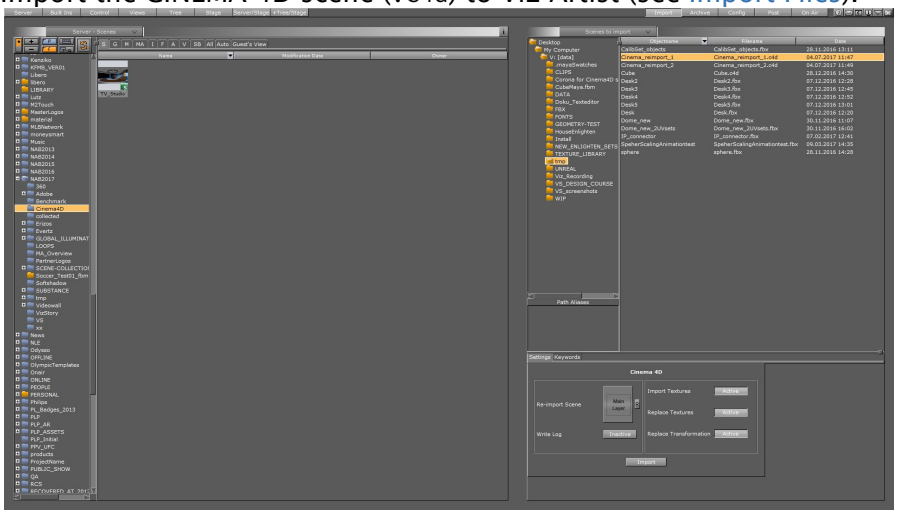
This section details how to create a CINEMA 4D scene for import into Viz Artist.

To Create a CINEMA 4D Scene for Import

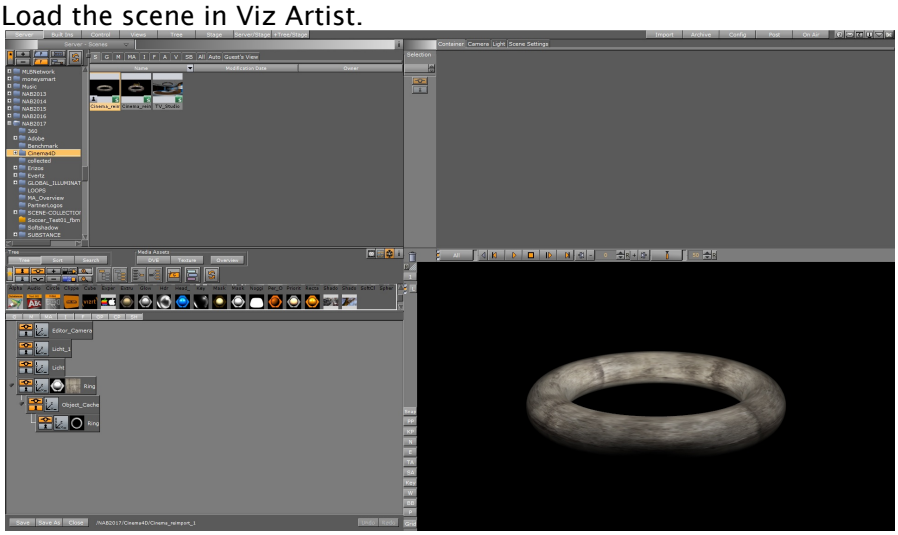
1. Create a scene in CINEMA 4D.



2. Select **File** -> **Save/Save As...**, and save the scene to your desired location.
3. Import the CINEMA 4D scene (`.c4d`) to Viz Artist (see Import Files).



4. Load the scene in Viz Artist.

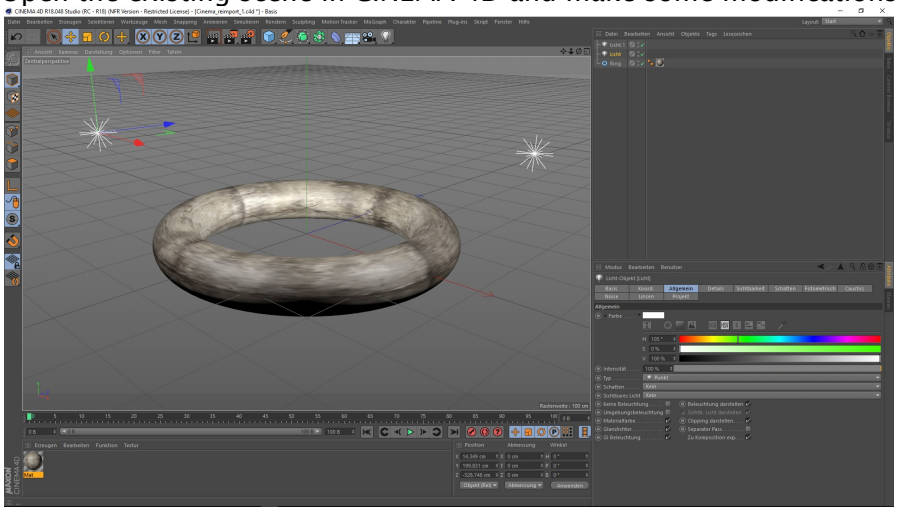


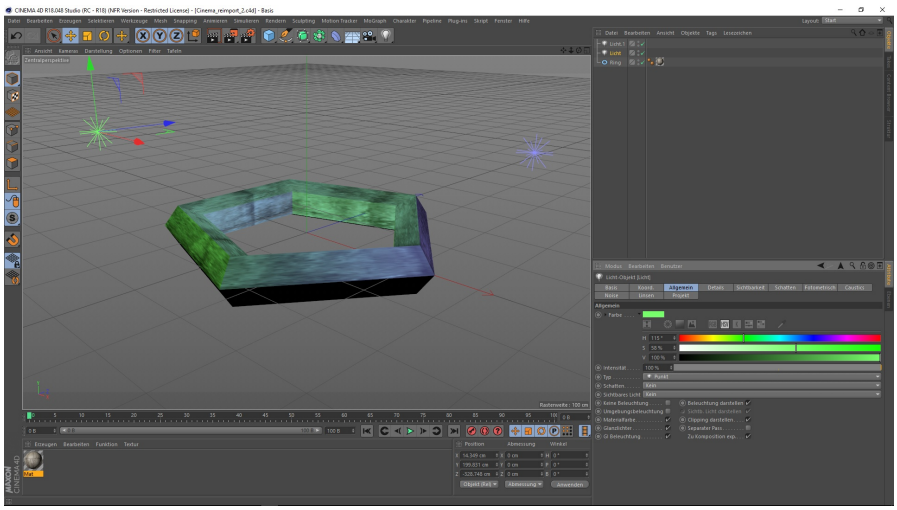
Reimporting CINEMA 4D Scenes

When you change a CINEMA 4D scene already imported to and changed in Viz Artist, you can reimport the updated scene without compromising the changes in Viz Artist.

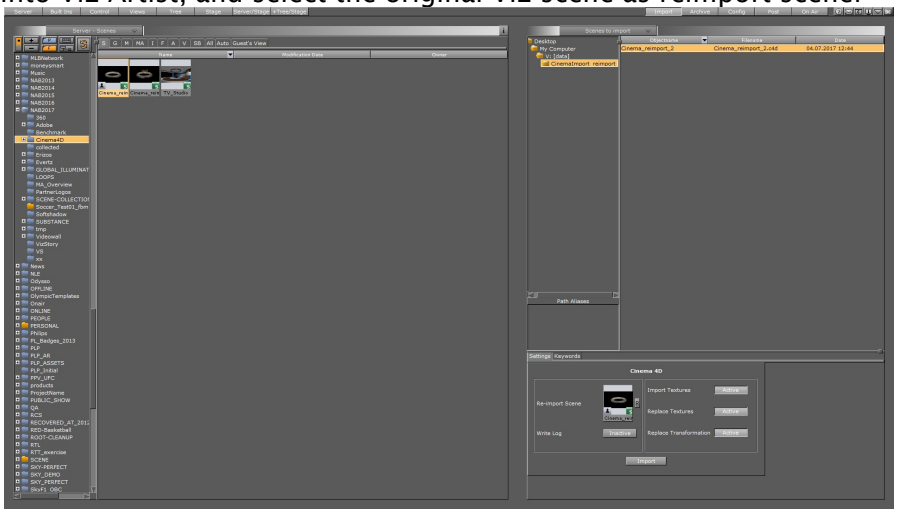
1. Open the existing scene in Viz Artist and make some modifications.

2. Open the existing scene in CINEMA 4D and make some modifications.

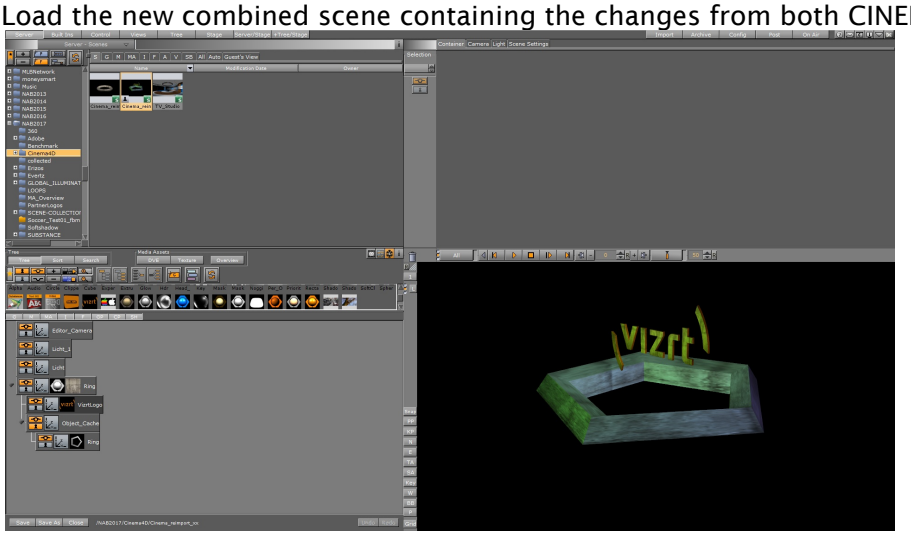




3. Select **File** ->**Save As...**, and save the scene to your desired location. Then import the scene into Viz Artist, and select the original Viz scene as reimport scene:



4. Load the new combined scene containing the changes from both CINEMA 4D and Viz Artist:



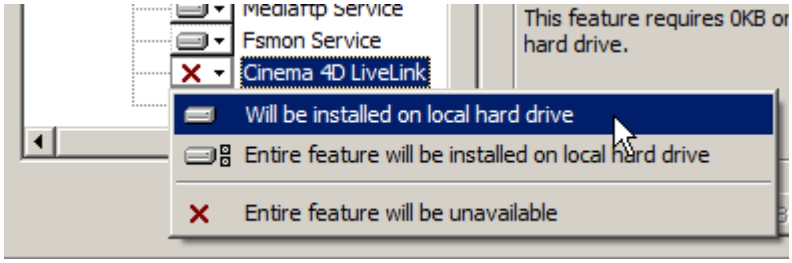
21.2.3 CINEMA 4D LiveLink

To install CINEMA 4D LiveLink, you must first install CINEMA 4D.

The Viz Artist installation files include the CINEMA 4D LiveLink tool, but it is not selected for installation by default.

To install CINEMA 4D LiveLink during Viz Artist installation

1. Install CIMEMA 4D.
2. Start the Viz Artist installation.
3. Click on **CINEMA 4D LiveLink**.



4. Select **Will be installed on local hard drive**.
5. Continue with the Viz Artist installation.

To install CINEMA 4D LiveLink to an existing Viz Artist installation

1. Install CIMEMA 4D.
2. Start the Viz Artist installation in **Repair Mode**.
   The Viz Artist installer looks for the CINEMA 4D installation folder, and copies the LiveLink files to the CINEMA 4D plug-in folder.

⚠ **Example:** `C:\Program Files\MAXON\CINEMA 4D R16\plugins\vizrt-live-link-0.1.3`

After installing the LiveLink files, they will remain in the CINEMA 4D installation folder even if you remove the LiveLink feature later on.

Troubleshooting CINEMA 4D LiveLink installation

If the installation displays the **CINEMA 4D LiveLink will not work** warning, check the following possible solutions:

- **The Cinema 4D installation location is not the default folder**
  - Manually copy the LiveLink files to the Cinema 4D **plugin** folder.
  - The LiveLink files are in the Viz Artist installation folder: `<viz install folder>\vizrt-live-link-0.1.3`
- **CINEMA 4D has not been installed prior to Viz Artist installation**
  - Complete the Viz Artist installation.
  - Install CINEMA 4D.
  - Start the Viz Artist installer again and select **Repair mode**. This makes a copy of the LiveLink files to the CINEMA 4D plugin folder.

To Open a LiveLink

⚠ **IMPORTANT!** Open the same files in Viz Artist and CINEMA 4D. Because Viz Artist and CINEMA 4D communicate using UUIDs, opening different file versions will cause the live link to fail.

1. Open Viz Artist and CINEMA 4D.
2. Open the same scene in Viz Artist and CINEMA 4D.
3. Set Viz Artist to On Air Mode.
4. In CINEMA 4D:
   a. Click on any entry in the CINEMA 4D Scene Tree.
   b. Click on **Tags.**

      ⚠ **Tip:** The Tag properties panel always shows the current connection status.

   c. Click on **Vizrt LiveLink.**
   d. In the Vizrt Realtime Demo properties panel, click on **Enable.**
   e. Enter the Viz Artist/Engine **Host** name.
   f. Enter the Viz Artist/Engine **Port** number.
   g. Click on **Reconnect.**
   Changes made to the scene in CINEMA 4D now reflects automatically in the scene in Viz Engine. The Live Link works only one way, from CINEMA 4D to Viz Artist. Moving objects in Viz Artist will not transfer back to CINEMA 4D.

21.2.4 CINEMA 4D Integration Limitations

The following limitations apply to the CINEMA 4D and Viz Artist integration when importing CINEMA 4D scenes:

- Cameras
- Lights
- Materials & Textures
- General Limitations
- CINEMA 4D LiveLink Tool

Cameras

- Maximum number of cameras in the scene: 16, including the editor camera.

Lights

- Maximum number of lights in the scene: 8.
  There is no support for a scale value other than 1.0 for X, Y, Z.
- Supported CINEMA 4D light types:
  - Omni
  - Spot
  - Distant
- Unsupported CINEMA 4D light types:
  - Parallel
  - Rectangular Spot
  - Parallel Spot
  - Parallel Spot Rectangular
  - Photometric
  - Tube
- There is no support for:
  - Light intensity parameter
  - Shadows
  - Near/far clipping
  - Noise
  - Caustics
  - Lens Effects
  - Reflection
  - Visible Light

Materials & Textures

- Multi-texturing is not supported.
- The texture that will be imported into Viz Artist is only the one on the Color channel of the last texture tag of an object (the one which shows on the top layer).
- Textures generated by a CINEMA 4D shader (and CINEMA 4D shaders in general) are not supported.
- There is no support for:
  - Polygon selections
  - Sides

- Seamless or Repetitions settings

General Limitations

- By default, Viz Engine does not render the backface of objects. To do this, add the Expert container plug-in.
- The **Save Animation for Melange setting** in CINEMA 4D generates Key Frames for animations produced by CINEMA 4Ds MoGraph toolset, among others. This setting bakes the animations as Key Frames into the timeline, potentially leading to an enormous amount of Key Frames. This can slow down importing and playback of the scene in Viz Artist. By disabling **Save Animation for Melange**, you avoid baking the MoGraph animations and those Key Frames, while keeping other animation Key Frames intact.
- Sky, Environment, Background, Foreground, and Physical Sky objects, are not supported.
- The Floor Object is imported as a rectangle of the size 2000x2000, just as shown in the Editing mode in CINEMA 4D. After import, you can scale its size, as you require.

CINEMA 4D LiveLink Tool

LiveLink only updates changes in the position, rotation and scale of objects. It does not support changes in other parameters.

21.3 FBX Files

Viz Artist's built-in FBX Importer reads Autodesk's exchange file format for 3D assets. FBX files (.fbx) are normally saved in a binary (or native) format, but they can also be saved in ASCII format. Binary FBX files and ASCII FBX files both use the `.fbx` filename extension.

Which features of a scene, that can be carried over to a Viz Scene, depends on:

- **FBX support of a 3D modeling application**
  The FBX file format supports many common features of a 3D scene. But not every feature offered by the major 3D modeling packages can be covered by an interchange format. Refer to the documentation of the 3D modeling application to discover which features can be exported to the FBX format.
- **Availability of the contained features in Viz Engine**
  Viz Engine's feature is optimized for real-time. Features which are not available in Viz Engine cannot be imported and are ignored.

  > ⚠ **Note:** Certain effects can be technically possible in Viz Engine but would require an elaborate conversion which may not be available.

  To import an FBX file to Viz Artist, see Import of Files and Archives.

This section contains information on the following topics:

- Scene Graph Traversal
- Transformations
- Export FBX Files from 3ds Max and Maya

- Reimporting Maya FBX files

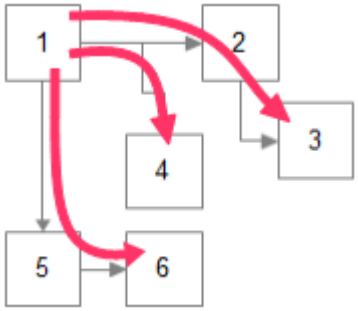
21.3.1 Scene Graph Traversal

The Scene representation in the form of a Scene Graph is common to all 3D applications, the FBX file format is also built this way.

When an FBX file is imported to Viz the Scene Graph is traversed node by node. All child nodes are traversed before advancing to the next sibling node.

This is how FBX nodes are transformed into Viz containers:

Scene graph traversal, child nodes first



FBX node types and how they are translated into Viz scene tree items

FBX Node	Child Nodes	Viz Tree
Empty	Imported	Group
Camera	Ignored	Imported into global camera
Light	Ignored	Imported into global light
Mesh	Imported	Geometry
Other	Imported	Group

21.3.2 Transformations

The computation of transformations is more streamlined in Viz Engine than in FBX because the focus is on real-time graphics. To limit computing time spent on transformations the number of parameters incorporated in computing the final transformation matrix has deliberately been kept low in Viz.

In contrast, in a 3D modeling application the number of parameters incorporated in computing the final transformation matrix is high, to give maximum flexibility.

⚠ **Note:** If unsupported transformation features are contained in a FBX scene, Viz Engine will try to provide them by combining several transformations into one (maintaining the transformation order). This limits animation and can only work to a certain degree, after which the engine will fall back to baking the imported transformation onto the imported vertices.

- **Baking transformations:** This is the fallback which may be chosen for imported nodes as necessary. It implies computing the transformation matrix with the FBX formula and to apply the result to the imported vertices.
  - **The good:** To Bake the imported transformations onto the imported vertices basically means that arbitrarily complex transformations can be imported into Viz Engine.
  - **Drawback:** Any information on how the result of a complex transformation has been obtained is discarded through baking. In particular individual items of the transformation cannot be animated anymore.
    This section also contains information on the following topics:

- Avoid Transformations Being Baked
- Avoid Transformations Being Combined

Avoid Transformations Being Baked

If unsupported features are detected the transformations of those nodes are Baked, which means that individual transformations are not retained in a Viz Scene. The transformed vertices will be in the Viz Scene, but with all transformation parameters reset to their default values.

To avoid this make sure that unsupported features are not used in a Scene and, ideally, that the FBX properties which import into the same transformation in a Viz Scene are not used at the same time:

- **Do not use a scaling offset**
  An extra offset applied after rotation and before scaling. It depends on the modeling application if it is available and under which name. To be on the safe side try to limit this to the regular translation and the rotation pivot.
- **Do not use a separate scaling pivot**
  This can be tricky, depending on the 3D modeling application. In Maya the two pivots move together by default, which keeps the possibility to map them both to a single axis center in Viz Engine.
  If the modeling application moves the rotation pivot separately by default, and objects are scaled afterwards, there will be two transformations based on different pivots. Viz Engine will have to bake the results to reflect them correctly.

FBX Import: Avoid separate scaling pivot

Rotation and scaling settings	State	Result
Rotation and scaling pivot identical	Object rotated and/or scaled	Ok, fully retained in a Viz Scene
Rotation and scaling pivot different	Object not scaled	Ok
	Object scaled	Baked

⚠ **Note:** More complex transformation hierarchies can be built and multiple pivots simulated by placing the objects in groups and apply additional transformations to the group or groups. This is reproduced in the scene tree when importing to Viz Engine.

Avoid Transformations Being Combined

If, during import, two FBX properties have to be combined into a single property in Viz, they cannot be animated separately anymore. Limitations are not fully specified yet, since animation import is still under development, but to keep 100% the same control over transformations as in the original scene it is a good idea to consider these tips:

- Do not use a rotation offset
- Do not use geometric translation/rotation/scaling in 3ds Max
- Do not use unit conversion (see Export FBX Files from 3ds Max and Maya)
- Do not use axis conversion
  - FBX uses a Y-up axis system, this is also the default setting for the FBX export. If the modeling application uses a different axis system it will apply pre-rotation of ± 90° to all objects to change the orientation of the Scene to match the FBX axis system. This can interfere with animation.
  - Maya uses a Y-up axis system by default so all pre- and post-rotations will be set to zero, which is perfect. If the modeling application uses another axis system (e.g. 3ds Max uses Z-up) and problems occur it is possible to select a different option in the 'Axis Conversion' section in the FBX export settings, so that no corrective rotation has to be applied.

21.3.3 Export FBX Files from 3ds Max and Maya

To Export FBX Files from 3ds or Maya

1. From the File menu in Maya 2013, select Export (All)... or Export Selection....
2. Select a directory and enter a filename.

⚠ **Note:** Make sure that FBX is selected as the export file type to make export options show.

This section also contains information on the following topics:

- Units
- Materials
- Other Materials
- Textures
- Camera
- Lights
- Spotlights
- Exponent in Maya

Units

It depends on each modeling application how units in virtual 3D space correspond to real world units. The default working unit is centimeters. When a Scene is exported choose a scale factor in the FBX export dialog, the option is named **File units converted to** and can be found in section **Advanced Options > Units**.

The table below shows how units import from Maya into Viz Artist when different settings are used:

Maya units imported in to Viz Artist

Maya Working Units	Object size in Maya	File units converted to	Size in Viz Artist	Scaling factor
Centimeter	6	Centimeters	6	1
Centimeter	6	Millimeters	60	10

⚠ **Caution:** It is recommended that Scenes are built so that a scaling factor of 1 will do or, if this is not possible, scale the Scene up in Viz Artist after it is imported. **The reason**: If scaling is enabled during export Maya will automatically alter the Scene to preserve the hierarchical transformation results. In some cases this will cause problems, e.g. create scaling pivot points, which are not supported in Viz.

Materials

Viz Engine supports one material per mesh node. The first material found on a FBX node is imported into a Material object, which is then applied to the Viz Container.

⚠ **Note:** Multiple or layered materials are not supported.

Supported FBX material classes are:

- Lambert Shading
- Phong Shading

Both types of shading have there own set of parameters which are described in more detail in the tables below:

Lambert Shading

FBX	3ds	Maya	Viz
Ambient Color	Ambient	Ambient Color	Ambient
Diffuse Color	Diffuse	Color	Diffuse
Diffuse Factor	-	Diffuse	Diffuse Color Attenuation
-	-	-	Specular =0,1%
Emissive Color	Self-Illumination	Incandescence	Emission
Transparency Factor/Color	Opacity (%)	Transparency (Color)	Alpha

Phong Shading

FBX	3ds	Maya	Viz
Ambient Color	Ambient	Ambient Color	Ambient Color
Diffuse Color	Diffuse	Color	Diffuse Color
Diffuse Factor	-	Diffuse	Diffuse Color Attenuation
Emissive Color	Self-Illumination	Incandescence	Emission Color
Specular Color	Specular	Specular Color	Specular Color
Specular Factor	Specular Level	-	Specular Color Attenuation

FBX	3ds	Maya	Viz
Shininess	Glossiness	Cosine Power	Shininess
Transparency Factor/ Color	Opacity (%)	Transparency (Color)	Alpha

Other Materials

Other kinds of material which the modeling application supports may be mapped to a FBX Lambert or Phong material by the exporter, at least the parameters which are compatible to those material types.

Textures

FBX can associate a texture with each color channel on a material.

> ⚠ **Note:** Currently one texture for the diffuse color channel is supported.

Textures are referenced by the FBX file through their path and filename by default. This feature is broken in the current FBX SDK provided by Autodesk, i.e. the exported file may contain an absolute path and in some cases this path will not even be correct on the machine where the file was created.

Also, texture images can be embedded into the FBX file which is a lot more robust. Tick the check-box Embed Media in the export dialog to enable this.

> ⚠ **Note:** Regardless of the export parameters chosen, the FBX file can still contain incorrect texture references. If texturing problems occur use a free FBX viewer (e.g. the QuickTime plug-in) to check if the texture import really failed or if there is a problem with the Viz import.

Camera

Currently up to 16 cameras are imported. The import is limited and can give little more than the correct camera position and direction. Dependant on the transformations that have been applied to a camera, and additional parameters unique to the modeling application, the camera position and viewing parameters can deviate a lot after import.

Lights

The lights detailed below are supported, but not all their parameters are imported:

Supported lights

	Lights
3ds	Target Spot
	Free Spot
	Target Directional (Switch to 'Overshoot' to make them resemble Viz Artist's 'Infinite' lights)
	Directional (Switch to 'Overshoot' to make them resemble Viz Artist's 'Infinite' lights)
	Omni (Switch to 'Overshoot' to make them resemble Viz Artist's 'Infinite' lights)
Maya	Directional Light
	Spot Light
	Point Light

The only animation properties imported for Lights are:

- **Translation**
- **Cutoff / Cone Angle** (because hardware accelerated OpenGL lights are limited, this is unlikely to produce the exact same effect produced in the modeling software)
  Color animation is not imported because it requires separate channels for red, green and blue components.

Spotlights

For performance reasons Viz Engine only supports the standard OpenGL light parameters.

OpenGL spotlight parameters and corresponding light parameters in Viz

OpenGL	Viz	Description
Cutoff	Cone Angle	Specifies the angle between the axis of the light cone and a ray along the edge of the cone. Can also be named 'outer cone angle'.
Exponent	Penumbra	Controls how concentrated the light is. The light's intensity is highest in the center of the cone. It's attenuated toward the edges of the cone by the cosine of the angle between the direction of the light and the direction from the light to the vertex being lit, raised to the power of the spot exponent.

FBX translates a modeling application's spotlight parameters to an inner and outer cone angle. During import into Viz the exponent is approximated based on the inner angle of the spotlight. Results can be different from what is seen in Maya or 3ds.

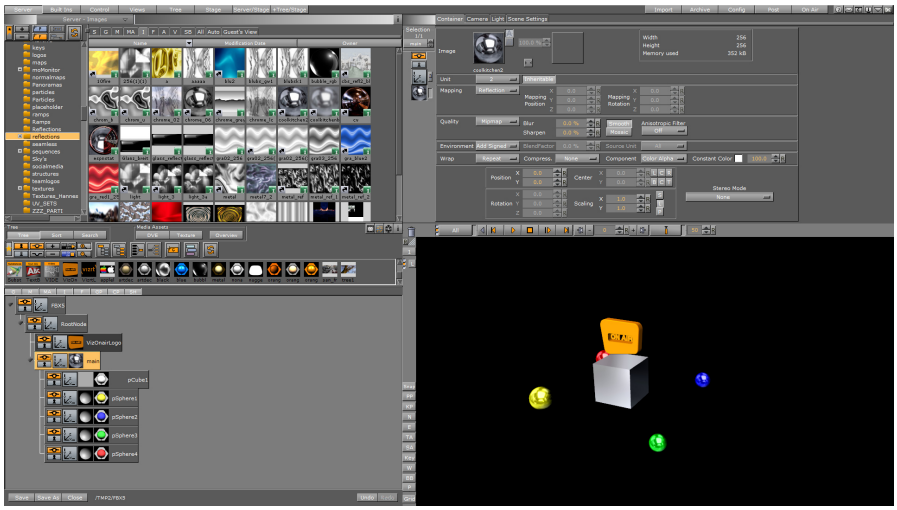
Exponent in Maya

Maya has an alternative spotlight parameter, 'Dropoff', which corresponds directly to the OpenGL parameter 'Exponent'. With the Viz Artist VizFbxExtension plug-in, this parameter can be exported. This allows work with real GL-spotlight parameters in Maya and gives a viewport lighting much more faithful to the actual rendering results in Viz. Without the VizFbxExtension plug-in, Maya does not export that parameter, because no other Autodesk product supports importing it.

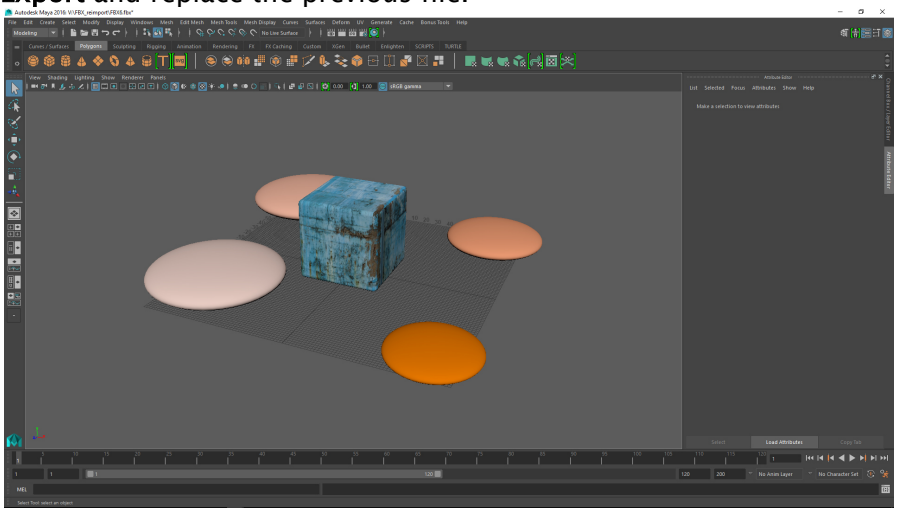
21.3.4 Reimporting Maya FBX files

When you change a Maya scene that is already in use in Viz Artist, you can reimport the Maya scene without losing changes made in your Viz Artist scene. When reimporting scenes, changes in the .fbx file and the Viz Artist scene are combined. This requires that you install a separate plug-in for Maya, which is included with the Viz Artist installer. After installing the plug-in, activate the `VizUUID.py` plug-in in Maya. When the plug-in is active, you will find a new section called **Viz UUID** in the **Extra Attributes** tab when exporting from Maya. Based on this UUID, Viz Artist determines which scenes or objects to reimport. Reimport requires that the UUIDs are identical.

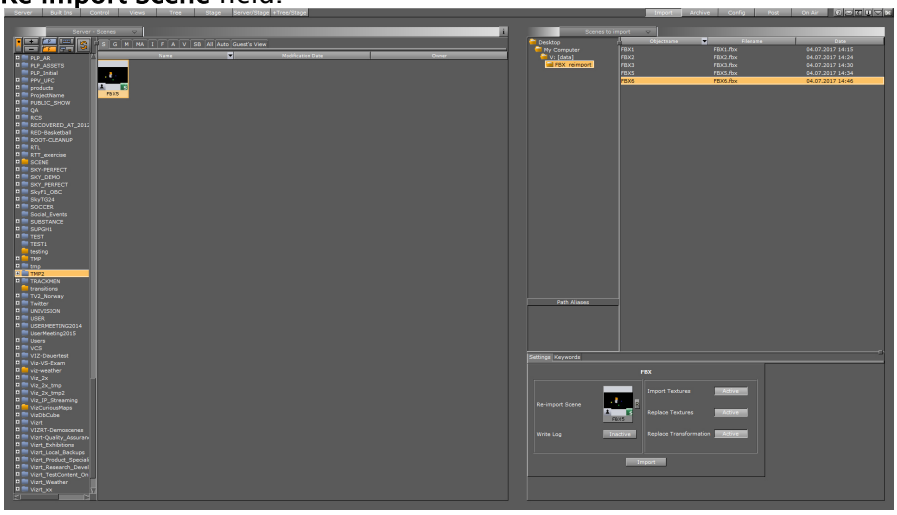
1. Create a scene in Maya and export it as `.fbx`.
2. Import the scene into Viz Artist.
3. Add different features to the Viz Artist scene, such as animations, textures, or effects. Then save the scene.



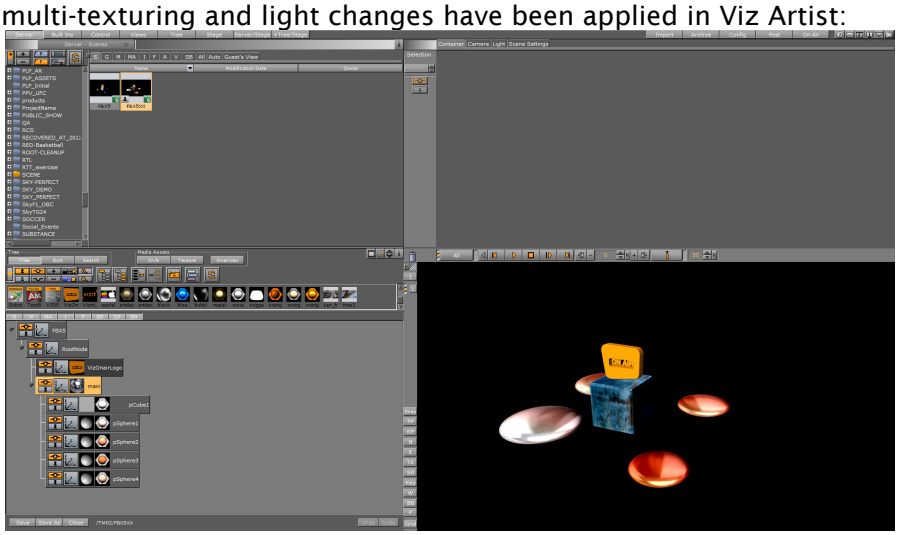
4. Make some changes to the original scene in Maya, then export it again using **Export as** or **Export** and replace the previous file.



5. Open the **Import** pane in Viz Artist. Reimport the changed .fbx file by first selecting the scene to import. Then, drag the target scene where the changes will be imported to, to the **Re-import Scene** field.



6. In this example, a changed .fbx file has been imported to the original Viz scene, where multi-texturing and light changes have been applied in Viz Artist:



See Also

- Import of Files and Archives
- Export FBX Files from 3ds Max and Maya

21.4 TriCaster

Newtek's TriCaster is a popular vision mixer for small to medium budget operations. With the Viz Engine integration for TriCaster, Viz Engine provides a fill and key IP stream which is automatically detected by TriCaster as a source for graphics overlay. Two different graphic streams can be assigned in the TriCaster (net1 and net2). Each stream is a fill key combination.

As part of the Viz Artist and TriCaster integration package, Viz Trio is included. If the integration package runs on a Trio One Box, animated preview is available within Trio. Viz Trio can also be controlled directly from the TriCaster through macros that can be mapped to keyboard and control panel shortcuts, and are routed to Viz Trio by Viz Engine. See To Set TriCaster Macros.

A second integration package includes Viz Trio on a HP ZBook, with still preview to create an ultra-portable graphics and video production solution for mobile production trucks (see Viz Trio and Viz Engine for a TriCaster Production on the Same Laptop).

This section contains the following topics and procedures:

- System Requirements
- NDI Protocol
- To Connect a Viz Engine to TriCaster
- To enable Audio pass through from Viz Engine to TriCaster
- To Set TriCaster Macros
- Viz Trio and Viz Engine for a TriCaster Production on the Same Laptop

21.4.1 System Requirements

- Viz Artist/Engine: 3.6.3 and higher, Video version

- TriCaster: All versions of TriCaster since:
  - **Build Number**: 2-2-150507B
  - **Driver** = 1.99 1-Apr-2014
  - **Core** = 24e8 29-Jul-2014
    The Viz Engine/TriCaster integration is supported for both 32 and 64 bit versions.

21.4.2 NDI Protocol

Support for the NDI protocol from Newtek was introduced in Viz Artist/Engine 3.8.3, with the latest version of the TriCaster software. This enables dual channel output to the TriCaster on one system via named video pipes. Other Viz systems may also be sending on the same network.

Systems still running the TriCaster Standard Edition software need to upgrade to the Advanced Edition 2.0 version. Contact Newtek to obtain the required installer.

Within the TriCaster, all Viz systems on the network can be seen and freely selected to any inputs. With NDI it is possible to check the number of clients actually connected to the Engine, and only render when they are connected. Thus it is possible to expose more channels than you want to actually render and only dedicate rendering time to those that are being used.

Similar to how Viz Artist configuration files are named, a particular Viz Engine appears to the TriCaster with the following naming convention:

`computerName-engineNumber`.

Examples of named VizEngines in the TriCaster/NDI world:

`Z420VIZ-WMH-0Z420VIZ-WMH-1`

The Newtek Video Monitor tool, available from Newtek, allows any video source or output on the TriCaster and/or NDI network to be viewed.

To run Viz Engine in Tricaster mode with a video board installed on the system, set **Check Video Card** to `None` in the **Video Board** section of Viz Configuration.

21.4.3 To Connect a Viz Engine to TriCaster

1. Open **Viz Configuration**.
2. Select the **Video Board** section.
3. Set **Check Video Card** to `None`.
4. Set **Software I/O Mode** to `NewTek TriCaster`.
5. Make sure the Viz Artist dongle has the TriCaster feature enabled.
6. In the TriCaster user interface, click on the drop-down menu (next to **net1** or **net2**).
7. Using the drop-down menu from step 5, connect either net1 or net2.

21.4.4 To enable Audio pass through from Viz Engine to TriCaster

To enable audio pass through from the Viz Engine to TriCaster, set `AudioOn=1` in the Viz configuration file. This is most easily done via *Artist > Config > Audio Settings > Audio Active* button = on. With this option set, the network stream to the TriCaster will have audio embedded in the signal.

21.4.5 To Set TriCaster Macros

1. Open the macro configuration panel from the TriCaster Macros menu.
2. Create a new macro called **Trio Take**.
3. Edit the macro:
   - **Shortcut**: net1 (tells TriCaster that the network address is the one where Viz Trio is installed)
   - **Value**: vizrt
   - **Key0**: trio
   - **Value0**: take
4. Assign a keyboard shortcut to the macro.
5. Start streaming from the Viz Engine, which is on the same host as Viz Trio.
6. Connect TriCaster **net1** to it.

21.4.6 Viz Trio and Viz Engine for a TriCaster Production on the Same Laptop

On a laptop, there is only one GPU and it must be shared between Viz Trio preview and Viz Engine program out. Since this will use one GPU to create the preview and the program, it is advisable to use a ringbuffer on the Viz Engine output. Depending on how heavy the Scenes are, this can go up to 15 frames.

To run Viz Trio and Viz Engine, Viz Engine should be installed for a Trio One Box setup by selecting the **Trio Box CG** platform type during installation.

1. Start Viz Trio client with the `-vizpreview localhost:50010` command line option.

   > ⚠ **Note:** The 50010 port is a dedicated preview port (check in the Viz Config file (*<viz data folder>*).

2. In the Viz Config file set:
   - `create_default_renderer = 1`
3. Start Viz Engine.

21.4.7 TriCaster NDI Support

The support for Newtek's TriCaster NDI comes in two flavors:

- Stand alone video solution
- In combination with a Matrox X.mio3 or Matrox DSX LE4, using the TriCaster NDI outputs as input channels for Viz Engine.

TriCaster Video Version

In this case, TriCaster is treated as the sole video hardware. Using the NDI protocol provided by Newtek, the TriCaster handles both inputs and outputs. The DirectShow Clip Player is used for playback of compatible files.

Matrox/TriCaster Combination

This configuration adds TriCaster NDI streaming inputs to a Viz Engine video version with Matrox hardware. The Matrox card is used for both input and output, and also handles clip playback.

Enabling the NewTek TriCaster Integration

The integration between Viz Engine and NewTek's TriCaster supports up to four inputs and outputs based on the NDI protocol. To enable the NewTek TriCaster integration, first set **Check Video Card** to **User Defined** in the **Video Board** section of Viz Configuration. Then select NewTek TriCaster as the **Software I/O Mode**.

21.5 Ncam AR For Unreal Engine 4

The Viz Engine integration with NcamAR for Unreal Engine 4 (UE4) provides an interface between the Viz Engine and UE4. This integration lets you combine the photo realistic rendering capabilities of UE4 with the seamless functionality of template-based graphics provided by Vizrt. Here you find information on the technical aspects of installation and configuration, as well as instructions for end users:

- Requirements
  - System Requirements
  - Hardware Requirements
  - Software Requirements
  - Additional Licenses
- Installation
- Configuration
- Use Cases
  - Basic Workflow
  - Augmented Reality (AR)
  - Virtual Studio (VS)
- Scene Setup in Viz Artist
  - AR Scene Setup
  - VS Scene Setup
- Camera Control
  - Camera Controlled by a Camera Tracking System
  - Camera Controlled by Viz Artist

21.5.1 Requirements

The following are the minimum requirements to use the Viz Engine integration with NcamAR for UE4:

System Requirements

- HP Z4 G4 or HP Z8 G4

Hardware Requirements

- Matrox X.mio3 or DSX LE4 video board.
- Nvidia Quadro P6000.

Software Requirements

- Windows 10 64-bit.
- Viz Artist 64-bit Video version.
- Epic Games Launcher (to install the Unreal Engine).
- NcamAR for UE4.

Refer to the following version compatibility table for an overview of supported versions of the different software components:

Viz Artist	Unreal Engine 4	NcamAR
3.11.0	4.16.1, 4.16.2, 4.16.3	2.0.0
3.11.1	4.20.0, 4.20.1	3.0.0

Additional Licenses

- NcamAR license

21.5.2 Installation

The NcamAR UE4 integration requires that Viz Artist and Engine is installed as platform type **Video** (see the Viz Engine Administrator Guide). For instructions on how to install NcamAR for UE4 and Unreal Engine 4, refer to the NcamAR reference manual provided with the NcamAR installation package.

21.5.3 Configuration

These are the necessary configuration settings to activate the NcamAR UE4 integration in Viz Artist:

1. Configure the render options:
    a. In Viz Configuration, select **Render Options**.
    b. Activate the auxiliary rendering pipeline by setting **Aux Rendering** to `Active`.

   c. Set **Aux Camera** to the camera number which is used to control the Camera Actor in UE4.
   d. Set **Render Scale** to `1.2`.
2. Activate the video input:
   a. In Viz Configuration, select **Video Input**.
   b. Activate a **Live** or **Clip** input and set it to the same video format as the configured output format of the Viz Engine.
3. Activate input **Stream 1**.
   a. In Viz Configuration, select **Video Input**.
   b. Activate input Stream 1 and set it to:
      i. `HD 720p` if configured output format of the Viz Engine is set to `720p HD Progressive`.
      ii. `HD 1080p`if configured output format of the Viz Engine is set to `1080i HD Interlaced` or `1080p HD Progressive`.
4. Configure input **Stream 1** as Aux-Return channel.
   a. In Viz Configuration, select **Video Input: Stream Input**.
   b. Set **Shared Memory > Unique Identifier** to `viz_aux_in_01`.
   c. Save the configuration changes and close Viz Artist.
   d. Open the Viz configuration file and set **stream_in_type** to `3`.

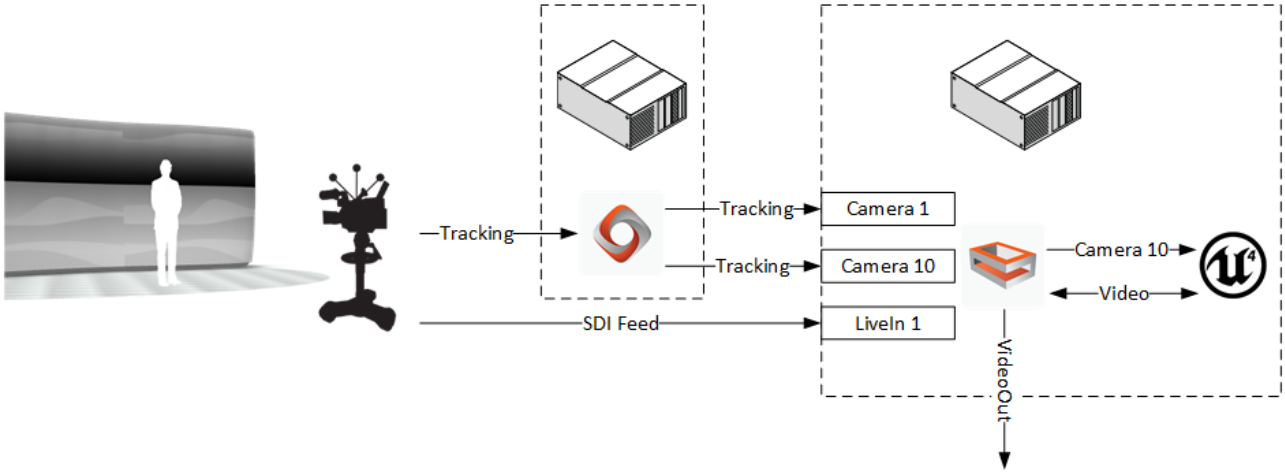
21.5.4 Use Cases

This section provides an overview of the supported use cases and their work flows. In addition, it outlines the scene set up and operation in Viz Artist.

Basic Workflow

The following provides a simplified overview of the software and hardware components, as well as the signal flows in a typical setup for the Viz Engine integration with NcamAR.

On a typical setup, both render engines, Viz Engine and UE4, run at the same time and on the same machine, which is connected to a camera tracking system through a Vizrt Tracking Hub. The image below illustrates the signal flow of the camera tracking data and video signal through the system.



1. The Tracking Hub receives the tracking data of a camera from a tracking system. Usually, the incoming tracking data is mapped to two Viz Engine cameras inside the Tracking Hub,

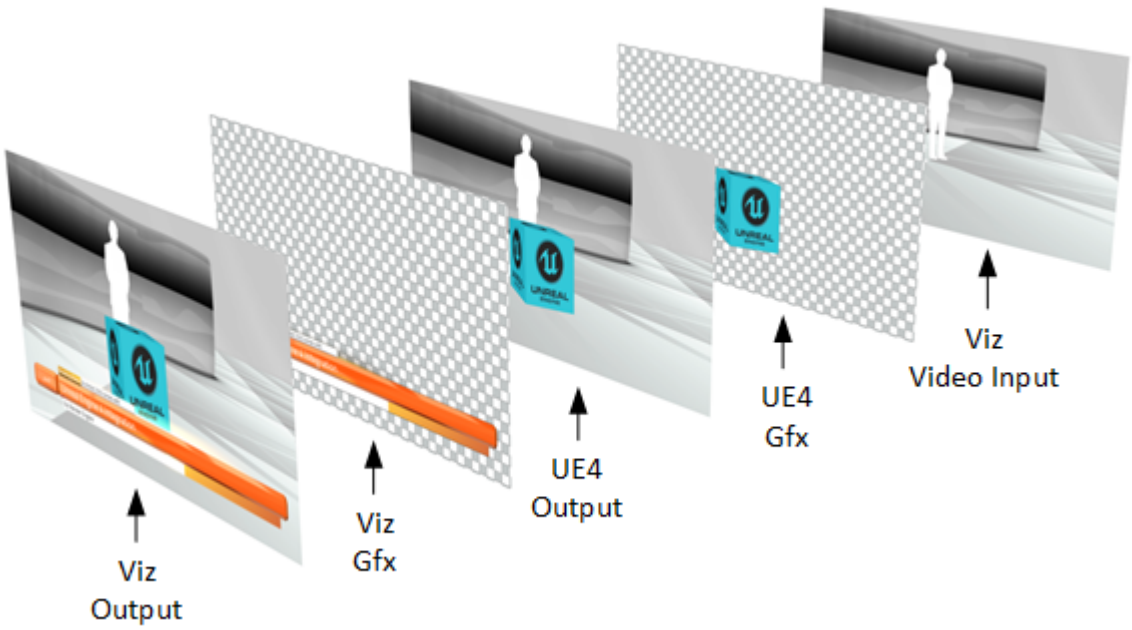
where one camera controls the rendering camera inside the Viz Engine (Viz-Camera) and the other the Camera Actor inside UE4 (Aux-Camera). This allows the user to adjust the delay between the two rendering engines using the Tracking Hub.
2. The Viz Engine receives a video signal from a camera over SDI as well as the tracking data of the Viz-Camera and Aux-Camera. The Viz Engine then applies the tracking data of the Viz-Camera to its internal rendering camera. Afterwards, the Viz Engine forwards the video signal and the tracking data of the Aux-Camera to the NcamAR I/O controller inside UE4.
3. The NcamAR I/O controller inside UE4 uses the supplied tracking data to render a matching viewpoint of the scene in UE4. Next, UE4 renders the UE4-Gfx layer and overlays it in the foreground (AR use case) or background (VS use case) of the video.
4. The NcamAR I/O controller inside UE4 sends the composed video signal (Video + UE4-Gfx) back to the Viz Engine, where additional graphic layers can be rendered on top of the received signal.
5. The Viz Engine sends the final rendering result to its SDI output.

Augmented Reality (AR)

The AR use case adds an additional compositing layer to the scene that is loaded in the main layer of Viz Artist. This additional layer is referred to as the UE4-Gfx layer, and is overlaid onto the media asset that is set as scene background of the main layer scene in Viz Artist.

The image below illustrates the compositing steps and layers of the AR use case:

1. Viz Engine supplies a RGBA image of the video input with a full-screen key to the NcamAR I/O controller.
2. UE4 renders and overlays the UE4-Gfx layer onto the video image.
3. Viz Engine renders additional graphics and overlays them onto the UE4 output.

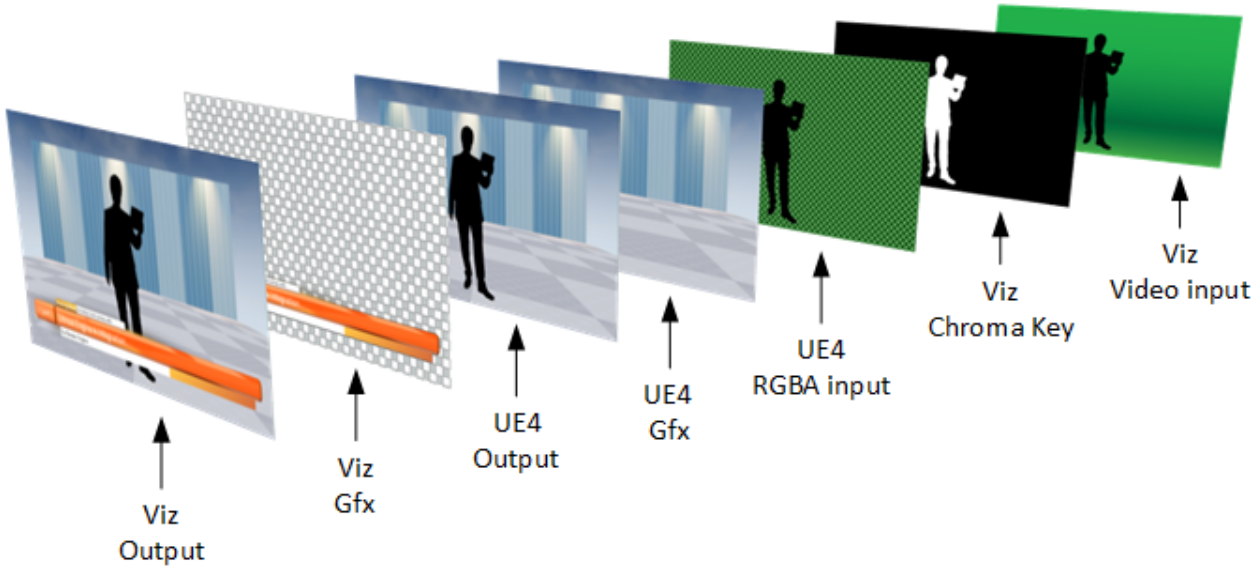


Virtual Studio (VS)

The VS use case adds an additional compositing layer to the scene that is loaded in the main layer of Viz Artist. This additional layer is referred to as the UE4-Gfx layer, and is placed behind the media asset that is set as scene foreground of the main layer scene in Viz Artist.

The image below illustrates the compositing steps and layers of the VS use case:

1. Viz Engine generates the key/alpha channel for the video input by using its internal chroma keyer.
2. Viz Engine supplies a fill and key RGBA image of the video input to the NcamAR I/O controller.
3. UE4 renders and overlays the UE4-Gfx (VS backdrop) layer behind the video image.
4. Viz Engine renders additional graphics and overlays them onto the UE4 output.



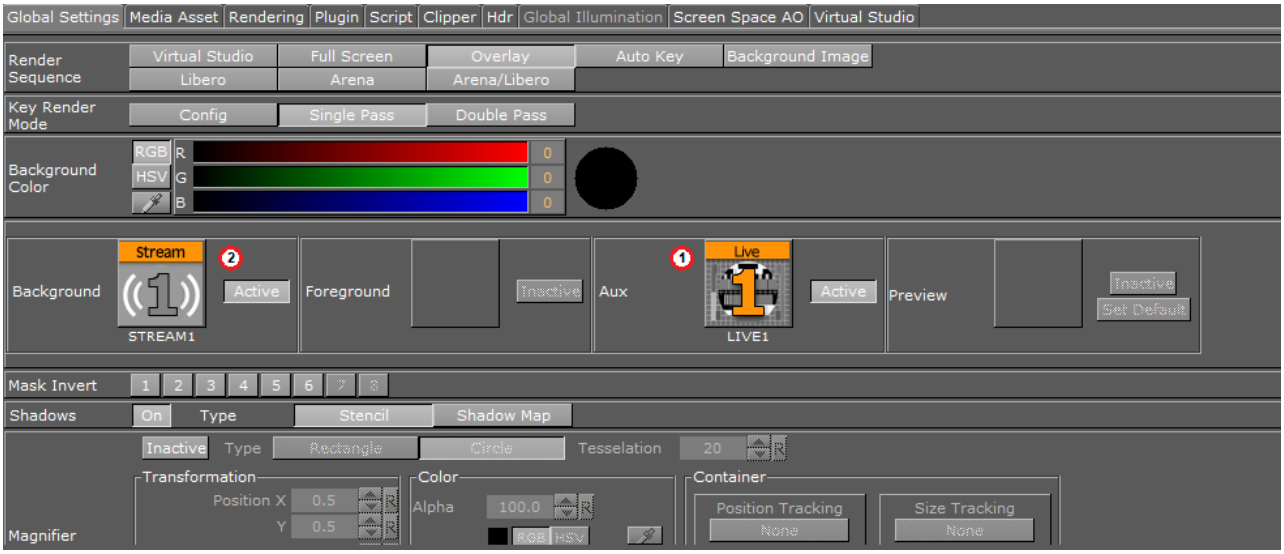
21.5.5 Scene Setup in Viz Artist

This section describes the scene setup in Viz Artist for Augmented Reality and Virtual Studio use cases.

AR Scene Setup

This section describes the scene setup in Viz Artist for the AR use case:

1. Set a video input as aux image (1).
    a. Go to **Built Ins > Media Assets**.
    b. Select the **Live** or **Clip** folder.
    c. In the **Properties** pane, go to **Scene Settings > Global Settings**.
    d. Select a Media Asset and drag it onto the **Aux Image** drop-zone.
2. Set stream input 1 as scene background (2).
    a. Go to **Built Ins > Media Assets**.
    a. Select the **Stream** folder.
    b. Select a Media Asset `Stream 1` and drag it onto the **Background Image** drop-zone.



VS Scene Setup

This section describes the scene setup in Viz Artist for the VS use case:

1. Set a video input as aux image.

1. Go to **Built Ins > Media Assets**.
2. Select the **Live** or **Clip** folder.
3. In the **Properties** pane, go to **Scene Settings > Global Settings**.
4. Select a Media Asset and drag it onto the **Aux Image** drop-zone.

- Set stream input 1 as scene foreground.
    1. Go to **Built Ins > Media Assets**.
    2. Select the **Stream** folder.
    3. Select a Media Asset `Stream 1` and drag it onto the **Foreground Image** drop-zone.
- Enable chroma keying of the video input:
    1. Click on the Media Asset in the Media Asset Manager. This opens the **Media Asset** tab in the **Scene Settings** pane.
    2. In the **Keying Mode** row, click the **Chroma** button.
    3. Set the chroma keying configuration as described in Keying Mode Configuration.

21.5.6 Camera Control

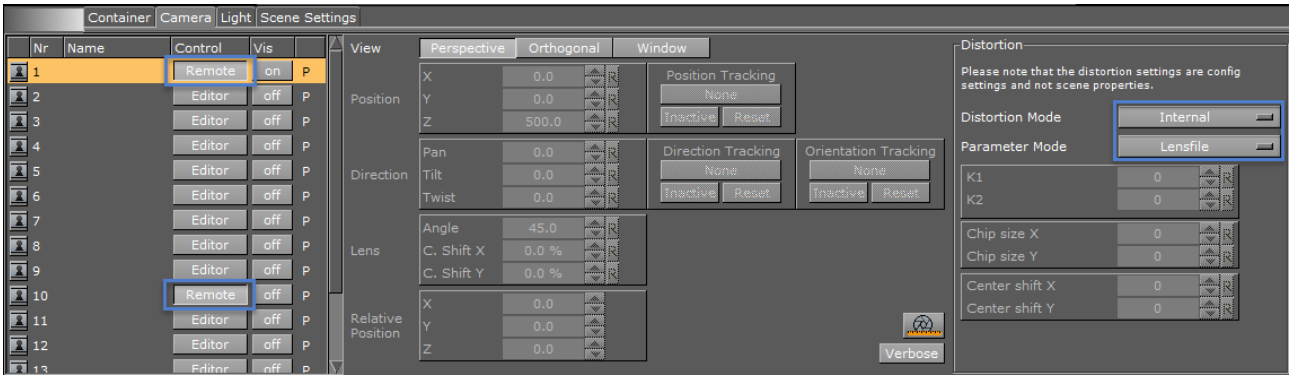
This section contains information on how to setup the camera control in Viz Artist for setups that are connected to a camera tracking system, as well as information on how to use the NcamAR for UE4 integration without a camera tracking system.

Camera Controlled by a Camera Tracking System

Enable remote control of the Viz-Camera and Aux-Camera:

1. Click **Views** and select the **Camera** tab.
2. Set the **Control** mode of the Viz-Camera (Camera 1) and the Aux-Camera (Camera 10) to `Remote`.
3. Set **Distortion Mode** for both cameras to `Internal`.

4. Set **Parameter Mode** for both cameras to `Lensfile`.
5. Set the Viz-Camera (Camera 1) as active camera.



Camera Controlled by Viz Artist

Enable internal control of the Aux-Camera.

1. Click **Views** and select the **Camera** tab.
2. Set the **Control** mode of the Aux-Camera (Camera 10) to `Editor`.
3. Set **Distortion Mode** of the Aux-Camera (Camera 10) to `Inactive`.
4. Set the Aux-Camera (Camera 10) as active camera.

22 Keyboard And Mouse Shortcuts

This section lists various keyboard and mouse shortcuts which can be used in Viz Artist.

When a keyboard shortcut is used, always make sure that the keyboard 'Caps Lock' is set to Off, unless otherwise stated.

This section contains information on the following topics:

- Application Controls and Shortcuts
- Integer and Float Controls
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- Import Shortcuts
- On Air Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.1 Application Controls And Shortcuts

Function	Key and/or Mouse
Display Context Menu	Right mouse button click
Quit Viz Artist	Double-Esc Alt+Q Alt+F4
Show/Hide Viz Command Console	Alt+C
Show Video Output Configuration Window	Alt+V
Show Performance Analyzer Window	Ctrl+Right Mouse click on Quit button
Show Import Window	F9
Show Archive Window	F10
Show Viz Configuration Panel	F11
Show On Air Window	F12

Function	Key and/or Mouse
Quit Viz Engine	Double-Esc Alt+Q Alt+F4

See Also

- Integer and Float Controls
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- Import Shortcuts
- On Air Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.2 Integer And Float Controls

Function	Key and/or Mouse
Increase/Decrease very large steps (100)	Shift+Alt Hold left mouse button and move
Increase/Decrease large steps (10)	Alt Hold left mouse button and move
Increase/Decrease normal steps (1)	Hold left mouse button and move
Increase/Decrease small steps (0.1)	Ctrl Hold left mouse button and move
Increase/Decrease very small steps (0.01)	Shift+Ctrl Hold left mouse button and move
Reset whole group	Double click one of the Rs (Reset Button)

See Also

- Application Controls and Shortcuts
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- Import Shortcuts

- On Air Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.3 Server Panel Shortcuts

This section contains the following topics:

- General Shortcuts
- General item Panel Shortcuts
- Server Panel Tree Shortcuts
- Built-in View Shortcuts

22.3.1 General Shortcuts

Function	Key and/or Mouse
Jump to top of tree	Home
Jump to bottom of tree	End
Move a page down in the tree	PageDown
Move a page up in the tree	PageUp
Scroll up the tree	Cursor up
Scroll down the tree	Cursor down
Open or close the selected tree	Return Numeric keyboard Return
Delete a folder	Delete (after confirmation)

22.3.2 General item Panel Shortcuts

Function	Key and/or Mouse
Jump to Top item	Home
Jump to Last item	End

Function	Key and/or Mouse
Move a page down	PageDown
Move a page up	PageUp
Scroll up the item list	Cursor up
Scroll down the item list	Cursor down
Scroll left in the item list	Cursor left
Scroll right in the item list	Cursor right
Open or close the selected tree	Return Numeric keyboard Return
Delete an item	Delete (confirmation required)

22.3.3 Server Panel Tree Shortcuts

Function	Key and/or Mouse
Open Graphics Hub Search Window	Ctrl+F
Show/Hide Properties of a File	Ctrl+P
Toggle Detail View	Ctrl+Y
Create new Material or new Scene	Ctrl+A This works only in Material or Scene View.
Copy File(s)	Ctrl+C
Cut File(s)	Ctrl+X
Paste File(s)	Ctrl+V This will create a link. File will not be duplicated.
Paste File(s)	Ctrl+Shift+V This will duplicate the file(s).
Export	Ctrl+E This works only in Image View.

Function	Key and/or Mouse
Rename File	F2
Duplicate Selected File(s)	Ctrl+Left mouse button click and drag
Link File(s) to a different Folder/Project	Alt+Left mouse button click and drag
Multidrop	Hold left mouse button and move Middle mouse button click

22.3.4 Built-in View Shortcuts

Function	Key and/or Mouse
Jump to plug-in which starts with a given letter	Press the first letter of the plug-in name.
Jump to Default folder for a category of plug-ins	Double-click the plug-in folder tab.
Scroll up the plug-in list	Cursor up
Scroll down the plug-in list	Cursor down
Scroll left in the plug-in list	Cursor left
Scroll right in the plug-in list	Cursor right

22.4 Scene Tree Shortcuts

This section contains the following topics:

- General Shortcuts
- Basic View Controls and Shortcuts
- Sort View Shortcuts

22.4.1 General Shortcuts

Function	Key and/or Mouse
Show Tree View	Ctrl+1

Function	Key and/or Mouse
Show Sort View	Ctrl+2
Show Search View	Ctrl+3 or Ctrl+F
Show DVE Media Asset Manager	Ctrl+4
Show Texture Media Asset Manager	Ctrl+5
Refresh	Ctrl+R
Select all Containers	Ctrl+A

<Ctrl-Shift-A> toggles the auto grab focus. This is toggles the auto grab focus to and from the Property Editor. When auto grab focus is on the Property Editor, the shortcuts below do not work.

⚠ **Tip:** This enables a user to work with objects without the need to click into the Property Editor again.

Function	Key and/or Mouse
Go to Top Container	Home
Go to Last Container	End
Go to top of page	Page Up
Go to bottom of page	Page Down
Scroll up the Container list	Cursor up
Scroll down the Container list	Cursor down
Open or close the Container tree	Return, or the numeric keyboard Return

22.4.2 Basic View Controls and Shortcuts

Function	Key and/or Mouse
Insert New Container	Ctrl+Insert The new Container will be created above the first selected Container.
Multi select Containers	Ctrl+Right click
Delete Selected Container(s)	Ctrl+Delete
Open Tree	Ctrl+O
Collapse Tree	Ctrl+I
Lock Selected Container(s)	Ctrl+L
Unlock Selected Container(s)	Ctrl+Shift+L
Group Selected Container(s)	Ctrl+G
Ungroup Selected Container(s)	Ctrl+Shift+G
Hide Selected Container(s)	Ctrl+H
Show Selected Container(s)	Ctrl+Shift+H
Undo	Ctrl+Z
Redo	Ctrl+Y
Save the Scene	Ctrl+S
Copy Selected Container(s)	Ctrl+Left click on container name and drag
Copy Selected Container(s) to Clipboard	Ctrl+C
Paste Containers from Clipboard	Ctrl+V They will be created one hierarchical level underneath selected container.
Cut Selected Container(s)	Ctrl+X

Function	Key and/or Mouse
Move Selected Container(s)	Left click on container name and drag
Move Selected Container and retain transformation properties	Alt+Left click on container name and drag
Copy Properties of Container	Left click on container Properties and drag
Merge Group	Ctrl+M
Split Container	Ctrl+Shift+M

22.4.3 Sort View Shortcuts

Function	Key and/or Mouse
Sort by Vertices	Ctrl+Q
Sort by Render Time	Ctrl+W
Sort by Texture Size	Ctrl+E
Toggle Unique for Texture Size	Ctrl+T

See Also

- Application Controls and Shortcuts
- Integer and Float Controls
- Server Panel Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- Import Shortcuts
- On Air Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.5 Scene Editor Shortcuts

This section contains the following topics:

- General Shortcuts
- Camera Controls and Shortcuts

- Orthogonal View Controls and Shortcuts
- Animation Controls and Shortcuts
- Text Editor

22.5.1 General Shortcuts

Function	Key and/or Mouse
Add new Key Frame in Stage of selected container	Return
Copy container(s)	Ctrl+Left-click and drag
Create Snapshot in RGBA	Shift+click on Snap Alt+click on Snap
Delete container	Ctrl+Delete Backspace
Cancel selection of container(s)	SPACE Left-click
Cancel selection of one out of multiple selected Containers	Ctrl+Left-click
Move Container on the z axis	Hold left and right button down
Select multiple Containers	Ctrl+Left-click
Scene Editor Shortcut Help	Ctrl+F1
Select next container down in tree hierarchy	Cursor right
Select next container in tree	Cursor down
Select parent container	Cursor left
Select previous container in tree	Cursor up
Switch additional performance information on	Middle mouse click on P
Toggle bounding box on/off	B

Function	Key and/or Mouse
Zoom in / out	Ctrl + Plus sign on the numeric keypad Ctrl + Minus sign on the numeric keypad

22.5.2 Camera Controls and Shortcuts

Function	Key and/or Mouse
Switch Camera from 1 to 10	1 - 0
Switch to Back Camera	Numeric Keypad 0
Switch to Bottom Camera	Numeric Keypad 2
Switch to Front Camera	Numeric Keypad 7
Switch to Left Camera	Numeric Keypad 4
Switch to Top Camera	Numeric Keypad 8
Change Position of Camera in X	P+Left mouse button click and drag (left-right)
Change Position of Camera in y	P+Middle mouse button click and drag (up-down)
Change Position of Camera in z	P+Right mouse button click and drag (up-down)
Orbit in X (Rotate Camera around Selected Object in X)	O+Right mouse button click and drag (up-down)
Orbit in y (Rotate Camera around Selected Object in y)	O+Left mouse button click and drag (left-right)
Orbit in z (Rotate Camera around Selected Object in z)	O+Middle mouse button click and drag (left-right)
Orient Camera against Selected Container	T
Pan Camera	I+Left mouse button click and drag (left-right)

Function	Key and/or Mouse
Reset Camera Transformation	R
Switch to Right Camera	Numeric Keypad 6
Tilt Camera	I+Middle mouse button click and drag (up-down)
Twist Camera	I+Right mouse button click and drag (left-right)
Zoom Camera	U+Left mouse button click and drag (left-right)

22.5.3 Orthogonal View Controls and Shortcuts

Function	Key and/or Mouse
Pan	Left mouse button click and drag on background
Rubberband Zoom In	Z+Left mouse button click and drag
Rubberband Zoom Out	X+Left mouse button click and drag

22.5.4 Animation Controls and Shortcuts

Function	Key and/or Mouse
Manipulate single handle	Ctrl+Left mouse button click handle and drag
Mirror Handles	Shift+Left mouse button click handle and drag
Reset handle of selected Key Frame in animation path	D

22.5.5 Text Editor

Function	Key and/or Mouse
Copy	Ctrl+C
Cut	Ctrl+X
Paste	Ctrl+V
Redo	Ctrl+Y
Select all	Ctrl+A
Undo	Ctrl+Z

See Also

- Application Controls and Shortcuts
- Integer and Float Controls
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Stage Shortcuts
- Import Shortcuts
- On Air Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.6 Stage Shortcuts

This section contains the following topics:

- General
- Stage Tree
- Stage Views

22.6.1 General

Function	Key and/or Mouse
Show Obj. Editor	Ctrl+1

Function	Key and/or Mouse
Dopesheet View	Ctrl+2 Shift+Page Down
Spline View	Ctrl+3 Shift+Page Up
Dope/Spline View	Ctrl+4

22.6.2 Stage Tree

Function	Key and/or Mouse
Jump to Top Container	Home
Jump to Last Container	End
Move a page down in the tree	Page Down
Move a page up in the tree	Page Up
Scroll up the tree	Cursor up
Scroll down the tree	Cursor down
Jump to an item (Director, Actor, or Channel), in the tree	Press the first letter of the item name
Open or close the selected tree	Return, or the numeric keyboard Return

22.6.3 Stage Views

Function	Key and/or Mouse
Start/Continue Animation Forward	Space
Start Animation Backwards	Shift+Enter
Set Time-line Marker/Value to Start	Ctrl+Home

Set Time-line Marker/Value to End (not when infinity is selected)	Ctrl+End
Set View to 0	Home
Set View to End director	End
Move the Time-line Marker one field	<Ctrl> and the left or right arrow key
Move the Time-line Marker more than one field at a time	Left or right arrow key. This, for example, in 1080i50 format, moves the time-line marker 5 fields, left or right, and in 1080i60 the movement is 6 fields, left or right
<Alt> and the left or right arrow key. This, for example, in 1080i50 format, moves the time-line marker 50 fields, left or right, and in 1080i60 the movement is 60 fields, left or right	
Delete Selected Key Frame(s)	Delete
Pan View	Middle mouse click and drag
Zoom Time Axis In or Out	Mouse wheel
Zoom Time Axis to Selected Channel	Shift+Z, Shift+Y
Zoom Time Axis to whole Animation	Shift+X
Zoom Spline/Dopesheet View In	Mouse wheel, or Z, orY, orCtrl + Plus sign on the numeric keypad
Zoom Spline/Dopesheet View Out	Mouse wheel, orX, orCtrl + Minus sign on the numeric keypad
Zoom Spline View to whole Spline	Shift+A
Zoom Time Axis and Spline View to view all Key Frames of selected Channel	Shift+Q
Zoom Time and Value Axis to View All Key Frames	Shift+W
Zoom Time Axis In/Out	Z+Middle mouse click and drag

Zoom Value Axis in Spline Editor In/Out	A+Middle mouse click and drag
Copy Selected Key Frame	C+Left mouse click and drag
Bring View to Time-line Marker	M
Reverse the Order of Selected Key Frames	Ctrl+R
Snap to Grid on/off	Ctrl+G
Set Time Axis to Seconds	Ctrl+8
Set Time Axis to Frames	Ctrl+9
Set Time Axis to Fields	Ctrl+0
Move Time-line Marker to nearest Key Frame	J
Scale a Portion of Selected Director	T+Left mouse click on Director Bar This shows a region where Key Frames can be scaled.
Rubber band Selection	S+Left mouse click and drag
Zoom in onto specific region	Z+Left mouse click and drag
Insert a new Stop Point	S+Left mouse click on Director Bar
View Spline 1 (Rotation Channel Only)	Ctrl+Z
View Spline 2 (Rotation Channel Only)	Ctrl+X
View Spline 3 (Rotation Channel Only)	Ctrl+C
Jump to next Key Frame	Tab (a Key frame must be highlighted first)
Jump to previous Key Frame	Shift+Tab
Escape from moving a Key Frame	Esc

See Also

- Application Controls and Shortcuts
- Integer and Float Controls

- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Import Shortcuts
- On Air Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.7 Import Shortcuts

This section contains the following topics:

- File Shortcuts
- Archive Shortcuts

22.7.1 File Shortcuts

Function	Key and/or Mouse
Select Fonts for import	Ctrl+1
Select Images for import	Ctrl+2
Select Geometries for import	Ctrl+3
Select Scenes for import	Ctrl+4
Select Audio for import	Ctrl+5
Select Video for import	Ctrl+6
Select Archives for import	Ctrl+7

22.7.2 Archive Shortcuts

Function	Key and/or Mouse
View Scenes for import	Ctrl+1
View Geometries for import	Ctrl+2

Function	Key and/or Mouse
View Materials for import	Ctrl+3
View for Materials Advanced for import	Ctrl+4
View Image for import	Ctrl+5
View Fonts for import	Ctrl+6
View Video Clips for import	Ctrl+7
View Substances for import	Ctrl+8
View All files	Ctrl+9
Entry Point View	Ctrl+E
Folder View	Ctrl+O

See Also

- To Import Selected Files
- Application Controls and Shortcuts
- Integer and Float Controls
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- On Air Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.8 On Air Shortcuts

Function	Key and/or Mouse
Return to Server Panel	Esc
Show or hide the VGA Preview window (not available on VGA version)	Hold Shift+Backspace

See Also

- On Air Mode

- Application Controls and Shortcuts
- Integer/Float Controls
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- Import Shortcuts
- Polygon plug-in Editor Shortcuts
- Script Editor Shortcuts

22.9 Polygon Plug-In Editor Shortcuts

Function	Key and/or Mouse
Open Shortcut list	<Ctrl+F1>
Sharpen Edge of Selected Point or Points	<Shift> + Left click
Delete Selected Point(s)	<Delete>
Add New Anchor Point	Left click
Add new Bezier point	Hold Left mouse button down and drag
Create New Spline	<Insert> + Left click
Close/Open Active Spline	C
Close Spline	Click on the first point to toggle open / closed
Enable/Disable Spline Info	<H>
Move Active Spline	<Alt>
Copy	<Ctrl+C>
Cut	<Ctrl+X>
Paste	<Ctrl+V>
Select Point or Spline	<S>

Function	Key and/or Mouse
Move Anchor point	<Ctrl> + Left mouse button
Move handles	<Ctrl> + Left mouse button
Move single handle	<Shift> + Left mouse button
Remove an Anchor point	Click on existing point with Left mouse button
Rectangular selection	Hold <S> + Left mouse button down, and drag
Convert straight point to Bezier	<Shift> + Left mouse button drag on point

See Also

- Application Controls and Shortcuts
- Integer/Float Controls
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- Import Shortcuts
- Script Editor Shortcuts

22.10 Script Editor Shortcuts

Function	Key and/or Mouse
Copy	Ctrl-C
Cut	Ctrl-X
Paste	Ctrl-V, Shift-Insert
Copy and Paste	Click the Middle mouse button to paste the selected text at the Mouse pointer location if enabled in the Viz Engine Configuration's User Interface section
Search for Search String	F3
Search for Selected Text	F4

Function	Key and/or Mouse
Comment	F5
Uncomment	F6
Increase Indent	Tab
Decrease Indent	Shift-Tab
Undo	Ctrl-z

See Also

- Application Controls and Shortcuts
- Integer and Float Controls
- Import Shortcuts
- Server Panel Shortcuts
- Scene Tree Shortcuts
- Scene Editor Shortcuts
- Stage Shortcuts
- The **Configuring Viz** section of the Viz Engine Administrator Guide