



grass valley

A **BELDEN** BRAND

XSTUDIO

BROADCAST GRAPHICS AUTHORIZING SYSTEM

User Manual

Vertigo Suite v5.2

M852-0500-521

2019-03-21

www.grassvalley.com

Copyright and Trademark Notice

Copyright © 2019, Grass Valley Canada. All rights reserved.

Belden, Belden Sending All The Right Signals, and the Belden logo are trademarks or registered trademarks of Belden Inc. or its affiliated companies in the United States and other jurisdictions. Grass Valley Canada, Miranda, Vertigo Suite, Xstudio are trademarks or registered trademarks of Grass Valley Canada. Belden Inc., Grass Valley Canada, and other parties may also have trademark rights in other terms used herein.

Terms and Conditions

Please read the following terms and conditions carefully. By using Vertigo Suite documentation, you agree to the following terms and conditions.

Grass Valley hereby grants permission and license to owners of Vertigo Suites to use their product manuals for their own internal business use. Manuals for Grass Valley products may not be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose unless specifically authorized in writing by Grass Valley.

A Grass Valley manual may have been revised to reflect changes made to the product during its manufacturing life. Thus, different versions of a manual may exist for any given product. Care should be taken to ensure that one obtains the proper manual version for a specific product serial number.

Information in this document is subject to change without notice and does not represent a commitment on the part of Grass Valley.

Warranty information is available in the Support section of the Grass Valley web site (www.grassvalley.com).

| | |
|------------------|---------------------|
| Title | Xstudio User Manual |
| Software Version | Vertigo Suite v5.2 |
| Part Number | M852-0500-521 |
| Revision | 2019-03-21, 13:38 |

toc

Table of Contents

| | | |
|----------|--|-----------|
| 1 | Introducing Xstudio | 1 |
| | Xstudio's role in the Vertigo Suite workflow | 2 |
| | Types of Xstudio projects: Templates, Panels & Pages | 4 |
| | Procedural overview for creating Template+Scene and Panel+Scene projects | 5 |
| | Procedural overview for creating Pages | 7 |
| 2 | Getting started with Xstudio | 9 |
| | Installation and licensing information | 10 |
| | Starting Xstudio and connecting to the Xmedia Server | 10 |
| | Configuring Xstudio's settings | 11 |
| | Servers settings | 12 |
| | Logging settings | 13 |
| | Workspace settings | 14 |
| | Asset Defaults settings | 15 |
| | Device settings | 16 |
| | Browsers settings | 17 |
| | Connecting Xstudio to an output device | 18 |
| | Verifying Xstudio's installation and setup | 21 |
| 3 | Orientation to Xstudio's desktop | 23 |
| | Starting Xstudio | 24 |
| | Design Mode vs. Production Mode | 26 |
| | Xstudio's menu commands | 27 |
| | Xstudio's desktop icons and toolbars | 33 |
| | Xstudio's primitives | 36 |
| | Control side primitives | 36 |
| | Scene side primitives | 37 |
| | Asset Browser and Gallery Browser | 38 |
| | Properties Browser | 39 |
| | Action Editor and Animation Editor | 41 |
| | Project Windows - Control and scene sides | 42 |
| | Safe areas and resolution format | 43 |
| | Zooming in/out of the canvas | 44 |
| | Panning across the canvas | 45 |
| | Production mode environment | 46 |
| 4 | Creating and opening Xstudio projects and assets | 47 |
| | Creating new projects and assets | 48 |
| | Setting the canvas' properties | 48 |

| | |
|---|------------|
| Opening existing projects | 53 |
| Searching for projects/assets on the Xmedia Server..... | 54 |
| Search query syntax supported by the Search XMS | 55 |
| Importing assets from another Vertigo system..... | 57 |
| Ingesting files created using third-party applications | 59 |
| Ingesting image or audio files..... | 59 |
| Ingesting multiple image files or creating cel animations or clips upon ingest .. | 61 |
| Ingesting video files | 66 |
| | |
| 5 Adding and preparing objects on the Xstudio canvas | 69 |
| Adding assets to the canvas | 70 |
| Adding primitives to the canvas..... | 71 |
| Removing assets or primitives from the canvas | 73 |
| Previewing a scene's layout on a device during development..... | 74 |
| Selecting, grouping, and/or layering objects on the canvas | 75 |
| Selecting a single object on the canvas..... | 75 |
| Selecting multiple objects on the canvas | 76 |
| Grouping and ungrouping objects on the canvas..... | 77 |
| Layering objects on the canvas | 78 |
| Resizing objects using object zones | 79 |
| Moving and aligning objects on the canvas | 80 |
| Cutting, copying and pasting objects to/from the canvas | 83 |
| Modifying the font properties of text objects | 84 |
| Editing an image | 85 |
| Undoing and redoing edits on the canvas | 86 |
| | |
| 6 Working with scene objects | 87 |
| Text2D objects on the scene canvas..... | 88 |
| Creating simple clocks and timers using Text2D scene elements | 92 |
| Using embedded text tags within Text2D scene objects | 94 |
| Clip objects on the scene canvas | 96 |
| Image objects on the scene canvas..... | 98 |
| Cel Animation objects on the scene canvas..... | 100 |
| Video In objects on the scene canvas..... | 102 |
| Crawl and Roll objects on the scene canvas..... | 104 |
| Creating a basic crawl | 105 |
| Making basic changes to a crawl or roll..... | 106 |
| Creating a controllable crawl with Xstudio | 107 |
| Using scene objects as a mask | 109 |
| | |
| 7 Animating scene objects | 111 |
| Animation Editor's interface components | 112 |
| Animation Editor toolbar | 112 |
| Keyframe Editor | 113 |
| Preview Mode controls | 117 |
| Animation Tree and Animations | 117 |

| | |
|---|------------|
| Animation Timeline and Keyframes | 124 |
| Creating scene animations..... | 129 |
| Displaying the animated properties in the Scene tree..... | 130 |
| Opening the Animation Editor..... | 130 |
| Choosing a type of animation..... | 131 |
| Adding the object's animated properties to your animation | 131 |
| Adding and setting keyframes | 132 |
| Adding and setting a keyer channel | 134 |
| Adding and setting a command channel | 134 |
| Saving and publishing the template/scene..... | 136 |
| Previewing and scrubbing the animation | 136 |
| Setting up an animation action | 137 |
| Editing existing animations..... | 139 |
| Renaming animations | 139 |
| Deleting channels from the Animation Tree | 139 |
| Cutting, copying, and pasting animation elements | 140 |
| Resizing and moving channels on the Timeline..... | 142 |
| Moving a keyframe | 142 |
| Deleting a keyframe | 142 |
| Animated property descriptions | 143 |
| General animated object properties..... | 143 |
| Cel animation properties | 157 |
| Clip properties | 158 |
| Image Properties | 159 |
| Text2D properties | 160 |
| 8 Working with control objects..... | 161 |
| Connecting control objects to scene objects | 162 |
| Text objects on the control canvas | 163 |
| Dropbox objects on the control canvas..... | 165 |
| Button objects on the control canvas..... | 167 |
| Textbox objects on the control canvas | 169 |
| Image objects on the control canvas | 171 |
| Checkbox & Radio Button objects on the control canvas | 173 |
| ActiveX objects on the control canvas..... | 175 |
| Timer objects on the control canvas..... | 176 |
| 9 Linking objects to a data source | 177 |
| Data Manager screen items..... | 178 |
| Schemas | 179 |
| Creating schemas | 179 |
| Editing schemas..... | 181 |
| Renaming or saving a copy of a schema with a new name | 181 |
| Deleting schemas | 181 |
| Data entities: snapshots, tickers, multi-tickers and classes..... | 183 |
| Creating a snapshot | 183 |
| Creating a ticker..... | 184 |
| Creating a multi-ticker..... | 187 |

| | |
|--|------------|
| Creating a class..... | 187 |
| Setting dynamic keys..... | 188 |
| Editing and deleting data entities..... | 189 |
| Creating data links..... | 190 |
| Prefixes (Scene:, Control:, Data:, L:, G:) | 190 |
| Creating and applying lookup tables..... | 191 |
| Editing lookup tables..... | 194 |
| Using social media messages and online polls as a data source..... | 195 |
| Acquiring and installing the Social Media Interface license..... | 195 |
| Creating a social media data source..... | 195 |
| Creating a snapshot or ticker for social media content..... | 196 |
| Data tables for social media snapshots and tickers..... | 200 |
| Creating a template that retrieves social media content..... | 206 |
| | |
| 10 Adding logic to control objects..... | 211 |
| Opening the Action Editor..... | 212 |
| Action Editor interface components..... | 213 |
| Editing and ordering buttons..... | 213 |
| Add Actions and Events..... | 213 |
| Action Table components..... | 214 |
| Using the object's default events..... | 216 |
| Creating custom events..... | 218 |
| User Defined events..... | 218 |
| Data Arrival events..... | 219 |
| Assigning an action to an object..... | 221 |
| Creating custom actions..... | 222 |
| Setting the Target parameter..... | 222 |
| Using the Expression Builder..... | 224 |
| Deleting actions from events..... | 226 |
| Default actions and action parameters..... | 227 |
| | |
| 11 Saving, publishing and exporting Xstudio projects..... | 251 |
| Saving Xstudio projects and assets..... | 252 |
| Publishing assets to output devices..... | 253 |
| Manually publishing assets to devices in Xstudio..... | 253 |
| Automatically publishing assets to devices in Xstudio..... | 254 |
| Exporting assets to be used on another Vertigo system..... | 255 |
| | |
| 12 Playing out a scene using Xstudio's Production mode.... | 257 |
| Playing out a scene associated with a template or page..... | 258 |
| Playing out a scene controlled by a control panel..... | 259 |
| Playing out a standalone scene..... | 260 |
| | |
| Appendix A Asset Browser..... | 261 |
| Opening, closing, and resizing the Asset Browser..... | 262 |

| | |
|--|-----|
| Structure of the Asset Browser | 263 |
| Creating Asset Browser categories | 266 |
| Deleting Asset Browser categories | 267 |
| Renaming an Asset Browser category | 268 |
| Publishing a category's assets to one or more output devices | 268 |
| Moving or copying assets to different categories | 269 |
| Removing assets from categories | 270 |
| Opening and editing assets using the Asset Browser..... | 271 |
| Deleting assets using the Asset Browser | 272 |
| Viewing and editing asset properties using the Asset Browser..... | 273 |

Appendix B Gallery Browser275

| | |
|--|-----|
| Opening, closing, and resizing the Gallery Browser | 276 |
| Switching between Proxy and Detail view in the Gallery Browser..... | 277 |
| Proxy View | 277 |
| Detail View..... | 278 |
| Switching between Proxy and Detail views..... | 278 |
| Viewing assets in the Gallery Browser | 279 |
| Adding and deleting asset categories to the Gallery Browser..... | 280 |
| Refreshing the Gallery Browser..... | 281 |
| Opening and editing assets using the Gallery Browser | 281 |
| Deleting assets using the Gallery Browser | 282 |
| Viewing and editing asset properties using the Gallery Browser | 282 |

Appendix C Xstudio properties285

| | |
|---|-----|
| Control object properties | 286 |
| Scene object properties | 293 |
| Group object properties | 311 |
| Formats for text, numbers, dates and times..... | 314 |
| Formatting numbers | 315 |
| Formatting dates and times..... | 316 |

Appendix D Image Editor317

| | |
|---|-----|
| Opening the Image Editor | 318 |
| Zooming in/out and repositioning the image..... | 319 |
| Zooming in/out | 319 |
| Reposition the image by panning | 320 |
| Cropping an image..... | 321 |
| Making color adjustments | 323 |
| Adjusting the sharpness of an image..... | 324 |
| Using the Image Modification buttons to flip an image | 324 |

Appendix E Script Editor325

| | |
|----------------------------------|-----|
| Accessing the Script Editor..... | 326 |
| Script Editor screen items | 326 |

Creating scripts..... 328
 Editing commands 328
 Performing a search within the script 328
 Adding comments to scripts 329
Opening existing scripts 329
Saving scripts 329

Appendix F External Controls331

About external controls..... 332
 Serial tab..... 332
 GPI tab..... 332
 Settings tab..... 333

Contact Us335

1 Introducing Xstudio

Xstudio is the authoring component of the Vertigo Suite. You can use Xstudio to:

- Design templates which combine graphical layout, links to data, and production logic. These reusable templates can be updated with new information and included in playlists for sequential playout.
- Build control panels for application areas such as info channels, live tickers, channel branding, community news channels, and more.
- Preview the playout of the scene on a software CG or hardware playout device using Xstudio's Production mode.

Xstudio facilitates the authoring process by allowing you to drag and drop objects for instant backgrounds, tickers, lower thirds, station bugs and more. These can then easily be connected to external data sources such as databases, data feeds, Excel files and text files. What's more, what you design can be used over and over again for different broadcasts.

Used in conjunction with other Vertigo Suite applications (i.e. Xbuilder, Xnews and Xplay), Xstudio provides a complete authoring system that's both flexible and easy to use.

The main purpose of this manual is to provide practical reference and procedural information for how to create and control attractive and effective broadcast graphics using the Xstudio authoring environment.

The following sections provide a high-level description of Xstudio's role in the Vertigo Suite workflow, as well as a procedural overview for creating the most common types of Xstudio projects.

- [Xstudio's role in the Vertigo Suite workflow](#), on page 2
- [Procedural overview for creating Template+Scene and Panel+Scene projects](#), on page 5
- [Procedural overview for creating Pages](#), on page 7

Xstudio's role in the Vertigo Suite workflow

Xstudio is the authoring component of the Vertigo Suite of products. Figure 1-1 shows in summary form how Xstudio works with Xnews and Xplay.

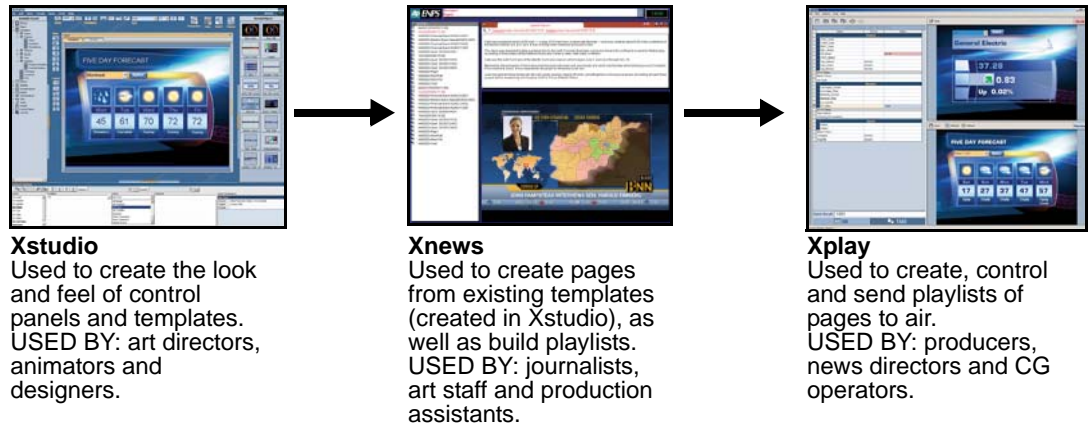


Fig. 1-1: Xstudio elements are passed on to Xnews and Xplay

To put it into a real world context, consider the following scenario that demonstrates how all of the Xmedia Suite components work together, including servers and drivers:

A schedule board needs to display upcoming shows at junction points throughout the week.

- 1 Using **Xstudio**, the art team could create a reusable **template** to act as a master copy.



Fig. 1-2: Schedule board Template built in Xstudio

- 2 Once basic layouts exist in the template, any number of copies can be made from it to enter specific show name, image, day, time, etc. These copies are called pages, and they allow users to make limited changes to template elements, e.g. text and image modifications. Pages are usually created in Xnews or Xbuilder, but they can also be created in Xstudio.



Fig. 1-3: Schedule board Pages created from the template

- 3 Once pages have been created from templates, they can be ordered into playlists and taken to air with Xplay.

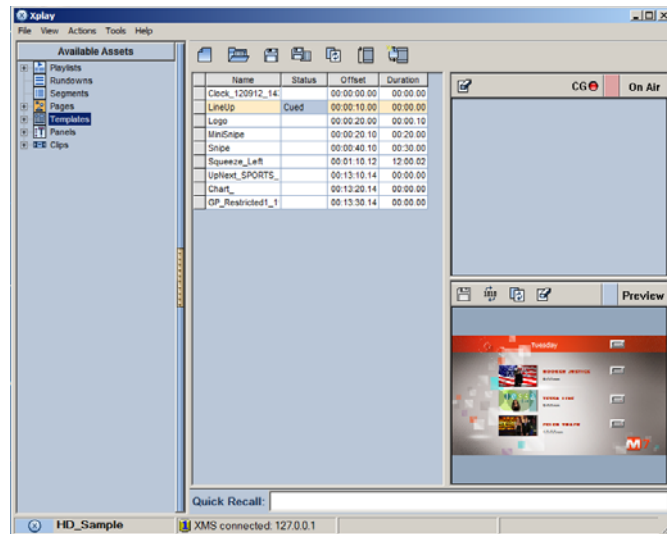


Fig. 1-4: Pages added to a playlist and brought to air using Xplay

In addition to templates, which are used to create pages that are played out sequentially in a playlist, Xstudio can also be used to create a panel. A panel is a user interface that sends commands to the output device, such as turning a bug on or off.

Typically, a panel is loaded at startup into Xplay and used for the entire duration of a show.

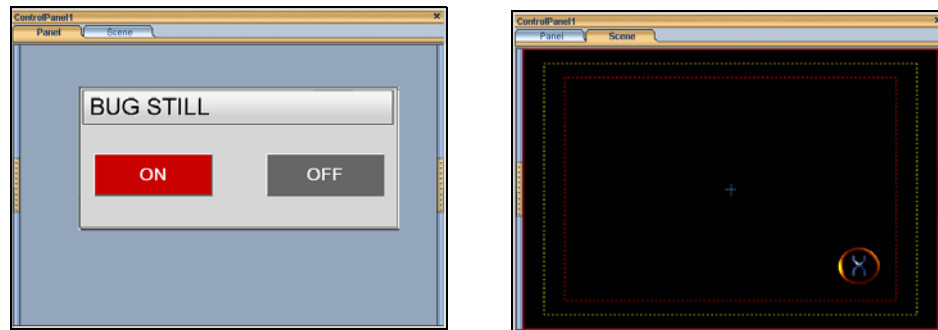


Fig. 1-5: Control Panel + Scene used to display/hide the station's logo

The XmediaServer connects all applications in the Xmedia suite, making it possible to share assets, such as panels, templates, pages, images and clips. For example, once an NHL template has been created with Xstudio, a user at a different workstation can use it to create a page in Xnews. Another user could then include this page in a playlist built in Xplay.

Meanwhile, the Data Server manages data coming from various feeds (e.g. SportsTicker, Reuters), and sends it out to the appropriate recipients. For example a playlist containing pages containing stock values needs the Data Server to provide live data.

Types of Xstudio projects: Templates, Panels & Pages

Xstudio is the Vertigo Suite's application for creating projects that control and display the broadcast graphics that will be displayed on air.

In general, a project contains a scene and an associated control side. A scene is a collection of graphical objects such as text, images, video clips, and cel animations that are displayed on air.

The control side is a user interface associated with the scene. It may be a template, in which case it is used to enter specific data when creating pages; or it may be a panel, in which case it is used by an operator to control the scene and its objects during playout. Both templates and panels can contain logic for animating specific elements of the scene and updating data that is displayed in the scene. Often the control side is made to resemble its associated scene to give the operator an indication of what is on air.

While templates and panels are both used to control the scene, templates are typically used as the basis for pages which are taken to air in sequential order without the operator interacting with the control side. On the other hand, panels are intended to provide a user interface that allows an operator to manually control the graphics during playout.

As mentioned above, another type of Xstudio project is a page, which is created from a template by entering specific information. Rather than recreating the control and layout of the project, multiple pages can borrow the layout from an existing template and scene. All that is left to do is enter the new text, images, clips, and/or animations specific to the broadcast. Although Xstudio can be used for creating pages, we highly recommend using the Vertigo Suite's Xbuilder application for creating and editing pages.

The following sections provide more information about each type of Xstudio project, as well as an overview of the procedure for creating and playing out these projects:

- [Procedural overview for creating Template+Scene and Panel+Scene projects](#), on page 5
- [Procedural overview for creating Pages](#), on page 7

Procedural overview for creating Template+Scene and Panel+Scene projects

Xstudio allows you to create and control television broadcast graphics using a Template/Panel + Scene project. The Template + Scene project consists of two parts: the template and the scene.



The Template (control-side) is used to set editable content and contains logic for controlling scene objects during playback



The scene contains a collection of graphic objects (static, animated or data driven) that are displayed on air during playback and controlled by the template's objects and logic.

Fig. 1-6: The two sides of a Template + Scene project

The following table provides a high-level procedure for creating television broadcast graphics using Xstudio's Template + Scene project. The procedure for creating a panel is identical, except the panel's control-side often contains user interface objects that can be used during playback to control the scene.

| Step | Task Description |
|------|---|
| 1 | <p>Open a new Template + Scene or Panel +Scene project</p> <p>Launch Xstudio and open a new project.</p> <ul style="list-style-type: none"> • Creating new projects and assets, on page 48 |
| 2 | <p>Set the template/panel and scene's properties</p> <p>Select the scene-side and set the scene's properties, including the scene's format (NTSC, HDTV-1080/30...etc.). Then select the control-side and set its properties.</p> <ul style="list-style-type: none"> • Setting the canvas' properties, on page 48 |
| 3 | <p>Add and position objects (assets and/or primitives) on both the scene side and the control side.</p> <p>Create your graphic's desired "playout" look by adding and arranging assets and/or primitives objects to the scene's canvas.</p> <p>Add primitives to the control side to provide operators with a means to control the scene.</p> <ul style="list-style-type: none"> • Adding assets to the canvas, on page 70 • Adding primitives to the canvas, on page 71 • Selecting, grouping, and/or layering objects on the canvas, on page 75 • Resizing objects using object zones, on page 79 • Moving and aligning objects on the canvas, on page 80 • Cutting, copying and pasting objects to/from the canvas, on page 83 • Modifying the font properties of text objects, on page 84 • Editing an image, on page 85 |

| Step | Task Description |
|------|--|
| 4 | <p>Set the objects' properties and/or animate the scene objects</p> <p>Using the Xstudio's Properties Browser, set the properties which determine the individual object's static look and behavior.</p> <p>Using Xstudio's Animation Editor, create scene animations by changing the properties of the objects on the scene over a period of time to create visual effects, like fading an object's visibility in/out or moving the object around the screen.</p> <ul style="list-style-type: none"> • Working with scene objects, on page 87 • Animating scene objects, on page 111 • Working with control objects, on page 161 |
| 5 | <p>Optional: Link the control objects to a data source</p> <p>You can create data-driven graphics by linking control objects to data sources (databases or data feeds) that provide live data to the scene upon playout. Xstudio also allows you to create and use lookup tables to define replacement values for incoming data.</p> <ul style="list-style-type: none"> • Linking objects to a data source, on page 177 |
| 6 | <p>Connect the control objects to the scene objects and add control logic</p> <p>Connect control objects to the scene objects that will be controlled and/or edited by objects on the control-side.</p> <p>Use the Action Editor the logic that will later be applied to the control objects will be linked to the appropriate scene objects.</p> <ul style="list-style-type: none"> • Working with control objects, on page 161 • Adding logic to control objects, on page 211 |
| 7 | <p>Save the project to the Xmedia Server</p> <p>When you save the project to the Xmedia Server, both the template and scene will be stored as one "Template" project within the Templates category. Similarly, both the panel and scene will be stored as one "Panels" project within the Panels category</p> <ul style="list-style-type: none"> • Saving Xstudio projects and assets, on page 252 |
| 8 | <p>Publish the project to the connected output device</p> <p>Before being able to playout the scene on an output device, you must publish the project (Template + Scene or panel + Scene) to the output device. Publishing gathers the project's assets from the Xmedia Server and copies them into your output device.</p> <ul style="list-style-type: none"> • Publishing assets to output devices, on page 253 |
| 9 | <p>Preview the playout in Xstudio's Production mode</p> <p>Xstudio allows you to preview and/or playout the scene to the connected playout device.</p> <ul style="list-style-type: none"> • Playing out a scene using Xstudio's Production mode, on page 257 |

Procedural overview for creating Pages

While Xstudio is the Vertigo Suite's application for authoring template-based graphics, it can also be used to create Pages, which are built from templates. A template is a master copy that determines the layout and contents of graphics that will be taken to air in sequential order (see [page 5](#)). A page is an instance of a template which means that the layout is already done and all you have to do is enter text, images, clips, and/or animations specific to the broadcast. Pages are then loaded into playlists and taken to air in a pre-determined order using Xplay.

Note: While you can certainly create pages in Xstudio, Xbuilder is the recommended application within the Vertigo Suite that allows you to populate the templates and convert them into graphic pages. Xbuilder can also be used to create playlists, rundowns, and automated interstitial promotions, which can then be played out on-air using Xplay.

To create a page from an existing template:

- 1 Select the **File>New** menu command to open the **Create New** dialog box.
- 2 In the **Create New** dialog box, select the **Assets>Page** and then click **OK**.
- 3 In the **Select a template for the page** dialog box, navigate through the templates stored in the Templates category on the Xmedia Server.
 - Double-click the template asset listed.

Or,

- Select the template and click **OK**.

The template opens as a page in a project-type window. Note that you cannot change any of the formatting or add/remove objects. You can only edit the fields (i.e. textboxes, images, drop-down list selections...etc.) that were configured on the template.

- 4 Set or edit the variable fields in the template as required. Depending upon the type of object/field, this may involve typing text in a textbox, making a selection from a drop-down list, right-clicking on an image, cel animation, or clip to choose another one.
- 5 Once you are satisfied with the content of the page, select the **File>Save** command to save the page to the Xmedia Server.

The **Save Page** dialog box appears and allows you to specify the page's name, specify the author's name, enter a short description and determine where within the Pages root category the page will be stored.

- 6 Click **OK** to save the page and exit the dialog box.
- 7 In order to preview and playout the page in Xstudio's Production mode, you must first publish the page to the connected playout device. See [Publishing assets to output devices](#), on page 253.
- 8 Once the page has been published, switch to Production mode and take the page to air. See [Playing out a scene using Xstudio's Production mode](#), on page 257.

Getting started with Xstudio



This chapter provides you with information about installing and licensing the Xstudio software on a client workstation. Additional information and instructions are provided for performing a basic setup and configuration of the Xstudio software.

Once you have completed the installation and quick setup, Xstudio will be properly configured to create and save projects (i.e. templates, control panels, and scenes) to the Xmedia Server, as well as preview a scene's ployout on an output device using Xstudio's Production mode.

This chapter is organized into the following topics:

- [Installation and licensing information](#), on page 10
- [Starting Xstudio and connecting to the Xmedia Server](#), on page 10
- [Configuring Xstudio's settings](#), on page 11
- [Connecting Xstudio to an output device](#), on page 18
- [Verifying Xstudio's installation and setup](#), on page 21

Installation and licensing information

Xstudio is packaged with the Vertigo Suite software installation wizard. It can be installed as part of the full installation or as an individual (standalone) component. In both cases, you simply need to run the installation wizard on a workstation that is on the same network as the Xmedia Server.

Information regarding the Vertigo Suite's hardware and network requirements, as well as instructions for installing and/or upgrading Xstudio, are available in the Vertigo Suite's Release Notes, which can be downloaded from the Documentation Library on the Grass Valley web site (www.grassvalley.com).

To operate in its full capacities, Xstudio requires that a software application license be installed and validated on the Xmedia Server. Therefore, once Xstudio is installed, you must contact our Technical Support team (support@grassvalley.com) to request a license key for Xstudio. Then you must apply and validate the licenses using the XmediaServer Control Panel. Instructions for acquiring and validating an Xstudio application license are also provided in the Vertigo Suite's Release Notes.

Starting Xstudio and connecting to the Xmedia Server

When you initially launch Xstudio, a dialog box appears and asks you to configure the server location. As a client application, Xstudio must be connected to the Xmedia Server, which is the centralized asset management server for Vertigo applications and devices.

Since you have only just installed Xstudio, the first task that you must do is to connect Xstudio to the Xmedia Server.

To start Xstudio and connect to the Xmedia Server:

- 1 Click **Start>Programs>VertigoXmedia>Xstudio**.
- 2 When the message asking you to configure the server location, click the **Yes** button. The **Settings** dialog box appears.
- 3 Click **Servers** in the tree view at left. This displays the settings related to the Xmedia Server (see [page 12](#) for more information about Xstudio's Server settings).
- 4 In the **XmediaServer IP Address** field, replace **<server IP>** with the IP address of the machine hosting the Xmedia Server.
- 5 Unless the port number was changed when the Xmedia Server was installed, the default in the **XmediaServer Port** field will be correct. If it was changed, enter the new number.
- 6 Click **OK**.

Xstudio launches. If the program does not launch, consult your system administrator. The problem could be due to one of the following:

- The IP address or port was entered incorrectly.
- The machine hosting the Xmedia Server is down.
- The XmediaServer is not running on the host machine.
- The machine is not connected to the network.

Configuring Xstudio's settings

Although Xstudio's default settings are appropriate for most operating circumstances, you may want to customize the look and behavior of the application using Xstudio's settings. Note that modifying these settings is entirely optional.

To access Xstudio's settings:

- 1 From the **Tools** menu, select **Settings**.

This displays the **Settings** dialog box, which contains a tree-structure of thematic settings. Each of the settings is defined in the following sections:

- [Servers settings](#), on page 12
- [Logging settings](#), on page 13
- [Workspace settings](#), on page 14
- [Asset Defaults settings](#), on page 15
- [Device settings](#), on page 16
- [Browsers settings](#), on page 17

- 2 After making changes anywhere in the dialog box, click the **OK** button to register the information and close the dialog box.

Or,

Click the **Apply** button to register the information and keep the dialog box open to work on other settings.

Or,

Click the **Cancel** button to ensure that no changes made since the last time you clicked the **Apply** button take effect.

Servers settings

Servers settings specify the locations of the Xmedia Server and the Data Server. The Xmedia Server contains all assets (all building blocks used in the creation of broadcast graphics). This is where templates, control panels, images...etc., are stored. The Data Server is an application that manages all data coming in from sources specified in control panels. The Data Server distributes the data and provides live updates of data values when requested.

- **XmediaServer IP Address** - The IP address of the machine running the Xmedia Server.
- **XmediaServer Port** - The port on which the Xmedia Server is listening.
- **Backup XMedia Server IP Address** - Backup server's IP address used in case of failure of the primary Xmedia Server.
- **Backup XMedia Server Port** - Backup server's port used in case of failure of the primary Xmedia Server.
- **Data Server IP Address** - The IP address of the Data Server.
- **Data Server Port** - The port on which the Data Server is listening.
- **Backup Data Server IP Address** - Backup data server's IP address used in case of failure of the primary data server.
- **Backup Data Server Port** - Backup data server's port used in case of failure of the primary data server.
- **Data Server Event Port Min Value** - The lowest UDP port value in a range of ports used for communication between the Data Server and Xstudio.
- **Data Server Event Port Max Value** - The highest UDP port value in a range of ports used for communication between the Data Server and Xstudio.
- **Data Server Path** - If the Data Server IP Address is local (e.g. 127.0.0.1), then enter the file name and path of the Data Server's location on the PC. This will prompt Xstudio to start the Data Server when it launches.

Logging settings

Xstudio offers several logging options that you can use for observing Xstudio's behavior and/or troubleshooting unexpected problems. The Xstudio log files are saved in the following directory: C:\Documents and Settings\All Users\Application Data\VertigoXmedia\Logs

- **Show Debug Console** - When enabled, this setting opens the **Debug Console** window, which is used for debugging plug-ins, scripts and the like.
- **Save Local XML File** - When enabled, an XML file containing the error log is written upon saving.
- **Log Level** - Xstudio can be set to provide the following levels of logging events:
 - **Error** – A major failure that impacts the end-user.
 - **Warn** – An important but non-fatal error.
 - **Info** – A useful notification message to help users troubleshoot/track issues.
- **Max File Size (MB)** - The maximum size of a log file before rotating.
- **Max File Count** - How many log files to keep.
- **Debug Categories** – The list of categories that will emit DEBUG level logs.

Workspace settings

The Workspace settings allow you to configure various appearance and behavior settings that affect Xstudio's workspaces in design and production mode.

- **Production Mode Zoom** - Xstudio lets you determine how your control screens will appear when you switch to Production mode to take to air. The default is **Fit**, which scales up the control screen to fit the window. The other option is **100%**. This keeps objects at their actual size.
You can also zoom in and out on objects when in Production mode to make them appear larger or smaller. Either move the mouse wheel up and down or (if you don't have a mouse wheel) press **ALT** and the up or down key on your keyboard.
- **Show New Dialog on Startup** - Selected by default, this setting makes the **Create New** dialog box display when you launch Xstudio in Design mode (it will not appear if launching in Production mode). Deselecting it makes Xstudio open with a blank screen in default mode (specified by the Initial mode setting, explained below), although you can always access the **Create New** dialog box by selecting the **New>File** menu command.
- **Fit Gallery Objects to Extents** - Applicable to Xmedia objects. When this setting is selected, anytime an Xmedia object is opened for editing, the window automatically zooms into the space occupied by the Xmedia object rather than displaying the whole scene.
- **Auto Load Driver** - When enabled, this setting makes the output driver automatically launch when you launch Xstudio.
- **Driver** - If the Auto Load Driver check box has been selected, you must provide the location of the driver on your PC (e.g. `C:\Program Files\VertigoXmedia\VxCG2000`). The driver will be in the location specified during the driver installation.
- **Always Overwrite On Publish** - When enabled, this option forces an update by rewriting the scene on publish regardless of whether or not it has been modified.
- **Show Publish Dialog** - Sets the interface to automatically open and display the **Publish Progress** dialog box when an asset is published. The Publish Progress dialog box keeps track of which assets have been recently published and the status of the publication process for each.
- **Initial mode** - This setting determines whether Xstudio opens in **Design** or **Production** mode when launched.
- **Panel to load on startup** - This settings loads a specific control panel whenever you launch Xstudio. Click the browse button to display a dialog box that lets you select the control panel.
- **Spell Check On** - Identifies possible spelling errors in text box objects on the control side by underlining the misspelled word.
- **Spell Check Color** - The color of the line under a misspelled word when the **Spell Check On** setting is enabled.
- **Confirm Clear Device** - When enabled, this setting displays the **Clear Device** dialog box in production mode and prompts you to clear the device (equivalent to **Tasks>Clear Device**). When this setting is disabled, you get no prompt and you risk mistakenly clearing all your keyers on the device by hitting the **F4** shortcut key.

Asset Defaults settings

The Asset Defaults settings allow you to set the formatting and behavior defaults for the certain assets types.

- **Default Scene Format** - Sets the scene format (e.g. NTSC, PAL, HDTV_1080_30) that Xstudio use as a default when opening a new scene project.
- **Default Font Name** - Sets the font value for the **Font** toolbar, and is automatically applied whenever you work with text. You are not bound to your selection, however. You can select new font information at any time using the **Font** toolbar. This applies to all other default font settings (size, bold, italic).
- **Default Font Size** - Sets the font pixel size value for the **Font** toolbar.
- **Default Font Bold** - Sets the weight value for the **Font** toolbar.
- **Default Font Italic** - Sets the angle value for the **Font** toolbar.
- **Default Text Direction** - Sets the default direction for the text when new instances of text boxes and text objects are created.
- **Default Text Justification** - Sets the default justification for the text when new instances of text boxes and text objects are created.
- **Default Device Type** - The type of device that the system typically uses and is set up for.
- **Auto Load Scene Default** - Indicates whether or not the scene will automatically be loaded when the template or panel is cued.
- **Ingest Loop Default** - Sets the default looping value for .j2k and .oxi files that are ingested into the system using Xstudio.
- **Clip Object Loop Default** - Sets the default looping value for clip objects that are added to the scene.
- **Cel Object Loop Default** - Sets the default looping value for cel animation objects that are added to the scene.
- **Default Safe Title Area (90)%** - Sets the default value (percentage) for the Safe Title area on the scene side. The default value is 90%.
- **Default Safe Action Area (93)%** - Sets the default value (percentage) for the Safe Action area on the scene side. The default value is 93%.

Device settings

Device settings specify the keyers that the panels and templates will be loaded to on the CG.

- **Panel Keyer** - Specifies which CG keyer the panel is loaded when a Cue, Take, or Clear is executed on the CG.
- **Program Keyer** - Specifies which CG keyer the template is loaded when a Cue, Take or Clear is executed on the CG.

Browsers settings

These settings control how items display in the Gallery Browser and Asset Browser. They also let you determine whether the Gallery Browser and Asset Browser will be automatically hidden after an item has been opened from it (i.e. double-clicked), and whether the Action Editor will display when you launch the application.

- **Tile Width** - This determines the width in pixels of proxy tiles (the title below each thumbnail image) in the Gallery Browser.
- **Tile Height** - This determines the height in pixels of proxy's tiles in the Gallery Browser.
- **Image Width** - This determines the width in pixels of a proxy in the Gallery Browser.
- **Image Height** - This determines the height in pixels of a proxy in the Gallery Browser.
- **Show Assets in Asset Browser** - This determines whether or not assets are listed in the Asset Browser (at left). If your asset lists get too long you may want to deactivate the setting in order to prevent them from taking over the browser completely. Then if you want to see what a category contains, right-click on it and from the pop-up menu select **Open** to make its full contents display in the Gallery Browser (at right).
- **Auto Hide Asset Browser** - This determines whether or not the Asset Browser will be hidden after you open (i.e. double-click) one of its listed assets.
- **Auto Hide Gallery** - This determines whether or not the Gallery Browser will be hidden after you open (i.e. double-click) one of its listed assets.
- **Auto Hide Action Editor** - This determines whether or not the Action Editor will be hidden when you launch Xstudio.
- **VEX Export Path** - Specifies the directory path where the Vertigo Export file (.vex) file will be saved to when the Export as vex command is selected in the Asset Browser.

Connecting Xstudio to an output device

Xstudio must be connected to the output device (i.e. Software CG, Vertigo XG, Intuition XG) that it will be used to playout the graphics scenes that are produced in Xstudio. This connection must be active for publishing Xstudio projects/assets, as well as for previewing activities.

While you can set up as many output devices as required, Xstudio can only control one at a time. If you are interested in connecting to more than one output device at a time, please contact our Technical Support department.

Note: As well as being connected to Xstudio, the output device should also be connected to the Xmedia Server and a broadcast monitor.

To add and connect Xstudio to an output device:

- 1 Open the Device Configuration window by selecting **Tools>Devices>Configure Devices**.
- 2 In the **Device Configuration** window, select the **Add** button:
- 3 The **Device Configuration** windows expands to reveal two tabs (**General** and **Advanced**) with settings for identifying the device's configuration.
- 4 Specify the appropriate General Device Configuration settings:

| General | |
|---------------------|---|
| Device name | The name of the selected output device. This can only be modified when adding a device. |
| Device alias | An alternative name for the selected device on your network. |
| Description | An optional field that lets you label the device to distinguish it from other similar devices at your facility (for example). |
| Device Type | Identifies the type of device that these settings apply to. Choose from: VertigoXG, IntuitionXG, Preview (or other device), Remote Render, ImageStore, Intuition, HMP, FTP, LGK, DSK or Other . |
| Control IP | The network location (IP address) or hostname of the selected output device. The host must be available on the network. Warning: It is possible to use the localhost, but discouraged, since the XmediaServer will have no way to locate the device for publishing. |
| Control Port | The networking port that serves as a channel for sending commands to and from the output device. |


| General | |
|---------------------|---|
| Publish IP | The network location (IP address) or hostname of the selected output device. The host must be available on the network. Warning: It is possible to use the localhost, but discouraged, since the XmediaServer will have no way to locate the device for publishing. |
| Publish Port | The networking port that serves as a channel for publishing assets. Typically, this value is set to 15000 . Note: This value must correspond with the port used by the XpublishAgent on the output device. |

5 Specify the appropriate Advanced Device Configuration settings:

| Advanced | |
|----------------------|---|
| Clip Template | The name of the template that is used to play out clips. The clip template's control-side must have a texture linked to a scene-side clip object. Use the Browse button to navigate to the Clip template. Note that among the Vertigo Suite's system templates are valid clip templates for various scene formats. The system templates are export files that are usually stored in the following directory: <code>Program Files\VertigoXmedia\Apps\Support</code> . Once imported into the Xmedia Server, the templates are stored in the Templates>System Templates category. The Clip Template field only appears when one of the following device types is selected: Vertigo XG, Intuition XG, XPublishAgent, or Other. |
| Cel Template | The name of the template that is used to play out cel animations. The cel template's control-side must have a texture linked to a scene-side clip object. Use the Browse button to navigate to the Cel template. Note that among the Vertigo Suite's system templates are valid Cel templates for various scene formats. The system templates are export files that are usually stored in the following directory: <code>Program Files\VertigoXmedia\Apps\Support</code> . Once imported into the Xmedia Server, the templates are stored in the Templates>System Templates category. The Cel Template field only appears when one of the following device types is selected: Vertigo XG, Intuition XG, XPublishAgent, or Other. |

| Advanced | |
|-----------------------|--|
| Image Template | <p>The name of the template that is used to play out images. The image template's control-side must have a texture linked to a scene-side clip object. Use the Browse button to navigate to the Image template.</p> <p>Note that among the Vertigo Suite's system templates are valid image templates for various scene formats. The system templates are export files that are usually stored in the following directory: <code>Program Files\VertigoXmedia\Apps\Support</code>. Once imported into the Xmedia Server, the templates are stored in the Templates>System Templates category.</p> <p>The Image Template field only appears when one of the following device types is selected: Vertigo XG, Intuition XG, XPublishAgent, or Other.</p> |
| External Keyer | <p>Allows you to associate another device with the current device. This setting is specifically used when the CG output is keyed over the ImageStore output. In this way, specific commands can be sent to the ImageStore (the External keyer) when specific things are done on the CG.</p> |
| Publish Master | <p>Lets you select a master device.</p> |
| Clip Template | <p>The name of the template that is used to play out clips. The clip template's control-side must have a texture linked to a scene-side clip object. Use the Browse button to navigate to the Clip template.</p> <p>Note that among the Vertigo Suite's system templates are valid clip templates for various scene formats. The system templates are export files that are usually stored in the following directory: <code>Program Files\VertigoXmedia\Apps\Support</code>. Once imported into the Xmedia Server, the templates are stored in the Templates>System Templates category.</p> <p>The Clip Template field only appears when one of the following device types is selected: Vertigo XG, Intuition XG, XPublishAgent, or Other.</p> |

6 Click **Apply**.

7 In the **Device Configuration** dialog box, select the output device from the table at left and then click . The device appears in the **Active Device** box at right.

8 Click **OK**.

The output device name appears in the upper right-hand portion of the main Xstudio window. When connected, the LED beside it is green and when disconnected, it is red.



Fig. 2-1: Connectivity indicator

Note: If Xstudio disconnects from the output device (red LED), you can reconnect to it without having to access the **Device Configuration** dialog box. Simply select Xstudio's **Tools>Devices>Reconnect Active Device** menu command, or double-click the LED in the upper right-hand portion of the main Xstudio window.

Verifying Xstudio's installation and setup

Once you have installed and setup Xstudio on a client workstation, it is recommended that you perform the following quick procedure to verify that Xstudio was properly installed and is working properly.

The following procedure is intended to test the proper functioning of Xstudio by loading a template and scene into Xstudio (Design mode) and then previewing the layout on the connected device using Xstudio's Production mode.

- 1 Open Xstudio.
- 2 Verify that Xstudio is actively connected to the Xmedia Server and the output device.
- 3 Open the Asset Browser and navigate to the **Templates>Demo_Samples08>M11** category.
- 4 Right-click on one of the templates listed within the **M11** subcategory and select the **Publish To>Connected Device** command.
- 5 Again in the Asset Browser's **M11** subcategory, double-click the template that you just published.
The template and scene open in the project window.
- 6 Switch to Production mode by pressing **Ctrl+P** or by double-clicking the **Design LED** in the upper-right corner.
Xstudio switches over to Production mode and cues up the template in the Next To Air window.
- 7 Click the **Take** button and the scene should playout on the active output device.
- 8 Click the **Clear** button and the scene should be removed from the active output device.
- 9 Return to Design mode by pressing **Ctrl+P** or by double-clicking the **Design LED** in the upper-right corner.
- 10 Close the project window by clicking the project window's close button (**X**) or by pressing **Ctrl+F4**.
- 11 Click **No** when prompted to save the template.

3 Orientation to Xstudio's desktop

Xstudio's user-friendly authoring environment gives you easy access to the projects, assets, features, and tools that you need for creating and playing out to air compelling broadcast graphics displays.

Featuring a large area in which to work, Xstudio's key features are immediately accessible via menus, floating toolbars, buttons, icons and browsers.

Inspired by the natural workflow of creating and playing out broadcast graphics, Xstudio has two work modes: Design and Production. Since each mode is used for different purposes, the Xstudio interface provides unique interface features in each mode, which are among the topics discussed below.

While further chapters provide full instructions for using Xstudio, the following sections are intended to introduce and help you to become familiar with the Xstudio's interface features:

- [Starting Xstudio](#), on page 24
- [Design Mode vs. Production Mode](#), on page 26
- [Xstudio's menu commands](#), on page 27
- [Xstudio's desktop icons and toolbars](#), on page 33
- [Xstudio's primitives](#), on page 36
- [Asset Browser and Gallery Browser](#), on page 38
- [Properties Browser](#), on page 39
- [Project Windows - Control and scene sides](#), on page 42
- [Production mode environment](#), on page 46

Starting Xstudio

To start Xstudio:

- Click **Start>Programs>VertigoXmedia>Xstudio**.

Once the program opens, you will be prompted with the **Create New** dialog box (it will always display on startup unless you disable the **Show New Dialog on Startup** setting -- see [page 14](#)).

The **Create New** dialog box is an entry point to creating new projects and assets. While all of the options listed within this dialog box are all assets, they have been organized into separate tabs according to the type of functionality they provide. Definitions for the individual items listed on each tab are identified in the table below.

Note: For the purposes of this chapter's overview of the Xstudio interface, select **Template + Scene** and click **OK**.

| Tab | Options |
|-----------------|--|
| Projects | <p>Contains the Panel + Scene and Template + Scene options, which are basically assets that allow you to create and control the playout of broadcast graphics using a combined project containing a scene-side and a control-side.</p> <ul style="list-style-type: none">• The scene-side (Scene) contains objects that create the look and content of the broadcast graphic that is to be played on air.• The control-side (Template or Panel) contains objects that are used to control the playout of the scene objects, including the editing of editable content. <p>See page 5 and page 7 for more information about Template + Scene and Panel + Scene projects.</p> |
| Assets | <p>Contains the following options for creating standalone assets whose primary function is to build or support other assets.</p> <ul style="list-style-type: none">• Xmedia Object: Xmedia objects are pre-programmed broadcast graphics with built-in functionality that make them usable right away, as well as easily portable to other graphics projects. Examples of Xmedia objects include headlines, station bugs (or any other logo), tickers, and squeezes.• Panel: A panel (also known as control panel) is the software version of master control hardware that controls graphics on air. Panels are created for control purposes only, and as such need not include the scene (graphical output) that will display on the output device. If you have graphics built outside Xstudio that you would like to control, you can create a panel on its own (a standalone panel) by creating control objects and linking them by name to their external scene counterparts. Panels are ideal for controlling on air displays of sports scores, stock tickers, headlines, station bugs (or any other logo), video squeezes and more.• Template: Standalone templates are used to control pre-existing scenes in that were perhaps built outside of Xstudio. The template is a visual display, which contains editable fields and controls that operators can use prior to going on air to set the content and behavior of the scene objects. Templates may also contain logic for controlling the behavior of scene objects during playout. The template's logic is not manually accessible during playout and is designed to execute without any operator intervention. This makes templates ideal for sequential or order playout.• Page: Pages are built from templates. The template is a master copy that determines the layout and contents of graphics that will be taken to air in sequential order. Creating a page based on a template means the layout is already done and all you have to do is enter text, image files, audio, etc. specific to the broadcast. Pages are then loaded into playlists and taken to air in pre-determined order. See page 7 for more information about using Xstudio to create pages. |

| Tab | Options |
|---------------|---|
| Assets | <ul style="list-style-type: none"> • Scene: Scenes contain the collection of graphic objects that will be displayed on air during playout. Standalone scenes are appropriate if the scene's objects and playout do not require any control logic (static and/or timed playout). However, if the scene's objects are to be animated, data driven or controlled during playout, the scene should be associated with a control panel or template. |
| Other | <p>Contains the following options for creating assets which affect other assets' functionality.</p> <ul style="list-style-type: none"> • Script: For advanced users with specific requirements not addressed by the Xstudio application, the Script Editor can be used to create custom actions that can be called in response to any event. Commands can be written in VB Script. Once scripts have been written, they can be linked to control panels, pages, playlists, etc., using the Action Editor. See page 325 for more information about creating scripts. • Lookup Table: A lookup table is an asset that defines replacement values for incoming data. See page 191 for more information about creating scripts. |
| Recent | Lists the ten (10) assets most recently worked on. |

Design Mode vs. Production Mode

Xstudio has two work modes: **Design** and **Production**.

Design mode is Xstudio's authoring environment, which allows you to create and edit scenes, control panels, templates, and pages. You can also use it to modify the look of objects, animate scene objects, assign actions (an action executes a specific command), create links to data... etc.

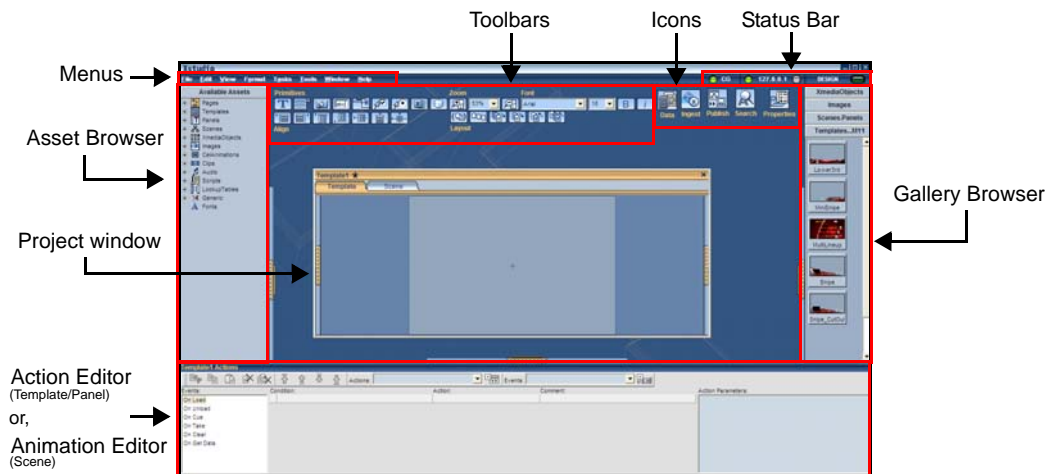


Fig. 3-1: Xstudio's Design mode environment components

When you have finished authoring, you can preview the playout by switching to **Production** mode and take your finished work to air. Production mode allows you to edit content (e.g. change text in text boxes and/or make new selections from drop-down lists) and control graphics (e.g. operate a control panel).



Fig. 3-2: Xstudio's Production mode environment components

Use one of the following methods to switch between Design and Production mode:

- From the **View** menu, select **Production Mode**.
- Press **CTRL+P**
- Click the **Design Mode** button  or **Production Mode** button  located in the upper-right corner of the menu bar.

Xstudio's menu commands

Many of Xstudio's mode key features are thematically organized within the menu bar command structure.

The following sections provide functional definitions of the Xstudio menu commands:

- [File menu commands](#), on page 27
- [Edit menu commands](#), on page 28
- [View menu commands](#), on page 28
- [Format menu commands](#), on page 30
- [Tasks menu commands](#), on page 31
- [Tools menu commands](#), on page 31
- [Windows menu commands](#), on page 32
- [Help menu commands](#), on page 32

File menu commands

The **File** menu contains commands that are related to opening, closing, saving, importing, and exporting existing files, creating new files, and exiting the application.

| | |
|----------------------|--|
| New | Opens the Create New dialog box, which allows you to create a new asset. While all of the options listed in this dialog box are assets, they have been organized into separate tabs according to the type of functionality they provide. |
| Open | Opens the MediaStore Browser , which allows you to navigate through the system's asset directories to find and open an existing template, control panel, Xmedia object, page, script, scene or lookup table. |
| Close | Closes the currently active window. This command works on most windows, except those that have their own Close or Cancel button. |
| Save | Saves changes made to the currently selected project/asset, as well as any specified metadata (e.g. author name, description, keywords, etc.). If saving the asset for the first time, it opens the Save dialog box (otherwise it simply saves using the current asset name). See Saving Xstudio projects and assets , on page 252 for more information. |
| Save As | Displays the Save As dialog box, which allows you to save the currently selected project/asset using another name and/or directory location. See Saving Xstudio projects and assets , on page 252 for more information. |
| Overwrite | Replaces (overwrite) the content of an existing (saved) project with the content of the currently selected project. The project that is overwritten maintains its existing name and asset ID. See Saving Xstudio projects and assets , on page 252 for more information. |
| Import Assets | Opens the Select VertigoXmedia Asset Export File dialog box, which allows you to navigate to and then import asset files that are used by, or stored on, another application within the Xmedia Suite. See Importing assets from another Vertigo system , on page 57 for more information. |

| | |
|----------------------|--|
| Export Assets | Opens the Export Assets dialog box, which allows you to export assets by sending a copy of the asset out of Xstudio and into a folder of your choice. See Exporting assets to be used on another Vertigo system , on page 255 for more information. |
| Exit | Allows you to quit Xstudio, prompting you first to save if any edits have not yet been saved. |

Edit menu commands

The **Edit** menu contains commands that are related to editing content, like undo, redo, cut, copy, paste, and delete.

| | |
|--------------------------------|---|
| Undo | Allows you to undo an edit on the currently active window. |
| Redo | Allows you to redo (repeat) an edit on the currently active window. |
| Cut | Cuts the selected object(s) on the currently active window to the clipboard. |
| Copy | Copies the selected object(s) on the currently active window to the clipboard. |
| Paste | Pastes the clipboard contents onto the currently active window. |
| Paste Scene to Template | Copies the selected objects from the Scene side and pastes them on the Control-side (Template or Panel). This is a quick way of creating a representative appearance of the scene on the control-side making it easy to identify project. |
| Delete | Deletes the selected object(s). |

View menu commands






The **View** menu contains commands that are related to activating or hiding Xstudio modes, browsers, and tools that are used by, or within, the Xstudio authoring environment.

| | |
|-------------------------|--|
| Production Mode | Switches the Xstudio interface from Design mode to Production mode, and vice versa. See Design Mode vs. Production Mode , on page 26 for more information. |
| Action Editor | Displays or hides the Action Editor when the control-side of the project is active or the Animation Editor when the Scene-side of the project is active (only available in Design mode). The Action Editor is a collapsible panel on the Xstudio interface when the control-side of a project is active. The Action Editor allows you to specify the actions that are executed when an event is sent to an object. |
| Property Browser | Displays or hides the Property Browser (only available in Design mode). The Property Browser displays and allows you to edit the attributes, or properties, of the currently selected object(s) on the canvas, including the canvas itself. For example, the Property Browser allows you to change the size and appearance of objects, transitions, images, cel animations, data sources, and more. |

| | |
|-----------------------------------|---|
| Search Assets | Opens the Search dialog box, which allows you to quickly locate an asset stored on the Xmedia Server. Offering several search parameters, like name, keywords, and type, you can tailor your search to your needs. |
| Data Manager | Displays or hides the Data Manager (only available in Design mode). The Data Manager allows you to change the data sources associated with an Xmedia object, or link data to a primitive. Data sources can be databases, Excel files, text files or RSS (an XML format that distributes news headlines, links and abstracts collected from the Web). See Linking objects to a data source , on page 177 for more information. |
| Asset Browser | Displays or hides the Asset Browser (only available in Design mode). The Asset Browser is a collapsible panel on the left-side of the Xstudio interface that allows you to view and access assets according to type and category. |
| Gallery | Displays or hides the Gallery Browser (only available in Design mode). The Gallery Browser is a collapsible panel on the right-side of the Xstudio interface that displays the contents of an Asset Browser category as proxies (thumbnails) or as a list of detailed asset information. The contents of the Gallery Browser can be dragged and dropped into control panels or templates. By default, only the root category of Xmedia objects appears in the Gallery Browser, and you have added other categories manually. |
| Publish Progress | Displays or hides the Publish Progress window, which allows you to see a list of all of the assets that have been published during the current Xstudio session. |
| Message Window | Displays the Messages window, which lists errors as they occur. These error messages provide valuable information when troubleshooting problems. |
| Hide Browsers/Tool Windows | In one action, this command hides all currently open browsers and tool windows, including the Asset Browser, Gallery Browser, Data Manager, Property Browser, Action Editor and Animation Editor. Note: The Hide Browsers/Tool Window command only closes the currently opened browser and tool windows. Project or asset windows can similarly be closed in one action using the Windows>Close All command (see Windows menu commands , on page 32). |
| Log File | Creates and displays a log file in a NotePad document. The log file records events and errors as the program runs, which is practical when troubleshooting. See Logging settings , on page 13 for related information. |

Format menu commands

The **Format** menu contains commands that are related to the positioning and sizing of objects placed on the template, control panel, or scene.

| | |
|-----------------------|--|
| Group | Takes several separate objects selected together on the currently selected window and groups them into a single grouped object. |
| Ungroup | Ungroups a single grouped object on the currently selected window back into separate objects. |
| Bring To Front | Brings an object to the front layer on the canvas. |
| Bring Forward | Brings an object one layer closer to the front on the canvas. |
| Send Backward | Sends an object one layer toward the back on the canvas. |
| Send To Back | Sends an object to the back layer on the canvas. |
| Align | <p>Using the first object that is selected as a reference (primary object), the following commands allow you to set the alignment of the selected objects on the canvas.</p> <ul style="list-style-type: none"> • Top:  Aligns all selected objects on the currently selected window with the top border of the primary object. • Bottom:  Aligns all selected objects on the currently selected window with the bottom border of the primary object. • Middle X:  Aligns all selected objects on the currently selected window with the horizontal centre of the primary object. • Middle Y:  Aligns all selected objects on the currently selected window with the vertical centre of the primary object. • Centers:  Aligns selected objects with both the horizontal and vertical centers of the primary object. |
| Size | <p>Allows you to edit the size of objects selected on the canvas.</p> <p>Choose from:</p> <ul style="list-style-type: none"> • Same Width: Makes all selected objects the same width as the primary object (first object selected). • Same Height: Makes all selected objects the same width as the primary object (first object selected). • Same Width/Height: Makes all selected objects the same width and height as the primary object (first object selected). • Biggest: Makes all selected objects the same size as the biggest object. • Smallest: Makes all selected objects the same size as the smallest object. • Natural Size: Reverts scaled images, cel animations and clips back to their natural size (i.e. the size they were when they were ingested). |
| Add Guide | <p>Allows you to display virtual lines on the canvas that you can use as a reference when arranging or aligning objects. See Positioning objects using guides, on page 81 for more information.</p> <p>Choose from:</p> <ul style="list-style-type: none"> • Horizontal: Makes a horizontal guide available for dragging into the canvas from the top of the window. • Vertical: Makes a vertical guide available for dragging into the canvas from the left side of the window. |

Tasks menu commands

The **Tasks** menu contains commands that are used for setting up and previewing the playout of a scene on air. As such, these commands are disabled in Design mode and they are only available when Xstudio is in **Production** mode. See [Playing out a scene using Xstudio's Production mode](#), on page 257 for more information.

| | |
|-----------------------|---|
| Cue | Cues the selected scene or page as the next to air or execute. |
| Take | Executes the cued action, or takes the cued scene/page to air, and then cues the next action or scene/page in the playlist. |
| On Air Out | Takes the currently airing graphics off air, but does not take the next cued page to air. |
| Clear Playlist | Empties the playlist. |
| Reset Playlist | Reloads the last saved version of the playlist and then cues the first item. |
| Clear Device | Stops the current playout on the output device and clears the device display. |

Tools menu commands

The **Tools** menu contains commands that allow you to set and configure Xstudio to interact with other devices and applications.

| | |
|-------------------------|--|
| Devices | <p>Provides access to configuration and settings tools that allow Xstudio to interact with external devices and applications.</p> <p>Choose from:</p> <ul style="list-style-type: none"> • Configure Devices: Displays the Device Configuration dialog box, which allows you to either define a new output device, or select one to be connected to Xstudio. See Connecting Xstudio to an output device, on page 18. • Reconnect Active Device: Re-sends the connection commands to reconnect an active device that has been disconnected. Note that the device's connection status is easily known by verifying if the LED is on or off on the device's label in the Xstudio status bar in the top-right corner of the Xstudio interface. • Active Device Settings: Functionality not currently implemented. • Quantel Ingest: Opens the Quantel FTP client dialog box, which is used to select and download an image file in Targa format. • Ingest Scene: Allows you to add a scene that was not created in Xstudio to the system. First a dialog box is displayed warning that scene ingesting overwrites all currently active scene data. When asked if you would like to proceed click the Yes button. A File dialog box appears, allowing you to navigate to the folder containing the output device scenes. See Ingesting files created using third-party applications, on page 59 for related information. |
| Settings | Displays the Settings dialog box, which allows you to modify Xstudio' settings as required. See Configuring Xstudio's settings , on page 11 for more information. |
| Plugins | Displays the Plugin Manager dialog box, which allows you to register plugins that can be used in conjunction with Xstudio. |
| External Control | Displays the External Control/GPI Mapping Manager , which allows you to set up a method of controlling Xstudio externally. See External Controls , on page 331 for related information. |

Windows menu commands

By default, the Windows menu contains only one command: **Close All**. As you open project or asset windows, the Window menu lists their names as submenu items. When multiple windows are opened during a session, selecting the window's name from the Windows menu provides you with a quick way of bringing that specific window to the front.

Rather than closing each window separately, the **Close All** command closes all of the currently open windows in one action.

Note: The **Windows>Close All** command only closes the currently opened project or asset windows. Browser and tool windows can similarly be closed in one action using the **View** menu's **Hide Browsers/Tool Window** command (see [View menu commands](#), on page 28).

Help menu commands

The **Help** menu contains commands that provide Xstudio product information, like licensing and software version identification.

| | |
|------------------------|--|
| Contents | Functionality not currently implemented. |
| License Manager | Opens the License Manager dialog box, which allows you to copy the customer ID and send it to our Technical Support team. Once you receive your license key, paste it into the License Key box. Please consult the Vertigo Suite's Release Notes documentation for further details. |
| About | Displays information about the version of Xstudio that you are using. |




Xstudio's desktop icons and toolbars








In Design mode, Xstudio's authoring environment provides you with immediate access to key features by placing floating toolbars, buttons, icons, and browsers directly on the Xstudio interface.


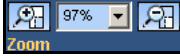


You can move the Xstudio toolbars and icons by dragging them by their titles (e.g. the word "Font") to its new location. You can also change the orientation of the floating toolbars by double-clicking on their titles making them line up vertically or horizontally.

Should you find yourself short on space in the interface because too many browsers and tool windows are open at the same time (e.g. Asset Browser, Gallery, Data Manager, Action Editor, Property Browser), you can instantly close all of them at once by selecting **Hide Browsers/Tool Windows** from the **View** menu, or by clicking **CTRL+D**.

The following table identifies and describes the toolbars and icons that are accessible in Design mode:

| | |
|---|---|
| <p>Data Manager icon</p>  | <p>Displays or hides the Data Manager. The Data Manager allows you to change the data sources associated with an Xmedia object, or link data to a primitive. Data sources can be databases, Excel files, text files or RSS (an XML format that distributes news headlines, links and abstracts collected from the Web).</p> <p>See Linking objects to a data source, on page 177 for more information.</p> |
| <p>Ingest icon</p>  | <p>Opens the Ingest Files dialog box, which allows you to create Xstudio assets from files that are not native to Xstudio, e.g. image and audio files. While these files are not actually modified in any way when ingested, they are made into assets -- in other words, they become usable within Xstudio. See Ingesting files created using third-party applications, on page 59 for more information.</p> |
| <p>Properties Browser icon</p>  | <p>Displays or hides the Properties Browser (only available in Design mode). The Property Browser displays and allows you to edit the attributes/properties, of the object selected on the canvas, or the canvas itself. For example, the Property Browser allows you to change the size and appearance of objects, transitions, images, cel animations, data sources, and more.</p> <p>See Setting the canvas' properties, on page 48 and Xstudio properties, on page 285 for more information.</p> |

| | |
|---|---|
| <p>Publish icon</p>  <p>Preview icon</p>  | <p>Publishes the scene to your output device. The progress of the publishing process is reported in the status bar until publishing is complete.</p> <p>See Publishing assets to output devices, on page 253 for more information.</p> <p>The Preview icon replaces the Publish icon when you click the Publish icon while holding down the Ctrl key (repeat to return to the Publish icon).</p> <p>The Preview icon allows you to preview of the scene you are working on it in Xstudio's Design Mode. The preview is displayed in real-time on the currently connected output device, so you can see the results of any edits that you make to the scene right away.</p> <p>See Playing out a scene using Xstudio's Production mode, on page 257 for more information.</p> |
| <p>Search icon</p>  | <p>Opens the Search XMS window, which allows you to quickly locate assets stored on the Xmedia Server.</p> <p>See Searching for projects/assets on the Xmedia Server, on page 54 for more information.</p> |
| <p>Primitives toolbar</p> |  <p>→ Control-side primitives</p>  <p>→ Scene-side primitives</p> <p>Adds primitive assets, like custom labels, drop-down lists, buttons, text boxes and image objects to your template, control panel, and/or scenes. A different set of primitives are used when working with the control side (templates and control panels), than with the scene side. This means that the toolbar switches automatically when the active window also switches to/from a scene or template/control panel. See Xstudio's primitives, on page 36 for more information.</p> |
| <p>Layout toolbar</p> |  <p>Groups and ungroups objects, and moves selected objects above or below others. See Grouping and ungrouping objects on the canvas, on page 77 for more information.</p> |
| <p>Align toolbar</p> |  <p>Allows you to easily align two or more objects, with precision, in relation to the primary object (the first object selected). See Moving and aligning objects on the canvas, on page 80 for more information.</p> |








| | |
|---------------------|---|
| Font toolbar |  <p>Modifies the font properties of objects, including the font type, size, and style. See Modifying the font properties of text objects, on page 84 for more information.</p> |
| Zoom toolbar |  <p>Zooms in and out of the canvas (the area within a window for a scene, template, control panel, or Xmedia object). The drop-down list in this toolbar allows you to select from preset percentage options, or select Fit (which resizes the currently open window's home area and its contents to fit the canvas). You can also click  to zoom in, or  to zoom out.</p> <p>Alternative access:</p> <ul style="list-style-type: none">• Press ALT + the (+) or (-) keys on your keyboard• Move the mouse wheel up and down <p>See page 42 for more information about using the canvas, including the home area.</p> |



Xstudio's primitives

Primitives are objects that act as building blocks in Xstudio, e.g. text box objects, button objects, image objects, clip objects. Certain primitives are only available for the **Scene** tab, whereas others are exclusive to the **Control** side.








- [Control side primitives](#), on page 36
- [Scene side primitives](#), on page 37

Control side primitives

| Primitive | What it does |
|--|---|
| Text  | Adds a text object to the canvas. Double-click on the object to enter text. This object differs from a text box object (explained below) in the sense that it's intended more for labelling purposes. For example, you could use the text object to add a title or label to the canvas, whereas the text box object would more suited to adding a field that the user could use to enter breaking headlines. |
| Drop-box  | Creates a drop-down list of choices on the canvas. Double-click on the object to enter text, one choice per line. |
| Button  | Adds a button object to the canvas that can be set up to trigger actions when clicked. If desired, you can apply an image to the button. Instructions for doing so are provided in Adding primitives to the canvas , on page 71. You can also access Button properties from the Property Browser (e.g. color, font). See Working with control objects , on page 161. |
| Text box  | Adds a text box to the canvas. Double-click on the object to enter text. While the text object (explained above) should be used for labelling purposes, the text box object is ideally suited for controlling text on the scene. For example, you could have a text box object connected to a Text 2D object (explained below) on the scene. Changing its contents on the control side would make corresponding changes on the scene side. |
| Image  | Adds an image object to the canvas, to which an image asset is assigned. Instructions for assigning an image asset to the object are provided in Adding primitives to the canvas , on page 71. |
| Check box  | Adds a check box object to the canvas that can be set up to trigger actions when selected. |
| Radio button  | Adds a radio button object to the canvas that can be set up to trigger actions when selected. |

| Primitive | What it does |
|---|--|
| ActiveX  | <p>Adds an active X component to the canvas. You can select any third party or custom component available on your PC. Once in the canvas, display the Property Browser and enter the registered class name of the active X object in the Prog ID field. For example, if adding the Microsoft Media Player, you would enter MediaPlayer.MediaPlayer.1.</p> <p>For details on using the Property Browser, refer to Working with control objects, on page 161.</p> |
| Timer  | <p>Adds a timer object to the canvas. Timer functionality is set using specific properties in the Property Browser. Timers are explained in more detail on page 176.</p> |

Scene side primitives

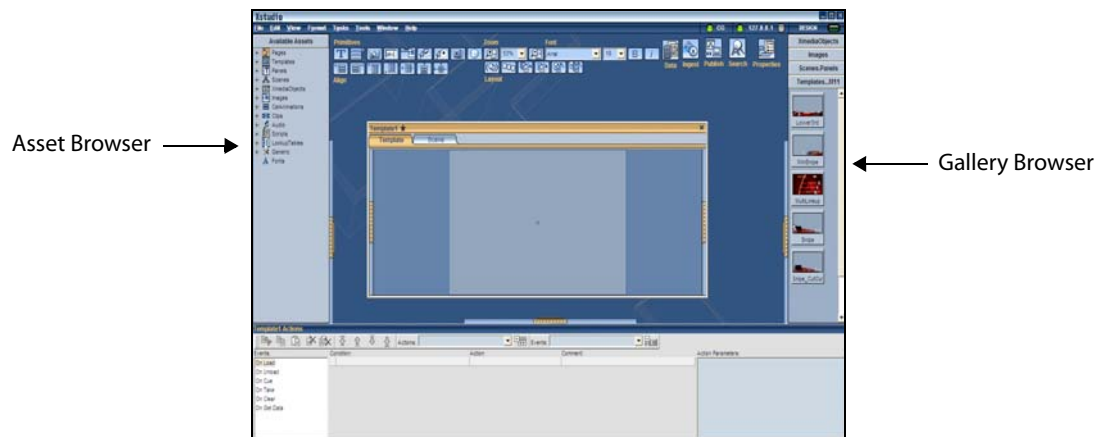
| Primitive | What it does |
|---|--|
| Text 2D  | <p>Adds a text object to the scene. These are usually controlled by a text box object on the control side.</p> |
| Cel animation  | <p>Adds a cel animation to the scene, to which a cel animation asset is assigned.</p> <p>Instructions for assigning a cel animation asset to the object are provided in Adding primitives to the canvas, on page 71“.</p> |
| Image  | <p>Adds an image to the scene, to which an image asset is assigned.</p> <p>Instructions for assigning an image asset to the object are provided in Adding primitives to the canvas, on page 71.</p> |
| Clip  | <p>Adds a video clip to the scene, to which a clip asset is assigned.</p> <p>Instructions for assigning a clip asset to the object are provided in Adding primitives to the canvas, on page 71.</p> |
| Video In  | <p>Adds a video in display to the scene.</p> |
| Crawl  | <p>Adds a crawl to the scene.</p> <p>Double-click in the crawl to add text, images and/or cel animations. To edit the crawl once it's been added to the scene, double-click it. This opens the CrawlGroup window, which is explained in detail in Crawl and Roll objects on the scene canvas, on page 104.</p> |
| Roll  | <p>Adds a credit roll to the scene.</p> <p>Double-click in the roll to add text. To edit the roll once it's been added to the scene, double-click it. This opens the RollGroup window, which is explained in detail in Crawl and Roll objects on the scene canvas, on page 104. Use this along with the Property Browser to make changes to the roll.</p> |

Asset Browser and Gallery Browser

The Asset Browser and the Gallery Browser are the two collapsible panels that occupy the left and right-hand sides of the Xstudio workspace respectively.

While much of their functionality overlaps, the main purpose of the Asset Browser and the Gallery Browser are:

- **Asset Browser** - Displays a list of all the assets stored on the Xmedia Server that are available and/or relevant to Xstudio. The assets are organized in a tree structure according to their type (e.g. scenes, panels, templates, images, clips, scripts, etc.). The Asset Browser allows you to further organize assets by creating categories within each type's root category (type). Having a logical organization of asset, easily accessible allows you to quickly drag and drop assets from the Asset Browser onto an open canvas.
- **Gallery Browser**- Displays the contents of an Asset Browser category. Since the Asset Browser only displays assets in a tree structure, the Gallery Browser provides you with a better view of a category's assets in the form of thumbnail images or a detailed view. The Gallery Browser allows you to better identify the exact assets you want to drag and drop onto your canvas.



Information and instructions related to using the Asset Browser and Gallery Browser are provided in the following sections:

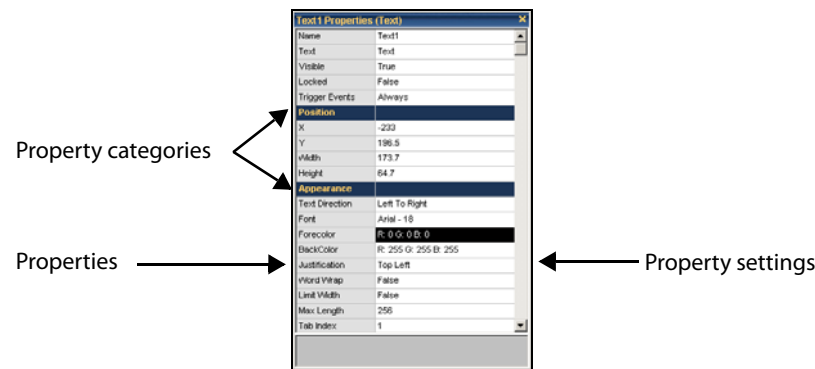
- [Opening existing projects](#), on page 53
- [Adding assets to the canvas](#), on page 70
- [Asset Browser](#), on page 261
- [Gallery Browser](#), on page 275

Properties Browser

The **Properties Browser** lists the current property settings for an object selected on the canvas, a selected group of objects, or for the canvas itself. Editing the properties allows you to change the appearance and behavior of the objects placed on the canvas, or the canvas itself.

While some properties are common to all object types, others are specific to the type of object that is selected on the canvas or to the canvas itself. As such, the Properties Browser only lists those properties that are relevant to the currently selected object.

Note: If an object has both a control side and scene side, you can only access the properties for one side at a time. The Properties Browser's user interface mainly groups the properties into thematic categories. For example, properties that define the aesthetics of a text object are grouped under the **Appearance** category.



Each property listed in the Properties Browser has a settings text box to the right of the property's name, which displays the property's current setting. Editing these settings allows you to change the appearance or behavior of the objects or the canvas. Depending on the type of property setting, clicking in the property's settings text box allows you to enter a setting value directly in the text box, select from a drop-down list , or click a browse button to open a dialog box for selecting a setting.

The following sections provide information and instructions for using the Properties Browser to change the appearance and behavior of objects and/or the canvas:


- [Setting the canvas' properties](#), on page 48
- [Working with scene objects](#), on page 87
- [Working with control objects](#), on page 161

While [Xstudio properties](#), on page 285 provides an alphabetical listing and descriptions for each of the properties associated with the Xstudio objects and canvas.

Opening the Properties Browser

The Properties Browser allows you to view and edit the properties associated with the currently selected object, or the canvas itself. The instructions below describe how to open the Properties Browser window.

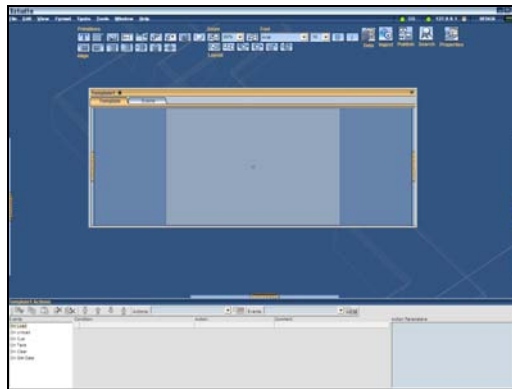
To open the Properties Browser:

- 1 Select the object, or the canvas, depending on whether you want to view the object's or the canvas' properties.
- 2 Perform one of the following:
 - Select the **Properties** icon  from Xstudio's desktop.Or,
 - Select the **View>Property Browser** menu command.Or,
 - Right-click on the selected object/canvas, and select the **Properties** command.Or,
 - Press **Ctrl+R**.
- 3 The Properties Browser window appears and lists the property settings for the selected object/canvas.

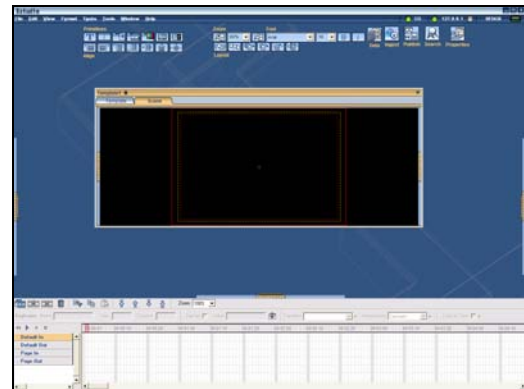
Note: It is not necessary to reopen the Properties Browser window each time you want to view the properties of a different object. If the Properties Browser window is already open, selecting another object or the canvas automatically updates the Properties Browser window and lists the properties of the selected item.

Action Editor and Animation Editor

The Action Editor and the Animation Editor occupy the same collapsible panel at the bottom of the Xstudio desktop in Design mode. When the control-side (Template or Panel) of the project window is active, the Action Editor is displayed; however, when the scene-side is active, the Animation Editor is displayed.



↑
When the control-side is active,
the Action Editor is displayed.



↑
When the scene-side is active,
the Animation Editor is displayed.

The **Action Editor** lets you set up actions to occur in response to an event. An event is a normal task within the payout that is used to also trigger an action. For example, possible events might be the user clicking a button on the control panel, a page being loaded, or Xstudio checking the database for updated information. Then the action executes a specific command.

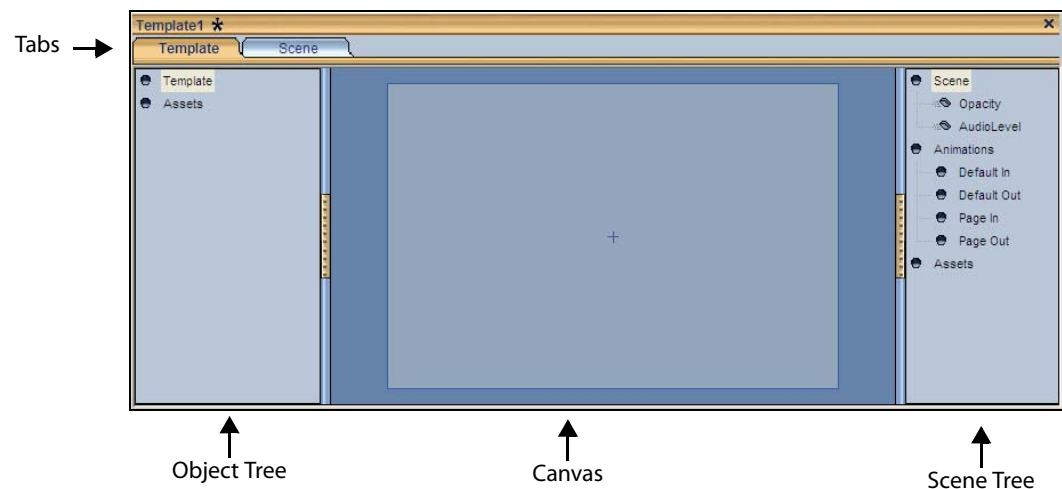
The Animation Editor allows you to create scene animations by changing the properties of the objects on the scene over a period of time to create visual effects. The Animation Editor also allows you to integrate and control the payout of secondary animations within the primary animation using keyer channels and command channels.

Information and instructions related to using the Asset Browser and Gallery Browser are provided in the following sections:

- [Animating scene objects](#), on page 111
- [Adding logic to control objects](#), on page 211

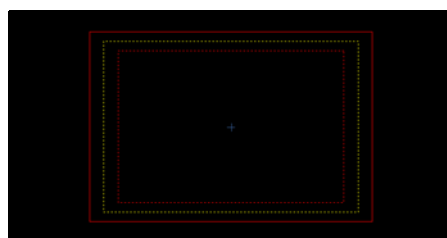
Project Windows - Control and scene sides

In Design mode, Xstudio projects (Templates, Control Panel and Scenes) are created and/or edited in the Project window. There are four (4) main parts to the Projects window:

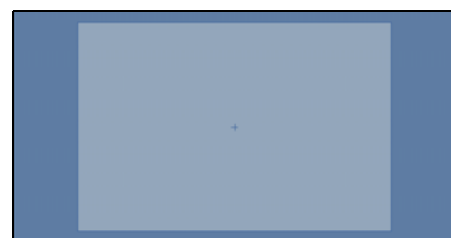


- **Tabs:** For projects that have a control and scene side, the Project window will feature two tabs (Template/Panel) and Scene, which allow you to switch between the control-side's canvas to the scene-side canvas.
- **Object Tree:** The Object Tree is located within the collapsible panel on the left-side of the Project window. Using two main categories, **Template** or **Panel** and **Assets**, the Object Tree displays the objects and assets that have been added to the control-side's canvas. Clicking on an object in the tree selects it on the canvas. See [Assigning an action to an object](#), on page 221 for related information.
- **Scene Tree:** The Scene Tree is located within the collapsible panel on the right-side of the Project window. Using three main categories, **Scene**, **Animations** and **Assets**, the Scene Tree lists the objects and assets that are currently located on the canvas. Items from the Scene Tree can be dragged and dropped to create connections with control panel or template objects. See [Displaying the animated properties in the Scene tree](#), on page 130 for related information.
- **Canvas:** The canvas is the work area onto which you drag-and-drop assets, primitives and/or Xmedia objects, and then manipulate them (e.g. aligned, layered, grouped) to create functional and attractive graphics displays and control panels.

There are significant differences in the appearance and functionality of the canvas for scenes and the canvas for control screens.



The canvas for scenes features safe areas



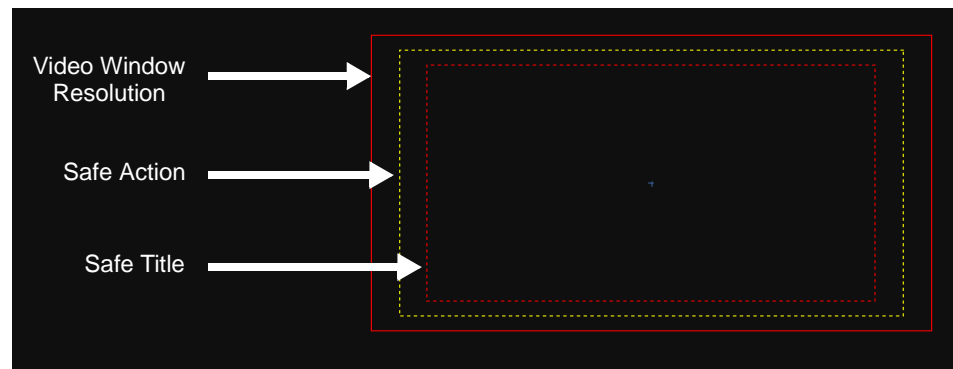
The canvas for templates, control panels, and Xmedia object windows features the home area

The scene's canvas (whether it is an independent scene, or one that has been built with an accompanying control panel or template) features safe areas which are indicated by the set of dotted and solid lined frames on the canvas. The safe areas help you to position objects within the area of the canvas that becomes visible when the scene is brought to air.

Control-side's canvas features a home area, which is a boxed section that defines the work area onto which you can place control objects. The placement of objects on the home area only affects the control-side and has no effect on the placement of objects on the scene or on air.

Safe areas and resolution format

The scene's canvas (whether it is an independent scene, or one that has been built with an accompanying control panel or template) features safe areas, which are designated on the canvas by the set of dotted and solid lined frames on the canvas. The safe area frames help you to position objects on the canvas to ensure that the objects are visible when the scene is brought to air.

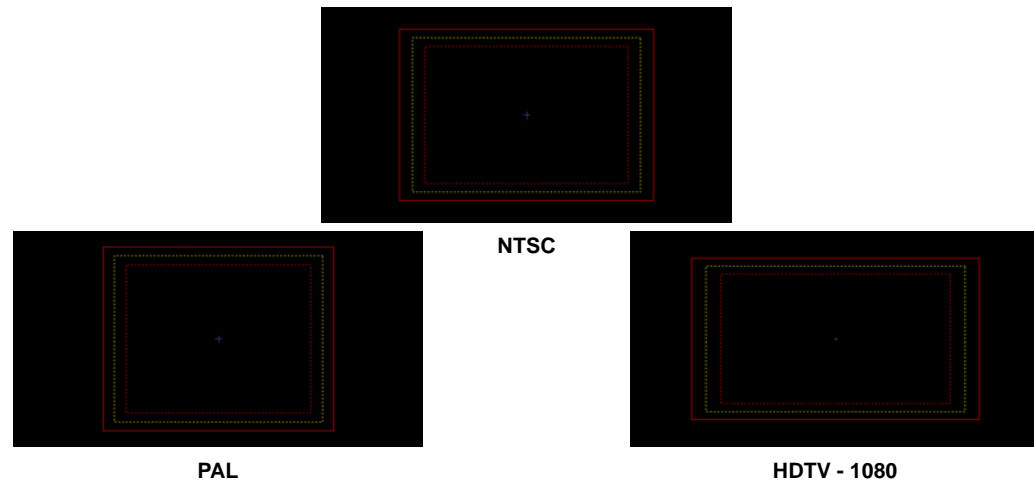


Each of the safe areas defines the likelihood that an object placed in the safe area will be visible in the on air broadcast. The following table indicate the location and meaning of each of the safe area frames.

| | |
|--|---|
| Video window resolution (outer solid red line) | The area within this frame represents the signal size, as determined by the scene format property. Any object placed between the Video Window Resolution frame and the Safe Action frame will not likely be displayed. Likewise, any object placed outside of the Video Window Resolution frame is guaranteed not to be visible on air. |
| Safe action (yellow dotted line) | The area within this frame represents ninety percent (90%) of the signal size. Whether or not objects placed within the area between Safe Action frame and Safe Title frame display depends on the type of television airing the broadcast. |
| Safe title (inner red dotted line) | The area within this frame represents eighty percent (80%) of the signal size. Anything placed within this border is guaranteed to display on any screen. |

When you are creating a scene, it will be the output device's (i.e. television) resolution format that determines the size and shape of scene format. This ensures that all of the objects placed on the scene appear in the appropriate places on air. Consequently, the size

and shape of the safe area frames is determined by the television format that you select for your scene using the **Scene Format** property field (see [SceneFormat](#), on page 51).



Zooming in/out of the canvas

Whether you are working on the scene-side or control-side canvas, you may want to see the characteristics of objects in greater detail by zooming in on the canvas. Similarly, you can get an expanded or distanced view of the canvas by zooming out of the canvas.

There are several methods for zooming in or out of the canvas:



Alt key sequence

- While holding down the **ALT** key, press the + (plus) key on your keyboard's number pad.
- While holding down the **ALT** key, press the - (minus) key on your keyboard's number pad.
- Hold down the **ALT** key and the right mouse button, then drag a rectangle around the specific area that you want to zoom in on.

Mouse wheel

- Click anywhere on the canvas and roll the mouse wheel forward to zoom in or roll the mouse wheel backward to zoom out.

Zoom toolbar

- Click the **Zoom In** button  or the **Zoom Out** button  on the Zoom toolbar.
- Select a higher or lower percentage scale factor from the Zoom toolbar's drop-down list.
- You can easily return to the canvas to this default scaling state by selecting the **Fit** command from the Zoom toolbar's drop-down list.

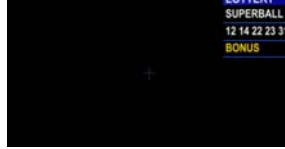
Note: When you zoom in on a portion of the canvas, you may not see certain areas of the canvas. Rather than zooming out and sacrificing the current zoomed view, you can maintain the zoom scale and pan across the canvas to the desired area. See [Panning across the canvas](#), on page 45 for more information.

Panning across the canvas

When you zoom in on a portion of the canvas, you may not see certain areas of the canvas. Rather than zooming out and sacrificing the current zoomed view, you can maintain the zoom scale by panning across the canvas to the area that was otherwise outside of the window's display.



Original scene (centered & Fit)



Canvas centered and zoomed in 150%, but the object is not completely visible



Panned to the right to view the entire object, while still zoomed

To pan across the canvas:

While holding down the **ALT** key on your keyboard:

- Use the arrow keys on your keyboard to move the canvas toward the desired location.

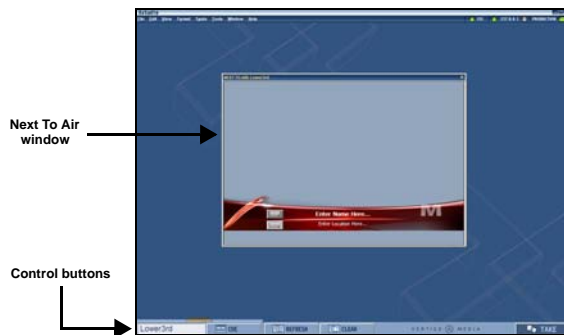
Or,

- Hold down the left mouse button while moving the mouse to drag the canvas to the desired location.

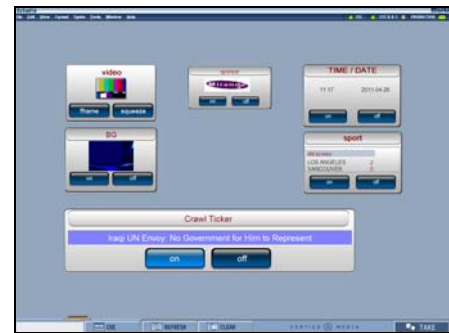
Production mode environment

Xstudio's Production mode allows you to preview the playout of the active project on the connected output device. See [page 26](#) for instructions on how to switch between Design and Production mode.

The Production environment is slightly different depending upon the type of project that is to be previewed.



Template or Page in Production mode



Control Panel in Production mode

When a template or page is previewed in Production mode, the template is displayed the **Next To Air** window, which allows you to make changes to the scene's editable fields before it goes on air (i.e. type or select specific text to display). The **Control** buttons (**Cue**, **Refresh**, **Clear**, **Take**) are available to templates and pages and allow you to control the playout on the device.

When a control panel is previewed in Production mode, it occupies the entire display and the Control buttons are not needed since you will control the playout by manually interacting with the panel's elements.

Further information and instructions for using Xstudio's Production mode are provided in [Playing out a scene using Xstudio's Production mode](#), on page 257.

4 Creating and opening Xstudio projects and assets

This chapter provides information and instructions for creating new Xstudio projects, as well as how to locate and open existing projects and assets that are stored on the Xmedia Server.

You may also want to use projects and assets that are stored on another system, or were created using a third-party application. In such cases, you must import or ingest these assets into the Xmedia Server before being able to use them in Xstudio.

The following topics are covered in this chapter:

- [Creating new projects and assets](#), on page 48
- [Opening existing projects](#), on page 53
- [Searching for projects/assets on the Xmedia Server](#), on page 54
- [Importing assets from another Vertigo system](#), on page 57
- [Ingesting files created using third-party applications](#), on page 59

Creating new projects and assets

When you launch Xstudio (**Start>Programs>VertigoXmedia>Xstudio**), the **Create New** dialog box automatically appears (if Xstudio's **Show New Dialog on Startup** setting is enabled). The **Create New** dialog box is an entry point to creating new projects and assets (definitions for the individual items listed on each tab are provided on [page 24](#)).

Note: The **Create New** dialog box can also be opened by selecting the **File>New** menu command or the keyboard shortcut: **Ctrl+N**.

To create a new project or asset in Xstudio:

- 1 Open the **Create New** dialog box.
- 2 Select the type of project or asset that you want to create and click **OK**.
An empty canvas appears. Before you begin to author, we recommend that you set the canvas' properties, especially the scene canvas' [SceneFormat](#) property.


Setting the canvas' properties

Each type of Xstudio canvas (template, panel, scene) has a series of properties which affect the canvas' look and behavior. You can edit an individual canvas' properties by opening the Properties Browser and clicking anywhere on the canvas area. See [page 39](#) for instructions on opening the Properties Browser.

The following table provides an alphabetical listing and descriptions for each of the properties associated with the Panel, Template, and/or Scene canvas:

| | |
|---|--|
| Auto Load Scene [AutoLoadScene] | <ul style="list-style-type: none">• Applies to: Panel or Template canvas• Description: Determines whether or not to automatically load the scene when you switch to Production mode. Select False if your output device automatically loads the scene, or if the scene is being controlled by more than one control panel (since reloading the scene every time a new panel goes to Production mode may cause the scene to flicker).• Accepted values: Select from the drop-down list: True or False. |
| Auto Name [AutoName] | <ul style="list-style-type: none">• Applies to: Template canvas• Description: Automatically generates a name for the pages created from the template that this property applies to.• Accepted values: To specify or edit the name that will automatically be assigned to the pages, simply type the name (text) in this property's text box. |
| Auto Take [AutoTake] | <ul style="list-style-type: none">• Applies to: Panel canvas• Description: Determines whether or not to automatically take the panel to air when you switch to Production mode. If you select False, the Take button will be amber when you switch to Production mode, which signifies that you must click the button to take the panel to air. Note that if a scene is loaded it appears in the output. However, animations will not run, nor will data tick until you click the Take button.• Accepted values: Select from the drop-down list: True or False. |

| | |
|--|---|
| <p>Available Takes [AvailableTakes]</p> | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Sets the number of takes (<Active>Take Number property) that must be issued to complete the display of the scene. This property is often used to display scenes containing bullet points that are intended to be taken to air one item at a time. Such a scene would have its Available Takes property set to the number of items. In simple scenes, the Available Takes property is usually set to 1. • Accepted values: Enter a value that represents the number of instances that will be required to completely payout the scene. In simple scenes, the Available Takes property is usually set to 1. |
| <p>BackColor [BackColor]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: Sets the backcolor for the canvas. • Accepted values: Click in the property's text box and then click the Browse button to display the Color dialog box. Select an RGB color definition (e.g. R: 0 G: 0 B: 0). |
| <p>Background Location [BgLocation]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: Identifies where the background asset is located within Xstudio and on the Xmedia Server. • Accepted values: This property can be automatically filled in when the background asset is dragged from the Asset Browser or Gallery Browser to the Background Name property in the Properties Browser. Or, If you can type XMS: to indicate the Xmedia Server, followed by the name(s) of the category or categories to which it belongs, e.g. XMS:Images.Backgrounds to indicate that the image is in the Backgrounds category under Images. |
| <p>Background Name [BgName]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: The name of the background asset. • Accepted values: Drag the background asset (e.g. image) from the Asset Browser or Gallery Browser and drop it in this property's text box. Or, If you would prefer, you can type the name and then the location in the ImageLocation property, or you can type both the name and location all in this row, e.g. XMS:Images.Backgrounds:Blue. |
| <p>Default Device [DefaultDevice]</p> | <ul style="list-style-type: none"> • Applies to: Scene canvas • Description: Indicates the intended device type that the scene was authored for. • Accepted values: Select from the drop-down list: XG, Intuition, ImageStore, or other. |
| <p>Duration [Duration]</p> | <ul style="list-style-type: none"> • Applies to: Template canvas • Description: The amount of time that the template is intended to stay on air. • Accepted values: |
| <p>Font [Font]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: Sets the font for the text characters. • Accepted values: Select the property's text box and then click the Browse button to display the Font dialog box. Select a font name, style, and size (e.g. Arial - 18) and click OK. |
| <p>Guides Active [GuidesActive]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: Determines whether or not the guides will have "snap-to" functionality. Guides are lines that can be added to the canvas to facilitate object alignment. See Positioning objects using guides, on page 81. • Accepted values: Select from the drop-down list: True or False. |
| <p>Guides Color [GuidesColor]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: Determines the color of the guides. Guides are lines that can be added to the canvas to facilitate object alignment. See Positioning objects using guides, on page 81. • Accepted values: Click in the property's text box and then click the Browse button to display the Color dialog box. Select an RGB color definition (e.g. R: 0 G: 0 B: 0) and click OK. |

| | |
|---|--|
| <p>Guides Locked [GuidesLock]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: Determines whether the guides will be locked in place, or available for repositioning. Guides are lines that can be added to the canvas to facilitate object alignment. See Positioning objects using guides, on page 81. • Accepted values: Select from the drop-down list: True or False. |
| <p>Guides Visible [GuidesVisible]</p> | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: Determines whether or not the guides will be visible. Guides are lines that can be added to the canvas to facilitate object alignment. See Positioning objects using guides, on page 81. • Accepted values: Select from the drop-down list: True or False. |
| <p>Home Area Color [HomeAreaColor]</p> | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Determines the color of the home area. The home area is a visual device intended to help position objects on the canvas. • Accepted values: Click in the property's text box and then click the  button to display the Color dialog box. Select an RGB color definition (e.g. R: 0 G: 0 B: 0) and then click OK. |
| <p>Home Area Height [HomeAreaHeight]</p> | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Determines the height of the home area. The home area is a visual device intended to help position objects on the canvas. • Accepted values: Enter the number of pixels. Or, If the <Active>Home Area Locked property is set to False, you can drag the top or bottom edges/corners of the home area to your desired height and the Home Area Height property is automatically set to the new height value. |
| <p>Home Area Locked [HomeAreaLock]</p> | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Determines whether or not the home area can be resized by dragging from its edges and corners. The home area is a visual device intended to help position objects on the canvas. • Accepted values: Select from the drop-down list: True or False. |
| <p>Home Area Visible [HomeAreaVisible]</p> | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Determines whether or not the home area is visible (i.e. boxed), or blended with the canvas. The home area is a visual device intended to help position objects on the canvas. • Accepted values: Select from the drop-down list: True or False. |
| <p>Home Area Width [HomeAreaWidth]</p> | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Determines the width of the home area. The home area is a visual device intended to help position objects on the canvas. • Accepted values: Enter the number of pixels. Or, If the <Active>Home Area Locked property is set to False, you can drag the right or left edges/corners of the home area to your desired width and the Home Area Width property is automatically set to the new width value. |
| <p>Home Key Frame [HomeKeyFrame]</p> | <ul style="list-style-type: none"> • Applies to: Scene canvas • Description: A Scene "transitions in" by executing its Default In animation when it is taken to air. The home key frame specifies a frame within the Default In animation that defines the "default appearance" of the scene. The Scene is saved in the state obtained by playing the Default In animation up to the home key frame. This determines the appearance of the Scene's thumbnail. See Default, Page and Manual animations, on page 118 for more information about Default In animations. • Accepted values: For most scenes, the home key frame is set to the end of the Default In animation when all elements have finished "transitioning in". In a timeline, the Default In may "transition in" the scene at the beginning of the animation and "transition out" at the end of the animation. In this case the home key frame may be chosen in the middle of the animation. Also, -1 means the end of the Default In animation. |

| | |
|--|--|
| Load Delay [LoadDelay] | <ul style="list-style-type: none"> • Applies to: Scene canvas • Description: Sets the number of seconds delay that will elapse before the scene is loaded on air. • Accepted values: Enter the number of seconds of the desired delay that will elapse before the scene is loaded on air. |
| Name [Name] | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: The name is used to identify the canvas itself. The default name given to canvas is the asset type followed by an incrementing number (i.e. Template1 or Panel1). • Accepted values: To edit or rename the canvas or object's name, simply type the new name (text) in this properties text box. |
| Persist [Persist] | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Determines whether a scene will persist on the device. This property allows for continuity as one scene transitions to another by maintaining the playout of objects that are common to both scenes (i.e. background or looping clips). • Accepted values: Select from the drop-down list: True or False. |
| Recall ID [RecallID] | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: This property allows you to set a unique identifier that used in other Vertigo applications (i.e. Xbuilder, Xplay) to quickly call up the specific Panel or Template. • Accepted values: Enter text and/or numbers to make up the identifier. |
| Recall ID Range Start | <ul style="list-style-type: none"> • Applies to: Template canvas • Description: Allows you to set the lower limit of the range of Recall IDs that can be used for pages based on this template. • Accepted values: |
| Recall ID Range End | <ul style="list-style-type: none"> • Applies to: Template canvas • Description: Allows you to set the upper limit of the range of Recall IDs that can be used for pages based on this template. • Accepted values: |
| SceneFormat [TVFormat] | <ul style="list-style-type: none"> • Applies to: Scene canvas • Description: Sets the television format for the scene. • Accepted values: Select one of the following from the drop-down list: <ul style="list-style-type: none"> • NTSC • PAL • HDTV-720/60 • HDTV-720/59.94 • HDTV-720/50 • HDTV-1080/30 • HDTV-1080/29.97 • HDTV-1080/25 • ITV • NTSC-16/9 • PAL-16/9 |
| Target Device [TargetDeviceType] | <ul style="list-style-type: none"> • Applies to: Template canvas • Description: The name of the template's target device, which is the name of a "virtual device" as defined in the Xplay configuration. For instance, you might configure Xplay to use devices "A", "B" and "C". You can create a template with "B" as its target device, meaning that you intend to take it to air on device "B" in Xplay. When the template is dragged into a playlist, its Device column is automatically filled in with "B". • Accepted values: Type the name of the intended device as configured in Xplay. |

| | |
|---|---|
| Template Type | <ul style="list-style-type: none"> • Applies to: Panel or Template canvas • Description: Sets the template type for the template. • Accepted values: Select one of the following template types from the drop-down list: Still, Crawl, or Clip. |
| User Properties | <ul style="list-style-type: none"> • Applies to: Panel, Template or Scene canvas • Description: User-defined properties. For example, can be used as a data container to be accessed by a script. Click in the grey cell below the User Properties category in the Properties Browser to add another User Property. • Accepted values: Type the name of the user-defined property and type a user defined value/setting. |
| Video Reset [VideoReset] | <ul style="list-style-type: none"> • Applies to: Scene canvas • Description: Determines whether or not the video will be reset when the scene is loaded. • Accepted values: Select from the drop-down list: True or False. |


Opening existing projects

If you do not want to create a new project, but rather work from existing (saved) project, you can access and open the project using any of the following methods:

- Select **File>Open** or **Ctrl+O**. The **MediaStore Browser** dialog box appears. Navigate through the asset category tree and double-click the name/proxy of the desired project.
Or,
- Open Xstudio's **Asset Browser**. Navigate through the asset category tree and double-click the name/proxy of the desired project/asset.
Or,
- Open Xstudio's **Gallery Browser**. If the project's category was added to the Gallery Browser, select the category and then double-click the name/proxy of the desired project/asset.
Or,
- Open the **Create New** dialog box (**File>New**) and select the **Recent** tab, which lists the ten (10) most recently opened projects/assets. Double-click the name of the desired project/asset.




Note: If you cannot located the desired project using the above methods, please consider searching for it using Xstudio's Search XMS feature. See [page 54](#) for more information.

Searching for projects/assets on the Xmedia Server

Clicking the **Search** button  on the Xstudio desktop or selecting the **View > Search Assets** menu opens the **Search XMS** window, which allows you to quickly search for assets stored on the Xmedia Server. Once a search is launched, assets that match the name, type, description, author, keywords and/or Recall ID are displayed as thumbnails in **Proxy View** or listed in a table in **Detail View**.

Note: The **Search** window uses **Solr**, which is a web-based solution by the Apache group that performs fast searches across large amounts of data. The Xmedia Server must maintain a connection to Solr so that it can create an index of the database. See the **Xmedia Server Configuration Guide** for more information.

To search for assets stored on the Xmedia Server:

- 1 Click the **Search** button  to open **Search XMS** window.
- 2 Type a search query in the **Search XMS** text box.
See [Search query syntax supported by the Search XMS](#) below for descriptions and examples of the syntax you can use to build more effective search queries.
- 3 Optional: Limit Search XMS results to specific asset types.
By default, all asset types are evaluated by the search query as indicated by the **Type** drop-down box set to **All Types**.
 - To limit the search results to one asset type, select an asset type from the **Type** drop-down list.
 - To limit the search results to multiple asset types, click the **Browse**  button to open the **Select Types To Search** dialog. Select the asset types that you want included in the search and uncheck the asset types that you want excluded from the search results.
- 4 Click the  button to launch the search.
Assets that satisfy the search query are displayed in the results pane of the Search XMS window.

Note: If an expected asset is not returned in the search results, it may be because user rights management is enabled on the Xmedia Server and the current user does not have permission to view the asset. See "User Rights Management" in the Xmedia Server Configuration Guide for more information.

- 5 Browse the results and select an asset of interest.
 - Right-clicking on an asset exposes menu commands that allow you to rename, delete or display the asset's properties.Depending on the asset type, here are some of the more common actions performed using the search results:
 - Add or replace an image, clip, cel animation or Xmedia Object by dragging an asset of the equivalent type from the search results onto a template, page, scene or panel open in Xstudio.

- Open a specific Template, Page, Scene or Panel asset in Xstudio by double-clicking the asset in the search results.
- Add or replace an asset associated with an action by dragging an asset from the search results onto the appropriate parameter in Xstudio's Action Editor.
- Open and edit a specific script or lookup table in Xstudio.

Search query syntax supported by the Search XMS

The search query syntax supported by the **Search XMS** text box is very similar to that supported by typical search engines. The following table provides descriptions and examples of syntax you can use to build more effective search queries.

| | |
|--|--|
| Single term and wildcard characters | <p>When a single term is specified in the search text box, any asset whose name, type, description, author, keywords and/or Recall ID matches the term is returned in the search results.</p> <p>Wildcards are supported when using single terms, but not within multi-term phrases.</p> <ul style="list-style-type: none"> • To perform a single character wildcard search use the "?" symbol. For example, ca?e matches cake as well as cape. • To perform a multiple character wildcard search use the "*" symbol. For example ap*, matches ape, apple, application, and apathetic. |
| Multiple terms | <p>When multiple terms are specified in the search text box, any asset whose name, type, description, author, keywords and/or Recall ID matches one or more of the terms is returned in the search result.</p> |
| Multi-term phrase | <p>To search for a multi-term phrase, enclose the phrase in double quotes. Only results with an exact match to the terms and the order of the terms are returned. For example: "family vacation"</p> |

| | |
|---------------------------------|--|
| <p>Boolean operators</p> | <p>Boolean operators allow terms to be combined in various ways. The following Boolean operators are supported:</p> <p>Note that the Boolean operations must be in ALL CAPS.</p> <ul style="list-style-type: none"> • AND: The AND operator matches assets that contain both terms. Examples: House AND Fire or Fire + House • +: The required operator is placed immediately before a term to indicate that the term must exist in the result. Example: +NASA weather Search results must contain <i>NASA</i> and may contain <i>weather</i>. If Assets that contain both terms receive a higher relevancy score than the assets with <i>NASA</i> only. • OR: This is the default operator between terms if no operator is specified. The OR operator matches assets that contain either of the specified terms. Assets having both terms have a higher relevancy score. Example: House OR Residence • NOT or -: The NOT '-' (hyphen character) operators exclude assets that contain the term after the operator. Example: House NOT Home or House -Home |
| <p>Asset fields</p> | <p>By default, basic search queries use the text entered to try to match against all of the following asset fields: name, asset type, description, author, keywords and Recall ID fields. It is possible to target specific asset fields in a search.</p> <p>Note that the searchable field names are all case sensitive and must include the colon : character.</p> <p>The following field names can be used to restrict search results:</p> <ul style="list-style-type: none"> • name: Returns only matches to the terms specified in the asset's name field. Example: name:"black box" • description: Returns only matches to the terms specified in the asset's description field. Example: description:weather • asset_type: Returns only matches to the specified asset type. Example: asset_type:clip • author: Returns only matches to the asset's author field. Example: author:jsmith <p>recall_id: Returns only valid Recall IDs. An asset is considered to have a valid Recall ID if the recall is not blank and not equal to 0. Example: recall_id:[" TO *] AND -recall_id:0</p> |

Importing assets from another Vertigo system

If you would like to use Vertigo assets that were created on a separate Vertigo Suite installation (e.g. stored on a different Xmedia Server), the assets must first be exported from the original system (see [page 255](#)), which creates a Vertigo Export file (.exp).



You can then use Xstudio (or Xplorer) to import the .exp file into your Vertigo system. Unlike ingesting files, which loads and converts files from third-party formats into Vertigo assets (see [page 59](#)), imported assets are already in the appropriate Vertigo formats and therefore require no conversion. The assets are also automatically assigned to their rightful place on the Xmedia Server according to their asset type and previous categorization.


There are two things to worth noting regarding importing asset files:

- **Links to data will not be maintained.** The same data sources need to be set up on your server and the links re-established (see [page 177](#)).
- **Existing assets may be overwritten.** This only occurs under a very specific set of circumstances. For example, if you import a file in a category that has the same name as a file in an identically named category, the imported file will overwrite the other one. Also, if you import a file from the root category and an identically named file is in the root category of the installation you are importing to, the imported file will overwrite the other one. You can prevent files from being overwritten by assigning them to different categories, since an imported file will only overwrite another file if both have the same name and belong to the root category or to an identically named category.

Note: If the assets that you are importing contains data schemas, then your data schema connect settings will likely need to be maintained. Do this before using any templates, scenes, or pages.

To import assets:

- 1 In Xstudio, select **File>Import Assets**.
This displays the select **VertigoXmedia Asset Export File** dialog box.
- 2 Navigate to the export folder that contains the asset(s) that you want to import (i.e. VxM_Export_21082008_125200).
- 3 Double-click the folder to display its contents.
- 4 Select the **.exp file** and then click the **Open** button.
- 5 The Import window appears and identifies all of the assets that will be imported from the export file to the target XMS. An icon in front of the asset's name indicates whether the asset is being added  or is being updated .

The assets are displayed in a tree-structure that represents the assets' destination among the target server's categories. Newly created categories are indicated the  icon.

The Import window can also contain a list of error or warning messages to alert users of problems that may occur if the import is performed. For example, some categories may have to be renamed and some assets will not be able to be categorized because of name conflicts. Clicking on an asset that shows either an error or warning icon highlights the associated messages. Similarly, clicking on a message will select the corresponding asset or category in the tree.

- 6 After inspecting the import contents, select the **Import** button to proceed with the import or select **Cancel** to abort the import.

The status bar at the top of the Xstudio window displays the progress as the file is imported into your system.

The Import window remains open after **Import** is clicked so that it can be referenced when looking for newly imported assets. Select the **Close** button to close the **Import** window.

- 7 Optional: Verify that the asset was properly imported by opening the Asset Browser and navigating to the category where you expected the asset to be stored. Open the asset by double-clicking its name in the Asset Browser.

Note: Transferring assets by exporting assets from one Xmedia Server (XMS) and then importing them into another XMS manually is ideal for low volume and infrequent transfers. However, if you are frequently transferring assets between servers, we recommend using propagation to transfer assets. See "Propagating assets from the XMS to another server" in the *Xmedia Server Configuration Guide* for more information.

Ingesting files created using third-party applications

Individual content files (i.e. images, cel animations, video clips, and audio clips) that make up broadcast graphics are normally produced by third-party applications outside of the Vertigo Suite. Therefore, you must ingest them into the Xmedia Server before they can be used by Xstudio as assets. **Ingesting** a file basically means converting the file into a Vertigo asset and saving it on the Xmedia Server according to its asset type.

While the procedure for ingesting image, video, and audio files is for the most part the same, ingesting cel animations requires a different procedure. The following sections provide information and instructions for ingesting files into the system.


- [Ingesting image or audio files](#), on page 59
- [Ingesting multiple image files or creating cel animations or clips upon ingest](#), on page 61
- [Ingesting video files](#), on page 66

Ingesting image or audio files

Images and audio clips that were created using third-party applications must be ingested into the Xmedia Server before they can be used in Xstudio.

Note: If you are ingesting multiple image files that have a sequential naming convention and/or ingesting image files for purposes of creating a cel animation or video clip, please refer to [Ingesting multiple image files or creating cel animations or clips upon ingest](#), on page 61.

To ingest an image or audio file:

- 1 Click the **Ingest** button  on Xstudio's desktop.
The **Ingest Files** dialog box appears.
- 2 Navigate through the categories in the **Ingest Files** dialog box and select the file(s) that are to be ingested into the Xmedia Server.
You can ingest multiple files by pressing the **SHIFT** key as you click on each file for consecutive selections, or pressing the **CTRL** key to select a grouping of non-consecutive files.

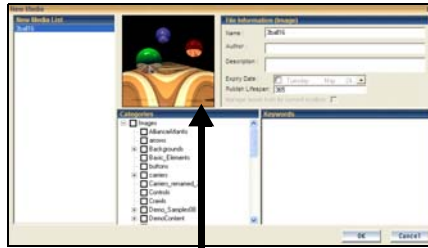
Note: If the first three letters in the name of the selected image file are identical to other files in the same directory (i.e. Bug001, Bug002, Bug003...), the **Ingesting a Cel Animation** message box appears and asks if your intention is to ingest the single image file or to create cel animation.

- Select **No** to ingest a single image file using the **New Media** dialog box.
 - Select **Yes** to create a cel animation from the set of identically named image files using the **Multi-image Ingest** window. See [page 61](#) for instructions on how to proceed in such a case.
-

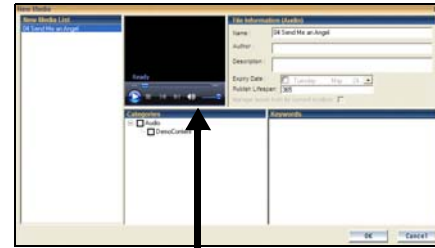
- 3 Click **Open**.

The **New Media** window appears, which allows you to specify and save details about the ingested asset file.

- 4 Selecting a file's name in the **New Media List** displays the file's information and all of the categories within that root category. When an image file is being ingested, a proxy of the image is displayed in the center pane of the Ingest window. If an audio file is being ingested, then the Ingest window features the Windows Media Player allowing you to preview the playout of the audio clip before ingesting it.



Displaying the proxy of the image file



Media Player for previewing the audio file

- 5 Select the first file name in the **New Media List** panel and complete the following tasks. Repeat this step for each file listed in the **New Media List** panel.
- In the **Name** field, assign a name to the asset to be called. By default, the file's name is used.
Note: If you are ingesting an edited version of a file that already exists on the selected device, saving it under the same name in the same category overwrites the previous version of the asset. To avoid overwriting the asset, rename it or save it to another category.
 - Enter your name in the **Author** field.
 - Optional - Enter a short description in the **Description** field.
 - Optional - Enable and set a **Expiry Date** and **Publish Lifespan** fields. The **Expiry Date** refers to date when the asset will be automatically deleted from the main XMS, spoke XMSs (hub & spoke model), as well as removing the asset from the devices that it was published to. The expiry date value is set in days and the purge is executed at 3:00 AM. The **Publish Lifespan** used by the XMS to determine when the asset should be removed from the playout device. The Publish Lifespan value is set in days, it has a one (1) day grace period, and the purge is executed at 2:00 AM.
 - The **Manage asset from its current location** check box will only be enabled if the asset that is to be ingested is currently stored on an external device and if the permission permit access. If this criteria is met, the parameter is available and you can save space on your local/central XMS by managing the asset from its current location. In most cases, this parameter should not be enabled.
 - Assign the asset to one or more categories by selecting check boxes in the **Categories** box. To create a new category, right-click in the **Categories** box and select **Add Category** from the pop-up menu.
 - Optional - Assign the file one or more keywords by selecting check boxes in the **Keywords** box. (Keywords are used by Xstudio's Search XMS feature for searching and retrieving assets.) To create a new keyword, right-click in the **Keywords** box and select **Add Keyword** from the pop-up menu.
- 6 Click **OK** to save and exit the dialog box.

Or,

Click **Cancel** to discard the information entered in the dialog box, exit the dialog box and return to Xstudio.

Ingesting multiple image files or creating cel animations or clips upon ingest

As explained on [page 59](#), you can use Xstudio to ingest a single image file. Additionally, you can use Xstudio's Multi-image Ingest dialog box to ingest multiple image files or create a cel animation or clip from a series of sequentially named images files.

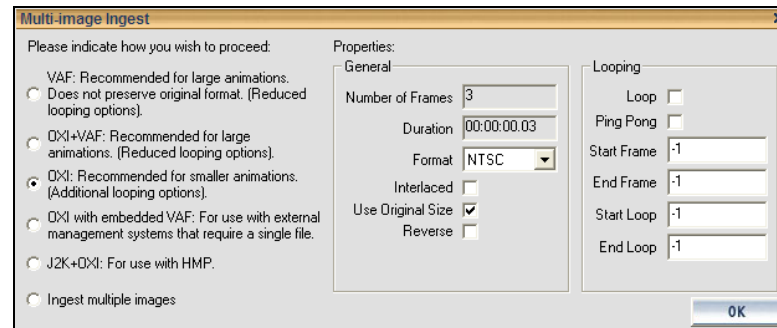



Fig. 4-1: The Multi-image Ingest dialog box

To ingest multiple images or convert the image files into a cel animation or a clip animation:

- 1 Click the **Ingest** button  on Xstudio's desktop. The **Ingest Files** dialog box appears.
- 2 Navigate through the categories in the **Ingest Files** dialog box and select the image file(s) that are to be ingested. Then click **Open**.

Note: You select multiple files by pressing the **SHIFT** key as you click on each file for consecutive selections, or pressing the **CTRL** key to select a grouping of non-consecutive files.

If you selected a single image file and the first three letters in the name of the file are identical to other files in the same directory (i.e. Bug001, Bug002, Bug003...), the **Ingesting a Cel Animation** message box appears and asks if your intention is to ingest the single image file or to create cel animation.

- Select **No** to ingest the single image file using the **New Media** dialog box. See [page 59](#) for instructions on how to proceed in such a case.
- Select **Yes** to create a cel animation or clip from the set of identically named image files using the **Multi-image Ingest** dialog box.

If however, you selected multiple image files, the **Multi-image Ingest** dialog box automatically appears and offers you several options to either ingest the individual image files, or to create a cel animation or clip asset from these image files.

- 3 When the **Multi-image Ingest** dialog box appears, it offers you six (6) options for how you can proceed to ingest the set of image files. Each option is described in the table below.

Note that the Xstudio takes into consideration what is trying to be ingested and makes a preliminary suggestion as to which option to use by pre-selecting an option for you. You may use the pre-selected option or select another.

| | |
|-------------------------------|---|
| VAF | The VAF option ingests the selected image files and converts their sequential playout into a video clip asset. The resulting video clip asset is saved using the Vertigo Animation Format (.vaf) and it is stored in the Xmedia Server's Clips category. This option is recommended for large animations. Note that this option does not preserve the original format and it offers a reduction in looping options. |
| OXI +VAF | Select the OXI +VAF option when ingesting large animations. In such a case, the OXI is the master, while the VAF as the alternate format. When publishing the OXI + VAF , the Xmedia Server determines whether to publish the OXI or the VAF based on the target device type. Note that this option results in a reduction of looping options, but there is no size limit for the VAF file. |
| OXI | The OXI option ingests the multiple images and creates a cel animation by converting the images into oxi format, which is a series of targa format files (.tga) that are lossless, uncompressed, and cropped. The resulting .oxi file will be saved as a cel animation asset in the Cel Animation category. The OXI option is recommended when ingesting smaller animations. Unlike the OXI +VAF option, the OXI option provides additional looping options. The recommended file sizes for this option are < 250 MB when the target device is a Vertigo XG and < 4 GB when the target device is an Intuition. |
| OXI with embedded VAF | The OXI with embedded VAF option is intended to be used with external management systems that require a single file. Embedding the VAF in the OXI allows the same file to be used on both the Intuition (Classic), as well as the Vertigo XG. Note that this option provides reduced looping options and that file sizes should be < 3.6 GB. |
| J2K+OXI | The J2K+OXI option is used with HMP devices. In such a case, the J2K is the master, while the OXI as the alternate format. |
| Ingest multiple images | The Ingest multiple images option ingests several images and saves them as individual image files in the Xmedia Server's Image category. |

- 4 If creating a cel animation, select from the **Format** drop-down list the target video format of the cel animation. This information indicates the desired full screen size and is used when the cel animation is cropped. Valid settings are: PAL, NTSC, 720p60, 720p60M, 720p50, 1080i30, 1080i30M, and 1080i25.
- 5 Enable or disable the **Interlaced** property.
Enable this setting if the images that make up the sequence being ingested contain fields, rather than frames. Disable this setting if the images that make up the sequence contain frames.

6 Enable or disable the **Use Original Size** property.

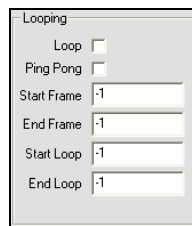
Enable this setting if the width and height of the cel animation will be equal to the width and height of the images from which it was created. When this setting is disabled, the cel animation will be cropped to the smallest size possible.

7 Enable or disable the **Reverse** property.

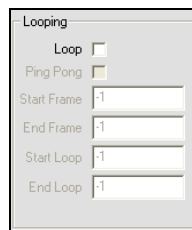
The reverse parameter determines whether the files that make up the sequence should be played in reverse order: **True** or **False** (default).

8 If creating a cel animation or a clip, set the **Looping** properties to ensure that the Start Frame/Loop and End Frame/Loop values fall on the full frame.

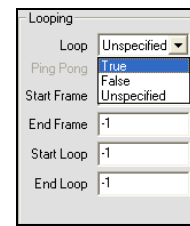
The availability and format of the **Looping** properties depends upon which ingest option is selected.



OXI



J2K + OXI



VAF, OXI + VAF, OXI with embedded VAF

- **Loop** - While the check box can be enabled or disabled, the **Loop** drop-down list has three (3) possible settings:
 - **True**: Enables or engages infinite looping. When this setting is used, the only way to stop looping is to send an action to stop it.
 - **False**: Disables looping.
 - **Unspecified**: Takes the looping setting that is already established by the scene object.

The **Loop** setting is used in conjunction with Start Loop and End Loop settings. If Loop is enabled and both Start Loop and End Loop are set to -1, then the entire cel animation loops continuously. If Loop is enabled and either Start Loop or End Loop is not -1, then the cel animation does a four point loop (in-loop-out). If Loop is disabled, the cel animation plays once from beginning to end. However, if loop is disabled and Start Loop and End Loop are both set to the same value (other than -1), then the cel animation displays the frame specified by Start Loop and End Loop and the cel animation can be controlled by automation. Note that if Ping Pong is enabled, and both Start Loop and End Loop are set to -1, then Loop is ignored.

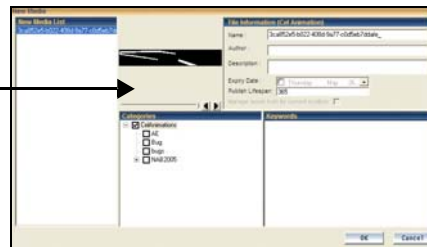
- **Ping Pong** - If enabled, and both Start Loop and End Loop settings are set to -1, then the cel animation plays in ping-pong mode. That is, it plays from beginning to end and then back to the beginning.
- **Start Frame** - Specify the frame at which the cel animation starts payout. A value of -1 means start at the beginning.
- **End Frame** - Specify the frame at which the cel animation ends payout. A value of -1 means play until the end.
- **Start Loop** - Specify the frame at which looping starts. A value of -1 means loop from the beginning of the cel animation.

- **End Loop** - Specify the frame where the cel animation loop ends. A value of -1 means loop until the end of the cel animation.

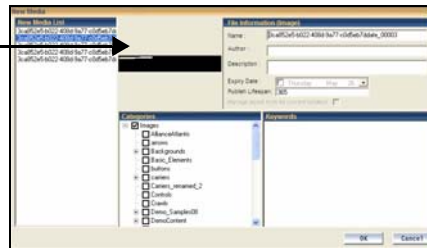
Note: All frames are specified starting from 0 for the first frame in the cel animation.

- 9 Click **OK** and the **New Media** window appears.
- 10 Optional - You can preview the cel animation or clip by dragging the slider or using the arrow buttons along the bottom of the preview display in the middle panel. When you are ingesting multiple images, you can preview an image file by selecting its file name in the **New Media List**.

Preview the cel animation or clip frame-by-frame



When ingesting multiple images, selecting an image's file name displays a preview of the image



- 11 Assign a name to the asset in the **Name** field. By default, the file's name is used. Note: If you are ingesting an edited version of a file that already exists on the selected device, saving it under the same name in the same category overwrites the previous version of the asset. To avoid overwriting the asset, rename it or save it to another category.
- 12 Enter your name in the **Author** field.
- 13 Optional - Enter a short description in the **Description** field.
- 14 Optional - Enable and set a **Expiry Date** and **Publish Lifespan** fields. The **Expiry Date** refers to date when the asset will be automatically deleted from the XMS as well as remove them from the devices that it has published them to. **Publish Lifespan** used by the XMS to determine when the asset should be removed from the playout device. The Publish Lifespan value is set in days, it has a one (1) day grace period, and the purge is executed at 2:00 AM.
- 15 The **Manage asset from its current location** check box will only be enabled if the asset that is to be ingested is currently stored on an external device and if the permission permit access. If this criteria is met, the parameter is available and you can save space on your local/central XMS by managing the asset from its current location. In most cases, this parameter should not be enabled.
- 16 Assign the asset to one or more categories by selecting check boxes in the **Categories** box. To create a new category, right-click in the **Categories** box and select **Add Category** from the pop-up menu.

- 17 Optional - Assign the file one or more keywords by selecting check boxes in the **Keywords** box. (Keywords are used by Xstudio's Search XMS feature for searching and retrieving assets.) To create a new keyword, right-click in the **Keywords** box and select **Add Keyword** from the pop-up menu.
- 18 Click the **OK** button to save and exit the dialog box.
Or,
Click the **Cancel** button to discard the information entered in the dialog box, exit the dialog box and return to Xstudio's desktop.


Ingesting video files

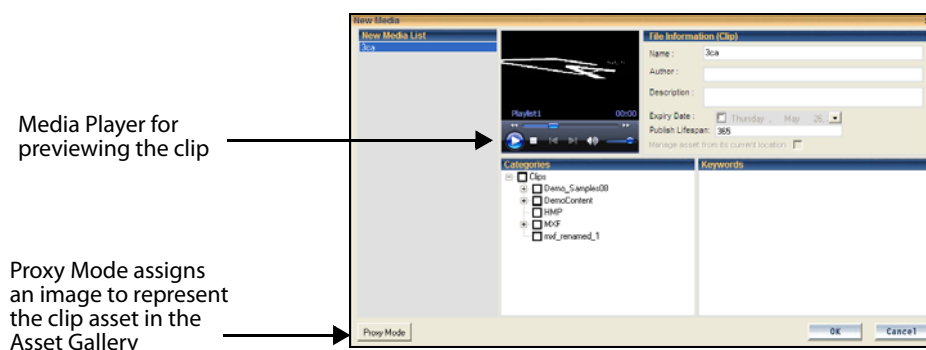
Video clips that are created using third-party applications must be ingested into the Xmedia Server before they can be used by the Xmedia Suite applications.

As the instructions below describe, when a video clip is ingested into the system, the New Media dialog box contains a Windows Media Player for previewing the video clip before ingesting it.

Also worth noting, the file that is to be ingested does not yet have a proxy image (thumbnail) associated to it. Although a proxy image is not required, it does make identifying the clip easier when it is displayed in Proxy View in the Asset Gallery. Xstudio allows you to assign a proxy image to the clip when you are ingesting the clip. Should you choose not to specify a proxy image for the clip, a generic proxy is assigned to represent the clip asset in Proxy View.

To ingest a video clip:

- 1 Click the **Ingest** button  on Xstudio's desktop.
The **Ingest Files** dialog box appears.
- 2 Navigate through the categories in the **Ingest Files** dialog box and select the file(s) that are to be ingested. Then click **Open**.
To select multiple files, press the **SHIFT** key as you click on each asset for consecutive selections, or press the **CTRL** key to select a grouping of non-consecutive files.
- 3 When the **New Media** dialog box appears, select the first file name in the **New Media List** panel and complete the following tasks. Repeat this step for each file listed in the **New Media List** panel.
 - a Optional - Preview the clip in the Windows Media Player in the middle panel to confirm that this is the correct video file to be ingested.
 - b Optional - Assign a proxy image to the clip by clicking the **Proxy Mode** button in the lower-left corner of the New Media dialog box. Click on the **Proxy Not Available** panel (where the Media Player was). The Assign Asset Proxy dialog box appears and lets you select the .jpg image that will be associated as the proxy image for the clip asset. Click the **Open** button. The image now appears in the middle panel of the New Media dialog box.



- c In the **Name** field, assign a name to the asset. By default, the file's name is used.
Note: If you are ingesting an edited version of a file that already exists on the selected device, saving it under the same name in the same category overwrites the previous version of the asset. To avoid overwriting the asset, rename it or save it to another category.

- d Enter your name in the **Author** field.
 - e Optional - Enter a short description in the **Description** field.
 - f Optional - Enable and set a **Expiry Date** and **Publish Lifespan** fields. The **Expiry Date** refers to date when the asset will be automatically deleted from the XMS as well as remove them from the devices that it has published them to. **Publish Lifespan** used by the XMS to determine when the asset should be removed from the playout device. The Publish Lifespan value is set in days, it has a one (1) day grace period, and the purge is executed at 2:00 AM.
 - g The **Manage asset from its current location** check box will only be enabled if the asset that is to be ingested is currently stored on an external device and if the permission permit access. If this criteria is met, the parameter is available and you can save space on your local/central XMS by managing the asset from its current location. In most cases, this parameter should not be enabled.
 - h Assign the asset to one or more categories by selecting check boxes in the **Categories** box. To create a new category, right-click in the **Categories** box and select **Add Category** from the pop-up menu.
 - i Optional - Assign the file one or more keywords by selecting check boxes in the **Keywords** box. (Keywords are used by Xstudio's Search XMS feature for searching and retrieving assets.) To create a new keyword, right-click in the **Keywords** box and select **Add Keyword** from the pop-up menu.
- 4 Click the **OK** button to save and exit the dialog box.
- Or,
- Click the **Cancel** button to discard the information entered in the dialog box, exit the dialog box and return to Xstudio's desktop.

5 Adding and preparing objects on the Xstudio canvas

In Design mode, Xstudio projects (Templates, Control Panel and Scenes) are created and/or edited on the canvas. As described on [page 42](#), the canvas is the work area onto which you drag-and-drop assets and/or primitives, and then arrange them (e.g. aligned, layered, grouped) to create functional and attractive graphics displays (scenes) and control panels.

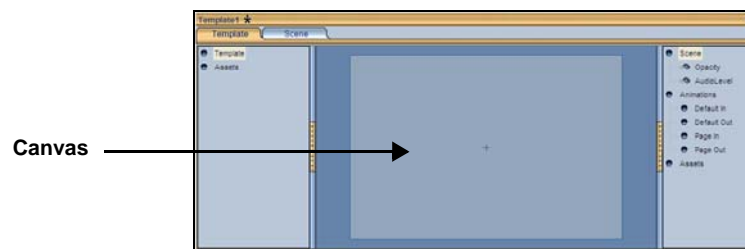


Fig. 5-1: The canvas is the work area onto which you drag-and-drop assets and/or primitives

Xstudio's Asset Browser and Gallery Browser provide you with immediate access to all of the assets (i.e. images, clips, cel animations...etc.) that are available on the system for creating compelling graphics displays. Alternatively, you can use Xstudio's primitives (e.g. text boxes, buttons, images, clip), which are empty objects that you can use to create or control graphics elements within Xstudio scenes and control panels.

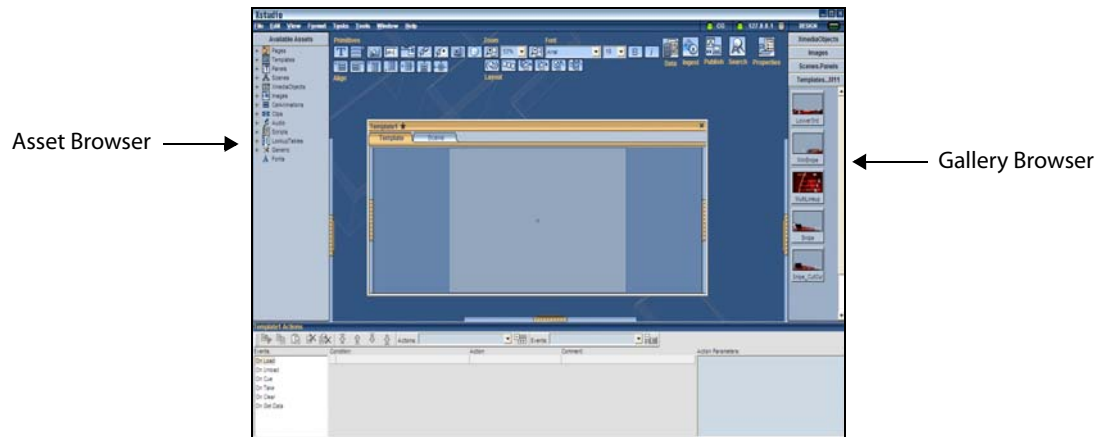
Since the scene represents the on-air look of the broadcast graphic, you will spend time carefully placing and arranging objects on the scene's canvas to achieve the desired look. To help you achieve the desired look, Xstudio's Preview feature allows you to see exactly what the scene will look like on air by previewing it on the output device as you are creating the scene. As you add, move and edit objects on the scene, the preview is updated on the output device in real-time, so you can see the results of any edits that you make to the scene right away.

The following topics provide information and instructions for adding, formatting and previewing assets and primitives on the Xstudio canvas:

- [Adding assets to the canvas](#), on page 70
- [Adding primitives to the canvas](#), on page 71
- [Removing assets or primitives from the canvas](#), on page 73
- [Previewing a scene's layout on a device during development](#), on page 74
- [Selecting, grouping, and/or layering objects on the canvas](#), on page 75
- [Resizing objects using object zones](#), on page 79
- [Moving and aligning objects on the canvas](#), on page 80
- [Cutting, copying and pasting objects to/from the canvas](#), on page 83
- [Modifying the font properties of text objects](#), on page 84
- [Editing an image](#), on page 85
- [Undoing and redoing edits on the canvas](#), on page 86

Adding assets to the canvas

Xstudio's Asset Browser and Gallery Browser provide you with immediate access to all of the assets (i.e. images, clips, cel animations...etc.) that are available on the system for creating compelling graphics displays (see [page 38](#) for more information about the Asset Browser and Gallery Browser). From these browsers, you can simply drag and drop a copy of an asset onto the canvas.



To add an asset to the canvas:

- 1 Open either the Asset Browser or the Gallery Browser.
- 2 Click, drag, and drop the desired asset object from the Asset Browser or Gallery Browser onto the canvas.

Note: If the asset is an Xmedia Object that has both control and scene side elements, the control object will appear on the control side's canvas, and/or the scene object will appear on the scene side's canvas.

You can now position the objects as desired by dragging the object around the canvas using your mouse (see [Moving and aligning objects on the canvas](#), on page 80). If you are working on an asset with both a control and scene side, changing object positioning on one tab will not affect corresponding items on the other.

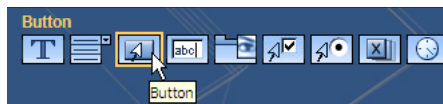
Adding primitives to the canvas

Xstudio's primitives are empty Xstudio objects that you can use to create or control graphics elements within Xstudio scenes and control panels. Some primitives are only available for the Scene, whereas others are exclusive to the Control side (see [page 36](#) for a description of each of the available primitives).



To add a primitive to the canvas:

- 1 Select the desired primitive from the **Primitives** toolbar.
The selected primitive's icon is highlighted in the toolbar.



- 2 Hover the mouse pointer over the canvas and the pointer changes. While holding down the left mouse button, drag the mouse to create the initial size of the primitive on the canvas.
- 3 When you release the button, the primitive appears on the canvas.

Assigning an asset to a primitive



Certain primitives exist to have assets assigned to them (i.e. image, cel animation and clip objects). In other words, once any one of these primitives is selected from the **Primitives** toolbar and added to the canvas, it will not be functional until an asset of a corresponding type is assigned to it.

Note that assigning an asset (e.g. an image file) to a button object is optional.



To assign an asset to a primitive:

- From either the **Asset Browser** or the **Gallery Browser**, drag and drop the asset directly onto the primitive object.
- Or,
- Select the primitive object, and then drag and drop the asset onto the appropriate row in the **Properties Browser** (CTRL + R). It will be one of the following:
 - **ImageName** - For image and button objects.
 - **ClipName** - For clip objects.
 - **CelName** - For cel animation objects.

To assign a file from your hard drive:

- 1 Select the primitive object (i.e. button) upon which you want to set the file.
- 2 Open the **Property Browser**, click in the **ImageName** row.
- 3 Click the  button, which now displays the **Select An Image** dialog box.
- 4 Navigate to the desired file on your hard drive by clicking the  button.

Note that you may be able to list or display the proxies of the contents of the folders by selecting the following buttons:

-  - View the contents of a selected folder or category in proxy format.
-  - View the contents of a selected folder or category in detailed file information format.

5 Once you have selected the image, click **OK**.

This displays the **New Media** dialog box.

This dialog box is used for ingesting the image into Xstudio, making it henceforth available within the application.

6 You can now modify the file name, add author information and a description for the image. You can also select one or more categories to link to the image; if no selections are made, Xstudio assigns the file to the **Images** root category. Selecting a keyword for the image is also optional. When done, click **OK**. This closes the dialog boxes and sets the image on your image object.

Removing assets or primitives from the canvas

If they are no longer wanted, you can remove assets or primitives from the scene or control panel's canvas.

Here are a couple of important things to note about removing assets and primitives from the canvas:

- Assets and primitives that are removed from the canvas are not deleted from the system if they have been previously saved. However, if they have not been saved and they are removed, then they will be deleted from the system.
- Xmedia objects often contain scene objects on the scene's canvas and control objects on the control side's canvas. When removing an Xmedia object, you have to delete the objects on each side separately by switching between the two tabs.

To remove an asset or primitive from the canvas:

- Select the object on the canvas and then select the **Edit>Delete** menu command.
Or,
- Select the object on the canvas and then press the **Delete** key on your keyboard.

Previewing a scene's layout on a device during development

As you are creating a scene in Design mode, Xstudio's Preview feature allows you to see exactly what the scene will look like on air by previewing it on the output device (i.e. Software CG) that is currently connected. This ensures the accuracy of your scene's layout.

In Preview mode, as you add, move and edit objects on the scene, the preview is updated on the output device in real-time, so you can see the results of any edits that you make to the scene right away. For example, you can add a Text2D primitive object to the scene, move, rotate, resize and even change some of the font properties. All of these edits will be previewed on the Software CG's output display.

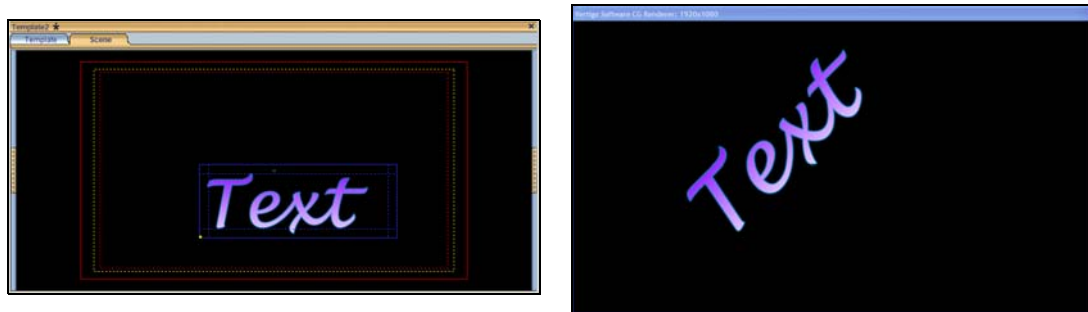




Fig. 5-2: Previewing the scene's layout on a Software CG device

Note: Xstudio's preview mode is a static display of the scene's layout, not its playout. It is intended to preview the look of the scene, not the scene's behavior. Therefore, dynamic assets like cel animations, clips and animated properties do not playout in preview mode. To preview playout, use Xstudio's Production mode.

The scene does not have to be saved or even published to the device to be previewed in Xstudio's Preview mode. However, assets (i.e. images, clips, cel animations...etc) from the Asset Browser must be published to the device in order to be viewed. If you add an asset to the scene that has not yet been published, the asset will appear on the scene, but not in the preview. See [page 253](#) for instructions on manually publishing an asset to a device.

To preview a scene's layout on a device during development:

- 1 Verify that Xstudio is actively connected to a playout device.
- 2 With the project window opened to the scene, click the **Publish** icon  on the desktop while holding down the **Ctrl** key. The **Preview** icon  replaces the Publish icon.
- 3 Add primitives or assets to the scene canvas, or move or edit objects that are already on the canvas.

The changes to the scene will appear on the output device's display immediately. Remember to save the project to retain any changes or edits.

Selecting, grouping, and/or layering objects on the canvas

While some graphic displays will contain only one object, others will contain multiple objects. To edit and position the objects properly on the canvas, you will have to select the individual objects. For edits or positioning that affects several objects, it may be quicker to group the objects before performing the edits.

The following sections provide instructions for selecting and/or grouping objects on the canvas:

- [Selecting a single object on the canvas](#), on page 75
- [Selecting multiple objects on the canvas](#), on page 76
- [Grouping and ungrouping objects on the canvas](#), on page 77
- [Layering objects on the canvas](#), on page 78

Selecting a single object on the canvas

The most common way to select an object that has been added to the canvas is by simply clicking once directly on the object. Once you have clicked the object, a blue grid (object zones) appears on top of the object indicating that it has been selected and that it is now the primary object on the canvas (see [Resizing objects using object zones](#), on page 79 for more information on object zones).



Fig. 5-3: An object zone grid on top of the object indicates that the object has been selected

An alternative technique for selecting individual objects is to hold down the left mouse button and drag the pointer completely around the object and then release. If the entire object was inside the dragged box, then the blue object zone grid appears on the object indicating that the object has been selected.

Note: It is possible to select multiple objects on the canvas at the same time. See [Selecting multiple objects on the canvas](#), on page 76 for more information.

Selecting multiple objects on the canvas

There are editing situations when it is faster, more effective, and even necessary to select multiple objects on the canvas. For example when creating an Xmedia object it may contain several objects, like buttons or text box primitives. If you wanted to move the Xmedia object to another location on the canvas, you would want to select them all, possibly group them, and then move them as one object, rather than as individual objects.

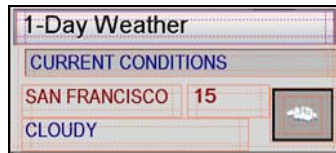


Fig. 5-4: Red object zone grids appear when multiple objects are selected on the canvas

To select multiple objects on the canvas:

- Hold down the **Ctrl** or **Shift** key on your keyboard, then click directly on each of the objects.

When the first object is selected, a blue grid (object zones) appears on top of the object indicating that it has been selected and that it is now the primary object on the canvas. However, when you select the next object, the first object's grid changes to red, indicating that it is still selected but no longer the primary object. Meanwhile the second object that was selected now appears with a blue object zone grid indicating that it was the last, and now primary, object selected. See [Resizing objects using object zones](#), on page 79 for more information about object zones.

An alternative technique for selecting multiple objects is to hold down the left mouse button and drag the pointer completely around all of the objects to be selected and then release. If all of the objects were inside the dragged box, the blue and red object zone grids appear on the objects indicating that these objects are selected.

Grouping and ungrouping objects on the canvas

There are editing situations when it is faster, more effective, and even necessary to select and group multiple objects on the canvas. Grouping objects basically means taking several separate objects that are selected together and makes them into a single object. Grouping objects is very useful when you want to keep several elements together, or when you want to apply the same operation to them all. At a later time, you can ungroup the object and return the object to being individual objects.

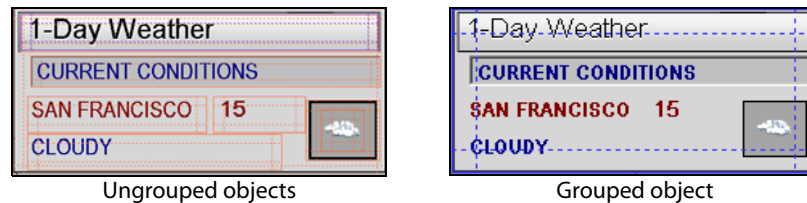




Fig. 5-5: Objects can be grouped so that they can be edited as one object

To group objects:

- 1 Select the objects to be grouped (see [Selecting multiple objects on the canvas](#), on page 76).
- 2 Select the **Format>Group** menu command, or click the **Group**  button on the Layout toolbar.

A blue border surrounds the grouped objects to indicate that the group is selected.

To ungroup objects:

- 1 Select the group (see [Selecting a single object on the canvas](#), on page 75).
A blue grid appears over the object.
- 2 Select the **Format>Ungroup** menu command, or click the **Ungroup**  button on the Layout toolbar.


The blue border disappears and each object is bordered in red.

Note: When objects are grouped, the group has a set of properties that can be edited to determine the group's positioning or enable the group as a mask. See [Group object properties](#), on page 311 for a listing and description of the group properties.


Layering objects on the canvas

Xstudio allows you to build graphics displays in which objects are layered one on top of another. The layout commands allow you to move objects to the top layer for display or editing, as well as sending the object behind others, like in the case of a background image. A practical example of layering objects is displaying a lower-third headline over a video.


To bring an object one step closer to the front:

- 1 Select the object (see [Selecting a single object on the canvas](#), on page 75).
A blue grid appears over the object.
- 2 Click the **Bring Forward** button  on the Layout toolbar, or select the **Format>Bring Forward** menu command.
The object moves one layer closer to the front.


To send an object one step toward the back:

- 1 Select the object (see [Selecting a single object on the canvas](#), on page 75).
A blue grid appears over the object.
- 2 Click the **Send Backward** button  on the Layout toolbar, or select the **Format>Send Backward** menu command.
The object moves one layer to the back.

To bring an object to the front:

- 1 Select the object (see [Selecting a single object on the canvas](#), on page 75).
A blue grid appears over the object.
- 2 Click the **Bring to Front** button  on the Layout toolbar, or select the **Format>Bring to Front** menu command.
The object moves to the first (top) layer.

To send an object to the back:

- 1 Select the object (see [Selecting a single object on the canvas](#), on page 75).
A blue grid appears over the object.
- 2 Click the **Send to Back** button  on the Layout toolbar, or select the **Format>Send to Back** menu command.
The object moves back to the furthest layer.

Resizing objects using object zones

When you select an object that has been placed on the canvas, a grid appears on top of the object, dividing it into multiple zones. The blue grid indicates that this is the primary object, which is the last or only object selected. When multiple objects are selected at the same time, the primary object displays a blue grid, while the others will have a red grid. Identifying the primary object is extremely helpful in formatting situations like aligning objects. If you were aligning objects all other selected objects (overlaid with a red grid to indicate that they are not the primary object) would align with the primary object.

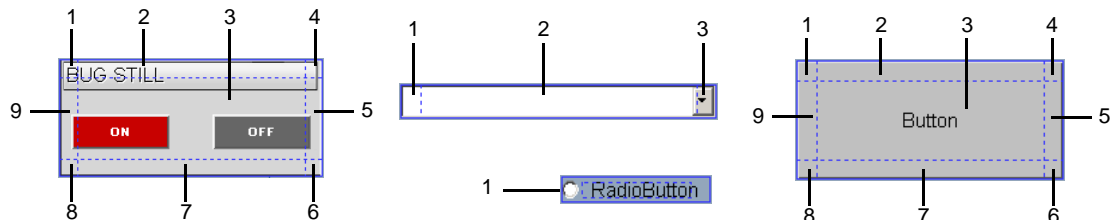


Fig. 5-6: An object's zones, indicated by a blue/red grid, can be used to resize or move the object

Based on their dimensions and purpose, the various types of primitives have a different number of object zones. For example, the Button primitive above is divided into nine (9) zones, while the drop-down box primitive is only divided into three (3) zones, and a radio button primitive only has one (1) zone.

When you hover the pointer over the object zones, the pointer transforms into a two-headed directional arrow. Clicking and dragging the object zones of objects, and some types of primitives, allows you to resize (stretch and shrink) the object in a free-form fashion in the direction that it is being dragged. To maintain the scaling of the object while resizing, simply hold down the **Shift** button on your keyboard, while dragging the outer zones of the object. As well, dragging from the central zone allows you to move the object around the canvas.

Note: Pressing **ESC** before releasing the mouse button aborts the move, copy or resize operation.

Moving and aligning objects on the canvas

The placement of objects within the scene's safe areas represents exactly how the object will appear on air (see [page 43](#)). While the placement of objects on the control-side has no affect on the placement of objects on the scene or on air, it does represent exactly how the objects will appear in production mode. Having said that, once you have dragged objects onto the canvas you will want to create an attractive and functional layout by performing the following tasks:

- [Moving objects on the canvas](#), on page 80
- [Positioning objects using the Align commands](#), on page 80
- [Positioning objects using guides](#), on page 81

Moving objects on the canvas

There are several ways to move and position objects around the canvas. Which method you choose to use depends on the precision of placement that you desire.

The most common and convenient way of moving objects on the canvas is to click the center zone of the object and drag the object to the new location. This method is convenient, but it lacks the precision that is sometimes necessary. Meanwhile, an ortho move provides the convenience of dragging the object, but strictly in a vertical or horizontal direction from its current position (right angles only).

If you require more precision than the dragging or ortho methods provide, you can nudge the object to the exact desired position. Nudging uses the arrow keys on your keyboard to move the object using predetermined increments. Coarse nudging moves the object by approximately 6 points with each keystroke, while fine nudging moves it only 1.4 points with each keystroke.

To perform an ortho move:

- Hold down the **SHIFT** key while dragging the object from its central zone.

To coarsely nudge an object:

- With the object selected, hold down the **SHIFT** key while pressing one of the arrow keys.

To finely nudge an object:

- With the object selected, press one of the arrow keys.

Positioning objects using the Align commands








Use the **Align** commands to place objects on the canvas so that they are vertically, horizontally, or center-aligned in relation to other objects on the canvas.

To align objects using the Align commands:

- 1 Select a group of objects, making sure that you select the primary object last.

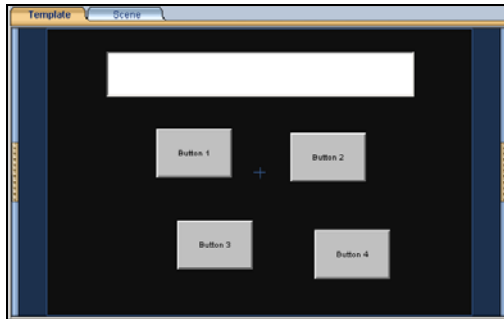
The primary object is the one that you would like all others aligned with, and its selection lines appear in blue, whereas those of others appear in red. You can change the primary object by clicking a selected object.

- 2 Select one of the following commands from the **Format>Align** menu, or select one of the icons from the **Align** toolbar on the Xstudio interface.

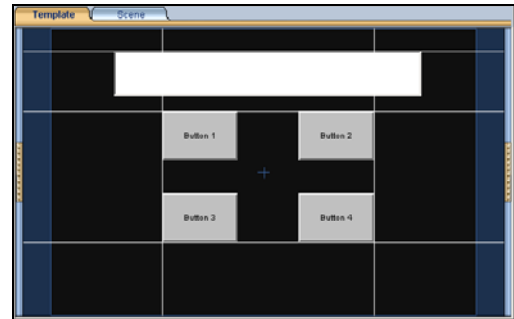
| Menu item | Icon | What it does |
|-----------------|---|--|
| Left |  | Aligns all selected objects with the left edge of the primary object. |
| Right |  | Aligns all selected objects with the right edge of the primary object. |
| Top |  | Aligns all selected objects with the top edge of the primary object. |
| Bottom |  | Aligns all selected objects with the bottom edge of the primary object. |
| Middle Y |  | Aligns all selected objects with the horizontal centre of the primary object. |
| Middle X |  | Aligns all selected objects with the horizontal centre of the primary object. |
| Centers |  | Aligns all selected objects with both the horizontal and vertical centers of the primary object. |

Positioning objects using guides

You can add guides to the canvas that allow you to align objects along common vertical or horizontal reference lines. Once the objects are added to the canvas, you can superimpose guides to the canvas and when an object is dragged near a guide, it automatically snaps to it. You can add one or many guides to the canvas.



Objects added to the canvas by drag-and-drop



The objects snap to the multiple guides, which helps organize and align the objects

Note: You can use the Property Browser to determine guide properties (i.e. color, whether or not they're visible, whether or not they can be moved around, and whether or not the "snap to" component is active).

To add a guide to the canvas:

- 1 Select **Format > Add Guide** and then either **Horizontal** or **Vertical**.
- 2 Add the guide to the canvas:
 - If you selected **Horizontal**, drag the guide from the top or bottom of the outer border of the canvas.

- If you selected **Vertical**, drag the guide from the left or right side of the outer border of the canvas.

3 Adjust the placement of the guide by dragging the guide up or down on the canvas.

To delete a guide:

- 1 Move the mouse over the guide until the cursor becomes a double-headed arrow.
- 2 Select the guide and drag it out of the window.

Cutting, copying and pasting objects to/from the canvas

You can move, duplicate, and/or replace objects on the canvas by using the **cut** and **copy** commands to place an object on the system's clipboard and then place the object on another canvas using the **Paste** command. When you use the **Cut** or **Copy** commands, the cut or copied object remains on the clipboard until you choose **Cut** or **Copy** again.

Note: Objects cannot be copied/cut from the control side and pasted to the scene side, or vice versa.

To cut, copy and paste objects:

- 1 Select the object by clicking on it once.
- 2 Select one of the following from the Xstudio interface's **Edit** menu:
 - The **Cut** command (**CTRL + X**) to remove the object and save a copy to the clipboard.Or,
 - The **Copy** command (**CTRL + C**) to keep the original object where it is and save a copy to the clipboard.
- 3 To paste a copy of the object, select the **Paste** (**CTRL + V**) command from the **Edit** menu.

To move a copy of the object out from the original:

- Hold down the CTRL key while moving the copy out from the original. If you release the CTRL key before the mouse button, the object will not be copied -- only moved.

To move a copy of the object out from the original at right angles only:



- Hold down the CTRL and SHIFT keys while moving the copy out from the original. If you release the CTRL key before the mouse button, the object will not be copied -- only moved.

Modifying the font properties of text objects



You can edit the font properties of text and text box primitives on the control side and text2D primitives on the scene side using Xstudio's **Font** toolbar or **Properties Browser**.

Note: You can select several text and/or text box objects and apply the same font properties to all at once. To select multiple text boxes, hold down either the **SHIFT** or the **CTRL** key while clicking each object to be grouped.

To modify an object's font properties using the Font toolbar:

- 1 Select the object by clicking on it once.
- 2 In the **Font** toolbar:
 - a Make a selection from the **font** drop-down list to select a font.
 - b Make a selection from the **pixel size** drop-down list to change text size.
 - c Click  to apply bold formatting and/or  to apply italic formatting.

To modify an object's font properties using the Properties Browser:

- 1 Select the object by clicking on it once.
 - 2 Open the Property Browser:
 - Select the **Properties Browser** icon 
- Or,
- Right-click the object and select the **Properties** from the context menu.
- 3 Select the **Font** property field in the Property Browser.
 - 4 Press the  button. The **Font** dialog box appears.
 - 5 Select the desired font type, style, and size, and then click **OK**.

Editing an image

Xstudio's Image Editor allows you to perform some basic image editing tasks on image assets like, adjusting the image's color and sharpness, cropping the image, among others.



Please see [Image Editor](#), on page 317 for a full description and instructions for using Xstudio's Image Editor.

Undoing and redoing edits on the canvas

Creating graphics displays often involves a certain amount of experimentation before achieving the desired layout and functionality. As such, Xstudio allows you to undo up to sixty (60) unsaved edits. Similarly, you can also redo the last unsaved edit performed a single, or multiple times.

To undo the last edit:

- Select the **Undo** command from the Xstudio interface's **Edit** menu.
- Or,
- Press **CTRL + Z** on your keyboard.

To redo the last edit:

- Select the **Redo** command from the Xstudio interface's **Edit** menu.
- Or,
- Press **CTRL + Y** on your keyboard.

Note: Edits will be lost if Xstudio or the project is closed before you save. See [Saving Xstudio projects and assets](#), on page 252 for more information.

Working with scene objects

6

The scene portion of a project contains the collection of graphic objects that will be displayed on air during playout. As described in previous chapters, you create the scene by adding objects (assets or primitives) to the scene's canvas and arranging them (e.g. align, layer, group) to create a functional and attractive graphics display (see [page 69](#)).

Once the objects have been added to the scene canvas, you can edit the object's properties to change the appearance and/or behavior of the object.

The following sections provide a brief introduction to each type of scene object and its properties:


- [Text2D objects on the scene canvas](#), on page 88
- [Clip objects on the scene canvas](#), on page 96
- [Image objects on the scene canvas](#), on page 98
- [Cel Animation objects on the scene canvas](#), on page 100
- [Video In objects on the scene canvas](#), on page 102
- [Crawl and Roll objects on the scene canvas](#), on page 104
- [Using scene objects as a mask](#), on page 109

Note: Scene object properties are fully defined in [Xstudio properties](#), on page 285.

Note that scene's objects can be static, animated or data driven, however the scene's playout is controlled by the objects and logic applied to the template or control panel. While this chapter mainly deals with static object features, the following chapters provide information and instructions for animating and associating scene objects control objects and external data sources:


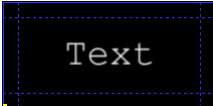

- [Animating scene objects](#), on page 111
- [Connecting control objects to scene objects](#), on page 162
- [Linking objects to a data source](#), on page 177







Text2D objects on the scene canvas



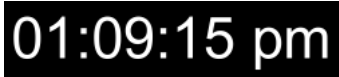
Adding a Text2D object  to the scene's canvas allows you to display a text string on air. While the text within the Text2D object can be "fixed" by typing directly into this object, the Text2D object can also host dynamic text, which can be controlled by a text box object on the control side.


Other methods for determining or changing the contents of the Text2D object during playback include using the **Set Text** action in response to an event or using the **Text** property within a keyframe on the scene's animation channel. Automation can also change the text on a scene.

You can change the default appearance and behavior of a Text2D scene object by selecting the object on the canvas and editing its properties settings in the Properties Browser. While full descriptions of all of the scene object properties are provided in [Scene object properties](#), on page 293, the following table summarizes the properties specifically associated with Text2D objects:

| Category | Text2D object properties |
|------------------------|---|
| | <p>The following properties determine the identity, content and control of the Text2D object: Name, Text, Visible, Locked, Dynamic, Use As Mask</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Transformations | <p>The following properties allow you to set the coordinates (location), spatial position and dimensions (size) of the Text2D object: X, Y, Z, RotateX, RotateY, RotateZ, ScaleX, ScaleY</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Appearance | <p>The following properties determine the style and formatting of the Text2D object. Text Direction, Shrink To Fit, Font, Justification, Opacity, WordWrap, Allow Word Break, Tracking, Leading, Basic Rendering</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> <p>The following examples demonstrate the properties that allow you to specify the text's font style and size.</p> <div style="display: flex; justify-content: space-around; align-items: flex-start;"><div style="text-align: center;"><p>Font: Arial - 86 Justification: Top Left</p></div><div style="text-align: center;"><p>Font: Courier New - 24 Justification: Mid Center</p></div><div style="text-align: center;"><p>Font: Playbill - 320 Justification: Mid Center Shrink To Fit: True</p></div></div> <p>You can also assign a font to the text by selecting the Text2D object on the canvas and using the Font toolbar on Xstudio's desktop (page 35). While the Font property applies the font style and size to the object's entire text string, you can force changes to portions of the text's style and appearance using embedded text tags. See Using embedded text tags within Text2D scene objects, on page 94 for more information.</p> |

| Category | Text2D object properties |
|----------------------|---|
| <p>Face</p> | <p>The following properties determine the style and color of the fill in the face of the characters in the Text2D object. FaceMaterialType, FaceGradient, FaceColor, FaceOpacity, FaceBlur Click on the links or see Scene object properties, on page 293 for a description of each of these properties. The following examples demonstrate the properties that allow you to specify the text's face color and style.</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p>FaceMaterialType: Color FaceColor: R: 255 G: 128 B: 0</p> </div> <div style="text-align: center;">  <p>FaceMaterialType: Gradient FaceGradient: Vertical FaceColor1: R: 255 G: 128 B: 0 FaceColor2: R: 0 G: 0 B: 255</p> </div> </div> |
| <p>Edge</p> | <p>The following properties determines the thickness, color and style of the edge that surrounds each character in the Text2D object. EdgeMaterialType, EdgeGradient, EdgeColor, EdgeOpacity, EdgeSize, EdgeBlur, EdgeShape, EdgeType Click on the links or see Scene object properties, on page 293 for a description of each of these properties. The following examples demonstrate the properties that allow you to specify the text's edge color and size.</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p>EdgeMaterialType: Color EdgeColor: R: 255 G: 255 B: 255 EdgeSize: 0</p> </div> <div style="text-align: center;">  <p>EdgeMaterialType: Color EdgeColor: R: 255 G: 255 B: 255 EdgeSize: 5</p> </div> </div> |
| <p>Shadow</p> | <p>The following properties determines the positioning, color and style of the shadow that is associated with each character in the Text2D object. ShadowMaterialType, ShadowGradient, ShadowColor, ShadowOpacity, ShadowType, ShadowDirection, ShadowDistance, ShadowBlur Click on the links or see Scene object properties, on page 293 for a description of each of these properties. The following examples demonstrate the properties that allow you to specify the text's shadow color, size and position.</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p>ShadowMaterialType: Color ShadowColor: R: 255 G: 255 B: 255 ShadowType: None ShadowDirection: 135 ShadowDistance: 0</p> </div> <div style="text-align: center;">  <p>ShadowMaterialType: Color ShadowColor: R: 255 G: 255 B: 255 ShadowType: Drop ShadowDirection: 135 ShadowDistance: 7</p> </div> </div> |

| Category | Text2D object properties |
|---------------------|---|
| <p>Frame</p> | <p>The following properties determines the color and style of the frame that behind the characters in the Text2D object. FrameMaterialType, FrameGradient, FrameColor, FrameOpacity, FrameShape</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> <p>The following examples demonstrate the properties that allow you to specify the text's frame color and style.</p> <div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  <p>FrameMaterialType: Color FrameColor: R: 255 G: 255 B: 255 FrameShape: None</p> </div> <div style="text-align: center;">  <p>FrameMaterialType: Gradient FrameGradient: Vertical FrameColor1: R: 255 G: 255 B: 255 FrameColor2: R: 0 G: 0 B: 255 FrameShape: Round</p> </div> </div> |
| <p>Time</p> | <p>The following properties allow you to create clocks and timers using the Text2D object. Type, Format, Marker, Offset, Start, Target</p> <p>See Creating simple clocks and timers using Text2D scene elements, on page 92 for complete instructions on using these properties.</p> <div style="text-align: center; margin: 10px 0;">  </div> <p>Type: Clock Format: hh:mm:ss Marker: am / pm Offset: 00:00:00</p> |

| Category | Text2D object properties |
|-------------------------------|--|
| <p>Transition</p> | <p>The following properties determine how the Text2D object transitions when the scene is played out. Trigger Transition, TransitionType, TransitionFeathering, TransitionDirection, TransitionDuration, TransitionOrientation, HingePosition, InTransitionDirection, InTransitionDuration, InTransitionFeathering, InTransitionHingePosition, InTransitionOffset, InTransitionType, OutTransitionDirection, OutTransitionDuration, OutTransitionFeathering, OutTransitionHingePosition, OutTransitionOffset, OutTransitionOrientation, OutTransitionType</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> <p>The following examples demonstrate the properties that allow you to specify how the text transitions on air.</p> <div style="text-align: center;">  <p>Trigger Transition: Always Transition Type: Reveal Transition direction: Left Transition Duration: 5</p> </div> |
| <p>User Properties</p> | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playback.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playback by calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

Creating simple clocks and timers using Text2D scene elements

Simple clocks and timers can be created by using a Text2D scene object to represent time. The following types of clocks and timers are possible:









- **Clock:** Displays the current system time in 12 or 24 hour time format. Optionally, you can display AM/PM and/or assign a relative offset to the displayed time.
- **Count up timer:** Counts time up, like a stopwatch.
- **Count down timer:** Counts time down, like a microwave.
- **Count down to target time:** Counts time down by displaying the time remaining from the current time until the specified target time.

Note: When using the Text2D object for a clock or timer, the Text2D object cannot contain any other text. In other words, the Text2D object must be completely dedicated to displaying time.

To create a clock or timer:

- 1 Add a Text2D object to the template's scene.
- 2 Select the Text2D object and open the Properties Browser.
The properties for Text2D object features a **Time** property category, which enables text elements to represent time.
- 3 Select from the **Type** property's drop-down list the type of time element to be displayed.
 - **None**
 - **Clock**
 - **Count Up Timer**
 - **Count Down Timer**
 - **Count Down Timer To Target Time**

4 Depending on the **Type** property's setting, additional properties are displayed in the **Time** properties category. The following table describes each of the **Type** property's options:

| | |
|----------------------------------|---|
| Clock | <p>When clock is selected, the following properties are exposed:</p> <ul style="list-style-type: none"> • Format: Determines how the clock's time will be displayed (e.g. hh:mm:ss). Selecting the Format property exposes the  button to open the Time Format dialog box. You can select whether hours, minutes, and/or seconds should be displayed, if the clock should be 12 or 24 hour, and/or if leading zeroes should be displayed. • Marker - Determines whether or not an AM/PM marker is displayed and in what format. Select from the following options: None, am/pm, AM/PM. • Offset - Determines if an offset is added to the clock's time. Selecting the Offset property displays the  button to open the Time Offset dialog box. Use the up/down arrows to set the offset value for hour, minute, and/or second relative to the current system time. Values can be positive or negative. |
| Count Up Timer | <p>When Count Up Timer is selected, the following properties are exposed:</p> <ul style="list-style-type: none"> • Format - Determines how the clock's time will be displayed (e.g. hh:mm:ss). Selecting the Format property exposes the  button to open the Time Format dialog box. You can select whether hours, minutes, and/or seconds should be displayed and/or if leading zeroes should be displayed. Hours are always displayed in 24 hour format. • Start - The time at which the timer will start counting up. Selecting the Start property exposes the  button to open the Counter Start dialog box. Use the up/down arrows to select the exact hours, minutes, and seconds values to be set as the displayed start time. Values can only be positive. |
| Count Down Timer | <p>When Count Down Timer is selected, the following properties are exposed:</p> <ul style="list-style-type: none"> • Format - Determines how the clock's time will be displayed (e.g. hh:mm:ss). Selecting the Format property exposes the  button to open the Time Format dialog box. You can select whether hours, minutes, and/or seconds should be displayed and/or if leading zeroes should be displayed. Hours are always displayed in 24 hour format. • Limit Leading Digits - Determines if the leading digits displayed in a count down timer can exceed the 24 (for hours) or 60 (for minutes & seconds) limitation. When the Limit Leading Digits property is set to True, the leading digits in the timer (i.e. minutes) are limited to the 24/60 limitation (e.g. 70 minutes = 1 hour 10 minutes). When the Limit Leading Digits property is set to False, the leading digits in the timer can exceed the 24/60 limitation and will display the specified value. • Start - The time at which the timer starts counting down. Selecting the Start property exposes the  button to open the Counter Start dialog box. Use the up/down arrows to select the exact hours, minutes, and seconds values to be set as the displayed start time. Values can only be positive. |
| Count Down to Target Time | <p>When Count Down To Target Time is selected, the following properties are exposed:</p> <ul style="list-style-type: none"> • Format - Determines how the clock's time will be displayed (e.g. hh:mm:ss). Selecting the Format property exposes the  button to open the Time Format dialog box. You can select whether hours, minutes, and/or seconds should be displayed and/or if leading zeroes should be displayed. Hours are always displayed in 24 hour format. • Target - The target time to which the timer will count down to. When the current time passes the target time, the count time will stop at zero when the target time has been reached. Selecting the Target property exposes the  button to open the Counter Target dialog box. Use the up/down arrows to select the exact hours, minutes, and seconds values to be set as the used as the target time. Values can only be positive. |

5 Save the template to apply and retain any edits to the Text2D object's Time properties.

Using embedded text tags within Text2D scene objects

The properties that determine the appearance and style of the text within a Text2D object are applied uniformly to all of the characters within the object's text string. Using embedded text tags within a Text2D object allows you to change the text's formatting within portions of the text string. For example, the following text string contains the **Size** embedded text tag, which allows us to change the size of the font within the tag:

```
Season premiere of <SIZE=24>Ghost Mountain</SIZE> starring Lisa Hope
```



Like the HTML markup tags, each embedded text tag consists of a pair of tags: a start tag and a terminator. The start tag introduces and sets the new setting to the text, while the terminator restores the text's attribute to its previous value.

You can also nest and/or overlap embedded text tags, which allows you to apply additional formatting to text that is already being treated with another text tag.

Example of nested text tags:

The **Size** and **Color** text tags are nested within the **FONT** text tag.

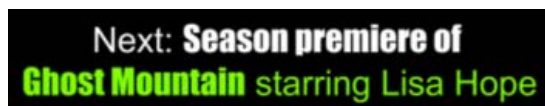
```
Season premiere of <CR><Font=Impact><Size=36>Ghost Mountain</Size>  
starring <Color=128,255,0>Lisa Hope</Color></Font>
```



Example of overlapped text tags:

Color overlaps the **Font** text tag (within the Font and also continues once Font is closed).

```
Next: <Font=Impact>Season premiere of <CR><Color=128,255,0>Ghost  
Mountain</Font> starring Lisa Hope</Color>
```




The following table identifies and describes each of the available embedded text tags:

| | |
|------------------------|---|
| Bold | <Bold=1> </Bold> Enables or disables the bold attribute of the font. Specify 1 to apply the bold attribute or 0 remove the bold attribute. The terminator restores the previous bold state. |
| Carriage Return | <CR> Inserts a hard carriage return into the text. No terminator is required. |
| Edge Color | <EdgeColor=255,255,255> </EdgeColor> Changes the edge color by applying new R,G,B color values to the text between the text tag and the tag's terminator. If the edge style type is gradient, then this tag changes the first color. The terminator restores the previous edge color. |

| | |
|---------------------|--|
| Edge Blur | <p><EdgeBlur=5> </EdgeBlur> Changes the blur effect applied to the edge style for the portion of text between the text tag and the tag's terminator. The value of the Edge Blur must be an integer and the terminator restores the previous edge blur.</p> |
| Edge Size | <p><EdgeSize=4> </EdgeSize> Changes the size of the edge for the portion of text between the text tag and the tag's terminator. A size of 0 results in no edge being visible. The terminator restores the previous edge size.</p> |
| Face color | <p><Color=255,255,255> </Color> Changes the face color by applying new R,G,B color values to the text between the text tag and the tag's terminator. If the face style type is gradient, then this tag changes the first color. The terminator restores the previous face color.</p> |
| Face Blur | <p><Blur=0></Blur> Changes the blur effect applied to the face style for the portion of text between the text tag and the tag's terminator. The Blur value must be an integer and the terminator restores the previous face blur.</p> |
| Font | <p><FONT=Impact> Changes the font face style. Specify the name of any installed font. The terminator restores the previous font face.</p> |
| Italic | <p><ITALIC=1> </ITALIC> Enables or disabled the italic attribute of the font. Specify 1 to apply italics or 0 to remove italics. The terminator restores the previous italic state.</p> |
| Shadow Color | <p><ShadowColor=255,255,255> </ShadowColor> Changes the shadow color by applying new R,G,B color values for the portion of text between the text tag and the tag's terminator. If the shadow style type is gradient, then this tag changes the first color. The terminator restores the previous shadow color.</p> |
| Shadow Blur | <p><ShadowBlur=6> </ShadowBlur> Changes the blur effect applied to the shadow style for the portion of text between the text tag and the tag's terminator. The value of the Shadow Blur must be an integer and the terminator restores the previous shadow blur.</p> |
| Size | <p><SIZE=50> </SIZE> Changes the size of the font, which is the character height in pixels. The terminator restores the previous font size.</p> |

Clip objects on the scene canvas

Adding a Clip object to the scene's canvas allows you to position and playout a video clip on air.

A clip object on the scene can either be clip asset that was added from the Asset Browser, or a Clip primitive . When a clip primitive is added to the scene, it is empty as it does not yet have a video clip file associated to it. See [page 71](#) for instructions on associating an asset with the Clip primitive.

Other methods for determining or changing which clip asset will appear in the Clip object include using **Clip Command** action in response to an event during playout or using the **ClipName** property within a keyframe on the scene's animation channel.


You can change the default appearance and behavior of a Clip scene object by selecting the object on the canvas and editing its properties settings in the Properties Browser. While full descriptions of all of the scene object properties are provided in [Scene object properties](#), on page 293, the following table summarizes the properties specifically associated with clip objects.

| Category | Clip object properties |
|------------------------|---|
| | <p>The following properties determine the identity, visibility and control of the clip object: Name, Visible, Use As Mask, Locked</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Transformations | <p>The following properties allow you to set the coordinates (location), spatial position and dimensions (size) of the clip object: X, Y, Z, RotateX, RotateY, RotateZ, ScaleX, ScaleY</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Clip | <p>The following properties determine which clip asset file is associated with the object, as well as other attributes that affect the clip's appearance and playout behavior. ClipName, ClipLocation, AutoPlay, Dynamic, Alignment, AntiAliasing, StartTime (sec), EndTime (sec), SendEvents, SendPlayListEvents, Loop, StartLoop, EndLoop, LoopTransition, HoldLastFrame, ClosedCaption, Opacity, Style, AudioLevel, AudioProfile. Playback State On Invisible</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> <p>See page 71 for instructions on assigning a clip asset file with the Clip object using the ClipName property.</p> |

| Category | Clip object properties |
|------------------------|--|
| Transition | <p>The following properties determine how the Clip object transitions (enters/exits) when the scene is played out.</p> <p>Trigger Transition, TransitionType, TransitionFeathering, TransitionDirection, TransitionDuration, TransitionOrientation, HingePosition, InTransitionDirection, InTransitionDuration, InTransitionFeathering, InTransitionHingePosition, InTransitionOffset, InTransitionType, OutTransitionDirection, OutTransitionDuration, OutTransitionFeathering, OutTransitionHingePosition, OutTransitionOffset, OutTransitionOrientation, OutTransitionType</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

Image objects on the scene canvas

Adding an Image object to the scene's canvas allows you to position and play out a image file on air.

Image objects on a scene can either be an image asset that was added from the Asset Browser, or an Image primitive . When an image primitive is added to the scene, it is empty as it does not yet have an image file associated to it. See [page 71](#) for instructions on associating an asset with the Image primitive.

Other methods for determining or changing which image asset will appear in the Image object include using **Set Image** action in response to an event during playout or using the **ImageName** property within a keyframe on the scene's animation channel.


You can change the default appearance and behavior of an Image object by selecting the object on the canvas and editing its properties settings in the Properties Browser. While full descriptions of all of the scene object properties are provided in [Scene object properties](#), on page 293, the following table summarizes the properties specifically associated with image objects.

| Category | Image object properties |
|------------------------|---|
| | <p>The following properties determine the identity, visibility and control of the image object:</p> <p>Name, Visible, Use As Mask, Locked</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Transformations | <p>The following properties allow you to set the coordinates (location), spatial position and dimensions (size) of the clip object:</p> <p>X, Y, Z, RotateX, RotateY, RotateZ, ScaleX, ScaleY</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Image | <p>The following properties determine which image asset file is associated with the object, as well as other attributes that affect the image's appearance and playout behavior.</p> <p>ImageName, ImageLocation, Dynamic, Alignment, AntiAliasing, Opacity, Style</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |

| Category | Image object properties |
|------------------------|---|
| Transition | <p>The following properties determine how the Image object transitions (enters/exits) when the scene is played out.</p> <p>Trigger Transition, TransitionType, TransitionFeathering, TransitionDirection, TransitionDuration, TransitionOrientation, HingePosition, InTransitionDirection, InTransitionDuration, InTransitionFeathering, InTransitionHingePosition, InTransitionOffset, InTransitionType, OutTransitionDirection, OutTransitionDuration, OutTransitionFeathering, OutTransitionHingePosition, OutTransitionOffset, OutTransitionOrientation, OutTransitionType</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playback.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playback by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

Cel Animation objects on the scene canvas

Adding an Cel Animation object to the scene's canvas allows you to position and play out a cel animation asset on air.

Cel animation objects on a scene can either be a cel animation asset that was added from the Asset Browser, or an Cel Animation primitive . When a cel animation primitive is added to the scene, it is empty as it does not yet have an animation file associated to it. See [page 71](#) for instructions on associating an asset with the cel animation primitive.

Other methods for determining or changing which cel animation asset will appear in the Cel animation object include using **Set Cel Animation** action in response to an event during playout or using the **CelName** property within a keyframe on the scene's animation channel.


You can change the default appearance and behavior of a cel animation object by selecting the object on the canvas and editing its properties settings in the Properties Browser. While full descriptions of all of the scene object properties are provided in [Scene object properties](#), on page 293, the following table summarizes the properties specifically associated with cel animation objects.

| Category | Cel animation object properties |
|------------------------|--|
| | The following properties determine the identity, visibility and control of the cel animation object: Name , Visible , Use As Mask , Locked Click on the links or see Scene object properties , on page 293 for a description of each of these properties. |
| Transformations | The following properties allow you to set the coordinates (location), spatial position and dimensions (size) of the clip object: X , Y , Z , RotateX , RotateY , RotateZ , ScaleX , ScaleY Click on the links or see Scene object properties , on page 293 for a description of each of these properties. |
| Cel Animation | The following properties determine which cel animation asset file is associated with the object, as well as other attributes that affect the animation's appearance and playout behavior. CelName , CellLocation , AutoPlay , Dynamic , Alignment , AntiAliasing , StartFrame , EndFrame , StartLoop , EndLoop , EndEvent , Loop , Reverse , PingPong , Opacity , Rate , Style , Use OXI Default Position Click on the links or see Scene object properties , on page 293 for a description of each of these properties. |

| Category | Cel animation object properties |
|------------------------|---|
| Transition | <p>The following properties determine how the Cel Animation object transitions (enters/exits) when the scene is played out.</p> <p>Trigger Transition, TransitionType, TransitionFeathering, TransitionDirection, TransitionDuration, TransitionOrientation, HingePosition, InTransitionDirection, InTransitionDuration, InTransitionFeathering, InTransitionHingePosition, InTransitionOffset, InTransitionType, OutTransitionDirection, OutTransitionDuration, OutTransitionFeathering, OutTransitionHingePosition, OutTransitionOffset, OutTransitionOrientation, OutTransitionType</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

Video In objects on the scene canvas

Adding a Video In object to the scene's canvas allows you to position and play out a video for an incoming video sources.

When a Video In primitive is added to the canvas, it is represented by a television test pattern object  until a video source is specified in the object's **Source** property.

You can change the default appearance and behavior of a Video In scene object by selecting the object and editing its properties settings in the Properties Browser. While full descriptions of all of the scene object properties are provided in [Scene object properties](#), on page 293, the following table summarizes the properties specifically associated with clip objects.

| Category | Video In object properties |
|------------------------|---|
| | The following properties determine the identity, content and control of the video in object: Name , Visible , Use As Mask , Locked , Source Click on the links or see Scene object properties , on page 293 for a description of each of these properties. |
| Transformations | The following properties allow you to set the coordinates (location), spacial position and dimensions (size) of the video in object: X , Y , Z , RotateX , RotateY , RotateZ , ScaleX , ScaleY Click on the links or see Scene object properties , on page 293 for a description of each of these properties. |
| Video | The following properties allow you to control the appearance and level of the Video In source's audio and visual aspects: Dynamic , AntiAliasing , Opacity , AudioLevel Click on the links or see Scene object properties , on page 293 for a description of each of these properties. |
| Crop | The following properties allow you to crop the outside edges of the source video that is displayed within the Video In object: CropLeft , CropTop , CropRight , CropBottom Click on the links or see Scene object properties , on page 293 for a description of each of these properties. |

| Category | Video In object properties |
|------------------------|--|
| Transition | <p>The following properties determine how the Video In object transitions (enters/exits) when the scene is played out.</p> <p>Trigger Transition, TransitionType, TransitionFeathering, TransitionDirection, TransitionDuration, TransitionOrientation, HingePosition, InTransitionDirection, InTransitionDuration, InTransitionFeathering, InTransitionHingePosition, InTransitionOffset, InTransitionType, OutTransitionDirection, OutTransitionDuration, OutTransitionFeathering, OutTransitionHingePosition, OutTransitionOffset, OutTransitionOrientation, OutTransitionType</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playback.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playback by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

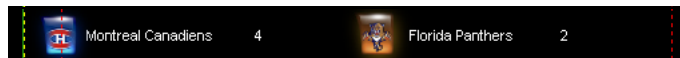
Crawl and Roll objects on the scene canvas

Crawl and roll objects function identically except for the fact that crawls move horizontally across the screen, whereas rolls move vertically. Both crawls and rolls differ from other primitives in that they can be composed of more than one element. Crawls and rolls always start with a Text2D object, but you can include more than one Text2D object, as well as images, cel animations, etc. All are automatically grouped so that they move together.

For example, the following crawl displays the latest headlines in the Text2D object and each headline is separated by the logo image.



Meanwhile, the following crawl is displaying sports results using multiple text boxes for the team names and the scores and images representing the teams. The team logos are linked to a lookup table. The lookup table replaces incoming data with text or in this case, with the name of an image. See [page 191](#) for more information about using lookup tables



Crawl and rolls objects on the scene side each need a corresponding text box object on the control side, which feeds content to the crawl or roll's Text2D object. In some cases, when the scene is taken to air, an operator will manually type content into the text box object from the control side. However, in most cases (i.e. tickers) the text box is linked to a data source, which feeds content to the crawl or roll's Text2D object.

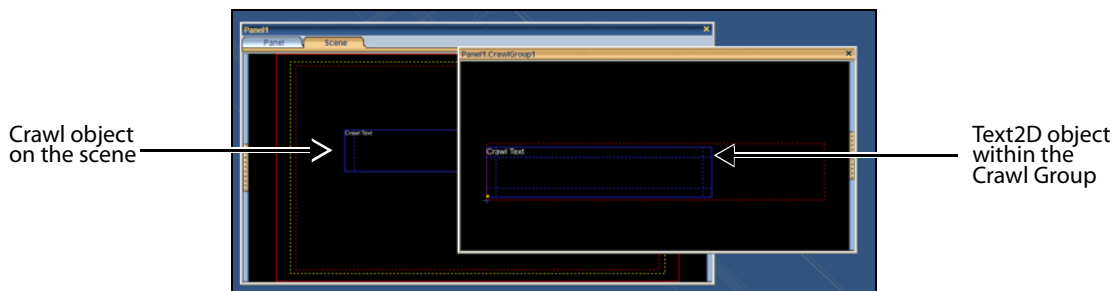
The following sections describe how to create a basic crawl, make basic changes to the crawl, as well as introduce how to configure a crawl to receive and display content from a data source:

- [Creating a basic crawl](#), on page 105
- [Making basic changes to a crawl or roll](#), on page 106
- [Creating a controllable crawl with Xstudio](#), on page 107

Creating a basic crawl

When a crawl primitive is added to the scene's canvas, it can be positioned and resized just like any other primitive. However, crawl (and rolls) primitives are unique in that they really represent a group of objects (1 or more) that move together within the boundaries of the crawl.

By default, a crawl object contains a single Text2D object that contains the text "Crawl Text". Double-clicking the crawl object opens the crawl group, which is the work area in which you can add other Text2D objects, images, cel animations, etc. Each object within the crawl group can be configured individually using the object's properties, as well as be linked to control objects that can determine the object's content and behavior.



To familiarize yourself with the components and behavior of a basic crawl, we suggest that you perform the following task to create a simple crawl that repeats a static text.

To create and playout a basic crawl:

- 1 With Xstudio actively connected to a playout device (i.e. Software CG), open a new project (i.e. Panel + Scene) in Xstudio.
- 2 On the scene side of a project, add a crawl primitive to the canvas and adjust it to the desired size and position on the scene.
- 3 Open Xstudio's Properties Browser and view the crawl's properties.

These properties determine the appearance and behavior of the crawl. Instructions for setting these properties are provided on [page 106](#). For the purposes of this example, leave the crawl's properties in their default state and ensure that the **Crawl>AutoStart** and **Crawl>Repeat** properties are set to True.
- 4 Switch to the control side of the project (i.e. Panel) and add a Text Box primitive to the canvas.
- 5 Link the crawl's Text2D object to the Text Box on the control side, so that text typed into the text box appears in the crawl during playout.
 - a Open the Scene tree panel.
 - b Expand the **CrawlGroup** object listed in the Scene category.
 - c Drag and drop the Text2D in the CrawlGroup object onto the Text box.
- 6 Save and publish the project to the active playout device (see [page 251](#)).
- 7 Switch to **Production** mode and type some text in the text box.

The text displays on the output and repeatedly moves across the screen within the boundaries defined by the crawl object.

Making basic changes to a crawl or roll

Like other primitives, selecting the crawl or roll object on the scene while the Properties Browser is open exposes the properties that affect the appearance and behavior of the overall crawl or roll object.

The following table summarizes the properties specifically associated with crawl and roll objects. Full descriptions of all of the scene object properties are provided in [Scene object properties](#), on page 293.


| Category | Crawl & Roll object properties |
|------------------------|--|
| | <p>The following properties determine the identity and visibility of the crawl object: Name, Visible, Opacity, Use As Mask, Locked</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Transformations | <p>The following properties allow you to set the coordinates (location), spacial position and dimensions (size) of the crawl or roll object: X, Y, Z, RotateX, RotateY, RotateZ, ScaleX, ScaleY</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Crawl or, Roll | <p>The following properties determine the formatting and playout behavior of the crawl or roll object, including when the crawl or roll should start playing. Rate, Spacing, Repeat, Autostart, TriggerPoint, TriggerEvent, ItemStartEvent, Direction</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| Transition | <p>The following properties determine how the crawl/roll object transitions (enters/exits) when the scene is played out. Trigger Transition, TransitionType, TransitionFeathering, TransitionDirection, TransitionDuration, TransitionOrientation, HingePosition, InTransitionDirection, InTransitionDuration, InTransitionFeathering, InTransitionHingePosition, InTransitionOffset, InTransitionType, OutTransitionDirection, OutTransitionDuration, OutTransitionFeathering, OutTransitionHingePosition, OutTransitionOffset, OutTransitionOrientation, OutTransitionType</p> <p>Click on the links or see Scene object properties, on page 293 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

To change the appearance and behavior of the individual objects within the crawl group or roll group, you must open the group and select the individual object.

By default, a crawl or roll always contains a single Text2D object. Double-clicking the crawl object opens the crawl group, which exposes the Text2D object within the crawl. You can use the Properties Browser to edit the properties of the Text2D object (i.e. size and position, color, font style...etc.) just like you would do for any Text2D object.

Note that when the default **Text2D** object is selected it does not take up the entire width of the crawl object.

You can also include more than one Text2D object, as well as images, cel animations... by dragging them onto the crawl group within the boundaries of the crawl or roll. As always, you can reposition the objects and change the object's properties by selecting the object and edit the settings in the Properties Browser.

Note: Shrink-wrapping determines the height of crawl groups and the width for roll groups. Meanwhile, users determine how wide a crawl is or how tall a roll is. The sum of the width of the objects in a crawl is allowed to be greater than the width of the crawl itself. When you are done making changes to the crawl or roll group objects, click  to close the **CrawlGroup** or **RollGroup** window. You will be prompted to save your changes. Click the **Yes** button to save, the **Cancel** button to return to the **CrawlGroup** or **RollGroup** window, or **No** to close the window without saving any changes and return to the scene.

When you return to the scene, you will notice that a **CrawlGroup** or **RollGroup** heading has been added to the Scene category in the Scene Tree. Within that group you will find the animation properties for the crawl or roll object, as well as the objects that belong to the group (i.e Text2D). These will be used to link the individual objects in the crawl to control objects. The control objects can then be configured to feed data or other behaviors to the crawl objects. For more information, see [Selecting, grouping, and/or layering objects on the canvas](#), on page 75 and [Linking objects to a data source](#), on page 177.

Creating a controllable crawl with Xstudio

The following example demonstrates how to create a controllable crawl in Xstudio. In this particular example, when the Start Crawl button is clicked, the local variable `L:ticking` is set to true, the reset command forces the ticker to start from the beginning and the tick command causes the first line to be loaded into the control panel textbox and displayed on the scene side. The `ItemStartEvent StartTick` is actioned causing another tick command to be issued and the next line of data to be displayed.

Pressing the Stop Crawl button sets the local variable to false and further ticks are disabled. The crawl will complete the current information already loaded but will then crawl off screen. As the last item crawls past position 0, the `StopTick` user event is triggered and the background will cut off. The Clear Crawl command does the same as Stop Crawl but in addition hides the crawl and background by setting the visibility to false causing them to disappear immediately.

Note: The Set Visibility actions can obviously be replaced by animations to enhance the display.

To create a controllable crawl:

- 1 Open a new Panel + Scene.

- 2 Add a crawl object to the scene side.
- 3 Add a graphic object to the scene side as a background for the crawl.
- 4 Add a textbox and a button to the control side. Label the button Start Crawl.
- 5 Set up a Data Schema to access the required data (see [Schemas](#), on page 179 for more information).
- 6 Set up a Ticker with the parameters shown below.
- 7 Display the Ticker data and link it to the textbox by 'drag and drop' as shown.
- 8 Open the scene objects display by clicking the gripper bar on the right of the control panel window. Expand the crawl group by clicking the + sign and link the scene crawl text box to the control panel by dragging and dropping as shown below.
- 9 Set the CrawlGroup properties to Repeat = False, TriggerPoint = 0 and add ItemStartEvents called StopTick and StartTick.
- 10 Open the Action Editor and create a User Event called StartTick and then click OK. Repeat this procedure to create a User Event called StopTick.
- 11 Click on the home area of the control panel and set up the condition, action and action parameters for StartTick as shown below. NOTE: The target for the Ticker Command can be dragged directly from the Data Schema.
- 12 Click the Start Crawl button and edit the actions for the On Click event as shown below.
- 13 Copy and paste the Start Crawl button and this will copy all of the actions as well.
- 14 Rename the copied button as Clear Crawl.
- 15 Select the Clear Crawl button and change both Set Visibility actions to false.
- 16 Delete one of the Ticker Command actions.
- 17 Change the Set Data value to false.
- 18 Change the Ticker Command to Stop.
- 19 Copy and paste the Clear Crawl button.
- 20 Rename the copied button as Stop Crawl.
- 21 Delete both the Set Visibility actions.
- 22 Click the ClearCrawl button and copy the Set Visibility action for the crawl background.
- 23 Click the StopTick user event and paste the Set Visibility action.

Using scene objects as a mask

When scene objects are layered on top of each other, the top object can be defined as a mask. Using the mask's alpha defined areas to reveal or block areas of the object behind it. Masks allow you to create impressive graphical effects by creating soft fades, decorative edges, and translucent effects.

Any object on the scene that has an alpha (image, cel, clips, text... etc.) is able to mask an object behind it. This means that masks can be irregular shaped since they are based on the exact shape of the mask object's alpha areas.

Note: If the objects have been grouped, you can use the entire group as a mask by setting the group's Mask properties. See [Group object properties](#), on page 311 for more information.

To specify that an object is to be used as a mask, you must set the **Use As Mask** parameter in the object's Xstudio properties. The following values are available for the **Use As Mask** parameter:

- **None:** Disables masking. In other words, the object will not be applied to the scene as a mask.
- **Positive:** Enables the object as a mask on the scene by displaying the lower image where the mask object's alpha value is 1. In areas where the object's alpha value is 0, the masked object (lower) is hidden (blocks the lower image).
- **Negative:** Enables the object as a mask on the scene by displaying the lower image where mask object's alpha value is 0. In areas where the object's alpha value is 1, the masked object (lower) is hidden (blocks the lower image).

The graphic below shows a scene where two of the three objects are grouped so that part of logo (back) will be masked by a white polka-dotted image (front). The white dots in the mask image have an alpha value of 1 and the transparent area has an alpha value of 0. When the dots image's **Use As Mask** parameter is set to **Positive** and the scene is played out, the white area (holes) reveal the logo image behind it. In the area where the alpha value is 0, the logo is not shown (blocked), but the background is displayed since it was not part of the group being masked.

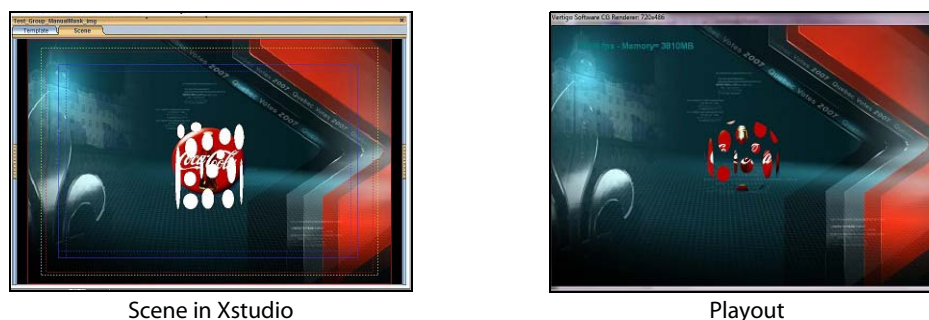


Fig. 6-1: Masking one object with another

If a mask is applied to a scene, but the masked object is not part of a group in the scene, the mask will punch through to black, which may include any keyer behind it.



Fig. 6-2: If the mask object is not part of a group, the mask drills down to black layer

Animation parameters that were available for the group or the individual objects are all still available and remain unchanged. You can animate a mask group as a whole group, or animate as individual objects within the group. Note however, that the mask is generated based on the alpha values within the group's original boundaries. That means that if the mask is animated out of the group's original area, the mask will not be visible as it moves beyond the original boundaries. Also, the bounding box of the mask is static. Therefore, any cropping will happen on the static box before any animation.



The original mask



As the mask is animated, the original boundary (straight line) is exposed.

Fig. 6-3: The mask is not be visible as it moves beyond the original boundaries

7 Animating scene objects

The Animation Editor allows you to create scene animations by changing the properties of the objects on the scene over a period of time to create visual effects. Some of the visual effects that can be achieved are:

- Fading an object in or out during playout using the object's Opacity property.
- Moving an object around the screen during playout using the object's X, Y, and or Z coordinate properties.
- Having an object appear or disappear from the scene during playout using the object's Visibility property and setting a transition style.
- Shrinking or expanding the size of the object during playout using the object's Scale X and Scale Y properties.
- Flip or rotate the object during playout using the object's Rotate X, Rotate Y, and Rotate Z properties.

Animations change the properties of objects in a scene over time to create a visual effect. For example, an image can have its opacity, X, and Y coordinates properties animated to produce the effect of an image fading in as it moves into the corner of the screen. An animation is composed of one or more channels, each of which describes how a single property of a single object changes over time. A channel is, in turn, composed of keyframes. Each keyframe defines the value of the property at a specific time and describes how to interpolate to the next keyframe.

The Animation Editor also allows you to integrate and control the playout of secondary animations within the primary animation using keyer channels, fader channels and command channels.

The following sections describe and provide instructions for creating, editing, and managing scene animations using the Animation Editor:

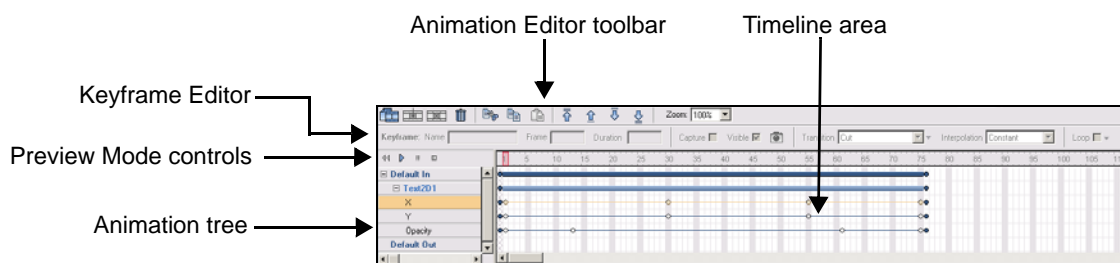
- [Animation Editor's interface components](#), on page 112
- [Creating scene animations](#), on page 129
- [Editing existing animations](#), on page 139
- [Animated property descriptions](#), on page 143

Animation Editor's interface components

The Animation Editor's interface organizes the scene's animations into an Animation tree consisting of three levels: animations, objects, and properties. A scene animation is created by adding scene object properties to the animation and then using keyframes to change the object's animatable properties over time to create a visual effect.

Keyer, fader and command channels can also be added to animations in the Animation Tree to control specific layout tasks and techniques from within the animation itself.

To track the behavior of the animation, the Animation Editor displays each animation, including its objects and objects' properties, as channels on a timeline. Adding keyframes to the property channels allows you use the Keyframe Editor to specify a change to the settings of the property at particular point during the scene's playout. You can then preview the playout of the scene animation on a device using the Animation Editor's Preview Mode controls.






Consult the following topics for a more detailed functional description of each part of the Animation editor's interface components:







- [Animation Editor toolbar](#), on page 112
- [Keyframe Editor](#), on page 113
- [Preview Mode controls](#), on page 117
- [Animation Tree and Animations](#), on page 117
- [Animation Timeline and Keyframes](#), on page 124

Animation Editor toolbar

The Animation Editor toolbar contains buttons that allow you to add, remove, and order the animations and their object properties in the Animation Tree.

The following table provides a functional description of each button in the Animation Editor toolbar:

| | |
|---|--|
| <p>New Animation</p>  | <p>Creates a new animation into which objects and their properties can be added. See Choosing a type of animation, on page 131 for usage instructions.</p> |
| <p>Insert KeyFrame</p>  | <p>Inserts a keyframe in the selected channel at the selected frame in the timeline. See Adding and setting keyframes, on page 132 for usage instructions.</p> |
| <p>Remove KeyFrame</p>  | <p>Removes the selected keyframe from the timeline. See Deleting a keyframe, on page 142 for usage instructions.</p> |

| | |
|--|--|
| <p>Delete</p>  | <p>Deletes the selected items (animation, object, channel, keyframe) from the animation tree and/or the timeline area.</p> <p>See Deleting channels from the Animation Tree, on page 139 and Deleting a keyframe, on page 142 for usage instructions.</p> |
| <p>Cut, Copy, Paste</p>    | <p>Allows you to cut or copy and then paste a selected item in the Animation Editor, whether it be an animation, an object channel, a property channel, or a keyframe. These features provide a quick and easy way of reproducing or moving items in the animation editor. See Cutting, copying, and pasting animation elements, on page 140 for usage instructions.</p> |
| <p>Ordering</p>  | <p>Allows you to reorder the items (animations, objects, properties) in the animation tree.</p> <ul style="list-style-type: none"> • Top: Moves the selected item to the very top of the animation tree's list. • Up: Moves the selected item up by one position in the animation tree's list. • Down: Moves the selected item to the very bottom of the animation tree's list. • Bottom: Moves the selected item down by one position in the animation tree's list. |
| <p>Zoom</p>  | <p>Allows you to zoom out on the Animation Editor's timeline. This function is useful when you have created a lengthy animation and you would like to see a larger portion of it on the timeline.</p> <p>Select from the drop-down list one following settings:</p> <ul style="list-style-type: none"> • 100% • 75% • 50% • 25% • 12% |

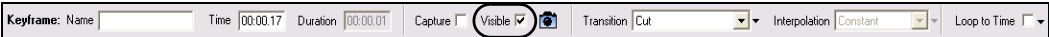
Keyframe Editor

The Keyframe Editor is located above the Animation Editor's timeline, and allows you to set the parameters for the selected keyframe on a property channel. A keyframe represents a particular point of change in time of the object property's setting.

Since the parameters differ depending on the type and function of the property, the Keyframe Editor's contents change according to the selected keyframe. In particular, the **value entry field** changes depending on the property displayed. For example, setting a keyframe for the **Opacity** property involves editing a value displayed in a text box, while the **Visible** property involves selecting or clearing a check box that represents either **On** or **Off**.

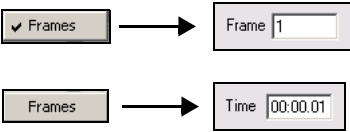








The Keyframe Editor's display for the Opacity property

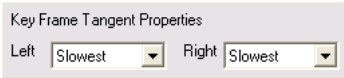



The Keyframe Editor's display for the Visible property

The following table provides a functional description of each of the Keyframe Editor's fields. In addition, The acceptable values and settings for the different property values is described in [Animated property descriptions](#), on page 143.

| | |
|------------------------|--|
| <p>Name</p> | <p>The Name field allows you to enter a name for the selected keyframe. A name is required only if the keyframe will be accessed using a SetKeyFrame action (see page 243). Keyframe names should be unique within an animation. For example, you cannot make an action execute on two keyframes simultaneously by giving them the same name.</p> |
| <p>Frame</p> | <p>Displays the keyframe's location on the timeline either as a frame number or as timecode. The relation between frame number and time comes from the Video Format property of the scene. For instance, if the NTSC format is chosen, there are 30 frames per second. As demonstrated by the images below, you can toggle back and forth between Frame and Time when assigning keyframes by right-clicking on Frame or Time field, then selecting or deselecting Frames from the context menu.</p>  |
| <p>Duration</p> | <p>Duration is display for the all Visibility keyframes, but it is enabled if the Transition field is set to anything except Cut. ClipName, Text, ImageName, CelName and Keyer channels also have the Duration property. Depending upon whether the Keyframe Editor is set to Frame or Time, the Duration field displays the number of frames or the length of time that it will take to complete the Visible property's transition. For example, if the transition field is set to Fade and the Duration field is set to 5 frames, then the object will begin to fade out upon encountering the keyframe and increasingly fade out over five frames at which point the object becomes no longer visible. When the Duration field is set, a yellow channel line extends to the right of the selected keyframe on the timeline. The yellow line's length represents the duration of the visibility transition. You can edit the Duration field's value by either typing a value in the text box, or by dragging the right edge of the yellow line. If you drag the yellow line, the Duration field is updated and displayed automatically in the Duration field.</p> |

| | |
|--|---|
| <p>Capture</p> | <p>When selected, the Capture check box indicates that the keyframe's value will be obtained from the value of the object property at the time of the animation's execution. This is useful when then the keyframe's required value is not known at edit time.</p> <p>When the Capture check box is not selected, you must enter an explicit value for the keyframe using the keyframe's property value field, which appears immediately to the right of the Capture check box.</p> <p>The following is an example of when the capture field might be useful: you may want to squeeze a video object so that it fills only the top left corner of the screen. However, the size of the video object when the animation starts may vary. In this case, the Capture check box should be selected for the initial keyframes in the Scale X and Scale Y property channels. This will cause the animation to start using whatever size the video object happens to be at.</p> |
| <p>Keyframe's Property Value field and Current Value button </p> | <p>When a new keyframe is added to a property channel, its value is automatically set to the current value of the object's property. A different value can be entered in the keyframe's property value field. The acceptable values and settings for the different property values is described in Animated property descriptions, on page 143.</p> <p>This field will be different depending on the property displayed. For example, setting a keyframe for the Opacity property involves editing a value displayed in a text box, while the Visible property involves selecting or clearing a check box that represents either On or Off.</p> <p>The Current Value button  allows you to set the property's value based on the object's value of the scene at the time of the animation's creation.</p> <p>If you would like to set keyframe values from the scene, you could use the Current Value button. For example, if animating an object to move from A to B, you could position the object on the scene at its starting point, select the keyframe for the object's X property, and then click . This automatically populates the field to the left of the button with the value of X. Next, you could drag the object to its finish location on the scene, select the next keyframe, and then click  again.</p> <p>The  button, beside the value entry field, provides an additional way of entering the keyframe's value. Clicking it sets the keyframe value to the current value of the object's property. For example, to set the X coordinate of an object you could move it to the desired spot, select the X keyframe and then click . The value field will populate with the X value.</p> |

| | |
|---|--|
| Interpolation | <p>The Interpolation field allows you to choose the type of interpolation that will be used to compute the values between the selected keyframe and the next keyframe.</p> <p>The possible values are:</p> <ul style="list-style-type: none">• Constant - The value remains constant between the selected keyframe and the next keyframe.• Linear - The value is interpolated linearly between the selected keyframe and the next keyframe.• Hermite - The value is interpolated, using a hermite curve, between the selected keyframe and the next keyframe. The Key Frame Tangent Properties settings offer a simple set of options to control the speed of the interpolation in non-linear mode through presets, but they do not provide full control of the Hermite curves.  <p>The interpolation between two keyframes is set by the right of the first keyframe and left of the second keyframe. The speed of the interpolation on either side of the keyframe is represented by left and right Key Frame Tangent properties.</p> <p>Since the HERMITE option determines the movement between two keyframes, the LEFT parameter is disabled when the first keyframe is selected. Similarly, the RIGHT parameter is disabled for the last keyframe.</p> <p>Note that for some properties such as Image Name or Text, interpolation between keyframes does not make sense, and the interpolation drop box is disabled.</p> |
| Loop to Time Loop to Frame | <p>Depending upon whether the Keyframe Editor is set to Time or Frame, the Loop field will be labelled either as Loop to Time or Loop to Frame. Both behave in a similar fashion, in that when the Loop check box is selected, the animated property settings that belong to the selected keyframe be will repeated, or looped, during playback according to the Loop parameters described below.</p>  <p>The following Loop parameters are displayed when you click the drop-down arrow next to the Loop check box.</p> <ul style="list-style-type: none">• Loop to Frame/Time: This check box is directly linked to and a replication of the Keyframe Editor's Loop check box. It allows you to activate or de-activate the loop function.• Frame: Select the exact frame number where you want the loop to begin from.• Time: Select the exact second where you want the loop to begin from.• Repeat: Allows you to specify how many times the loop will be executed. Choose Number and then the exact number (i.e. 3 means to repeat 3 times) to have the loop be repeated a specific number of times. Otherwise, select Infinite to have the animation loop continuously until the scene is cleared. <p>Note: Adding an End Loop Animation command channel to the primary animation allows you to stop all looping objects in an animation. See Command Channels, on page 119 for more information.</p> |

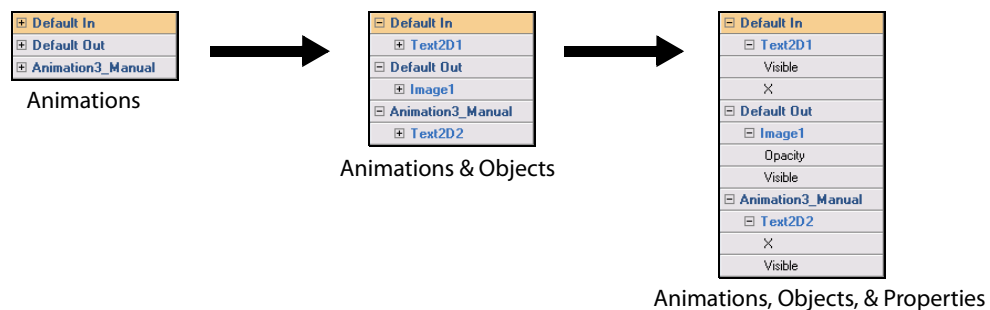
Preview Mode controls

It is often useful to preview the layout of the scene animation during the animation creation process to ensure that the animation is behaving as intended. When used with Xstudio's **Preview** mode, the Animation Editor's Preview Mode controls allow you to play out a preview of the animation on a connected device.

Only the **Play** and **Stop** buttons are currently enabled in Preview Mode. Instructions for publishing and previewing a scene animation are provided in [Previewing and scrubbing the animation](#), on page 136.

Animation Tree and Animations

The Animation Editor helps you create and manage scene animations using an expandable tree view called the Animation Tree. The Animation Tree organizes the main components of the scene's animation into three levels: animations, objects, and properties. Additionally, you can add keyer channels and command channels to the animations in the Animation Tree.



The Animation Tree is equipped with four default animations: Default In, Default Out, Page In and Page Out. As described below, these default animations are automatically triggered to start by calling and clearing the scene/page, respectively. Additional animations can be added to the Animation Tree, however, these animations are called manual animations because they must be explicitly called using the Action Editor's **Animation** action and an associated event.

The following topics provide a more detailed description of the elements belonging or related to the Animation Tree:

- [Default, Page and Manual animations](#), on page 118
- [Keyer Channels](#), on page 119
- [Fader Channel](#), on page 119
- [Command Channels](#), on page 119
- [Animation Tree's Context menu](#), on page 122

Default, Page and Manual animations

In its initial state, the animation tree consists of four (4) animations:

- **Default In:** The Default In animation is intended to be the template's first animated introduction and is automatically triggered the first time a template's scene is taken to air. More precisely, the Default In animation only executes if there is no page immediately preceding the current page (non-consecutive) in the playlist, or if the previous playlist item is based on a different template. Note that if pages based on the same template are taken back-to-back, then the Page In animation will be played, rather than the Default In animation (see below).
- **Default Out:** The Default Out animation is intended to be the template's concluding animation. The Default Out animation is automatically triggered when the scene is cleared from the output device or an additional Take is executed at the end of the playlist.
- **Page In:** The Page In animation is automatically triggered when pages from the same template are taken consecutively (back-to-back). If pages from different templates are taken back-to-back, then the Default In animation will be played.
- **Page Out:** The Page Out animation is automatically triggered when pages from different template are taken back-to-back. However, if the page is the last item in the playlist, the Default Out animation will be triggered, rather than the Page Out, upon a last Take or when the device is cleared.

Note: The use of the Page Out and Default Out animation affects the time at which pages are taken to air. The take of the cued page or the Clear will be delayed until the on air page has sufficient time to play the Page Out and Default Out animations, respectively.

The playout sequence of animations when consecutive (back-to-back) pages are taken to air. In this example, two (2) unique templates (Template1 & Template2) are used to create three (3) pages each. The pages are added to a new playlist and played out in sequential order, starting with Page1 (Template1) and ending with Page3 (Template2).

| | |
|---|--------------|
| Default In - Template1 - Page1 | Take |
| Page Out - Template1 - Page1 Page In - Template1 - Page2 | Take |
| Page Out - Template1 - Page2 Page In - Template1 - Page3 | Take |
| Page Out - Templat1 - Page3 Default In - Template2 - Page1 | Take |
| Page Out - Template2 - Page1 Page In - Template2 - Page2 | Take |
| Page Out - Template2 - Page Page In - Template2- Page3 | Take |
| Default Out - Template2 - Page3 | Take / Clear |

When the playlist is brought to air, the Default In animation is automatically executed. Upon the next take, the first Page Out animation is played out completely followed by the Page In animation for Page2. The Page Out and Page In animations continue upon each

Take command, until Page1 from Template2 is encountered. Since it is not based on the same template as the previously played page, the Default In animation is played. Again the Page Out and Page In animations continue until the last page in the playlist is played out and a Clear command or final Take command is executed. This action triggers the Default Out animation.

Note: If you cue a page before the Take is complete, the Page Out animation may not playout properly.

Another type of animation can also be added to the animation tree called a manual animation. Rather than being triggered automatically by internal events, manual animations must be explicitly called using the Action Editor's **Animation** action and an associated event (see [Choosing a type of animation](#), on page 131).

Note: See [Creating scene animations](#), on page 129 for instructions on how to create and execute animations.

Keyer Channels

A keyer channel is way to start playing a Page or Template from another Page or Template panel. You can drag on pages and templates from the Asset Browser directory onto the timeline to create these keyframes and adjust the duration accordingly.

The keyer channel contains an initial keyframe (white circle) that is empty, but using the Keyframe Editor you can specify at which frame you want the animation to start playing on the keyer, the duration, and which relative layer the animation will playout on.

See [Adding and setting a keyer channel](#), on page 134 for more information.

Fader Channel

The scene animation (Default In, Default Out, Page In or Page Out) can be configured to automatically fade-in and fade-out when they are played. The durations of the fade-in and the fade-out of the animation may be set using the keyframes along a Fader channel within the animation.

When a keyframe is selected along the fader channel, the **Fader** property appears along the top of the Animation Editor.

To fade the scene from opaque to transparent, you must set the first keyframe to a larger Fader value than that in the second keyframe. For example, the object transitions from solid to transparent when the first keyframe is set to 1.0 and the second keyframe is set to 0.0.

Similarly, to fade an object from transparent/translucent to opaque, you must set the first keyframe to a smaller Fader value than the second keyframe.

See [Previewing and scrubbing the animation](#), on page 136 for more information.

Command Channels

Command channels can be added to the scene's animation to control some basic playout tasks from within the animation itself. The majority of these command channels are intended to be used to control the playout of secondary animations from within the primary animation.

The following table provides descriptions for each command channel that is available to be added to the primary animation. See [Adding and setting a command channel](#), on page 134 for precise instructions for how to add and set the command channels in the primary animation.

| | |
|----------------------------------|--|
| Send Callback | Adding a Send Callback channel and setting its keyframe allows you to identify a point during the animation's playout where a flag is sent to the Action Editor as a user defined event, which triggers the execution of an action assigned to that event. |
| Play Animation | Allows you to initiate the start of the playout of a secondary animation at a particular point of a primary animation. The Play Animation command channel is intended to be used when multiple animations are played out together, but with different starting points. |
| Stop Animation | Allows you to stop the playout of a secondary animation at a particular point within the primary animation. The Stop Animation command channel is intended to be used when multiple animations are running together, and you want one to stop before the end of the primary animation. |
| End Loop Animation | Allows you to stop the looping of all the objects in a secondary animation at a specific point during playout of a primary animation. Note that since the End Loop Animation command is applied to all objects in the animation, it is recommended that you create a separate animation for objects that you do not want to be affected by this command. Then, use the Play Animation command channel to play both animations simultaneously. |
| Play Audio | Allows you to execute the playing of an audio file at a specific point during playout of an animation. |
| Set AFD | Allows you to set the Active Format Description (AFD). |
| Ancillary Source Priority | <p>Allows you to dynamically control what ancillary data is written to the output of the Vertigo XG or Intuition XG.</p> <p>The types of ancillary data that can be controlled are VANC and CDP. VANC represents the all available lines of the vertical ancillary data space. CDP represent caption data packets, which will be written on a specific line of the vertical ancillary data space. Ancillary data can be associated with three inputs: CLIP, INPUT A, and INPUT B.</p> <p>The highest priority ancillary data, if it exists, will be written to the output. If the Ancillary Source Priority is not set, the Vertigo XG and Intuition XG defaults to the following ancillary priority assignments:</p> <ul style="list-style-type: none"> • Clip => Priority 3 • Input A => Priority 2 • Input B => Priority 1 <p>Therefore, the clip's ancillary data would be written to the output first. If clip ancillary data does not exist, then Input A's ancillary data will be written to the output. If Input A's ancillary data does not exist, then Input B's ancillary data will be written to the output.</p> <p>When the Ancillary Source Priority is set, the PRIORITY field defaults to 0, but it can be set to values between 0 and 100. A PRIORITY value of 0 means do not include ancillary data in the output, while the rest of the values are relative to the other sources.</p> |

Custom command channels can also be added to enable animation techniques that are exclusively sent to ImageStore master control switchers.

When a keyframe is encountered on the timeline's command channel, the specified command is sent to the ImageStore where it performs the intended task. For example, there are custom commands that allow you to control the ImageStore's graphics keys, and others for controlling the DVEs.

You can preview all of the custom commands available by right-clicking on an object channel in the Animation Tree and selecting **Add Command Channel>Custom**.

- Audio fade rate
- Custom command
- DVE cut
- DVE move
- DVE move Default
- DVE move reverse
- DVE move reverse Default
- DVE reverse Cut
- Duck level voice-over 1
- Duck level voice-over 2
- Fade DSK1 down
- Fade DSK1 up
- Fade DSK2 down
- Fade DSK2 up
- Fade audio to A
- Fade audio to B
- Fade rate DSK1
- Fade rate DSK2
- Fade rate voice-over 1
- Fade rate voice-over 2
- Fade video to A
- Fade video to B
- Fade voice-over 1 off
- Fade voice-over 1 on
- Fade voice-over 2 off
- Fade voice-over 2 on
- HD DVE mode = AB Mixer
- HD DVE mode = AB Over DSK1
- HD DVE mode = Bypass
- HD DVE mode = None
- Load Live DSK1
- Load Live DSK2
- Load live DSK1
- Load live DSK2
- Play audio clip
- Preset level voice-over 1
- Preset level voice-over 2
- SD DVE mode = Behind
- SD DVE mode = None
- SD DVE mode = Reveal MG
- Set voice-over shuffles input channel 1
- Set voice-over shuffles input channel 2
- Set voice-over shuffles input channel 3
- Set voice-over shuffles input channel 4
- Unload Live DSK 1
- Unload Live DSK 2
- Unload live DSK 1
- Unload live DSK 2

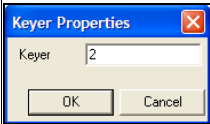
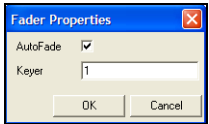
Descriptions of each command are available in the ImageStore user documentation. For instructions using command channels in your scene animations, see [Adding and setting a Custom command channel](#), on page 136.

Animation Tree's Context menu

When you right-click on the animations, objects, or, properties listed in the Animation Tree, a context menu appears providing you with immediate access to commands related to tasks that can be performed with the Animation Tree.

The following table provides a functional description of each command on the Animation Tree's context menu:

| | |
|-----------------------------|--|
| Preview | <p>The Preview command is only available if the context menu was opened by right-clicking an animation in the Animation Tree.</p> <p>When Xstudio is in Preview Mode, this command allows you to start the playout of the scene animation on the output device.</p> <p>An alternative access to this command is the Start button in the Animation Editor's Preview Controls. See Previewing and scrubbing the animation, on page 136 for more information about previewing animations.</p> |
| Rename | <p>The Rename command is only available if the context menu was opened by right-clicking a manual animation in the Animation Tree.</p> <p>When a manual animation is added to the Animation Tree, it is given a generic name like, Animation3. This command allows you to rename a manual animation. See Renaming animations, on page 139 for more information about renaming manual animations.</p> |
| Delete | <p>Deletes items selected in the Animation Tree (manual animation, object, property, keyer, or command channel).</p> <p>Note that you can delete the contents of the Default In/Out and Page In/Out animations, but you cannot delete, or remove, these animations from the Animation Tree. See Deleting channels from the Animation Tree, on page 139 for more details.</p> |
| Select All Keyframes | <p>This command allows you to quickly select all of the keyframes that belong to the property, keyer, or command channel that is selected in the Animation Tree. Keyframes turn yellow when they are selected.</p> <p>Select all keyframes is useful if you want to copy all the keyframes to another channel or delete all your keyframes from a channel.</p> <p>This command is not available for animation and object channels.</p> |
| Cut | <p>The Cut command allows you to remove the selected item from its current location in the Animation Tree and stores it on the system's clipboard until it is pasted into a new location using the Paste command.</p> <p>Note that the only the contents of the Default In/Out and Page In/Out animations can be cut, not the animations themselves.</p> |

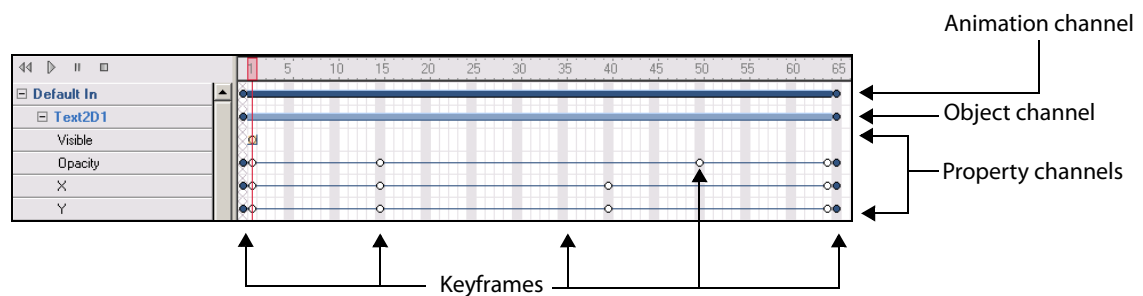
| | |
|-------------------------------------|---|
| <p>Copy</p> | <p>The Copy command allows you to duplicate the selected item in the Animation Tree and stores it on the system's clipboard until it is pasted into a new location using the Paste or Paste Generic Objects commands.</p> <p>This command is very useful when you want to create multiple animations that are similar (same objects and or properties), but require subtle differences. Copying and pasting avoids having to recreate everything from scratch.</p> <p>Note that if you copy and paste the contents of the Default In/Out and Page In/Out animations, they become manual animations when pasted.</p> |
| <p>Paste</p> | <p>The Paste command allows you to place the items that are currently on the system's clipboard to the selected location on the Animation Tree.</p> |
| <p>Properties</p> | <p>The Properties command is only available if the context menu was opened by right-clicking a keyer or fader channel in the Animation Tree.</p> <p>This command opens the Properties dialog box that allows you to select the exact keyer the animation will payout on by setting the Keyer property with the keyer's number. For Fader channels, the Properties dialog box is also where you enable/disable the AutoFade property.</p> <div style="display: flex; justify-content: space-around; align-items: center;">   </div> |
| <p>Paste Generic Objects</p> | <p>This command allows you to paste an existing object animation in a scene animation as a generic placeholder.</p> <p>Since the same object animation cannot exist twice in a scene animation, this command allows you to paste an existing object animation in the scene animation as a generic placeholder. You can then assign an object of the same type from the Scene Tree to the generic object animation by dragging the desired object from the Scene Tree onto the generic object. The object from the Scene Tree replaces the generic object in the scene animation and its animated properties are now identical to the other object animation.</p> <p>See Displaying the animated properties in the Scene tree, on page 130 for more information.</p> |
| <p>Add Keyer Channel</p> | <p>Adds a keyer channel to the animation selected in the Animation Tree. See Keyer Channels, on page 119 and Adding and setting a keyer channel, on page 134 for more information.</p> |
| <p>Add Fader Channel</p> | <p>Adds a fader channel to the animation selected in the Animation Tree. See Fader Channel, on page 119 and for more information.</p> |
| <p>Add Command Channel</p> | <p>Adds a command channel to the animation selected in the Animation Tree. See Command Channels, on page 119 and Adding and setting a keyer channel, on page 134 for more information.</p> |

Animation Timeline and Keyframes

The Animation Editor displays each animation, including its objects and object properties, as channels on a timeline. Adding keyframes to the property channels allows you to use the Keyframe Editor to specify a change to the settings of the property at particular point during the scene's playout. You can then preview the playout of the scene animation directly on the scene's canvas using the Timeline's slider to scrub the animation.

The following topics provide a more detailed description of the elements belonging or related to the Animation Editor's Timeline:

- [Types of channel lines](#), on page 124
- [Keyframes](#), on page 126
- [The Timeline's Ruler and Slider](#), on page 127
- [Timeline's Context menu](#), on page 128



Types of channel lines

The Animation Editor's timeline displays and tracks each animation, including its objects, object properties, keyer channels, and command channels as horizontal channel lines on the timeline. The following table describes each of the channels that may appear on the Animation Editor's Timeline.

| | |
|--------------------------|--|
| Animation channel | <p>The animation channel is the top-level channel in the Animation Tree and it is represented on the timeline by a thick dark blue bar.</p> <p>As the top-level channel, animation channels can contain objects with property channels, command channels and keyer and fader channels. The length of the animation channel on the timeline represents the duration of the animation's playout. Therefore its length is often determined by the longest object property within the animation. The animation channels themselves can be resized, but since they are the top-level channel, all channels contain within it will also be resized (see Resizing and moving channels on the Timeline, on page 142).</p> <p>Also note that keyframes cannot be added to animation channels.</p> |
| Object channel | <p>Each object that is added to the animation channel is represented on the timeline as an object channel (indicated by a medium blue bar). The name of the object channel is the same as the object's name on the scene, not the asset's name.</p> <p>Object channels are used to move and group object properties all at once (see Resizing and moving channels on the Timeline, on page 142).</p> <p>Also note that keyframes cannot be added to animation channels.</p> |

| | |
|-------------------------|---|
| Property channel | <p>Property channels on the timeline represent the object's properties that will be animated during the playout of the scene. The property channels appear as light blue blocks or thin blue lines on the timeline.</p> <p>The property channel contains an initial keyframe (white circle) that represents the current, or beginning, settings for that object property, which is displayed in the Keyframe Editor fields when the keyframe is selected.</p> <p>You can add additional keyframes, which allows you to specify a new setting for the property at that specific point during the animation. Depending on the property, you may be able to choose the interpolation, or transition, from one keyframe setting to the next keyframe. See Keyframes, on page 126 and Adding and setting keyframes, on page 132 for more information.</p> |
| Keyer channel | <p>Keyer channels on the timeline allows you drag a template/page onto the keyer and specify when it is to playout during the animation.</p> <p>The keyer channel contains an initial keyframe (white circle) that is empty, but using the Keyframe Editor you can specify at which frame you want the animation to start playing on the keyer, the duration, and which relative layer the animation will playout on.</p> <p>When you set the duration parameter for keyer channel's keyframe, a thick pink line appears along the keyer channel to indicate the length of time that the playout will occur on the keyer. See Keyer Channels, on page 119 for more information.</p> |
| Fader channel | <p>Fader channels on the timeline allow you to configure the scene to automatically fade-in and fade-out when it is played. The durations of the fade-in and the fade-out of the animation may be set manually using the keyframes along a Fader channel within the animation or by enabling the AutoFade property, which allows Xbuilder to modify and control the Fade In/Out behavior of the scene.</p> <p>For the Fader channel to be used by Xbuilder, you simply have to add the Fader channel to the animation and enable the AutoFade property in the channel's Properties dialog box (right-click on the Fader channel in the animation tree and select Properties). You do not need to add any additional keyframes to the Fader channel.</p> <p>To have full manual control over the fading in and out of a scene, add the Fader channel to the animation. The Fader channel contains an initial keyframe whose Fader property is set to transparent (0.0) by default. Add other keyframes to the channel and set Fader property. To fade the scene from opaque to transparent, you must set the first keyframe to a larger Fader value than that in the second keyframe. For example, the object transitions from solid to transparent when the first keyframe is set to 1.0 and the second keyframe is set to 0.0. Similarly, to fade an object from transparent/translucent to opaque, you must set the first keyframe to a smaller Fader value than the second keyframe.</p> <p>When you set the Fader parameter for Fader channel's keyframe, a thick blue line appears along the channel to indicate the length of time that the fade will occur on the keyer.</p> |

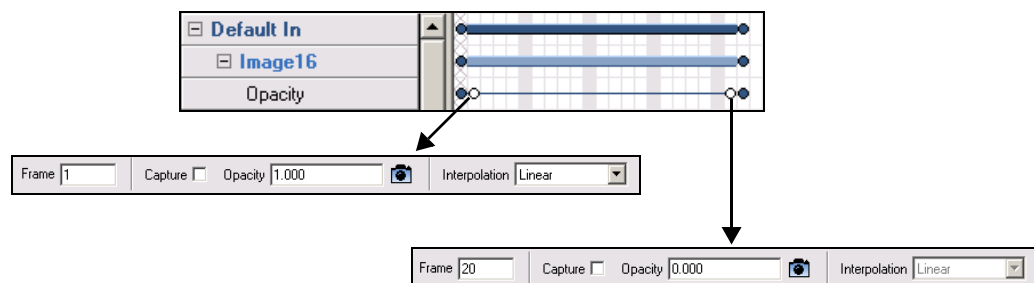
| | |
|------------------------|---|
| Command channel | <p>Command channels on the timeline allow you to embed commands within the animation that provide additional functionality, like starting the playout of a secondary animation or introducing an audio playout. Initially a command channel is represented on the timeline by a single keyframe in frame 1. However, once another keyframe is added to the same command channel, a thin blue line joins the keyframes along the channel.</p> <p>Command channels are considered to be at the same hierarchical level in the Animation Tree and Timeline as an object channel. Therefore, it can be resized independently, but it is also linked to any size changes applied to the animation channel. See Command Channels, on page 119 for more information.</p> |
|------------------------|---|

Keyframes

A keyframe is a snapshot of an object's property settings at a particular frame during the animation. The Animation Editor uses keyframes to change an object's properties over time to create an animated effect. The idea is to create sets of keyframes that define the starting and ending points of the desired effect. Then when the animation is played, the transition from one keyframe state to the other is executed. Depending on the property or the interpolation, some transitions might take effect immediately, while others might occur as gradual changes in values between keyframes (interpolation).

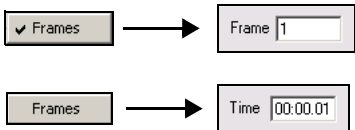
When a property channel is added to the animation's timeline, an initial keyframe (white circle) is automatically added to the first frame, and reports the current, or beginning, settings for that object property in the Keyframe Editor fields. You must then add additional keyframes to the channel and set their values to create the desired effect (see [Adding and setting keyframes](#), on page 132).

The graphic below demonstrates the use of two keyframes on an object's Opacity property channel that are set to produce the effect of fading out the object from being completely opaque (1.0) to completely transparent (0.0). The first keyframe's Interpolation field is set to Linear, which means that the fading effect will occur evenly over the duration between the two keyframes, starting at frame 1 and finishing at frame 20.



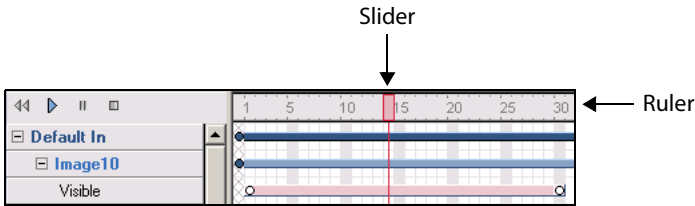
The Timeline's Ruler and Slider

The ruler above the Animation Editor's Timeline allows you to visually determine the duration of timeline channels, or the precise location of a keyframe on a property channel. The units that the ruler displays can be set to either **Frames** or **Time**. As demonstrated in the images below, you can toggle back and forth between **Frame** and **Time** when assigning keyframes by right-clicking on **Frame** or **Time** field, then selecting or deselecting **Frames** from the context menu.



The relationship between the frame number and time comes from the **Video Format** property of the scene. For instance, if the NTSC format is chosen, there are 30 frames per second.

A slider is also available on the ruler, which can be dragged along the Timeline. The slider's main function is to scrub (preview) the behavior of certain animated properties directly on the Scene window (see [Previewing and scrubbing the animation](#), on page 136). You can also use the slider as a guide to help you identify precise cells within the timeline, by visually finding the intersection of a channel and a frame.



Another important use of the slider is that key frames are added where it is positioned. For instance, if the slider is moved to frame 30 and then a new property channel is dragged (or add a new command channel or keyer or fader channel) onto the timeline, then the keyframe added is positioned at frame 30. This is useful as you don't want your first keyframe to start at 1.

Timeline's Context menu

When you right-click on a channel or keyframe on the Animation Editor's Timeline, a context menu appears providing you with immediate access to commands related editing the channel selected.

The following table provides a functional description of each command on the Timeline's context menu:

| | |
|------------------------------|---|
| Insert Keyframe | Inserts a keyframe at the exact frame/time location selected on the channel. This command is only available when the context menu is accessed on a frame/cel, keyer channel, or command channel. |
| Remove Keyframe | Removes the selected keyframe from the channel. |
| Select All Keyframes | This command allows you to quickly select all of the keyframes that belong to the property, keyer, or command channel that is selected on the Timeline. Keyframes turn yellow when they are selected. This command is not available for animation and object channels. |
| Cut | The Cut command allows you to remove the selected item from its current location in the Timeline and stores it on the system's clipboard until it is pasted into a new location using the Paste command. |
| Copy | The Copy command allows you to duplicate the selected item on the Timeline and stores it on the system's clipboard until it is pasted into a new location using the Paste or Paste Generic Objects commands. |
| Paste | The Paste command allows you to place the items that are currently on the system's clipboard to the selected location on the Timeline. |
| Paste Generic Objects | This command allows you to paste an existing object animation in a scene animation as a generic placeholder. Since the same object animation cannot exist twice in a scene animation, this command allows you to paste an existing object animation in the scene animation as a generic placeholder. You can then assign an object of the same type from the Scene Tree to the generic object animation by dragging the desired object from the Scene Tree onto the generic object. The object from the Scene Tree replaces the generic object in the scene animation and its animated properties are now identical to the other object animation. See Cutting, copying, and pasting animation elements , on page 140 for more information. |

Creating scene animations

The following table identifies the high-level steps involved in creating a new scene animation. Subsequent sections (identified by the links within the table) provide step-by-step instructions for performing each step. It is recommended that you first read through the entire procedure first using the table, and then perform each step by following the instructions in each section.

| Step | Task description |
|------|--|
| 1. | Build the scene by adding objects to the scene's canvas (i.e. primitives or assets). See Displaying the animated properties in the Scene tree , on page 130. |
| 2. | Open the scene tree and verify that the Animated properties are displayed. See Displaying the animated properties in the Scene tree , on page 130. |
| 3. | Open the animation editor. See Opening the Animation Editor , on page 130. |
| 4. | Choose whether you want to create a Default In, Default Out, Page In, Page Out or a manual animation. See Choosing a type of animation , on page 131. |
| 5. | Add objects and the object's animated properties to the animation. See Adding the object's animated properties to your animation , on page 131. |
| 6. | Add keyframes to the Property channels and set the keyframe's properties. See Adding and setting keyframes , on page 132. |
| 7. | Optional: Add and set keyers, fader and/or command channels. See Adding and setting a keyer channel , on page 134, Add Fader Channel , on page 123, or Adding and setting a command channel , on page 134. |
| 8. | Save and publish the template/scene. See Saving and publishing the template/scene , on page 136. |
| 9. | Preview the behavior and/or layout of the animation. See Previewing and scrubbing the animation , on page 136. |
| 10. | Optional: Set up an Animation action and an associated event to call and execute a manual animation. See Setting up an animation action , on page 137. |

Note: Although the above procedure guides you through the creation of a new scene animation, building a scene animation can be iterative process. As such, you may be required to return to steps 5 to 9 several times to edit the animation's settings before attaining the desired look and behavior.

Displaying the animated properties in the Scene tree

To build a scene animation you need to add specific scene objects and properties to the Animator Editor. As a preparatory step, you must first open the canvas' Scene Tree panel and display the Animated Properties for each scene object.

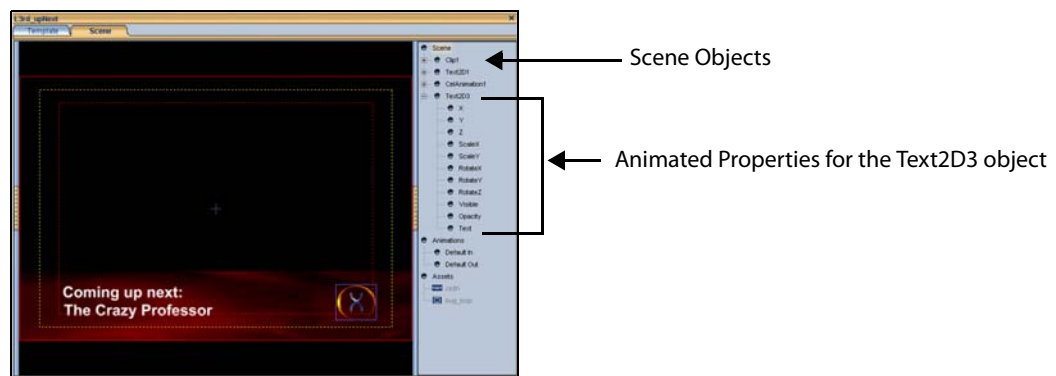
To open the Scene Tree and display the Animated Properties:

- 1 Open the Scene Tree panel by clicking the gripper bar on the right-hand side of the screen.

When the panel expands, the Scene Tree contains (at a minimum) the Scene category, which lists all of the objects that are currently placed on the scene. Additional categories, like Animations and Assets may also be displayed on the Scene Tree.

- 2 Display the animatable properties that belong to each object listed in the scene category by right-clicking inside the Scene Tree panel and selecting (checking) the **Animated Properties** command from the context menu.

Each object's name in the Scene Tree will now have a + sign in front of it. (Clicking the + sign exposes the object's animatable properties, which are the only properties that can be used to create animation effects.



Opening the Animation Editor

Building an animated scene occurs within the Animation Editor, which is an Xstudio tool located within a collapsible panel at the bottom of the Xstudio interface in Design mode.

Since the Animation Editor is used only to animate objects on scenes, it is only available when the Scene tab is selected. When the control-side is selected, the Action Editor appears in this collapsible panel.

To open the Animation Editor:

- With the scene-side open (the scene tab selected), click the center of the gripper bar the bottom of the screen.

Or,

- Press **Ctrl + E** on your keyboard.


Note: You can resize the Animation Editor by dragging the gripper's blue boundary along the top of the Animation Editor until it is the desired height.

Choosing a type of animation

When animating a scene, you have a choice from five types of animations: Default In, Default Out, Page In, Page Out and manual animations. See [Default, Page and Manual animations](#), on page 118 for description of each of these animations.

If you want to create a Default In/Out or a Page In/Out animation, continue to step 5 ([Adding the object's animated properties to your animation](#)). However, if you want to create a manual animation, you must first add a new animation to the Animation Tree.

To open and add a new manual animation to the Animation Editor's Animation Tree:

- Click the **New** button  in the Animation Editor toolbar.

A generically named animation (i.e. animation3) is added to the Animation Tree.

Note: At this point, you may consider performing some tasks associated with managing the Animation Tree, like renaming the manual animation, changing the order of the animation listing...etc. See [Editing existing animations](#), on page 139 for more details.

Adding the object's animated properties to your animation

At this stage, you will start building your scene animation by choosing from the Scene Tree the objects and their animated properties that you want to animate and then copying them into your animation in the Animation Editor.

To add an object's animated property to an animation:

- 1 In Scene Tree, click the + sign preceding the object's name to display the animatable properties for the object that you want to animate on your scene. For a description of the animated properties available to each type of object, see [Animated property descriptions](#), on page 143.
- 2 In the Scene Tree, select the property to be animated and drag it onto your animation in the Animation Editor's Animation Tree (i.e Default In, Default Out, Page In, Page Out or Animation#).

The object and its property is added to the animation tree and an animation, object, and property channel are added to the Editor's Timeline area. Each property that is added will have one keyframe (white circle) automatically added to its property channel on the Timeline.

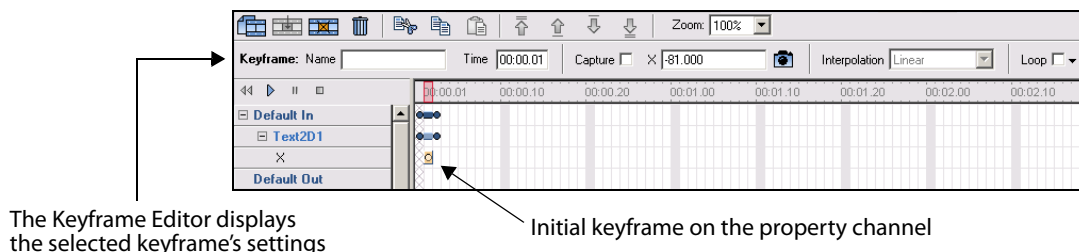
- 3 Optional: Add more objects and properties to the animation.
 - If you want to add additional animated properties to that same object in the animation, repeat step 2 above.
 - If you want to add another object and its properties to the animation, then repeat steps 1 and 2 above.

Adding and setting keyframes

Changing the property settings at two points (keyframes) along the property channel creates animated visual effects. Therefore, use the Keyframe Editor to either display the current settings for the object property at the selected keyframe location, or edit the property's values to transition the object's properties from one setting state to another.


See [Animated property descriptions](#), on page 143 for more information about each property's settings.

The following instructions describe how to add keyframes to a property channel and how to use the Keyframe Editor to set the keyframe values.



Note: Be sure to read [Animation Timeline and Keyframes](#), on page 124 familiarize yourself with the Animation Editor's Timeline before attempting the following procedure.

To add a keyframe to an animation:


- 1 Using the property channel and the Timeline's ruler as reference guides, click on a cell (frame) where you will want the setting of that property to change or become a particular value. For example, at frame 20 you want the **X** animated property to be -81.
- 2 Add a keyframe to the selected location:
 - Double-click on the cel/frame.
 - Or,
 - Right-click on the selected cell and select **Insert Keyframe** command from the context menu.
 - Or,
 - Click the **Insert Keyframe** button  in the Animation Editor toolbar.

A keyframe (circle) is added and the property channel extends to the furthest keyframe on that property channel.

If the keyframe's location is further along the timeline than the animation or object channels, then the object and animation channels will extend to that length as well.

Note: The blue circles that are located at each end of the channels (animation, object, or property) are handles that you can drag to either extend or shorten the length of the channel. Similarly, the keyframe circles can be moved to a new frame by dragging them along the property channel.

Using the Keyframe Editor to specify a keyframe setting:

- 1 Select the keyframe on the property channel.
The keyframe turns yellow indicating that it is selected and the keyframe's current settings are displayed in the Keyframe Editor's fields.
- 2 Optional: assign a name to the keyframe if the keyframe will be accessed using a **SetKeyFrame** action.
To name the keyframe, type the name in the **Keyframe Name** field located in the Keyframe Editor and press **Enter**.
- 3 Optional: change the location of the keyframe on the property channel.
The location of the keyframe on the timeline is displayed either as a frame number or a time, depending on the Keyframe Editor's setting (see [Keyframe Editor](#), on page 113). If you want to change the keyframe's location on the timeline, you can type the frame number or time directly in this field and press **Enter**, or drag the keyframe along the property channel.
- 4 Set the keyframe values:
 - Use the **Capture** check box to set the property's value from a specific point in time on the output device (see **Capture**, on page 115 for more information).
Or,
 - Depending on the property, either type the value directly in the text box (if you know the exact value desired), or select/clear the field's check box (see **Keyframe's Property Value field and Current Value button**, on page 115 for more information).
Or,
 - Use the **Capture** button  to set keyframe values from the scene (see **Keyframe's Property Value field and Current Value button**, on page 115 for more information).
- 5 Set the **Interpolation** field, which determines how the animation will transition from one keyframe to the next.
Select from the **Interpolation** drop-down list: **Constant**, **Linear**, or **Hermite** (see [Keyframe Editor](#), on page 113 for more details).
- 6 Optional: Set the **Loop** parameters.
Enabling and setting the Loop parameters allows you to repeat the property's behavior that led up to the key frame starting at a particular point and for as many times as specified (see [Keyframe Editor](#), on page 113 for more details).

Adding and setting a keyer channel

Adding and setting a keyer channel to the animation allows you to overlay other templates/pages on other keyers over the timeline frame accurately.

It is important to note that the Keyer property of a Keyer channel within the animation is a relative keyer. This means that the value entered, does not specify an absolute keyer on the device, but an offset from the keyer on which the timeline is loaded. For example, if a timeline with two keyer channels has its keyer property values set to 1 and 2 and the timeline is loaded on keyer 7, then the keyer channels use keyers 8 and 9.

To add and set a keyer channel to the animation:

- 1 Right-click the animation's name in the Animation Tree to display the context menu.
- 2 Select the **Add Keyer Channel** command.
The keyer channel is added to the animation in the Animation Tree and Timeline.
- 3 Open the **Keyer Properties** dialog box by right-clicking the keyer channel's name in the Animation Tree and select the **Properties** command from the context menu.
- 4 Set the keyer value in the **Keyer Properties** dialog box. As discussed above, the keyer number that is typed in the Keyer Properties dialog box is relative to the keyer the timeline is loaded in. For example, if you set the keyer channel to 1, if the timeline is loaded onto keyer 6, the template in the keyer channel will be loaded onto keyer 7 (6 + 1).
- 5 Select the keyer channel's keyframe and set its properties:
 - a Drag the template or page that is to be layered above the animation scene from the Asset Browser to the Keyer field in the Keyframe Editor toolbar. The template's location now appears in this field.
 - b Set the length of time (frames/time) for which the template will be layered over the animation by entering a value in the Keyframe Editor's **Duration** field.
 - c Set the precise starting point at which the keyer will layer the specified template over the animation in the **Time/Frame** field on the Keyframe Editor's toolbar.

Adding and setting a command channel

Adding a command channel to the scene's animation allows you to control some basic playout tasks from within the animation itself using keyframe settings. See [Command Channels](#), on page 119 for a description of each of the available command channels.

The following set of instructions describe generically how to add and set a command channel for a scene animation. The procedure for setting the Send Callback command channel and for Custom command channels is slightly different from the others and is described in the following sections:

- [Adding and setting a Send Callback command channel](#), on page 135
- [Adding and setting a Custom command channel](#), on page 136

To add a command channel to the animation:

- 1 Right-click on the animation's name in the Animation Tree to display the context menu.
- 2 Select **Add Command Channel** and then one of the following commands:
 - Send Callback

- Play Animation
- Stop Animation
- End Loop Animation
- Play Audio
- Set AFD
- Ancillary Source Priority
- Custom (and then select a custom command from the list)

The command channel is added to the animation in the Animation Tree and Timeline. As well, an initial keyframe is added to channel at the slider bar position.

- 3 Select the channel's keyframe and set its parameters in the Keyframe Editor:
 - a Set the Keyframe's Property Value field (i.e. Send Callback, Play Animation...etc.) by typing the name of the animation in the Keyframe's Property Value field, or by dragging the animation's name from the Scene Tree into the Keyframe Property field.
 - b In the Frame/Time field, specify exact frame or time instance when you want the keyframe to be first encountered on the timeline and the command's setting will be applied.

Adding and setting a Send Callback command channel

Adding a Send Callback channel and setting its keyframe allows you to identify a point during the animation's ployout where a flag is sent to the Action Editor as a user defined event, which triggers the execution of an action assigned to that event.

To add a Send Callback command channel to the animation:

- 1 Right-click on the animation's name in the Animation Tree to display the context menu.
- 2 Select the **Add Command Channel>Send Callback** command.

The Send Callback channel is added to the animation in the Animation Tree and Timeline. As well, an initial keyframe has been added to channel at the slider bar position on the Send Callback channel.
- 3 Select the Send Callback channel's keyframe and set the keyframe parameters in the Keyframe Editor:
 - a Type the name of the callback in the Send Callback field.
 - b Type in the Frame/Time field instance of when you want the callback to be sent to the Action Editor.
- 4 Select the Template tab on the Xstudio window to activate the control-side of the template.
- 5 Add a User Defined event to the Action Editor and give it the same name that you used in the Send Callback field in the Animation Editor (step 3A). The event is added to the Events list.
- 6 Select the event that was just added and add an action to the event.
- 7 Set the Action's parameters.

Adding and setting a Custom command channel

Custom command channels allow you to embed command calls within the animation that are exclusively sent to ImageStore master control switchers.

When a keyframe is encountered on the timeline's command channel, the specified command is sent to the ImageStore where it performs the intended task. For example, there are custom commands that allow you to control the ImageStore's graphics keyers, and others for controlling the DVEs.

To add and set a custom command channel to an animation:

- 1 Right-click on the animation's name in the Animation Tree to display the context menu.
- 2 Select the **Add Command Channel>Custom** command. Select the desired command from the extensive list of ImageStore commands that appears.
The command channel is immediately added to the Timeline with an initial keyframe.
- 3 Select the command channel's keyframe. Note that the command channel's field is automatically updated in the Keyframe Editor with the appropriate value.
- 4 If the command has parameters, a **Parameters** button appears in the Keyframe Editor. Select the **Parameters** button to expose and set the command's parameters.

Saving and publishing the template/scene

Although it is not completely necessary, it is a good habit to save and publish the scene at this point in the procedure as it will prevent your work on the animation from being accidentally lost or altered.

To save and publish the template and scene:

- 1 Press the **Publish** icon on the Xstudio interface.
The **Save Changes** dialog box appears asking you if you want to save.
- 2 Click **Yes**.
The **Save Template** dialog box appears.
- 3 Fill in the **Asset Information**, **Categories**, and **Keywords** sections.
At a minimum, type a name for the template and select **Templates** as the category where to store the template & scene.
- 4 Press **OK**.
Xstudio automatically begins publishing the template and scene as indicated by the progress bar and message in the status bar at the top of the Xstudio interface.

Previewing and scrubbing the animation

Xstudio and the Animation Editor provide you with a preview mode that allow you to view the playout of the animated scene on an output device without having to switch to Production mode.

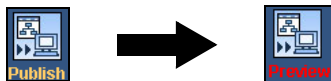
The Animation Editor also allows you to preview the behavior of the animated objects directly on the scene's canvas through a process referred to as scrubbing the animation. By moving and stopping the timeline's slider along the animation, the scene's canvas is updated to reflect some of the animation's objects' settings at that particular point in the animation.



Since the Scene window cannot perform 3-dimensional rendering, scrubbing the animation is limited to only displaying the behavior of 2-dimension animated properties. Only the following animated properties can be previewed by scrubbing the animation:

- X, Y
- Scale X, Y
- Visible (only if the **Transition** field is set to **Cut**)
- Opacity
- ImageName
- Text

To preview a scene animation on an output device:

- 1 Check that the output device is connected to Xstudio by verifying that the device's green light is illuminated in the upper right corner of the Xstudio interface.
- 2 Switch Xstudio to Preview mode by holding down the Control (**CTRL**) key while clicking the **Publish** icon. The Publish icon's heading turns red and now reads "Preview". The scene's Default In animation should begin to play out automatically on the output device.



- 3 To play any of the animations on the output device, select the animation's name in the Animation Tree and then click the **Play** button  located above the Animation Tree, or right-click on the animation's name and select the **Preview** command from the context menu.
- 4 You can stop the playout of an animated scene at any time by clicking the **Stop** button  located above the Animation Tree.

To scrub an animation using the Animation Editor's Timeline slider:

- 1 Select the animation's name in the Animation Tree.
- 2 Drag the Timeline's slider along the Timeline's ruler and release it on the frame/instance where you want to see the state of the animated scene object.

The scene's canvas updates the scene objects to represent the object property settings at that point in the animation playout.

Note: Objects that have an Opacity setting of zero (0.0) are identified on the scene by a colored placeholder box.

Setting up an animation action

You can then use the Action Editor to call a specific animation based on a playout event.

If you chose in step 4 to create a manual animation, you must use the Action Editor to create an **Animation** action to start the animation when a specific event is encountered.

To set up an Animation action:

- 1 Click the **Panel** or **Template** tab (depending on whether you're working on a control panel or template). If you had the Animation Editor displayed, it automatically switches to the Action Editor.

- 2 Create a object on the control-side canvas, like a button by dragging a button primitive onto the canvas.
- 3 With the new object selected, choose the trigger event in the Action Editor. In the example below, the On Click event is being used.
- 4 From the Action drop-down list select Animation.
- 5 Under Action Parameters select Play from the Type drop-down list and then enter the animation name in the Animation field.
- 6 Publish the scene, switch to Production mode, and use the control object to play the animation.

Editing existing animations

Building a scene animation can be an iterative process, in that you may have to edit the animation's settings and preview that animation's playout several times before attaining the desired look and behavior. The following sections provide instructions for performing common editing tasks that are required after the initial animated scene has been created and saved.

- [Renaming animations](#), on page 139
- [Deleting channels from the Animation Tree](#), on page 139
- [Cutting, copying, and pasting animation elements](#), on page 140
- [Resizing and moving channels on the Timeline](#), on page 142
- [Moving a keyframe](#), on page 142
- [Deleting a keyframe](#), on page 142

Renaming animations

While the Default In/Out and Page In/Out animations cannot be renamed, you can personalize the names of the manual animations that are added to the Animation Tree. When they are added, manual animations are given a generic name (i.e. Animation3). You can either rename the animation immediately when it is added, or you can access the **Rename** command later by right-clicking on the animation's name and choosing **Rename** from the context menu (see [Animation Tree's Context menu](#), on page 122).

Note: The Default In/Out and Page In/Out animations cannot be renamed. Similarly, object, property, and command channels cannot be renamed. However, keyer channels can be renamed using the **Rename** command in the Animation Tree's context menu.

Deleting channels from the Animation Tree

While the contents of the Default In/Out and Page In/Out animations can be deleted, the animations themselves cannot be deleted from the Animation Editor. Manual animations, however, can be deleted from the Animation Editor by selecting the animation in the Animation Tree and then selecting the **Delete** command from the Animation Tree's context menu, or by clicking the **Delete** key on your keyboard.

Similarly, any object, property, keyer, or command channel listed in the Animation Tree can be deleted, by selecting the channel in the Animation Tree or on the Timeline and then selecting the **Delete** command from the context menu, or by clicking the **Delete** key on your keyboard.

Note: To immediately restore the deleted a channel that was mistakenly deleted, select the **Edit>Undo** command from the Xstudio menu, or press **Ctrl + Z** on your keyboard.

Cutting, copying, and pasting animation elements

The experience of creating and editing animations can be more efficient by using common editing commands like copy, cut, and paste to duplicate or move elements within an animation or to another animation.

Since the cut, copy, and paste commands can only be used among the animation elements of the exact same type, the following sections describe how and when to use the cut, copy, and paste commands as they apply to the specific elements in the Animation Editor.

- [Cutting, copying, and pasting animation channels](#), on page 140
- [Cutting, copying, and pasting object channels](#), on page 140
- [Cutting, copying, and pasting property channels](#), on page 141
- [Cutting, copying, and pasting keyframes](#), on page 141

Cutting, copying, and pasting animation channels

The **Cut** command allows you to remove (cut) a manual animation from the Animation Tree and places the contents of the manual animation on the clipboard, where it can be added (paste) to another location on the Animation Tree. On a similar note, using the **Cut** command on one of the default animations (Default In/Out or Page In/Out) does not remove the animation from the Animation Tree, but places its contents on the clipboard, where it can be added (paste) to the Animation Tree as a new manual animation.

Unlike the **Cut** command, the **Copy** command does not remove the animation from the Animation Tree. Instead, it duplicates (copies) the animation and its contents and places them on the clipboard, until it is added (pasted) to the Animation Tree as a new animation. Using the Copy command on a default animation does not empty the contents of the default animation, like the Cut command does. Rather, the copy command duplicates (copies) the contents of the default animation and places them on the clipboard until it is added (pasted) to the Animation Tree as a new animation.

When animations are added to the Animation Tree using the Cut, Copy, and Paste commands, they are often given a generic name, like Animation3. See [Renaming animations](#), on page 139 for instructions on how to rename the new animation.

Cutting, copying, and pasting object channels

The **Cut** command allows you to remove (cut) an object channel from one animation and add (paste) it into another animation using the **Paste** command. Similarly, the **Copy** command allows you to duplicate (copy) an object channel from one animation and add (paste) it into another animation. In both cases, the copied/cut object channel retains all of its characteristics (i.e. property channels and keyframe settings).

One caveat regarding using the Copy and Paste commands is that the same object channel cannot exist twice in the same animation. In other words, you cannot use the **Paste** command to add an identical object channel to the same animation. However, the **Paste Generic Objects** command does provide a solution to this situation. The **Paste Generic Objects** command allows you to paste an existing object channel in the same animation as a generic placeholder. You can then assign an object of the same type from the Scene Tree to the generic object animation by dragging the desired object from the Scene Tree onto the generic object. The object from the Scene Tree replaces the generic object in the scene animation and its animated properties are now identical to the other object animation.

The **Paste Generic Objects** command is also quite useful for copying object animations between scenes since you can paste an object animation from one scene to another and then associate it with an object of the same type.

When the copied/cut property channel is pasted to its new location, if there is a conflict between the pasted channel and an existing channel in the animation (i.e. the same type of property channel already exists in the object animation where you are trying to paste another one), the **Channel Conflict** dialog box appears and asks if you want to overwrite the existing channels. Click **Yes** to apply the new property channel, or click **No** to retain the old property channel.

Cutting, copying, and pasting property channels

The **Cut** command allows you to remove (cut) a property from an object in an animation and add (paste) it into the same type of object in the same or a different animation. Similarly, the **Copy** command allows you to reproduce (copy) a property from an object in an animation and add (paste) it into an object in the same or a different animation. In both cases, the property channel retains all of its characteristics (i.e. keyframe settings).

When the copied/cut property channel is pasted to its new location, if there is a conflict between the pasted channel and an existing channel in the animation (i.e. the same type of property channel already exists in the object animation where you are trying to paste another one), the **Channel Conflict** dialog box appears and asks if you want to overwrite the existing channels. Click **Yes** to apply the new property channel, or click **No** to retain the old property channel.

The copy and paste commands offer a practical way of instantly adding an object to another animation that includes all of the properties and settings of an existing object. This saves you the time and effort required to build up a similar object channel.

Cutting, copying, and pasting keyframes

Rather than going through the whole process of adding and setting a new keyframe on a channel, you can copy keyframes from one channel on an animation and then paste them on an identical channel in another animation. Since keyframes are set to exist on a specific frame on the channel using the Frame/Time field, when you paste the keyframe onto another channel, the keyframe will appear at the same frame location. See [Moving a keyframe](#), on page 142 if you would like to move the keyframe to another frame location after it has been pasted onto the channel.

When the copied/cut keyframe is pasted to its new location, if there is a conflict between two keyframes (i.e. a keyframe already exists in the frame where you are trying to paste another one), the **Frame Conflict** dialog box appears and asks if you want to overwrite the existing channels. Click **Yes** to apply the new keyframe(s), or click **No** to retain the old keyframe(s).

Resizing and moving channels on the Timeline

Each channel on the Timeline has a set of handles (big blue dots) at either end of the channel. Dragging the handles to the right or left allows you to resize (extend or shorten) any channel on the Animation Editor's Timeline.

Since the channels are organized in a tree structure in the Animation Tree, there are certain hierarchical dependencies that must be respected. Therefore, when a high-level channel (i.e. animation or object) is resized on the Timeline, the lower-level channels (i.e. property channels) associated to the high-level channel are all stretched by the same amount as the higher-level channel. For example, if the object channel is lengthened by ten frames, each of the property channels will also be lengthened by ten frames.

Similarly, when a low-level channel (i.e. property channel) is extended further than the high-level channels, the high-level channels extend to the same length as the lower-level. This ensures, for example, that a property channel does not extend beyond the playout of the animation channel.

To move any of the channels completely to a new location along the Timeline, you must click anywhere on the channel line where there are no keyframes and drag left or right. The lower-level channels move with the channel.

Moving a keyframe

Keyframes can be moved to a new location on the channel by dragging the keyframe to the left or right along the Timeline, or by selecting the keyframe and then setting a new value in the Frame/Time field in the Keyframe Editor.

Also note that the frame location of a keyframe can change as a result of the entire channel being moved or resized. In such cases, the keyframes maintain their relative positioning along the channel. See [Resizing and moving channels on the Timeline](#), on page 142 for more information.

Deleting a keyframe

If a keyframe is no longer needed on a channel, you can use the following methods to permanently remove the keyframe.

To remove a keyframe from a channel:

- Right-click on the keyframe and select the **Remove Keyframe** command from the context menu.

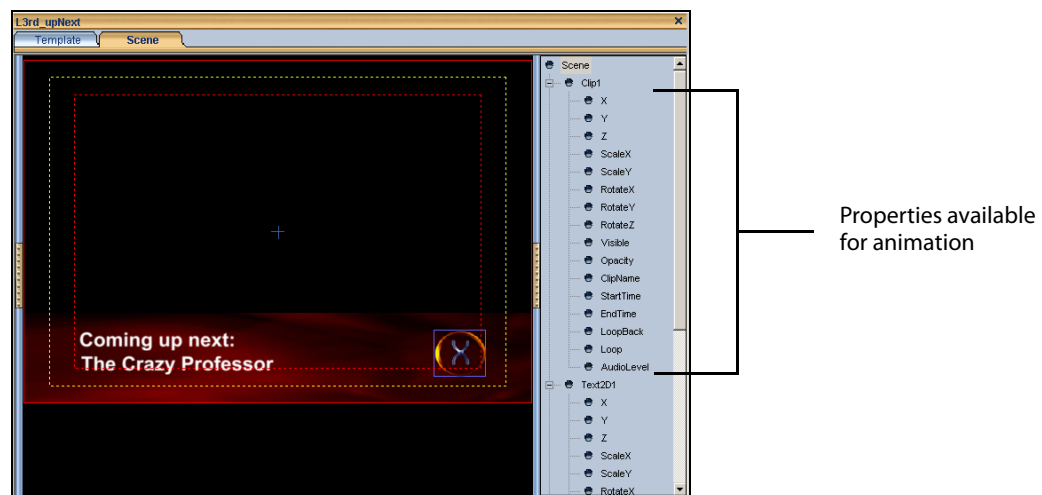
Or,

- Select the keyframe on the channel and press the **Delete** key on your keyboard.

Animated property descriptions

The Animation Editor allows you to create animated scenes (animations) by changing certain properties of the objects in a scene over a period of time to create visual effects. Not all object properties can be animated, and while some animated properties are common to all object types, others are available only to certain object types.

The properties that are eligible to be animated are listed in the Scene Tree when the Animated Properties setting is selected from the Scene Tree's context menu (see [Displaying the animated properties in the Scene tree](#), on page 130).



The properties that can be animated in all types of objects are identified and described in the General animated object properties section below, while animated properties that apply to a specific object types are identified and described in the following sections:

- [General animated object properties](#), on page 143
- [Cel animation properties](#), on page 157
- [Clip properties](#), on page 158
- [Image Properties](#), on page 159
- [Text2D properties](#), on page 160

General animated object properties

While not all object properties can be animated, there is a set of properties that can be animated on all object types. These animated properties are mainly responsible for determining the appearance, location, and spacial orientation of the object on the screen.

The general animated object properties and the following sections provide a description of each of the properties and their usage.


- [X, Y, Z](#), on page 144
- [Scale X and Scale Y](#), on page 144
- [RotateX, Rotate Y, and Rotate Z](#), on page 146
- [Visible](#), on page 147
- [Opacity](#), on page 156

X, Y, Z

The X, Y, and Z properties each represent the position of the object's lower-left corner on the scene's X axis, Y axis, or Z-axis. As such, these properties are used to specify the exact placement of the object on the scene.

The animation effect that these properties are able to achieve is moving the object from one location to another. For example, the X property can move a text object across the scene from the left-side to the right-side, the Y property can move a text object from the top corner to the bottom corner, and the Z property can move a text object from a back depth perspective to a close up. Combining these properties provides you with the ability to have objects travel anywhere on the scene.

This moving effect is achieved by setting two keyframes on the X, Y, or Z property channel to different values (a starting position and an ending position). During the playout of the scene animation, the object transitions from the location set by the first keyframe to the location set by the second keyframe.

To set a keyframe for an X, Y, or Z property, you can either enter a value (number of pixels from the point of origin) in the property's field in the Keyframe Editor, or move the object on the scene to the desired location and then click the **Capture** button  in the Keyframe Editor. The location of the object will automatically be recorded in the property's field.

Scale X and Scale Y

The **Scale X** and **Scale Y** animated properties allow you to create an animation effect that resizes the width and height an object over a period of time.

The actual dimensions of the asset are not changed, rather the **Scale X** and **Scale Y** property values that you assign are scale factors that are multiplied by the object's original dimensions to temporarily redefine the object's width and/or height.

The scaling formulas used are:

- **Original Width x Scale X = Object's animated width**
- **Original Height x Scale Y = Object's animated height**

The following table demonstrates the effect that the scale values can have on the size of an animation object.

| Scale X & Scale Y values | Effect on the object's size in the animation |
|--------------------------|---|
| 2.0 / 2.0 | Increases the object's width and height to twice its original size. |
| 1.0 / 1.0 | Original width and height remain unchanged. |
| 0.5 / 0.5 | Decreases the object's width and height to half of its original size. |

Note: The **Scale X** animated property is responsible for setting the object's width and the **Scale Y** property is responsible for setting the object's height. As such, each must have its own property channel in the Animation Editor. To maintain the object's proper aspect ratio (avoid distortion) both the **Scale X** and **Scale Y** properties should be set to the same values.

To create the animation effect that resizes an object:


- 1 Add the **Scale X** and **Scale Y** property channels to the object animation.
- 2 Add two keyframes to the **Scale X** and **Scale Y** property channels.
- 3 To enlarge an object, you must set the first keyframe to a smaller scale value than that in the second keyframe. For example, setting the first keyframe to 1.0 (original dimension) and the second keyframe to 2.0, results in the object transitioning to double its original size during the playout between the two keyframes.

To shrink an object, you must set the first keyframe to a larger scale value than the second keyframe. For example, setting the first keyframe to 1.0 (original dimension) and the second keyframe to 0.5, results in the object transitioning to half of its original size during the playout between the two keyframes.

There are two ways that you can set the **Scale X** and **Scale Y** values in the Keyframe Editor:

- You can manually enter the value to be used at the selected keyframe in the Keyframe Editor's Scale X and/or Scale Y field.

Or,

- You can resize the object as required and then click the Capture button  to automatically populate the field with the current value.

Note: To obtain the **Scale X** value required to achieve a specific width, divide the desired width by the original width. Similarly, to obtain the **Scale Y** value required to achieve a specific height, divide the desired height by the original height.

Rotate X, Rotate Y, and Rotate Z

The Rotate X, Rotate Y, and Rotate Z animated properties set the orientation or angular position of the object around its axes. Editing the object's spatial orientation over time allows you to create animation effects that rotate the object around its axes. For example, the Rotate X property allows you to flip the object vertically around the x-axis, while the Rotate Y property allows you to spin the object horizontally around the Y-axis, and the Rotate Z property allows you to rotate the object clockwise or counter-clockwise around the Z-axis.

Rotate X, Y, or X property values are reported in units of angular degrees, referenced from the point of origin (0.0) up to 360 degrees which provides a full rotation around the axis. The following table demonstrates that the direction of the rotation is determined by whether the Rotate property's value is a positive or negative number.

| Property | Rotation direction |
|-----------------|--|
| Rotate X | Positive values = front to back Negative values = back to front |
| Rotate Y | Positive values = right to left Negative values = left to right |
| Rotate Z | Positive values = counter-clockwise Negative values = clockwise |

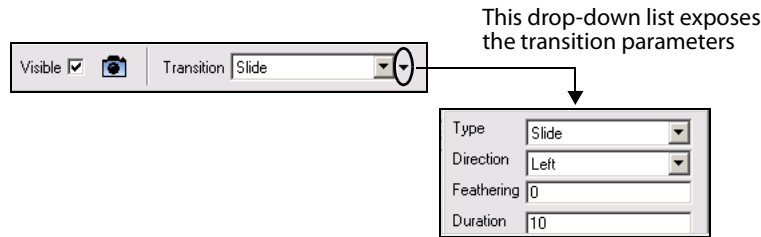
To create the animation effect that rotates an object:

- 1 Add the **Rotate X**, **Rotate Y**, and/or **Rotate Z** property channels to the object animation.
- 2 Add two keyframes to the property channels.
- 3 Set the first keyframe to object's initial orientation in the appropriate Keyframe Editor field. Remember to specify the rotation direction by setting a positive or a negative value.
- 4 Set the second keyframe to object's ending orientation in the appropriate Keyframe Editor field. Remember to specify the rotation direction by setting a positive or a negative value

During playback, the object will transition from the Rotate settings at the first keyframe to those at the second keyframe.

Visible

The **Visible** animated property controls whether or not the object is visible in the scene. Setting the Visible property's **Transition** parameters allows you to control how an object transitions into or out of an animation, or from one object asset to another. For example, you can create an animation where the object appears instantaneously using the **Cut** transition and then have the same object disappear gradually from the animation using the **Fade** transition. Or, you create an object animation where one object asset replaces another asset object using a stylish transition like revealing one as the other fades away.



To have an object appear (or transition) in an animation:


- 1 Add the **Visible** property channel to the object animation.
- 2 Select the Keyframe Editor's **Visible** check box.
- 3 Select a transition style from the Keyframe Editor's **Transition** field's drop-down list (see the table below for a listing and descriptions of the available Transition styles).
- 4 Expose and set the **Transition** parameters by selecting the second drop-down list arrow next to the Keyframe Editor's **Transition** field. See the table below for a listing and descriptions of the available Transition styles and their parameters.

To have an object exit (or transition) from an animation:













- 1 Add the **Visible** property channel to the object animation.
- 2 Clear the Keyframe Editor's **Visible** check box.
- 3 Select a transition style from the Keyframe Editor's **Transition** field's drop-down list (see the table below for a listing and descriptions of the available Transition styles).
- 4 Expose and set the **Transition** parameters by selecting the second drop-down list arrow next to the Keyframe Editor's **Transition** field. See the table below for a listing and descriptions of the available Transition styles and their parameters.

























The following list and table identifies and describes the **Transition** styles and parameters that are associated with the **Visible** animated property.

- Default
- Cut
- Fade or Dissolve
- Wipe or Reveal
- Push or Slide
- Box
- Flip or Valve
- Hinge Swing In or Hinge Swing Out
- Hinge Slide In or Hinge Slide Out
- Roll In or Roll Out
- Cube Roll In or Cube Roll Out
- Cube Slide In or Cube Slide Out

| <p>Default</p> | <p>The Default transition style setting uses the object's current Transition settings (i.e. TransitionType, TransitionDirection, TransitionDirection) that are set in the scene object's Properties.</p> <p>To access the object's Transition properties:</p> <ol style="list-style-type: none"> 1 Right-click the object on the scene and select the Properties command from the context menu. The Properties window appears. 2 Scroll down to the Transitions section. By default, this section will contain the TransitionType property, which allows you to set the object's transition. 3 To change the TransitionType property, select the properties text box and click the  button that appears. 4 Select the desired transition style and its parameters are added to the Transition section in the Properties window. | | | | | | | | | | |
|--------------------------------|---|---------|------|-------|----|------|----------|---|---|---|---|
| <p>Cut</p> | <p>The Cut transition allows the object to immediately appear or disappear from the scene when the keyframe is encountered on the timeline. There are no parameters for this property.</p> | | | | | | | | | | |
| <p>Fade or Dissolve</p> | <p>When the Visible animated property is enabled, the Fade and Dissolve transition styles introduce the object into the scene by gradually increasing the object's opacity from completely transparent to opaque. When the Visible animated property is disabled, the Fade and Dissolve transition styles removes the object from the scene by gradually decreasing the object's opacity from opaque to completely transparent.</p> <ul style="list-style-type: none"> • Duration: The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition. | | | | | | | | | | |
| <p>Wipe or Reveal</p> | <p>When the Visible animated property is enabled, the Wipe and Reveal transition styles introduce the object into the scene by removing a blind from overtop the object, gradually revealing the object beneath from hidden, to partially visible, then completely visible.</p> <p>When the Visible animated property is disabled, the Wipe and Reveal transition styles removes the object from the scene by using a blind on top to gradually hide the object beneath from visible, to partially visible, to completely hidden.</p> <ul style="list-style-type: none"> • Direction: Sets the direction of how the blind reveals/hides the object. Select from: Left, Right, Up, or Down. As the following bullets describe, there are slight behavior differences between the Wipe and Reveal styles' Direction settings. • Wipe: The direction setting represents the direction that the blind is moving in. When Visible is enabled, the blind reveals the object behind it, when Visible is disabled, the blind hides the object as it moves across it. <table border="1" data-bbox="605 1667 1141 1751"> <thead> <tr> <th>Visible</th> <th>Left</th> <th>Right</th> <th>Up</th> <th>Down</th> </tr> </thead> <tbody> <tr> <td>On / Off</td> <td>←</td> <td>→</td> <td>↑</td> <td>↓</td> </tr> </tbody> </table> | Visible | Left | Right | Up | Down | On / Off | ← | → | ↑ | ↓ |
| Visible | Left | Right | Up | Down | | | | | | | |
| On / Off | ← | → | ↑ | ↓ | | | | | | | |

| | <ul style="list-style-type: none"> • Reveal: When Visible is enabled, the direction setting represents the position that the blind starts from, not the direction that it moves toward. However, when Visible is disabled, the direction setting represents the direction that the blind moves toward. <table border="1" data-bbox="602 384 1138 512"> <thead> <tr> <th>Visible</th> <th>Left</th> <th>Right</th> <th>Up</th> <th>Down</th> </tr> </thead> <tbody> <tr> <td>On</td> <td>→</td> <td>←</td> <td>↓</td> <td>↑</td> </tr> <tr> <td>Off</td> <td>←</td> <td>→</td> <td>↑</td> <td>↓</td> </tr> </tbody> </table> <ul style="list-style-type: none"> • Feathering: This setting determines the transparency gradient that is applied to the leading edge of the object as it transitions. Assigning a positive value allows you to set this parameter as a percentage of the image. For example, 0 results in a zero percent or no gradient, while 1.0 is 100% transparent. Using a negative value allows you to specify the value in pixels, rather than a percentage of the object. • Duration: The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition. | Visible | Left | Right | Up | Down | On | → | ← | ↓ | ↑ | Off | ← | → | ↑ | ↓ | | | | | | | | | | |
|-----------------------------|--|---------|------|-------|----|------|----------|---|---|---|---|---------|------|-------|----|------|----|---|---|---|---|-----|---|---|---|---|
| Visible | Left | Right | Up | Down | | | | | | | | | | | | | | | | | | | | | | |
| On | → | ← | ↓ | ↑ | | | | | | | | | | | | | | | | | | | | | | |
| Off | ← | → | ↑ | ↓ | | | | | | | | | | | | | | | | | | | | | | |
| <p>Push or Slide</p> | <p>When the Visible animated property is enabled, the Push and Slide transition styles introduce the object into the scene by pulling the object out from behind a blind. This produces the effect of the object being hidden, to partially visible, then completely visible.</p> <p>When the Visible animated property is disabled, the Push and Slide transition styles removes the object from the scene by pushing the object to behind a blind. This produces the effect of the object being visible, to partially visible, to completely hidden.</p> <ul style="list-style-type: none"> • Direction: Sets the direction of how the object enters or exits the scene. Select from: Left, Right, Up, or Down. As the following bullets describe, there are slight behavior differences between the Push and Slide styles' Direction settings. • Push: The direction setting represents the position that the object starts from, not the direction that it is moving toward. For example, whether Visible is enabled or disabled, if the direction is set to Left, the object is pulled (enters) or pushed (exits) from left to right. <table border="1" data-bbox="602 1255 1138 1341"> <thead> <tr> <th>Visible</th> <th>Left</th> <th>Right</th> <th>Up</th> <th>Down</th> </tr> </thead> <tbody> <tr> <td>On / Off</td> <td>→</td> <td>←</td> <td>↑</td> <td>↓</td> </tr> </tbody> </table> <ul style="list-style-type: none"> • Slide: When Visible is enabled, the direction setting represents the direction that the object is moving toward. When Visible is disabled, the direction setting represents the position that the object starts from, not the direction that it is moving toward. For example, if the direction is set to Left, the object is pushed (exits) from left to right. <table border="1" data-bbox="602 1507 1138 1635"> <thead> <tr> <th>Visible</th> <th>Left</th> <th>Right</th> <th>Up</th> <th>Down</th> </tr> </thead> <tbody> <tr> <td>On</td> <td>←</td> <td>→</td> <td>↑</td> <td>↓</td> </tr> <tr> <td>Off</td> <td>→</td> <td>←</td> <td>↓</td> <td>↑</td> </tr> </tbody> </table> <ul style="list-style-type: none"> • Feathering: This setting determines the transparency gradient that is applied to the leading edge of the object as it transitions. Assigning a positive value allows you to set this parameter as a percentage of the image. For example, 0 results in a zero percent or no gradient, while 1.0 is 100% transparent. Using a negative value allows you to specify the value in pixels, rather than a percentage of the object. • Duration: The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts it's length on the timeline to represent the duration of the transition. | Visible | Left | Right | Up | Down | On / Off | → | ← | ↑ | ↓ | Visible | Left | Right | Up | Down | On | ← | → | ↑ | ↓ | Off | → | ← | ↓ | ↑ |
| Visible | Left | Right | Up | Down | | | | | | | | | | | | | | | | | | | | | | |
| On / Off | → | ← | ↑ | ↓ | | | | | | | | | | | | | | | | | | | | | | |
| Visible | Left | Right | Up | Down | | | | | | | | | | | | | | | | | | | | | | |
| On | ← | → | ↑ | ↓ | | | | | | | | | | | | | | | | | | | | | | |
| Off | → | ← | ↓ | ↑ | | | | | | | | | | | | | | | | | | | | | | |

| <p>Box</p> | <p>When the Visible animated property is enabled, the Box transition style introduces the object into the scene by revealing it from its center first, and then progressing outward.</p> <p>When the Visible animated property is disabled, the Box transition style removes the object from the scene by hiding its extremities and moving inwards.</p> <ul style="list-style-type: none"> • Orientation: The Orientation parameter specifies the direction(s) that the Box transition will use to reveal or hide the object. • Horizontal: When Visible is enabled, the Horizontal setting exposes the object horizontally (left-right), starting from the center and moving outward. When Visible is disabled, Orientation setting hides the object horizontally (left-right), starting from the outsides and moving inward. • Vertical: When Visible is enabled, the Vertical setting exposes the object vertically (up-down), starting from the center and moving outward. When Visible is disabled, Orientation setting hides the object vertically (up-down), starting from the outsides and moving inward. • Center: When Visible is enabled, the Center setting exposes the object both horizontally (left-right) and vertically (up-down), starting from the center and moving outward. When Visible is disabled, Center setting hides the object both horizontally (left-right) and vertically (up-down), starting from the outsides and moving inward. • Feathering: This setting determines the transparency gradient that is applied to the leading edge of the object as it transitions. Assigning a positive value allows you to set this parameter as a percentage of the image. For example, 0 results in a zero percent or no gradient, while 1.0 is 100% transparent. Using a negative value allows you to specify the value in pixels, rather than a percentage of the object. • Duration: The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition. | | | | | | | | | | |
|-----------------------------|--|---|--|---|----|------|----------|---|---|--|---|
| <p>Flip or Valve</p> | <p>When the Visible animated property is enabled, the Flip and Valve transition styles start with the object in a geometric position that renders the object completely invisible. The transition then uses the object's center as a pivot point and rotates the object by 90 degrees along either its vertical or horizontal axes, making it completely visible.</p> <p>When the Visible animated property is disabled, the Flip and Valve transition styles start with the object in a geometric position that makes it completely visible. The transition then uses the object's center as a pivot point and rotates the object by 90 degrees along either its vertical or horizontal axes, so that it becomes invisible.</p> <ul style="list-style-type: none"> • Direction: Sets the direction of how the object will rotate around its vertical or horizontal axes. Select from: Left, Right, Up, or Down. As the following describes there are slight behavior differences between the Flip and Valve styles' Direction settings. • Flip: Whether the Visible is set to On or Off, the direction setting represents the true rotation direction that the object is moving toward. <table border="1" data-bbox="605 1577 1138 1701"> <thead> <tr> <th>Visible</th> <th>Left</th> <th>Right</th> <th>Up</th> <th>Down</th> </tr> </thead> <tbody> <tr> <td>On / Off</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> | Visible | Left | Right | Up | Down | On / Off |  |  |  |  |
| Visible | Left | Right | Up | Down | | | | | | | |
| On / Off |  |  |  |  | | | | | | | |

| | <ul style="list-style-type: none"> • Valve: When Visible is set to On, the direction setting represents starting pivotal position, rather than the direction that it moves toward. For example, Left means that the object starts rotating from left to right. When Visible is Off, the direction setting represents the destination direction of the rotation. For example, Left means the object rotates from right to left. <table border="1" data-bbox="602 407 1138 611"> <thead> <tr> <th>Visible</th> <th>Left</th> <th>Right</th> <th>Up</th> <th>Down</th> </tr> </thead> <tbody> <tr> <td>On</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Off</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> <ul style="list-style-type: none"> • Duration: The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition. | Visible | Left | Right | Up | Down | On |  |  |  |  | Off |  |  |  |  |
|---|---|---|--|---|----|----------|---------------|---|---|--|---|-----------|---|---|--|---|
| Visible | Left | Right | Up | Down | | | | | | | | | | | | |
| On |  |  |  |  | | | | | | | | | | | | |
| Off |  |  |  |  | | | | | | | | | | | | |
| <p>Hinge Swing In or Hinge Swing Out</p> | <p>When the Visible animated property is enabled, the Hinge Swing In and Hinge Swing Out transition styles start with the object in a geometric position that renders the object completely invisible. The transition then uses one side of the object as a fixed pivotal point and rotates the object by 90 degrees along either its vertical or horizontal axes, making it completely visible.</p> <p>When the Visible animated property is disabled, the Hinge Swing In and Hinge Swing Out transition styles start with the object in a geometric position that makes it completely visible. The transition then uses one side of the object as a fixed pivotal point and rotates the object by 90 degrees along either its vertical or horizontal axes, so that it becomes invisible.</p> <p>The following tables demonstrate that the direction in which the object swings, inward or outward, depends on the Transition style and whether the Visible property is enabled or disabled.</p> <table border="1" data-bbox="602 1241 1078 1499"> <thead> <tr> <th>Visible</th> <th>Transition</th> <th>Swing direction</th> </tr> </thead> <tbody> <tr> <td>On</td> <td>Swing In</td> <td>Back-to-front</td> </tr> <tr> <td>Off</td> <td>Swing In</td> <td>Front-to-back</td> </tr> <tr> <td>On</td> <td>Swing Out</td> <td>Front-to-back</td> </tr> <tr> <td>Off</td> <td>Swing Out</td> <td>Back-to-front</td> </tr> </tbody> </table> <ul style="list-style-type: none"> • Position: Specifies which edge of the object remains fixed (pivotal point) during the rotation. Select from: Left, Right, Top, or Bottom. • Duration: The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition. | Visible | Transition | Swing direction | On | Swing In | Back-to-front | Off | Swing In | Front-to-back | On | Swing Out | Front-to-back | Off | Swing Out | Back-to-front |
| Visible | Transition | Swing direction | | | | | | | | | | | | | | |
| On | Swing In | Back-to-front | | | | | | | | | | | | | | |
| Off | Swing In | Front-to-back | | | | | | | | | | | | | | |
| On | Swing Out | Front-to-back | | | | | | | | | | | | | | |
| Off | Swing Out | Back-to-front | | | | | | | | | | | | | | |

Hinge Slide In or Hinge Slide Out









When the Visible animated property is enabled, the Hinge Slide In and Hinge Slide Out transition styles start with the object in a geometric position that renders the object completely invisible. With the object's center as a pivot point, the object is rotated 90 degrees along its vertical or horizontal axes, making it completely visible. Then the object moves smoothly into position.

When the Visible animated property is disabled, the Hinge Slide In and Hinge Slide Out transition styles start with the object in a geometric position that makes it completely visible. The object moves smoothly away from its current position (slides), and then uses the object's center as a pivot point to rotate the object 90 degrees along its vertical or horizontal axes, so that it becomes invisible.

The following tables demonstrate that the direction in which the object swings, inward or outward, depends on the Transition style and whether the Visible property is enabled or disabled.

| Visible | Transition | Swing direction |
|---------|------------|-----------------|
| On | Slide In | Back-to-front |
| Off | Slide In | Front-to-back |
| On | Slide Out | Front-to-back |
| Off | Slide Out | Back-to-front |

- **Position:** Sets the direction of how the object will slide and rotate around its vertical or horizontal axes. Select from: Left, Right, Top, or Bottom. As the following table demonstrates, the object's Position settings are interpreted differently when the Visible property is enabled and when it is disabled.

| Visible | Left | Right | Top | Bottom |
|---------|---|---|--|---|
| On |  |  |  |  |
| Off |  |  |  |  |

- **Duration:** The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition.

Roll In or Roll Out









When the Visible animated property is enabled, the Roll In and Roll Out transition styles start with the object in a geometric position that renders the object completely invisible. With the object's center as a pivot point, the object is rotated 90 degrees along its vertical or horizontal axes, making it completely visible. Then the object moves smoothly into position.

When the Visible animated property is disabled, the Roll In and Roll Out transition styles start with the object in a geometric position that makes it completely visible. The object moves smoothly away from its current position (slides), and then uses the object's center as a pivot point to rotate the object 90 degrees along its vertical or horizontal axes, so that it becomes invisible.









The following tables demonstrate that the direction in which the object swings, inward or outward, depends on the Transition style and whether the Visible property is enabled or disabled.

| Visible | Transition | Swing direction |
|---------|------------|-----------------|
| On | Roll In | Back-to-front |
| Off | Roll In | Front-to-back |
| On | Roll Out | Front-to-back |
| Off | Roll Out | Back-to-front |

- **Direction:** Sets the direction of how the object will rotate around its vertical or horizontal axes. Select from: Left, Right, Up, or Down.
- **Roll In:** When Visible is set to On, the direction setting represents starting pivotal position, rather than the direction that it moves toward. For example, Left means that the object rotates from left to right. When Visible is Off, the direction setting represents the destination direction of the rotation. For example, Left means the object rotates from right to left.

| Visible | Left | Right | Up | Down |
|---------|---|---|---|---|
| On |  |  |  |  |
| Off |  |  |  |  |

- **Roll Out:** When Visible is set to On, the direction setting represents the direction that it moves toward. For example, Left means that the object starts rotating from right to left. When Visible is Off, the direction setting represents starting pivotal position, rather than the direction that it moves toward. For example, Left means the object rotates from left to right.

| Visible | Left | Right | Up | Down |
|---------|---|---|---|---|
| On |  |  |  |  |
| Off |  |  |  |  |

- **Duration:** The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition.

**Cube Roll In or
Cube Roll Out**









When the Visible animated property is enabled, the Cube Roll In and Cube Roll Out transition styles start with the object in a geometric position that renders the object completely invisible. With the object's center as a pivot point, the object is rotated 90 degrees along its vertical or horizontal axes, making it completely visible. Then the object moves smoothly into position.

When the Visible animated property is disabled, the Cube Roll In and Cube Roll Out transition styles start with the object in a geometric position that makes it completely visible. The object moves smoothly away from its current position (slides), and then uses the object's center as a pivot point to rotate the object 90 degrees along its vertical or horizontal axes, so that it becomes invisible.

The following tables demonstrate that the direction in which the object rotates, inward or outward, depends on the Transition style.

| Visible | Transition | Swing direction |
|---------|---------------|-----------------|
| On/Off | Cube Roll In | Front-to-back |
| On/Off | Cube Roll Out | Back-to-front |

- **Direction:** Sets the direction of how the object will rotate around its vertical or horizontal axes. Select from: Left, Right, Up, or Down.
- **Cube Roll In:** Whether Visible is set to On or Off, the direction setting represents the direction that it moves toward. For example, Left means that the object rotates from right to left.
- **Cube Roll Out:** Whether Visible is set to On or Off, the direction setting represents the starting pivotal position, rather than the direction that it moves toward. For example, Left means the object rotates from left to right.

| | Left | Right | Up | Down |
|---------------------------------|---|--|---|---|
| Visible On/Off Cube Roll In |  |  |  |  |
| Visible On/Off Cube Roll Out |  |  |  |  |

- **Duration:** The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition.

Cube Slide In or Cube Slide Out

When the Visible animated property is enabled, the Cube Slide In and Cube Slide Out transition styles start with the object in a geometric position that renders the object completely invisible. With the object's center as a pivot point, the object is rotated 90 degrees along its vertical or horizontal axes, making it completely visible. Then the object moves smoothly into position.





When the Visible animated property is disabled, the Cube Slide In and Cube Slide Out transition styles start with the object in a geometric position that makes it completely visible.

The object moves smoothly away from its current position (slides), and then uses the object's center as a pivot point to rotate the object 90 degrees along its vertical or horizontal axes, so that it becomes invisible.

The following tables demonstrate that the direction in which the object rotates, inward or outward, depends on the Transition style.

| Visible | Transition | Swing direction |
|---------|---------------|-----------------|
| On/Off | Cube Slide In | Front-to-back |
| On/Off | Cube Roll Out | Back-to-front |

- **Direction:** Sets the direction of how the object will rotate around its vertical or horizontal axes. Select from: Left, Right, Up, or Down. Whether Visible is set to On or Off, the direction setting for both the Cube Roll In and Cube Roll Out transitions represents the direction that it moves toward. For example, Left means that the object rotates from right to left.

| Visible | Left | Right | Up | Down |
|---------|---|---|--|---|
| On/Off |  |  |  |  |

- **Duration:** The length of time that it will take to complete the transition after it is initiated at the keyframe. Enter the number or frames, or the amount of time (00:00:00). The visibility keyframe adjusts its length on the timeline to represent the duration of the transition.

Opacity

The **Opacity** animated property allows you to set the opacity level of an object. Setting this property at two keyframes in an animation produces animation effects like, fading objects into the background, or drawing attention to an object by fading it from translucent to opaque. The Opacity property can be set to a value within the range of 0.0 to 1.0, where 0.0 indicates full transparency and 1.0 indicates full opacity.

To fade object in an animation:

- 1 Add the **Opacity** property channels to the object animation.
- 2 A default keyframe will already be present, so add an additional keyframe to the **Opacity** property channels.
- 3 To fade an object from opaque to transparent, you must set the first keyframe to a larger opacity value than that in the second keyframe. For example, the object transitions from solid to transparent when the first keyframe is set to 1.0 and the second keyframe is set to 0.0.

Similarly, to fade an object from transparent/translucent to opaque, you must set the first keyframe to a smaller opacity value than the second keyframe.

Cel animation properties

A cel animation is a sequence of images shown in rapid succession to form an animation. Each image in the cel animation is known as a frame, and each has a frame number indicating its position in the sequence, starting with frame 1.

A cel animation starts playing when its **Visible** property is selected in the Keyframe Editor. Note however, that the cel animation also has a **Visible** property in the scene that is accessible from the Property Browser. Be sure to set the **Visible** property in the Property Browser to **False**, otherwise it will start playing immediately upon entering Production mode.

The following table identifies the additional properties that determine how the cel animation will play. However, these properties need to be set before playing the cel animation. Once the cel animation is playing, changing these properties will have unpredictable effects. (Generally, these properties are set up in the frame preceding the frame on which the cel animation's Visible property is switched on.)

| Property | What it does and how to make it animate |
|-------------------|---|
| CelName | The cel animation to be displayed. Specify the cel animation to display at the selected keyframe by dragging it from the Asset Browser and then dropping it into the CelName field in the Keyframe Editor or the desired frame on the timeline. |
| EndFrame | Specifies the cel frame at which the cel animation ends. Manually enter the frame number in the EndFrame field in the Keyframe Editor. Entering 1 means that the animation should run to the end. |
| EndLoop | The cel frame at which the cel animation finishes or stops looping. Manually enter the frame number in the EndLoop field in the Keyframe Editor. For this property to work, the Loop property must also be activated in the animation. |
| Loop | When selected, indicates that the cel animation will restart from the frame indicated by the StartLoop property. |
| Rate | The duration of a cel animation is determined by the number of frames in the cel animation and the rate at which it is played. By default, the rate is the number of frames per second supported by the target output format. Manually enter the number of frames per second to be used at the selected keyframe in the Rate field in the Keyframe Editor. |
| StartFrame | Specifies the cel frame at which the cel animation starts playing. Manually enter the frame number in the StartFrame field in the Keyframe Editor. |
| StartLoop | The cel frame at which the cel animation starts when looping. Manually enter the frame number in the StartLoop field in the Keyframe Editor. For this property to work, the Loop property must also be activated in the animation. |

Clip properties

A clip is a pre-recorded video sequence. Positions in the clip are expressed in terms of seconds rather than frames.

Additional properties determine how the clip will play. However, these properties need to be set before playing the clip. Once the clip is playing, changing these properties will have unpredictable effects. (Generally, these properties are set up in the frame preceding the frame on which the clip's Visible property is switched on.)

| Property | What it does and how to make it animate |
|---------------------------------|--|
| AudioLevel | Sets the volume of the clip's audio payout. Specify a value within the range of 0.0 (mute) to 1.0 (100%). To amplify a low-level audio payout, you can set this property to a maximum value of 8.0. |
| AudioProfile | * This property is currently not supported in animations on the CG. An index (integer) into a set of pre-defined audio profiles. These profiles are defined in the CG control panel and define how the various audio channels are mixed. |
| ClipName | The clip to be displayed. Specify the clip to display at the selected keyframe by dragging it from the Asset Browser and then dropping it into the ClipName field in the Keyframe Editor. |
| EndTime | Specifies the end time for the clip. Manually enter the time in the StartTime field in the EndTime field in the Keyframe Editor. It is recommended that the EndTime be set at the same keyframe as the StartTime. |
| Loop | When selected, restarts the clip from the time indicated by the Loopback property. |
| PlaybackStateOnInvisible | * This property is currently not supported in animations on the CG/XG. Determines what a clip does when it becomes invisible. It can be one of <ul style="list-style-type: none">• PLAY: Continue playing• PAUSE: Pause• RECUE: Stop and recue itself at the beginning |
| StartTime | Specifies the time at which the clip begins. |

Image Properties

In addition to the [General animated object properties](#), image objects have access to an exclusive animated property: **ImageName**. The **ImageName** animated property identifies the name and location of the image asset that is to be displayed.

The image that displays as an image object on the scene can be set by dragging it directly from the Asset Browser or Gallery Browser directly onto the scene's canvas, or by setting the object's **ImageName** property (right-click on the object, select **Properties**). The **ImageName** animated property provides extra functionality that allows you to change or replace the object's displayed image with another image during the playout of an animation. For example, when a scene animation begins the image object can be set to display "Image1". Later in the animation, you can set the **ImageName** animated property to replace the original image with "Image2". The Image object and all of its associated settings remain unchanged, it is only the displayed image that changes.

To replace an image in a image object using the ImageName animated property:

- 1 Add an image asset to the scene's canvas by dragging it from the Asset Browser or Gallery Browser.
- 2 Drag the image object's **ImageName** animated property from the Scene Tree to the animation in the Animation Tree.
- 3 Select the ImageName channel's keyframe and drag the image asset from the Asset Browser, Gallery Browser, or the Scene Tree to the Keyframe Editor's **ImageName** field.
- 4 Set the length of time (frames/time) that you want the image to be active by typing a value in the Keyframe Editor's **Duration** field. The keyframe adjusts its length on the timeline to represent the new duration.
- 5 Drag the second image from the Asset Browser or Gallery Browser directly onto the **ImageName** channel at the location where you would like the replacement image to occur.
- 6 Set the length of time (frames/time) that you want the image to be active by typing a value in the Keyframe Editor's **Duration** field. A thick line appears on the Timeline representing the duration of the image's display.

The transition type and duration that is used to replace the displayed image with another in the image object is set in the image object's properties (right-click on the object, select **Properties**).

Text2D properties

In addition to the [General animated object properties](#), text objects have access to an exclusive animated property: **Text**. The **Text** animated property identifies the text to be displayed in the text object.

The text that displays in the text object can be set by selecting and typing directly in the object, or by setting the object's **Text** property (right-click on the object, select **Properties**). The **Text** animated property provides extra functionality that allows you to change or replace the object's displayed text with another text. For example, when a scene animation begins the text object can be set to display "Coming up next". Later in the animation, you can set the **Text** animated property to replace the original text with "The evening news". The text object and all of its associated settings remain unchanged, it is only the displayed text that changes.

To replace the text in a text object using the Text animated property:

- 1 Add a text primitive to the scene's canvas.
- 2 Drag the text object's **Text** animated property from the Scene Tree to the animation in the Animation Tree.
- 3 Select the Text channel's keyframe and type the desired text in the Keyframe Editor's **Text** field.
- 4 Set the length of time (frames/time) that you want the text to be active by typing a value in the Keyframe Editor's **Duration** field. The keyframe adjusts its length on the timeline to represent the new duration.
- 5 Add a keyframe to the Text channel at the location where you would like the replacement of the text to occur.
- 6 Select the second keyframe and type the desired text in the Keyframe Editor's **Text** field.
- 7 Set the length of time (frames/time) that you want the text to be active by typing a value in the Keyframe Editor's **Duration** field. The keyframe adjusts its length on the timeline to represent the new duration.

Note: The transition type and duration that is used to replace the displayed text with the new text can be found in the Transition section of the Text object's properties (right-click on the object, select **Properties**).

8 Working with control objects

The control-side (template or panel) of an Xstudio project provides operators with an interface which helps them identify, edit and control the scene and its objects before and during playback.

Control objects on a template are usually designed to provide operators with fields (i.e. textboxes, dropboxes, images...etc.) that are linked to objects on the scene side. Prior to playback, the control objects can be edited which feeds new data or assets to the scene side. This allows one template to produce scenes with different content.

The nature of a Template + Scene project is that once it is taken to air, there is no manual intervention permitted to change the scene's content or behavior. Therefore, the template's control objects can be programmed using Xstudio's Action Editor to execute actions based on events during playback. These actions allow the scene and its objects to react and behave without any manual intervention.

Unlike templates, the nature of control panels is to provide operators with an interface to edit and control the scene and its objects during playback. Therefore, the control objects on the panel are often interactive (i.e. buttons, textboxes, radio buttons...etc.) and they are more likely to be associated with actions to trigger an edit or behavior on the scene-side of the project.

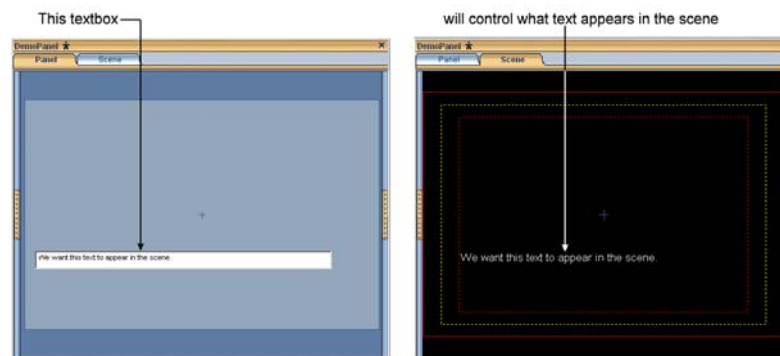
The following sections describe the types of objects that can be used on the Xstudio projects control-side and how to link them to objects on the scene side.

- [Connecting control objects to scene objects](#), on page 162
- [Text objects on the control canvas](#), on page 163
- [Dropbox objects on the control canvas](#), on page 165
- [Button objects on the control canvas](#), on page 167
- [Textbox objects on the control canvas](#), on page 169
- [Image objects on the control canvas](#), on page 171
- [Checkbox & Radio Button objects on the control canvas](#), on page 173
- [ActiveX objects on the control canvas](#), on page 175
- [Timer objects on the control canvas](#), on page 176

Note: See [Adding logic to control objects](#), on page 211 for information and instructions on how to program control objects to execute actions based on user or playback events.

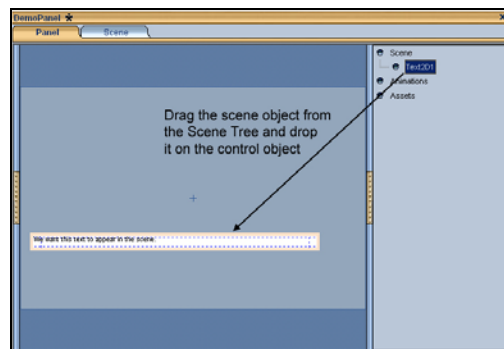
Connecting control objects to scene objects

When a template or control panel is associated with a scene, it is usually because objects on the project's scene-side are to be controlled by elements on the control-side. For example, a text box on the control side can dictate the text that will be displayed as text in the scene. For this relationship and behavior to occur, you must connect the control element with the scene element.



To connect a control object to a scene object:

- 1 Display the **Scene Tree** and then display the control-side of the project by selecting the Template or Panel tab.
- 2 Drag the scene object from the **Scene Tree** and drop it onto its corresponding object on the control-side tab.




- 3 Display the **Property Browser** (CTRL + R).

Note: You could also forego the first two steps of the procedure by simply entering the scene name in the **Scene Element** row of the **Property Browser** (press CTRL + P to display it).

- 4 Ensure that the scene name appears in the Property Browser's **Scene Element** row.

Text objects on the control canvas

Adding a Text primitive  to the control-side canvas allows you to display a text string, which is useful for labeling other objects on the template/control panel, or for providing instructions to the operator.

The text object provides instructions to the operator.




When a Text object is selected on the canvas, the Properties Browser displays the following properties:

| Category | Text object properties |
|-------------------|--|
| | <p>The following properties determine the identity, content and control of the text object:</p> <p>Name, Text, Visible, Locked, Trigger Events</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> <p>You can type the desired text into the Text property, or you can double-click the text object on the canvas and type the content directly into the object.</p> |
| Position | <p>The following properties allow you to set the coordinates (location) and dimensions (size) of the text object:</p> <p>X, Y, Width, Height</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> <p>You can also set the text object's location and size settings by dragging the text object and/or its boundaries directly on the canvas (see page 79 and page 80). The text object's properties are automatically updated in the Properties Browser when this method is used.</p> |
| Appearance | <p>The following properties allow you to set the style and formatting of the text object. For example, some of the properties allow you to specify the text's font style, size and color.</p> <p>Text Direction, Font, BackColor, Justification, WordWrap, FaceColor, Limit Width, Max Length, Tab Index, Format, Basic Rendering, Allow Edits</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> <p>You can also assign a font to the text object's content by selecting the text object on the canvas and using the Font toolbar on Xstudio's desktop (page 35).</p> |

| Category | Text object properties |
|------------------------|---|
| Links | <p>The following properties allow you to associate a lookup table and/or data source which will determine the content of the text object: Lookup Table, Lookup Location, Data Source, Data Update, Data Target, Data Auto Write, Scene Element, Take Number, Scene Update</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> <p>See Linking objects to a data source, on page 177 for more information.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add addition user properties.</p> |

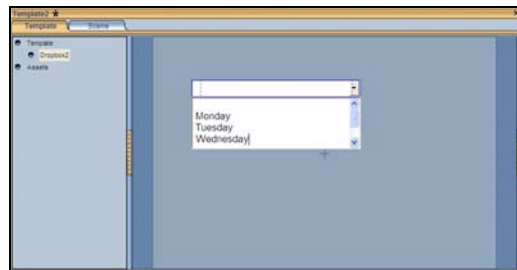
Dropbox objects on the control canvas

Adding a Dropbox primitive  to the control-side canvas creates a drop-down list which can be populated with multiple selections. Dropboxes are often used to provide operators with a quick and limited list of options that affect what displays or happens on the scene side (e.g. a list of show titles).

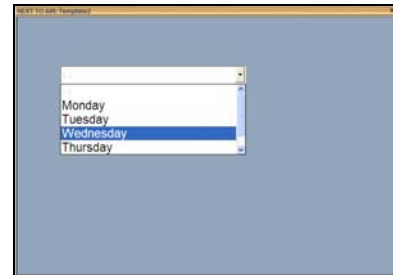
When associated with the **On Update** event, a change to the dropbox's selection can trigger an action that might affect the scene's content or playout behavior. See [On Update](#), on page 217 for more information.

To populate the dropbox object:

- 1 Drag the Dropbox primitive onto the control-side canvas.
- 2 Double-click on the dropbox object and the object expands to display an empty list.
- 3 Type the text for the first item in the list and then press Enter.
(For a blank selection press the spacebar and then Enter.)
- 4 Repeat for the remaining items in the list.
- 5 To preview the behavior of the dropbox, switch to Production mode and click the arrow on the dropbox.



Design mode




Production mode

When a Dropbox object is selected on the canvas, the Properties Browser displays the following properties:

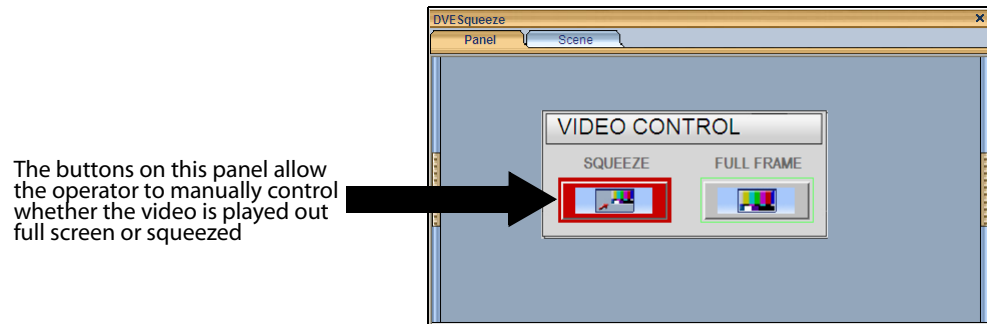
| Category | Dropbox object properties |
|-----------------|---|
| | <p>The following properties determine the identity and control of the dropbox object: Name, Visible, Locked, Trigger Events, Selection Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Position | <p>The following properties allow you to set the coordinates (location) and size of the dropbox object: X, Y, Width You can also set the dropbox object's location and size settings by dragging the object and/or its boundaries on the canvas (see page 79 and page 80). Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |

| Category | Dropbox object properties |
|------------------------|---|
| Appearance | <p>The following properties allow you to set the style and formatting of the dropbox object: Font, ForeColor, BackColor, Tab Index</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> <p>You can also assign a font to the dropbox object's content by selecting the dropbox object on the canvas and using the Font toolbar on Xstudio's desktop (page 35).</p> |
| Links | <p>The following properties allow you to associate a lookup table and/or data source which will populate the content of the dropbox object: Lookup Table, Lookup Location, Data Source, Data Update, Data Target, Data Auto Write, Scene Element, Take Number, Scene Update</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> <p>See Linking objects to a data source, on page 177 for more information.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playback.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playback by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

Button objects on the control canvas

Adding a Button primitive  to the control-side canvas creates a button object which can be programmed to execute actions when an operator clicks the button. The actions that executed upon a click may affect the content or behavior of the scene or scene objects, and even the button itself can be programmed to provide feedback to the operator (i.e. change color to indicate a state).

When associated with the **On Click** event, the clicking of the button can trigger an action that might affect the scene's content or playout behavior. See [On Click](#), on page 216 for more information.




See [Connecting control objects to scene objects](#), on page 162 and [Adding logic to control objects](#), on page 211 for information and instructions for adding logic to a button object to control the scene or scene objects.

When a Button object is selected on the canvas, the Properties Browser displays the following properties:

| Category | Button object properties |
|-----------------|---|
| | <p>The following properties determine the identity, content and control of the button object:</p> <p>Name, Text, Visible, Locked</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Position | <p>The following properties allow you to set the coordinates (location) and dimensions (size) of the button object:</p> <p>X, Y, Width, Height</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Image | <p>The following properties allow you to display an image on the button object:</p> <p>ImageName, ImageLocation, Style, Alignment</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |

| Category | Button object properties |
|------------------------|---|
| Appearance | <p>The following properties allow you to set the style and formatting of the button object:</p> <p>Font, ForeColor, BackColor, Tab Index</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playback.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playback by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |

Textbox objects on the control canvas

Adding a Textbox primitive  to the control-side canvas allows you to type and display text inside a recessed box. The Textbox object is usually linked to and determines the text that is displayed within a Text2D object on the scene-side. Text box objects allow for interactive editing in Production mode and they can be populated with information entered manually or fed in from a data source.

The Textbox object on the control-side allows operators to specify the text that will appear in the Text2D on the scene-side during playback



When associated with the **On Update** event, a change to the textbox's selection can trigger an action that might affect the scene's content or playback behavior. See [On Update](#), on page 217 for more information.


See [Connecting control objects to scene objects](#), on page 162 and [Adding logic to control objects](#), on page 211 for information and instructions for adding logic to a Textbox object for populating Text2D objects on the scene.

When a Textbox object is selected on the canvas, the Properties Browser displays the following properties:

| Category | Textbox object properties |
|-------------------|---|
| | <p>The following properties determine the identity, content and control of the textbox object: Name, Text, Visible, Locked, Trigger Events</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Position | <p>The following properties allow you to set the coordinates (location) and dimensions (size) of the textbox object: X, Y, Width, Height</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Appearance | <p>The following properties allow you to set the style and formatting of the textbox object: Text Direction, Font, BackColor, Justification, Word Wrap, FaceColor, Limit Width, Max Length, Tab Index, Format, Basic Rendering, Allow Edits</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |

| Category | Textbox object properties |
|------------------------|---|
| Links | <p>The following properties allow you to associate a lookup table and/or data source which will determine the content of the textbox object: Lookup Table, Lookup Location, Data Source, Data Update, Data Target, Data Auto Write, Scene Element, Take Number, Scene Update</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add additional user properties.</p> |


Image objects on the control canvas

Adding an Image primitive  to the control-side canvas creates an Image object which can act as a static visual representation (image, clip, or cel animation) on the control-side (template or panel), or like a button which can be programmed to execute actions (i.e. if the image object's Allow Clicks property is enabled it can act like a button).

When the image object is associated with an **On Update** event or an **On Click** event, it can be used to trigger actions that can affect the content or behavior of the scene or its objects. See [Add Actions and Events](#), on page 213 for more information.



The station logo is an image which quickly identifies which station the scene applies to, but it can also be programmed so that when the operator changes the image on the control-side, it is also changed on the scene-side.

As described on [page 70](#), image objects can either be an image asset that was added from the Asset Browser, or an Image primitive . When an image primitive is added to the scene, it is empty as it does not yet have an image file associated to it. See [page 71](#) for instructions on associating an asset with the Image primitive.



Other methods for determining or changing which image asset will appear in the Image object include using **Set Image** action in response to an event during playout ([page 242](#)) or using the **ImageName** property within a keyframe on the scene's animation channel ([page 159](#)).

When an image object is selected on the canvas, the Properties Browser displays the following properties:

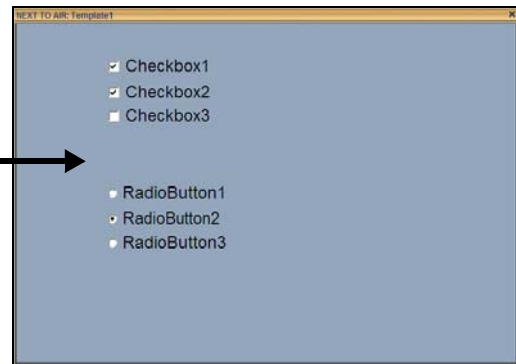
| Category | Image object properties |
|-----------------|---|
| | <p>The following properties determine the identity and control of the image object:</p> <p>Name, Visible, Locked, Allow Clicks</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Position | <p>The following properties allow you to set the location of the checkbox or radio button object:</p> <p>X, Y, Width, Height</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |

| Category | Image object properties |
|------------------------|---|
| Image | <p>The following properties allow you to specify image asset that will be displayed and positioned within the Image object: ImageName, ImageLocation, Allow Edits, Restricted, Alignment, Style, Auto Create Job</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Links | <p>The following properties allow you to associate a lookup table and/or data source which will determine the content of the image object: Data Source, Data Update, Scene Element, Take Number, Scene Update, Lookup Table, Lookup Location</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add addition user properties.</p> |

Checkbox & Radio Button objects on the control canvas

The purpose of adding the Checkbox  and Radio button  primitives to the control-side of a project is so that you can assign actions to the object in order to make it trigger commands when selected/deselected by the operator either prior or during playout. Therefore, checkbox objects are most often associated with actions that are executed on the [On Checked](#) and [On Unchecked](#) events, while radio button objects are most often associated with actions that are executed on the [On Selected](#) and [On Unselected](#) events. See [Adding logic to control objects](#), on page 211 for more information.

Multiple checkboxes can be selected at any one time, but only one radio button can be selected at a time.




When a checkbox or radio button object is selected on the canvas, the Properties Browser displays the following properties:

| Category | Checkbox / Radio Button object properties |
|-------------------|---|
| | <p>The following properties determine the identity and control of the checkbox or radio button object: Name, Text, Visible, Locked, Checked (checkbox only), Selected (radio button only) Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Position | <p>The following properties allow you to set the coordinates (location) of the checkbox or radio button object: X, Y Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Appearance | <p>The following properties allow you to specify the color and style of the text that displayed beside the checkbox or radio button object: Forecolor, Font, Tab Index Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |

| Category | Checkbox / Radio Button object properties |
|------------------------|---|
| Links | <p>The following properties allow you to associate a data source which will determine the content of the scene object associated with the checkbox or radio button object:</p> <p>Data Source, Data Update, Data Target, Data Auto Write</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add addition user properties.</p> |


ActiveX objects on the control canvas

Adding an ActiveX primitive  to the control-side canvas allows you to integrate a third party or custom component into the control panel. Once added, display the Property Browser and enter the registered classname of the ActiveX object in the **Prog ID** field. For example, if adding the Microsoft Media Player, you would enter `MediaPlayer.MediaPlayer.1`

When an ActiveX object is selected on the canvas, the Properties Browser displays the following properties:

| Category | ActiveX object properties |
|------------------------|---|
| | <p>The following properties determine the identity and control of the ActiveX object:</p> <p>Name, Visible, Locked, ProgID</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Position | <p>The following properties allow you to set the coordinates (location) and dimensions (size) of the ActiveX object:</p> <p>X, Y, Width, Height</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| Appearance | <p>The following properties allow you to specify whether the ActiveX object is transparent or opaque, as well as assign a tab index value:</p> <p>Transparent, Tab Index</p> <p>Click on the links or see Control object properties, on page 286 for a description of each of these properties.</p> |
| User Properties | <p>User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playout.</p> <p>To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playout by a calling the Set Property action (see page 244).</p> <p>Repeat the above steps to add addition user properties.</p> |

Timer objects on the control canvas

Adding an Timer primitive  to the control-side canvas allows you to create and display a timer object that can be set up to count up or down to/from specific times.

When associated with the **On Timer** event, an update to the timer object can trigger an action that might affect the scene's content or layout behavior. See [On Timer](#), on page 217 for more information.

When a Timer object is selected on the canvas, the Properties Browser displays the following properties:

| Category | Timer object properties |
|------------------------|---|
| | The following properties determine the identity and control of the Timer object: Name , Visible , Locked Click on the links or see Control object properties , on page 286 for a description of each of these properties. |
| Position | The following properties allow you to set the location and dimensions of the Timer object: X , Y , Width , Height Click on the links or see Control object properties , on page 286 for a description of each of these properties. |
| Timer | The following properties allow you to set the Timer object's display format, interval, start and end times: Format , Interval , StartTime , EndTime , Infinite , AutoStart Click on the links or see Control object properties , on page 286 for a description of each of these properties. |
| Appearance | The following properties allow you to specify the color and font style of the digits displayed in the Timer object: Forecolor , Font Click on the links or see Control object properties , on page 286 for a description of each of these properties. |
| User Properties | User-defined properties. You can associate your own properties with an object. For example, the property may store data that can then be recalled by a script during playback. To add a user property, click the grey cell in the first column and type the property's name. Enter the property's initial value in the cell next to it. You can also leave it blank and perhaps populate it during playback by a calling the Set Property action (see page 244). Repeat the above steps to add additional user properties. |

Linking objects to a data source



Xstudio's Data Manager allows you to set up data sources that can be linked to objects on the control-side of your canvas. Often the control object is linked to a scene object which then displays the data on air. Data sources can be databases, Excel files, text files, RSS feeds (an XML format that distributes news headlines, links and abstracts collected from the Web) or social media message and polls.

Note: Information and instructions for using Data Manager to retrieve and display social media messages and online polls are provided in [Using social media messages and online polls as a data source](#), on page 195.

To define data sources that can be linked to objects:

1 Open Xstudio's Data Manager:

- Select the **View**> **Data Manager** menu command.

Or,

- Click the Data Manager icon  on Xstudio's desktop.

2 Set up a **schema**.

A schema is a holding space for one or more data sources. To link live data to control objects, Xstudio must first be told where the data is coming from. This is done by setting up one or more data sources within a container called a schema. Once Xstudio recognizes the data source's existence within the schema, the source can be used over and over again to create, for example, tickers. See [page 179](#) for more information about creating and using schemas.

3 From your new schema, create a **data entity**.

A data entity combines specific data from a data source (e.g. a particular table) with instructions for displaying them (e.g. in ticker format, as an unchanging piece of information). There are four types of data entities:






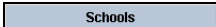
- [Creating a snapshot](#)
- [Creating a ticker](#)
- [Creating a multi-ticker](#)
- [Creating a class](#)

4 Link items from data entities to control objects.

Data always goes through a control object, which may be linked to a scene object. See [page 190](#) for more information.

Data Manager screen items

The following table briefly explains the functionality of each of the Data Manager screen items. All are discussed in further detail over the course of the chapter.

| Screen item | What it does |
|---|---|
| Schema icon  | Allows you to set up a schema. A schema contains one or more related data sources. A data source can be a database, Excel file, .txt file, etc. You cannot extract data directly from a source within a schema. Instead, sources within a schema become the basis for data entities (snapshots, tickers, multi-tickers and classes), which you can connect directly to objects in control panels, templates, etc. |
| Snapshot icon  | Allows you to select a single data item from a schema and have it display in a selected object (e.g. text box). |
| Ticker icon  | Allows you to select multiple data items from a single source and have them update at pre-set intervals in a selected object (e.g. text box). |
| Multi-Ticker icon  | Allows you to select multiple data items from several data sources and have them update at pre-set intervals in a selected target (e.g. text box). |
| Class icon  | Used when none of the other data source types (i.e. database, Excel spreadsheet, text file) suit your requirements, and/or you would like something other than a snapshot, ticker, or multi-ticker. |
| Schema button  | Once schemas have been created, they will be listed in the left-hand portion of the Data Manager as buttons. Once snapshots, tickers, multi-tickers or classes have been created from the schema, clicking the schema icon will list these derivative items. |
| Upper data table (white) | Each cell in this table represents a single data value that can be used as the data source of an object in a template or panel. |
| Lower data table (grey) | In the case of tickers and multi-tickers, the grey tables show the full set of data that the ticker cycles through. At any time, only the subset of rows in the upper white table is active. |

Schemas


For data links to work, Xstudio must first recognize where the data is coming from. This is done by setting up one or more data sources within a schema. Once the data source exists, it can be used over and over again to create data entities (i.e. snapshots, tickers, multi-tickers and classes). When you create a schema, you specify at least one data source and its identifying information (e.g. location, whether it is a database, Excel file, text file, social media provider etc.).

The following topics provide more information and instructions for creating and working with schemas:

- [Creating schemas](#), on page 179
- [Editing schemas](#), on page 181
- [Renaming or saving a copy of a schema with a new name](#), on page 181
- [Deleting schemas](#), on page 181

Creating schemas

To create a schema:

- 1 Click the **Create Schema** icon . This displays the **Schema Settings** dialog box with the **Builder** tab active.

Note: The **XML** tab is for modifying code generated from your data source configuration. You can use this tab, or make your changes on the **Builder** tab.

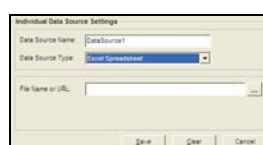
- 2 In the **Schema Name** field enter the name you would like to assign to the schema.

Note: At any time during the data source creation process you can click the **Clear** button to enter all new information in the fields, or the **Cancel** button to cancel the creation process altogether.

- 3 Under **Data Sources**, click the **Add** button.
This enables the fields under **Individual Data Source Settings**.
- 4 In the **Data Source Name** field, enter a name for the individual data source being created in the schema. Bear in mind that data source names cannot be modified once created.
- 5 Select the data source type: **Database**, **Excel Spreadsheet**, **Delimited Text File**, **RSS**, or **Social Media**. Depending on your choice, different fields will display below.



Database



Excel Spreadsheet



Delimited Text File



RSS

- 6 If you chose **Database** from the **Data Source Type** drop-down list:
 - a Make a selection from the **Database Type** drop-down list:
 - **Data Source Name (DSN)**
 - **Microsoft Access (*.mdb)**
 - **Microsoft SQL Server**
 - **Oracle Server**
 - **Microsoft Access 2007 (*.accdb)**
 - b If you chose **Data Source Name**, select either the **User/System DSN** or the **File DSN** radio button. If opting for **User/System DSN**, make a selection from the drop-down list; if opting for **File DSN**, enter the file name in the field provided or click to browse. If required, enter the **user name** and **password**.
 - c If you chose **Microsoft Access**, provide the file name in the **Access Database** field (click to browse).
 - d If you chose **Microsoft SQL Server** or **Oracle Server**, provide the name of the computer hosting the server in the **Server Name** field, the name of the database in the **Database** field, and the **User Name** and **Password** to connect to the server in the appropriate fields.
 - e Click the **Test Connection** button to ensure the connection to the database works (unless you opted for Microsoft Access, in which case it is not necessary). If successful, the database's tables will automatically display.
 - f If desired, you can skip testing the connection and simply click the **Get Tables** button to display the various tables within the database.
 - g Select one or more tables from the **Select related tables from the following list** section to make them available in the data source (hold down the **SHIFT** or **CTRL** key for multiple selections).
 - h Enter the **Database Connection Pool Size**. This specifies the number of database connections that are permitted to be reused.
 - i Enter the **Connect Timeout**: this is a timeout in seconds
- 7 If you chose **Excel Spreadsheet** from the **Data Source Type** drop-down list, provide the required information in the **File Name or URL** field (click to browse).

Note: Only .xls files are supported. Excel 2007 .xlsx files are not supported.
- 8 If you chose **Delimited Text File** from the **Data Source Type** drop-down list:
 - a Provide the required information in the **File Name or URL** field (click to browse).
 - b Specify the characters that will act as the column and row delimiters in the text file. By default, the **Column Delimiter** is set to a comma character and the **Row Delimiter** is set to `\r\n` (carriage return and line feed).
 - c Set the **URL Timeout** property in milliseconds.
 - d If the Column delimiter is set to comma and your text file data contains a text string with an intentional comma (i.e. Smith, Daniel, Jones, Mary), you can avoid the unintentional breaking on commas by putting the text between double quotations (i.e. "Smith, Daniel", "Jones, Mary") and enabling the **Double quotes escape column delimiter** parameter.
- 9 If you chose **RSS** from the **Data Source Type** drop-down list, enter the **URL** in the field provided.

RSS stands for **Rich Site Summary**. This commonly-used XML format distributes news headlines, links and abstracts collected from the Web.

- 10 Click the **Save** button to save the new data source. It displays at left under **Data Sources**.
- 11 Repeat steps 3 to 10 until all data sources for the schema have been created, then click **OK**. This closes the **Schema Settings** dialog box and brings you back to the **Data Manager**. The new schema is listed at left.

Editing schemas

When editing a schema, you are editing data source details. You cannot modify the data source's name -- to do so, you need to access either the Rename or Save As function, which are both explained in the following section.

To edit a schema:

- 1 In the **Data Manager**, right-click on the schema to be edited. A pop-up menu displays.
 - 2 Select **Edit**. This displays the **Schema Settings** dialog box.
 - 3 Select a data source from the list at left.
 - 4 Click the **Edit** button. The **Individual Data Source Settings** fields are enabled. For information on how to use them, refer to the previous section on creating a schema.
- Or,
- Click the **Delete** button to delete the data source.
 - 5 When done making changes, click the **Save** button and then click **OK** to exit the dialog box and return to the Data Manager.

Renaming or saving a copy of a schema with a new name

You can rename a schema by selecting one of two items in the pop-up menu that displays when you right-click a schema:

- **Rename** - This option should only be used if you have just created your schema and would like to rename it.
- **Save As** - This option creates a copy of the schema while keeping the old one. Any links to templates or control panels, for example, are preserved as long as the original schema exists. However, if you delete the old version all data links will break.

Note: If you have already used your schema to create tickers, snapshots etc., it is not recommended that you use the Rename option, since the data entities will be affected by then name change, and data links to objects will be broken. This is due to the fact that the ticker that used to be identified by Xstudio as **Old.MyTicker** becomes **New.MyTicker**.

Deleting schemas

To delete a schema:

- 1 In the **Data Manager**, right-click on the schema to be edited. A pop-up menu displays.

2 Select **Delete**.

Data entities: snapshots, tickers, multi-tickers and classes

Snapshots, tickers, multi-tickers and classes are all data entities -- combinations of specific data from a data source (more than one data source in the case of multi-tickers), and instructions on how to display the data once it has been linked to an object (e.g. in ticker format, as a snapshot value).

The dialog boxes for building a snapshot, ticker and multi-ticker all have the following two tabs:

- **Builder** - Allows you to specify the data to be displayed in the snapshot, ticker or multi-ticker without having to use a query string. The following sections explain how to use the **Builder** tab when setting up data entities.
- **Advanced** - For those who prefer to use a query string to the selection methods provided on the **Builder** tab. The query string is entered in the space provided. The two buttons along the bottom each display a dialog box. Click the **Show XML** button to display a dialog box that shows the XML code (non-editable) being sent to the output device.

Note: The **Class Settings** dialog box does not contain the **Builder** and **Advanced** tab.

The following sections describe how to create and edit data entities:

- [Creating a snapshot](#), on page 183
- [Creating a ticker](#), on page 184
- [Creating a ticker](#), on page 184
- [Creating a multi-ticker](#), on page 187
- [Creating a class](#), on page 187
- [Editing and deleting data entities](#), on page 189

Creating a snapshot

A snapshot is a table of data values that is periodically refreshed. Use a snapshot to display a single data value, in contrast to a ticker which cycles through a set of values.

To create a snapshot:

- 1 From the list of schema names listed at left in the **Data Manager** window, click on the one that the snapshot will be created from and then click the **Create Snapshot** icon



Or,

Right-click on the schema that the snapshot will be created from. When the pop-up menu displays, select **Add > Snapshot**.

This displays the **Data Source Settings** dialog box for snapshots with the **Builder** tab active.

- 2 In the **Snapshot Name** field enter a name for your snapshot.
- 3 Make a selection from the **Select Data Source** drop-down list.
- 4 Select one or more tables from the **Select table(s)** section (hold down the SHIFT or CTRL key for multiple selections)


- 5 Select the columns to be returned from the query in the **Select Return Column(s)**. To change the titles of selected columns, click the **Edit Column Names** button. When the **Column Headers** dialog box displays, click in a column name to make the text editable.
- 6 To have only specific records returned, use the **Set Key** area. To use stock information as an example, you could display only stocks that have gone up or down by \$1.00 or more. Each line in the Set Key area represents a filter on the available data and is expressed as a condition on the value of a column. Select a column name and operator from the drop-down list in their respective columns. The value can either be entered as a constant or can be set to the value of another column by choosing a column name from the drop down list in the **Value** column. Alternatively, it is possible to set a DYNAMIC VALUE by choosing <DYNAMIC> from the drop-down list.
- 7 To sort returned records, use the **Set Sorting Order** table. Click in the **Name** column to display a drop-down list that contains all column names of the selected tables in the data source. Once the column has been selected, click in the **Order** row to display the drop-down list that contains your two choices: **Ascending** and **Descending**.
- 8 Enter the time interval in seconds that will separate data refreshes in the **Refresh Rate** field.
Or,
Select the **Never Refresh** check box. This deactivates the **Refresh Rate** field.
- 9 The **Garbage Collect** field the time interval at which the system will purge outdated instances of data variables.
- 10 Enter the maximum number of rows to be returned by the query in the **Max Rows** field. This will ensure that the system's performance will not be hampered by the retrieval of unnecessarily large tables. If you would prefer that there be no limit on the number of rows returned by the query, enter 0.
- 11 The value of the **Stale Interval** field represents the frequency at which the system checks to see if the data has been updated. If the data has not been updated with the specified time, a warning is sent to the Xstudio log file. Note that the Stale Interval value must either be left blank (default) or be a value greater than the Refresh Rate value.
- 12 Click **OK**.
This closes the **Snapshot Settings** dialog box and brings you back to the **Data Manager**. The new snapshot is listed at left under its schema.

Creating a ticker

Use a ticker when you want to scroll through a data set, displaying one or more items at preset intervals. The item or items will be replaced with new content from a selected source.

Note: A standard ticker shows one full data item at a time. A crawling ticker displays data items moving from right to left, one after the other. For instructions on making your ticker crawl, see [Crawl and Roll objects on the scene canvas](#), on page 104.

To create a ticker:

- 1 From the list of schema names listed at left in the **Data Manager** window, click on the one that the ticker will be created from and then click the **Create Ticker** icon .

Or,

Right-click on the schema that the ticker will be created from. When the pop-up menu displays, select **Add > Ticker**.

This displays the **Ticker Settings** dialog box.
- 2 In the **Ticker Name** field enter a name for your ticker.
- 3 Enter the number of seconds you would like to elapse before displaying the next data item in the **Interval** field.
- 4 Enter the number of data rows that are available for display in the **Items** field.
- 5 Enter the number of rows you would like to increment in the **Increment** field. The number of items must always be equal to or less than the increment number to avoid data items being skipped.
- 6 Specify the number of data items you would like initially sent from the **Data Server** to the output device in the **Burst** field. If no burst is required, leave the count at 0.

Note: For time-sensitive data such as stock quotes, the **Burst** method may not be appropriate, since data items that have been “bursted” may be out of date by the time they make it to air. To get around this, see [Crawl and Roll objects on the scene canvas](#), on page 104.

The **Burst** setting is used with crawling or rolling tickers, and ensures that the first data item will not repeat itself if subsequent data items are not sent to the device in time. If using the **Burst** setting, it must be reset every time you go to **Production** mode. The steps for doing so are:

- a Click anywhere in the control panel background, and then display the **Action Editor** (CTRL+E).
- b Select **On Load** from the events listed at left, and then select **Ticker Command** from the drop-down list of actions.
- c In the **Action Parameters** that display at right, click in the **Type** field to access the drop-down list, and then select **Reset**. This is what will restart the crawl when you go to Production mode.

In the **Target** field, enter the **data source**, e.g.

Data:Schools:SchoolClosures(SchoolName). The data source can be dragged from the Data Manager dialog. Select the ticker to be used as the target and drag the name of the ticker from the top of the right side of the Data Manager dialog.

- d Add another **Ticker Command**. For this one, select **Start** as the **Type**, and enter the same **Target** as in the previous step.
- 7 To refresh ticker data only after it has gone through the entire table, select the **Refresh only at end** check box.
 - 8 To have the ticker start running as soon as its host control panel or page is activated, select the **Start By Default** check box (the only way to start it otherwise is by setting up an action; see [Assigning an action to an object](#), on page 221 for details).

- 9 To have the ticker wrap, select the **Wrap around** check box. This feature is useful if your ticker is showing, say, three data items at a time and you want to ensure that if you reach the end of your table and there are only two items left, that you don't end up with a blank item. In this case, if **Wrap around** were selected the first item from the table would go to air as the last item.
- 10 To have the ticker start at the beginning of the table and play again after it has reached the end, select the **Loop** check box.
- 11 Enable the **Individualised** property when you want to allow each client (template) to control the state of the ticker independently. For example, if two templates are using an individualized ticker, one template may choose to stop the ticker while the other may continue ticking. If the ticker was not individualized, if one template stopped the ticker, updates would stop on the second template as well.
- 12 Make a selection from the **Select Data Source** drop-down list.
- 13 Select one or more tables from the **SELECT TABLE(S)** section (hold down the **SHIFT** or **CTRL** key for multiple selections).
- 14 To change the titles of selected columns, click the **Edit Column Names** button. When the **Column Headers** dialog box displays, click in the column name to be changed to make the text editable.
- 15 To have only specific records returned, use the **Set Key** area. To use stock information as an example, you could display only stocks that have gone up or down by \$1.00 or more. Each line in the Set Key area represents a filter on the available data and is expressed as a condition on the value of a column. Select a column name and operator from the drop-down list in their respective columns. The value can either be entered as a constant or can be set to the value of another column by choosing a column name from the drop down list in the **Value** column. Alternatively, it is possible to set a **DYNAMIC VALUE** by choosing <DYNAMIC> from the drop-down list.
- 16 To sort data, use the **Set Sorting Order** table. Click in the **Name** column to display a drop-down list that contains all column names of the selected tables in the data source. Once the column has been selected, click in the **Order** row to display the drop-down list that contains your two choices: **Ascending** and **Descending**.
- 17 Enter the time interval in seconds that will separate data refreshes in the **Refresh Rate** field.
Or, select the **Never Refresh** check box. This deactivates the **Refresh Rate** field.
- 18 The **Garbage Collect** field the time interval at which the system will purge outdated instances of data variables.
- 19 Enter the maximum number of rows to be returned by the query in the **Max Rows** field. This will ensure that the system's performance will not be hampered by the retrieval of unnecessarily large tables. If you would prefer that there be no limit on the number of rows returned by the query, enter 0.
- 20 The value of the **Stale Interval** field represents the frequency at which the system checks to see if the data has been updated. If the data has not been updated with the specified time, a warning is sent to the Xstudio log file. Note that the Stale Interval value must either be left blank (default) or be a value greater than the Refresh Rate value.
- 21 Click **OK**.


This closes the **Ticker Settings** dialog box and brings you back to the Data Manager. The new ticker is listed at left under its schema.

Creating a multi-ticker

A standard ticker shows one full data item at a time. A crawling ticker displays data items moving from right to left, one after the other. For instructions on making your ticker crawl, see [Crawl and Roll objects on the scene canvas](#), on page 104.

A multi-ticker looks and acts exactly like a ticker, except that it takes its data from more than one data source within a single schema. This allows you to run through all data from one and then immediately move on to the next one.

To create a multi-ticker:


- 1 From the list of schema names listed at left in the **Data Manager** window, click on the one that the multi-ticker will be created from and then click the **Create Multi-Ticker** icon .
Or,
You are limited to the data sources within a single schema for multi-tickers.
Right-click on the schema from which the multi-ticker will be created. When the pop-up menu displays, select **Add > Multi-Ticker**.
This displays the **Data Source Settings** dialog box.
- 2 In the Ticker Name field enter a name for the ticker.
- 3 Follow steps 3 to 10 for [Creating a ticker](#), on page 184 to complete the **Ticker Settings** fields.
- 4 Under **Sub-Tickers** click the **Add** button. This enables the fields under Individual **Sub-Ticker Settings**. The sub-tickers are individual tickers that make up the multi-ticker.
- 5 Under **Individual Sub-Ticker Settings** enter a name for the sub-ticker in the **Sub-Ticker Name** field.
- 6 Follow steps 11 to 20 for [Creating a ticker](#), on page 184 to complete the rest of the sub-ticker information.
- 7 Click the **Save** button to save the new sub-ticker. It displays at left under **Sub-Tickers**.
- 8 Repeat steps 4 to 7 from this procedure until all data sources for the multi-ticker have been created, then click **OK**. This closes the **Multi-Ticker Settings** dialog box and brings you back to the Data Manager. The new multi-ticker is listed at left under its schema.

Creating a class

A class is another way of bringing data into the Data Server that can be retrieved by Xstudio. If none of the standard data entity types suit your needs, or if you would like actions performed on the data so that it appears on air differently than it does in the database, then you will require a class. For example, let's say you have a live data source bringing in numbers of votes for the top five music videos of the day. Any of the standard data entity types would show the numbers. However, with a class you could translate them into percentages. Classes are created from XML code.

Classes are an advanced feature requiring knowledge of class definition XML. Contact our Technical Support team if you would like a class created for you, or if you would like to obtain the Developer's Kit to see examples of classes on which you can base your own.

To create a class:

- 1 From the list of schema names listed at left in the **Data Manager** window, click on the one that the class will be created from and then click the **Create Class** icon .

Or,

Right-click on the schema that the class will be created from. When the pop-up menu displays, select **Add > Class**.

This displays the **Data Source Settings** dialog box.

- 2 In the **Class Name** field enter a name for your class.

Note: Make sure that what you enter in the **Class Name** field matches what you enter in the **class definition**.³ Enter the **Class Definition XML** in the box provided.

- 4 Click **OK**.

Setting dynamic keys

There may be cases where you would like the operator to call up specific data by entering particular information (e.g. entering a political candidate's ID number to display corresponding first and last name, election riding, etc.). This would be done by setting up a dynamic key.

To set a dynamic key (example):

- 1 Create the data entity as usual. For this example, a snapshot will be used.
- 2 In the **Select Return Column(s)** area of the **Snapshot Settings** dialog box, select the columns containing the information you would like returned (e.g. first and last name) when the operator enters the key information.
- 3 In the **Set Key(s)** area:
 - a From the **Name** column, select the column name containing the data that the operator will enter in order to call up the specific data from the selected return columns.
 - b From the Operator column, **select =**.
 - c From the **Value** column, select **Dynamic**.
 - d Press **ENTER** to register the key.

When you have finished creating the data entity, it should look something like this:

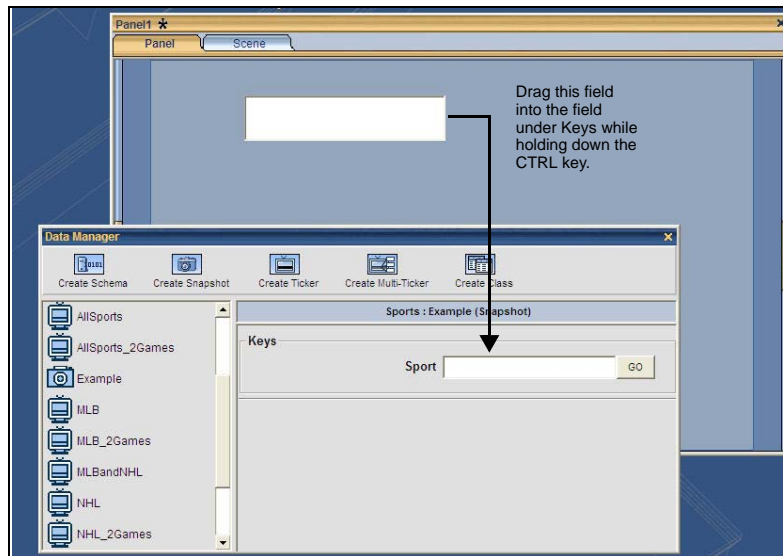
Entering a number in the **Key** field (in this case, the Sport field) yields corresponding data in the **Data Members** fields below.

How can this be used in a control panel? The following steps show you how.

To use key data in a control panel:

- 1 Open a new control panel or control panel + scene.
- 2 On the **Panel** tab, create one text box into which key data will be added, and another text box for every selected return column.

- 3 Holding down the **CTRL** key, drag the first text box into the **Key** field in the **Data Manager**. This designates the text box as the one that will provide the data.



- 4 Test what you have so far by switching to **Production** mode and entering data in the first field (the one set up as the data provider). The corresponding data appears in the other fields.
- 5 If desired, complete the exercise by creating Text 2D objects on the **Scene** tab, and then connect them to the text boxes displaying returned data on the **Panel** tab (see [page 162](#) for instructions).

Editing and deleting data entities

You can change or delete an entire schema or one of its data entities at any time.

To edit or delete data entities:

- 1 Right-click on the data entity. This displays a pop-up menu.
 - 2 Select **Edit** and modify as required, and then click **OK**.
- Or,
- Select **Delete**.

Renaming or saving a copy of a data entity with a new name

The same issues surrounding the **Rename** versus the **Save As** functions for schemas also applies to snapshots, tickers, multi-tickers and classes (see [Renaming or saving a copy of a schema with a new name](#), on page 181). Basically, renaming is risky unless you have just created the data entity. Any links you have made with it in templates and control panels will be broken.

Saving a data entity as something else creates a copy of it while keeping the original and any links created with it.

Creating data links

Data linking is the process of connecting data to objects. For example, to display headlines from a newswire feed you must link the data source to the object displaying the data (e.g. a text box). Data links are always made to an object on the control side, and then the control object is connected to a scene object. You cannot link directly to a scene object.

These links are made with the Data Manager, and are created from either a snapshot, ticker or class. See [page 177](#) for further information on data entities and the Data Manager.

To create a data link:

- 1 Select the control object.
- 2 In the **Data Manager**, click on a schema to display its data entities.
- 3 Select a data entity. Its contents are displayed at right.
- 4 Drag a cell from the upper white data table either directly onto the object on the template or control panel, or over to the DATA SOURCE row under LINKS in the Property Browser.

Prefixes (Scene:, Control:, Data:, L:, G:)

Xstudio uses prefixes to specify the origins of objects and data. For example, when assigning data to an object you must indicate whether the object is part of the scene or the control side.

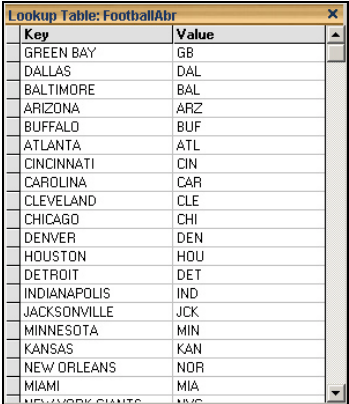
While you can type these in along with the object name or data source, Xstudio's drag and drop functionality makes it unnecessary for the most part by automatically filling all required information.

| Prefix | Signifies |
|-----------------|---|
| Scene: | A scene object, e.g. <code>Scene:Clip1Control:DropboxMonth</code> . |
| Control: | A control object, e.g. Control:MyControlObject . |
| Data: | A data item from the Data Server, e.g. Data:Schools:SchoolsClosed(1,1) . |
| L: | A local data server variable, e.g. L:SelectedStock . |
| G: | A global data server variable, e.g. G:StationID . |

Local and global database variables are created on the fly whenever a reference to one is made in a property value or action parameter. You are not required to create these variables in the Data Manager. It is also possible for controls to set the value of these variables such that other controls may read them. Local database variables can only be used within the control panel that created them and are destroyed when the panel exits Production mode. Global database variables, on the other hand, are shared amongst all control panels and remain in the database even after exiting Production mode.

Creating and applying lookup tables

A lookup table is a way of defining replacement values for incoming data. For example, say you have a live data source feeding in sports information. City names are provided in full by the data source, but you want it to display an abbreviation. You would set up a lookup table that looks something like this:



| Key | Value |
|-----------------|-------|
| GREEN BAY | GB |
| DALLAS | DAL |
| BALTIMORE | BAL |
| ARIZONA | ARZ |
| BUFFALO | BUF |
| ATLANTA | ATL |
| CINCINNATI | CIN |
| CAROLINA | CAR |
| CLEVELAND | CLE |
| CHICAGO | CHI |
| DENVER | DEN |
| HOUSTON | HOU |
| DETROIT | DET |
| INDIANAPOLIS | IND |
| JACKSONVILLE | JCK |
| MINNESOTA | MIN |
| KANSAS | KAN |
| NEW ORLEANS | NOR |
| MIAMI | MIA |
| NEW YORK GIANTS | NYG |

Note: An asterisk in the title of the lookup table indicates that unsaved changes have been made.

The **Key** column contains the items that require substitution, whereas the **Value** column contains the replacements.

One text string can be replaced by another. However, you can also use these tables to replace text strings with asset names, e.g. images. Assets are added to the lookup table by dragging them into the value column from the asset browser. They are displayed in bold to distinguish them from text.



| Key | Value |
|--------------------------|-----------------------|
| Eminem | pic_eminem |
| Linkin Park | pic_linkinpark |
| Eminem | pic_eminem |
| Pink | pic_pink |
| Eminem | pic_eminem |
| Mary J. Blige | pic_maryblige |
| Avril Lavigne | pic_avril |
| Michelle Branch | pic_michelle |
| No Doubt / Bounty Killer | pic_nodoubt |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

To create a new lookup table:

- 1 Select the **File>New** menu command. This displays the **Create New** dialog box.
- 2 Select the **Other** tab and **Lookup Table**. This opens a new lookup table.
- 3 In the **Key** column enter the string expressions to be flagged for substitution.
- 4 In the **Value** column enter the substitution value. This can be:
 - another text string (text or numbers)

- an asset (when dragged into the Value column it appears in bold to indicate that it's an asset)
- 5 To delete the contents of a row, click in the gray cell to the left of the row (this selects the entire row) and press the **DELETE** key.
 - 6 To save the lookup table, press **CTRL + S**.

Now that the lookup table exists it must be applied to a specific data source. Refer to [Applying lookup tables](#) below.

Applying lookup tables

A lookup table will not work until you apply it to specific data or text.

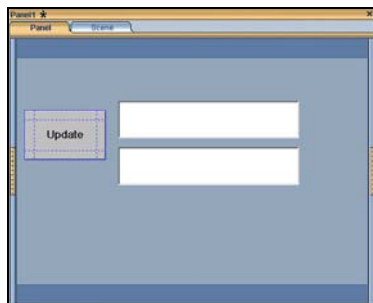
How you apply a lookup table depends on where the data is coming from:

- If data is being fed in from a data source, you would use the **Lookup Table** row in the **Properties Browser**.
- If the lookup table is replacing text coming from somewhere other than a data source, e.g. the contents of a text box being set into another text box, you would use the **Expression Builder** to create a LOOKUP expression. Since this requires using the Action Editor, you may want to read the chapter [Adding logic to control objects](#), on page 211 (which also includes a section on the Expression Builder) before proceeding to the instructions on how to create a LOOKUP expression.

To apply a lookup table to data from a data source:


- 1 Open the asset to which the lookup table will be applied (e.g. control panel or template).
- 2 Select the control object to which you would like the lookup table applied.
- 3 Display the **Properties Browser** (**CTRL + R**).
- 4 From the **Asset Browser**, drag the required lookup table over to the Property Browser and drop over the **Lookup Table** row. This automatically populates both the Lookup Table and Lookup Location rows.

Let's say you have a control panel with two text boxes and an update button. You want to take the text from Textbox1, apply a lookup table to it, and then set it into Textbox2 when you click the **Update** button.




To apply a lookup table to text coming from somewhere other than a data source (using the above example):

- 1 Select the button and display the **Action Editor** (**CTRL + E**).
For full details see the Action Editor on [page 211](#).
- 2 In the Action Editor, select the **On Click** event.

- 3 Select the **Set Text** action.
- 4 In the Action Parameters section, click in the **Source** field and then click . This displays the Expression Builder.
- 5 Click the **Clear** button to delete **Control:Button1** from the field at the top of the Expression Builder.
- 6 From the pane at left, select **Function**. This displays a list of functions at right.
- 7 From the list of functions, double-click **LOOKUP**. This displays the function in the field at the top of the Expression Builder.
- 8 In the field at the top of the Expression Builder, select **<string>**.
- 9 From the pane at left, select **Control**. This displays a list of the control-side contents.
- 10 From the **Contents** list, double-click **Textbox1**. This automatically replaces **<string>** in the field up top.

Next you will enter a lookup table name. This example will use the **DaysOfTheWeek** lookup table.

| Lookup Table: DaysOfTheWeek | |
|-----------------------------|-------|
| Key | Value |
| Monday | MON |
| Tuesday | TUE |
| Wednesday | WED |
| Thursday | THU |
| Friday | FRI |
| Saturday | SAT |
| Sunday | SUN |

- 11 In the field at the top of the **Expression Builder**, select **<lookupTable>**, and replace it with the name of the lookup table in quotation marks.
 - 12 Click **OK** to close the Expression Builder and apply the expression to the **Source** field in the Action Editor.
 - 13 In the Action Parameters section, click in the **Target** field and then click . This displays the **Target/Source Selector**. You will use this to select where you would like to set the text from **Textbox1**.
 - 14 From the left window pane select **Control**, and then from the list of contents that displays at right double-click **Textbox2** and then click the **OK** button. This closes the Target/Source Selector.
- The **On Click** action for the button is now complete.
- 15 Switch to **Production mode (CTRL + P)**. Enter days of the week in **Textbox1**, and note how the lookup table equivalent appears in **Textbox2** when you click the **Update** button.

Editing lookup tables

To edit a lookup table:

- 1 Open the lookup table by double-clicking it under the **LookupTable** root category in the **Asset Browser**.
- 2 Click in the rows to be edited and re-type as required.
Or,
Drag and drop assets from the **Asset Browser**.
- 3 Press **CTRL + S** to save your changes.

Using social media messages and online polls as a data source

Xstudio supports the retrieval of social media messages and online poll results using the Data Manager. Social media content is obtained from providers who aggregate, sort and filter raw content from original sources such as Twitter, Facebook and Instagram and make the content available on a website. Currently, three social media providers are supported: Telescope, Spredfast and Twitter Curator.

The following sections provide information and instructions for using these social media features:

- [Acquiring and installing the Social Media Interface license](#), on page 195
- [Creating a social media data source](#), on page 195
- [Creating a snapshot or ticker for social media content](#), on page 196
- [Data tables for social media snapshots and tickers](#), on page 200
- [Creating a template that retrieves social media content](#), on page 206

Acquiring and installing the Social Media Interface license

Xstudio's support of social media messages and polls as a data source requires that a valid **Social Media Interface** license be registered in the License Manager. If a valid license is not present, the Social Media will not appear in the **Data Source Type** drop-down list in the **Schema Settings** window.

Please contact Grass Valley's Support team to acquire a license and then consult the Vertigo Suite's Release Notes for installation instructions.

Creating a social media data source

To use social media content, you must first define the data sources that retrieve information from social media providers in Xstudio's Data Manager.

To create a social media data source:

- 1 Open the **Data Manager**.
- 2 Create a new schema by clicking the **Create Schema** button ([page 179](#)), or open an existing schema.
- 3 Click **Add** in the **Data Sources** section of the Schema Settings window.
- 4 In the Data Source Name field, enter a name for the data source.
- 5 In the Data Source Type drop-down list, select **social media**.

More fields automatically appear within the Individual Data Source Settings section.

Note: If **Social Media** does not appear in the Data Source Type drop-down list, verify the validity of your **Social Media Interface** license (see [Acquiring and installing the Social Media Interface license](#), on page 195).

- 6 In the Provider drop-down list, select **Spredfast**, **Telescope** or **Twitter Curator**.
- 7 In the **Type** drop-down list, select **Messages** or **Poll**.

- If you selected **Spredfast** and **Messages** or **Polls**, enter the Spredfast account name in the **Account** field and click **Save**.
- If you selected **Telescope** and **Messages**, no other fields appear and you simply have to click **Save**.
- If you selected **Telescope** and **Polls**, complete the **API Key** and **Secret Key** fields with the information provided by Telescope and click **Save**.
- If you selected **Twitter Curator** and **Messages** or **Polls**, enter your Twitter Curator authorization token in the **Authorization Token** field and click **Save**.

The Data Source is immediately added to the list of Data Sources in the Schema Settings window.

- 8 Click **OK** to close the Schema Settings window and the new schema is added to the Data Manager window.

Creating a snapshot or ticker for social media content

Once a social media data source has been defined, you can create a ticker or snapshot based on the data source. Typically, tickers are used for message streams and snapshots are used for polls.

To create a snapshot associated to a social media data source:

- 1 Open the Data Manager.
- 2 Select the schema that contains the social media data source.
- 3 Click **Create Snapshot** or **Create Ticker**.

The Settings window appears.

- 4 In the **Snapshot Name** or **Ticker Name** field, enter a name for the snapshot or ticker.
- 5 If you are creating a ticker, complete the following properties in the **Ticker Setting** section:
 - **Interval (sec)**: The number of seconds you would like to elapse before displaying the next data item in the **Interval** field.
 - **Items**: The number of data rows that are available for display in the **Items** field.
 - **Increment**: The number of rows you would like to increment. The number of items must always be equal to or less than the increment number to avoid data items being skipped.
 - **Burst**: The number of data items you would like initially sent from Data Server to the output device. If no Burst is required, leave the value set to 0. See [page 185](#) for more information.
 - **Refresh only at end**: Check the check box to refresh ticker data only after it has gone through the entire table.
 - **Start by default**: Check the check box to have the ticker start running as soon as its host control panel or page is activated.
 - **Wrap around**: Check the check box to have the ticker wrap. See [page 185](#) for more information.
 - **Loop**: Check the check box to have the ticker start at the beginning of the table and play again after it has reached the end. See [page 185](#) for more information.

- **Individualised:** Check the check box when you want to allow each client (template) to control the state of the ticker independently. See [page 185](#) for more information.

6 In the **Select Data Source** drop-down list, select a social media data source.

Additional fields automatically appear which allow you to identify the specific message stream or poll. These fields vary according to the **Provider** and **Type** defined in the selected data source.

7 Complete the fields specific to the social media provider:

| Field Name | Description |
|--|---|
| <ul style="list-style-type: none"> • Stream • Poll Stream • Topic ID • Campaign ID | Specify the specific message stream name or poll identifier and leave the Dynamic field directly below unchecked.. |
| <ul style="list-style-type: none"> • Collection ID (Twitter Curator messages only) | Specify the Twitter Curator Collection ID that contains the message stream that is to be displayed. Leave the Dynamic field directly below unchecked. |
| <ul style="list-style-type: none"> • Project ID • Stream IDs (Twitter Curator Polls only) | <p>If you want poll vote counts for every stream in the project, enter the Twitter Curator project ID in the Project ID text box. Leave the Dynamic field directly below unchecked.</p> <p>If you want to poll vote counts for only specific streams, select Stream IDs from the drop-down list and type a comma separated list of Stream IDs (with no spaces between items) in the text box. Leave the Dynamic field directly below unchecked.</p> |
| Dynamic | <p>Leave the check box empty if you have entered a specific message stream name or poll identifier in the fields above.</p> <p>Check the check box if you want the message stream or poll identifier to be undefined, so that it can be specified later in the Template that uses the snapshot or ticker.</p> <p>A dynamic identifier permits the creation of a Template that can be reused to show different message streams or polls. For example, it is possible to create a standard Template for displaying social media messages that contains a text box in which the desired message stream identifier is entered. It is then easy to create a set of Pages based on the Template where each Page displays a different message stream. This is accomplished by entering the desired message stream identifier into the text box in each page.</p> <p>If a ticker or snapshot with a dynamic identifier is selected in the Data Manager, then an entry field is displayed where a specific identifier can be entered. After the identifier is entered, click Go and the data corresponding to that identifier is displayed in the data table. The identifier can be changed to show different data.</p> |

| Field Name | Description |
|---|--|
| Start Time (Twitter Curator Polls only) | <p>A Twitter Curator poll counts votes that are cast starting from a specific time. The start time at which poll counting begins is specified in the Start Time text box.</p> <p>The Start Time value is a date and time expressed as a UNIX Timestamp in milliseconds. Go to http://www.epochconverter.com/ to generate UNIX timestamps from a date. The maximum value for the time span is 30 days.</p> <p>Alternatively, a Start Time value of 0 indicates that the poll only counts votes from the last hour. Note that this time span is a "moving window". One hour after the poll is created, older votes will start to be discarded.</p> |
| Poll > Ordered by | <p>Poll results are retrieved from the provider and stored in a table with one row for each poll choice. The Order By property indicates how the rows are ordered in the table. Certain orderings may be more convenient to view in the Data Manager or may make the creation of Templates more convenient.</p> <p>Select one of the following options from the drop-down list:</p> <ul style="list-style-type: none">• Default: The rows are kept in the same order as returned from the provider's web site.• Votes: The rows are ordered by number of votes from greatest to least number of votes. Rows with the same number of votes are ordered in ascending alphabetical order by choice name.• Alphabetic: The rows are ordered in ascending alphabetical order by choice name. <p>Note: The ordering selected does not restrict the manner in which the results can be displayed.</p> |

| Field Name | Description |
|---|--|
| Poll > Percent Decimal Places | <p>The Percent Decimal Places property indicates how to round percent values.</p> <p>Select one of the following options from the drop-down list: 0, 1, 2, or 3.</p> <p>The percent value for a choice is calculated by dividing the number of votes for the choice by the total number of votes and then rounding the result. If no votes have been cast for any choice, the percent value is set to 0. If one or more decimal places are selected, percent values are padded with zeros, if necessary, so that all percent values show the same number of decimal places.</p> |
| Poll > Poll Choices (Telescope Polls only) | <p>Provides the list of all possible choice names in the poll. This property is present because, in the case of Telescope polls, the results returned by the provider do not contain entries for choices that have zero votes. The names in this list are used to provide the missing entries. The names provided must match exactly the choice titles (as opposed to hash tags or keywords used to vote for the choice) entered in the poll definition on the provider's web page. It is important that the case of the choice names entered match those in the poll definition. However, the order of the choice names is not important.</p> <p>This property is only available when a static poll identifier is used. If a dynamic poll identifier is used, the Poll Choices are disabled, because the poll referred to by the ticker or snapshot is unknown until the dynamic poll identifier is specified in a template.</p> <p>Note: It is possible to take steps to ensure that the poll results returned by Telescope will always contain an entry for every choice (consult Telescope for details). If these steps are taken, the choice names list can be left blank and a dynamic poll identifier can be used.</p> |

- 8 In the **Refresh Rate** field, enter the time interval in seconds that will separate data refreshes.
Or, select the **Never Refresh** check box. This deactivates the **Refresh Rate** field.
- 9 In the **Garbage Collect** field, specify the time interval at which the system will purge outdated instances of data variables.
- 10 In the **Max Rows** field, enter the maximum number of rows to be returned by the query. This will ensure that the system's performance will not be hampered by the retrieval of unnecessarily large tables. If you would prefer that there be no limit on the number of rows returned by the query, enter 0.
- 11 The value of the **Stale Interval** field represents the frequency at which the system checks to see if the data has been updated. If the data has not been updated with the specified time, a warning is sent to the Xstudio log file. Note that the Stale Interval value must either be left blank (default) or be a value greater than the Refresh Rate value.
- 12 Click **OK**.
This closes the **Settings** window and brings you back to the **Data Manager**. The new snapshot or ticker is listed at left under its schema.

Data tables for social media snapshots and tickers

The data tables used by snapshots and tickers have a fixed set of columns determined by the choice of social media provider and type. The number of rows in the table varies according to the data that is present. The columns present for the different types of tables are outlined in the following sections:

- [Telescope and Twitter Curator messages data table columns](#), on page 200
- [Spredfast messages data table columns](#), on page 202
- [Spredfast, Telescope and Twitter Curator polls data tables columns](#), on page 205

Telescope and Twitter Curator messages data table columns

The table below identifies the columns that are populated in the data tables for Telescope and Twitter Curator messages. There is one row in the table for each message.

The example below demonstrates one way of using the information provided in the different columns to display a message.



MessageId

A unique message identifier that is guaranteed to be different for every message.

Example: 4783947938943

UserName

The unique identifier of the message poster. In the case of Facebook messages, this is the user id number.

Example: dmcormick5

ScreenName

The readable identifier of the message poster.

Example: Doug McCormick

Avatar

The url of the message poster's avatar image.

Example:

http://pbs.twimg.com/profile_images/473465519791360/c9iPjGKx_normal.jpeg

Source

The source from where the message was posted.

- twitter
- facebook
- instagram

Example: twitter

Message

The text message that was posted.

Example: Fire Marshal Wieclawek speaks to CTV about the importance of having CO & smoke alarms in #Ontario homes.

Image

The url of an image included with the message. This field is blank if no image is provided.

Example: http://scontent-b.cdninstagram.com/hphotos-xpf1/t51.2885-15/10375679_1427997170813710_24520238_n.jpg

IsRetweet

Indicates if the post is a retweet. This is always false for Facebook and Instagram posts.

- true
- false

Example: false

Spredfast messages data table columns

The table below identifies the columns that are populated in the data tables for Spredfast, messages. There is one row in the table for each message.

MessageId

A unique message identifier that is guaranteed to be different for every message.

Example: 4783947938943

UserName

The unique identifier of the message poster. In the case of Facebook messages, this is the user id number.

Example: dmcormick5

ScreenName

The readable identifier of the message poster.

Example: Doug McCormick

Avatar

The url of the message poster's avatar image.

Example:

http://pbs.twimg.com/profile_images/473465519791360/c9iPjGKx_normal.jpeg

Source

The source from where the message was posted.

- twitter
- facebook
- instagram

Example: twitter

Message

The text message that was posted.

Example: Fire Marshal Wieclawek speaks to CTV about the importance of having CO & smoke alarms in #Ontario homes.

Image

The url of an image included with the message. This field is blank if no image is provided.

Example: http://scontent-b.cdninstagram.com/hphotos-xpf1/t51.2885-15/10375679_1427997170813710_24520238_n.jpg

IsRetweet

Indicates if the post is a retweet. This is always false for Facebook and Instagram posts.

- true
- false

Example: false

Note: When using Spredfast messages, in the case of a retweet the table contains information from the original post and not the retweeted post. However, information about the retweeter is available in the `RetweeterUserName`, `RetweeterScreenName` and `RetweeterAvatar` columns. The `IsRetweet` column indicates whether or not a post is a retweet.

RetweeterUserName

When the post is a retweet, gives the unique identifier of the user that retweeted the message.

Example: leyfre

RetweeterScreenName

When the post is a retweet, gives the readable identifier of the user that retweeted the message.

Example: Frederic Leymarie

RetweeterAvatar

When the post is a retweet, gives the avatar of the user that retweeted the message.

Example:

http://pbs.twimg.com/profile_images/47352360/c9iPjGKx_normal.jpeg

Kind

Indicates the kind of post in Facebook or Instagram.

The following choices only apply to Facebook:

- **status:** A simple message that is contained in the Message column. The LinkName, LinkCaption, and LinkDescription columns are blank.
- **comment:** A simple message that is contained in the Message column. The LinkName, LinkCaption, and LinkDescription columns are blank.
- **link:** A link to a URL. There may be text in the Message column. The LinkName, LinkCaption and LinkDescription columns contain information about the link.
- **photo:** A link to a photo. There may be text in the Message column. The LinkName, LinkCaption and LinkDescription columns contain information about the photo.
- **video:** A link to a video. There may be text in the Message column. The LinkName, LinkCaption and LinkDescription columns contain information about the video. The Image column will contain the URL of an image that shows one frame from the video.

The following choice only applies to Instagram:

- **instagram_media:** A simple message that is contained in the Message column. The LinkName, LinkCaption, and LinkDescription columns are blank.

LinkName

This column is only used for Facebook messages where the kind is link, photo or video. This column provides the name of the link, photo or video. It may contain a short piece of text.

Example: Super Marcie Baseball Cap

LinkCaption

This column is only used for Facebook messages where the kind is link, photo or video. This column provides the caption of the link, photo or video. It may contain a short piece of text but may also be a lengthy message.

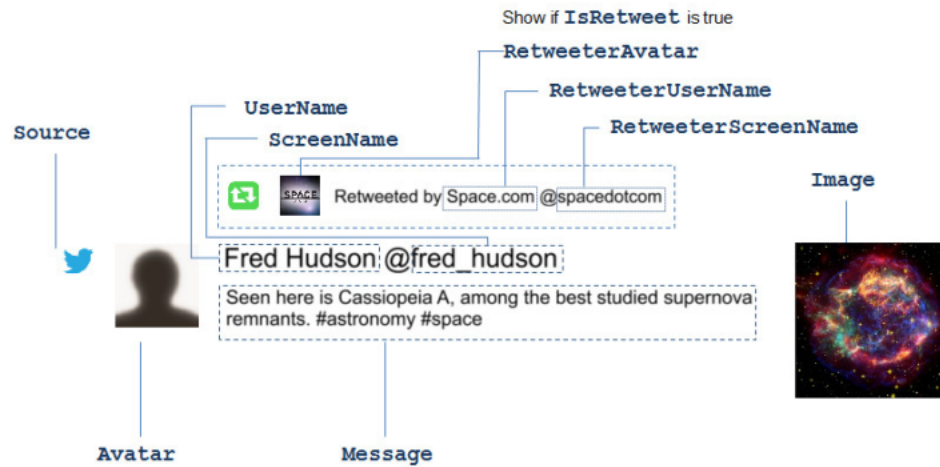
Example: Today marks the beginning of this year's Minor League Baseball season #MascotMania

LinkDescription

This column is only used for Facebook messages where the kind is link, photo or video. This column provides a description of the link, photo or video. It may contain a lengthy message.

Example: Flying Squirrels broadcaster Jon Laaser is joined by Wes McElroy and Jay Bernham as they talk about the potential post-season baseball.

The example below demonstrates one way using the information provided in the different columns to display a message:



The example below demonstrates one possible usage of the LinkName, LinkComment and LinkDescription columns for a Facebook message with kind set to link, photo or link:



Spredfast, Telescope and Twitter Curator polls data tables columns

Spredfast, Telescope and Twitter Curator polls provide the same set of columns, which are defined in the following table:

ChoiceName

The readable title of the choice entered in the poll definition, as opposed to hashtags or keywords used to identify votes.

Example: Germany

Votes

The number of votes cast for the choice.

Example: 352

Percent

The percent of the total number of votes cast for the choice. The percent value for a choice is calculated by dividing the number of votes for the choice by the total number of votes and then rounding the result. This number is rounded as indicated by the **Percent decimal** places property in the ticker or snapshot definition. If there are no votes cast at all in the entire poll, the percent is set to 0.

Example: 62

UniqueRank

This column contains a ranking of the choices from highest number of votes to lowest number of votes without any ties. When multiple rows have the same number of votes they are ordered in ascending alphabetical order by choice name. Every choice is guaranteed to have a unique rank that is in the range from 1 to the number of choices. This value can be used to order the choices when it is desired to display them in ranked order.

Example: 2

Rank

This column contains a ranking of the choices from highest number of votes to lowest number of votes. Unlike the **UniqueRank** column, multiple choices can have the same rank if there is a tie. This value can be used to indicate which choices ranked 1st or 2nd, with the possibility of indicating ties. Note that if, for example, there are two choices tied for 2nd place, the next choice will be in 4th place rather than 3rd place.

Example: 1

TieCount

this value is used in association with the **Rank** to indicate the number of choices tied with the same rank. When there is no tie for a given rank the tie count is 1, a two-way tie has a tie count of 2 etc.

Example: 1

The following example shows a world cup poll with five possible choices. Note that Portugal and Switzerland both have the same number of votes.

| ChoiceName | Votes | Percent | UniqueRank | Rank | Tie Count |
|-------------|-------|---------|------------|------|-----------|
| Argentina | 347 | 30 | 2 | 2 | 1 |
| Brazil | 529 | 46 | 1 | 1 | 1 |
| Cameroon | 23 | 2 | 5 | 5 | 1 |
| Portugal | 121 | 11 | 3 | 3 | 2 |
| Switzerland | 121 | 11 | 4 | 3 | 2 |

While the **Rank** column indicates that Portugal and Switzerland are tied for 3rd place, in the **UniqueRank** column Portugal is ranked before Switzerland based on alphabetical ordering. The **TieCount** is 1 where there is no tie, but is 2 for Portugal and Switzerland who are in a two-way tie.

Creating a template that retrieves social media content

Note: Consult the following website to obtain guidelines for using Tweets in broadcast: <https://support.twitter.com/articles/114233>.

After defining data sources, snapshots and tickers, the social media content can be used in a template. In template logic, a single item of data is specified using a *data descriptor*. This data descriptor can be used in the property of a control or in the parameter of an Action or Event.

The **DataSource** property of the text box is set to the data descriptor for the cell that was dragged out of the table: `Data:SocialMedia:Ticker(UserName, 1)`

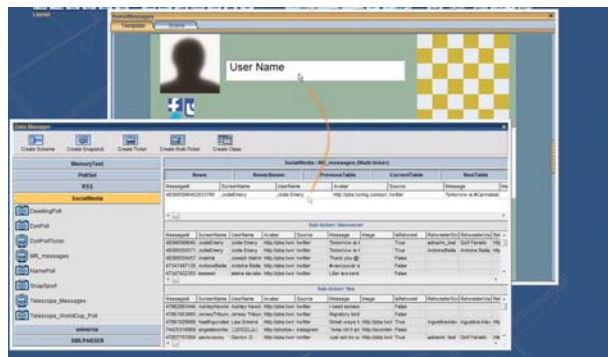


Fig. 9-1: Drag & drop to associate a data descriptor with a template control

The following sections describe how to use specific types of data descriptors to link social media data source to template controls or to an Action or Event:

- [Data linking using a basic data descriptor](#), on page 207
- [Data linking using a Dynamic identifier](#), on page 207
- [Data linking using the poll identifier from a local template variable](#), on page 208
- [Using Dynamic rows](#), on page 209

Data linking using a basic data descriptor

If the snapshot or ticker was configured with a specific message stream or poll identifier (no dynamic identifier), dragging a cell from the data table in the Data Manager to the property of a control or the parameter of an Action or Event creates a basic descriptor in the following form: `Data:Schema:TickerOrSnapshot(columnName, row)`

For example, dragging from the **UserName** cell from the first row of the table to the **DataSource** property of a text box results in the following data descriptor:

```
Data:SocialMedia:MR_Messages(UserName,1)
```

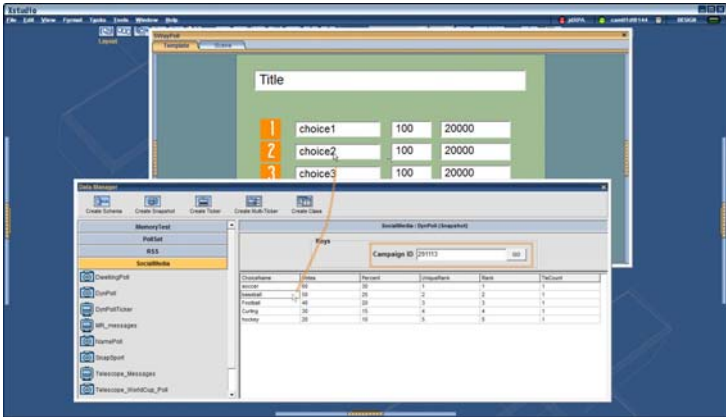
| Field | Description |
|--------------------|---|
| Data | A prefix used in template logic to indicate a data server variable, as opposed to a control property or constant. |
| SocialMedia | The schema name. The schema must have been defined in the Data Manager (see Creating a social media data source , on page 195). |
| MR_Messages | The name of a ticker in the specified schema. The ticker must have been defined in the Data Manager (see Creating a snapshot or ticker for social media content , on page 196). |
| UserName | The UserName column of the data table. |
| 1 | The first row of the data table. |

Data linking using a Dynamic identifier

When a ticker or snapshot uses a dynamic identifier (see [page 197](#)), the identifier must first be entered in the Data Manager to populate the data table. Once the table is populated, then you can drag from a cell in the data table to the property of a control or to parameter of an Action or Event. A data descriptor with a dynamic identifier has the following form:

```
Data:Schema:TickerOrSnapshot(dynamicIdentifier).tableName(columnName, row)
```

The example below shows a snapshot with a dynamic Campaign ID. To fill the data table, you must first enter the **Campaign ID** and click **GO**. Once this has been done, it is possible to drag a cell from the data table in the Data Manager into the location where it will be used.



In the above example, the **DataSource** property of the text box is set to the following data descriptor:

```
Data:SocialMedia:DynPoll("291113").m_DynPoll(ChoiceName,2)
```

| Field | Description |
|--------------------|--|
| Data | A prefix used in template logic to indicate a data server variable, as opposed to a control property or constant. |
| SocialMedia | The schema name. The schema must have been defined in the Data Manager (see Creating a social media data source , on page 195). |
| DynPoll | The name of a snapshot in the specified schema. The ticker must have been defined in the Data Manager of Xstudio (see Creating a snapshot or ticker for social media content , on page 196). |
| "291113" | The campaign ID of the poll. This is the value of the dynamic identifier. The quotes were added by the drag and drop operation, but are not required. |
| m_DynPoll | The name of the data table used by the DynPoll snapshot. |
| ChoiceName | The ChoiceName column of the data table. |
| 2 | The second row of the data table. |

Data linking using the poll identifier from a local template variable

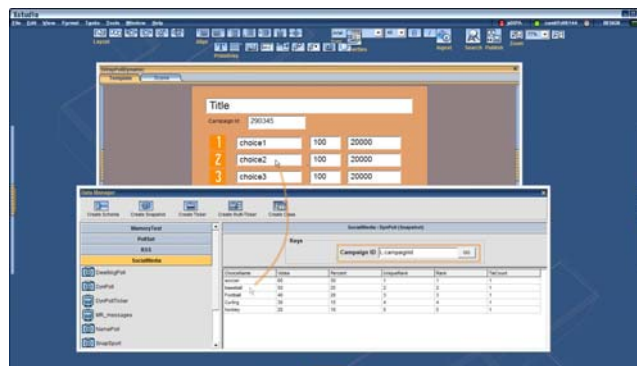
A more flexible data descriptor obtains the poll identifier from a local template variable. Typically, the poll identifier is entered into a control such as a text box. If the control has its **DataTarget** property set to a local variable such as `L:campaignId`, whatever is entered into the control is written to the corresponding local variable.

A data descriptor with a dynamic identifier that is contained in a local variable has the following form:

```
Data:Schema:TickerOrSnapshot(L:variable).tableName(columnName, row)
```

When a template is built using data descriptors of this form, it becomes possible to reuse the same template for different polls by typing in a different poll identifier in the text box.

Like in the example below, to populate the data table you would first enter a valid **Campaign ID** in the Data Manager and click **Go**. Next, type in the name of the local variable in place of the **Campaign ID** in the Data Manager. Once this has been done it is possible to drag a cell from the data table in the Data Manager into the location where it will be used.



In the example above, the **DataSource** property of the text box is set to the following data descriptor:

```
Data:SocialMedia:DynPoll(L:campaignId).m_DynPoll(ChoiceName,2).
```

| Field | Description |
|---------------------|--|
| Data | A prefix used in template logic to indicate a data server variable, as opposed to a control property or constant. |
| SocialMedia | The schema name. The schema must have been defined in the Data Manager (see Creating a social media data source , on page 195). |
| DynPoll | The name of a snapshot in the specified schema. The ticker must have been defined in the Data Manager of Xstudio (see Creating a snapshot or ticker for social media content , on page 196). |
| L:campaignId | The name of a local variable that contains the campaign ID of the poll. This is the value of the dynamic identifier. |
| m_DynPoll | The name of the data table used by the DynPoll snapshot. |
| ChoiceName | The ChoiceName column of the data table. |
| 2 | The second row of the data table. |

Using Dynamic rows

Dynamic rows are an advanced feature. When the **OrderBy** property of social media snapshots or tickers is used (see [page 198](#)), dynamic rows are usually not required.

Additional flexibility is provided by using a dynamic row in the data descriptor. A dynamic row identifies a row in the table using an expression rather than a constant number. For example, rather than setting the row to 2, set it to `@ChoiceName=France`. This specifies the row where **ChoiceName** is equal to `France`.

A dynamic row expression is of the form: `@<column_name>=<value>`

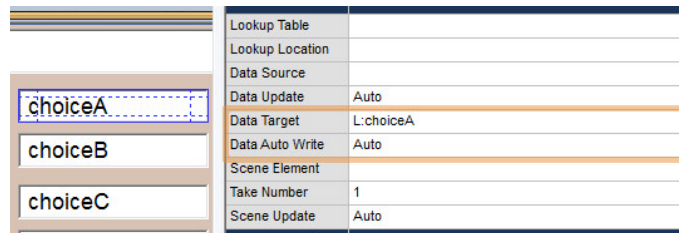
For example:

| | |
|--------------------------------|--|
| <code>@ChoiceName=Green</code> | The row where ChoiceName is equal to <code>Green</code> . |
| <code>@UniqueRank=2</code> | The row where UniqueRank is equal to 2. |

Dynamic rows are designed to be used in templates that display poll results. The only permitted column names in a dynamic row expression are **ChoiceName** and **UniqueRank** since these are the only columns where every row has a distinct value. (Note that the value used on the right hand side of the row expression is case insensitive. However, every other part of a data descriptor is case sensitive.) The value on the right hand side of the row expression may also be set to a local variable such as `L:choiceA`. Typically the local variable would be set from a control using its **Data Target** property.

Linking objects to a data source

Creating a template that retrieves social media content



Currently, there is no support in the user interface for creating data descriptors with dynamic rows. Data descriptors must first be created using one of the methods shown previously, and then edited to use a dynamic row.

Some examples of data descriptors with dynamic rows are shown below:

```
Data:SocialMedia:WorldCupPoll (L:campaignId) .m_WorldCupPoll (Votes,@ChoiceName=Spain)
```

In a poll snapshot with a dynamic campaignId, return the number of votes in the row where the ChoiceName is equal to Spain.

```
Data:SocialMedia:WorldCupPoll (L:campaignId) .m_WorldCupPoll (Percent,@ChoiceName=L:choiceA)
```

In a poll snapshot with a dynamic campaignId, return the percent in the row where the ChoiceName is equal to the value of the local variable named L:choiceA.

```
Data:SocialMedia:DynPoll (L:campaignId) .m_DynPoll (Votes,@UniqueRank=3)
```

In a poll snapshot with a dynamic campaignId, return the number of votes in the row where the UniqueRank is equal to 3.

10

Adding logic to control objects

The Action Editor allows you to specify actions that are executed in response to an event that is sent to an object that belongs to a control panel, template, or Xmedia object. A simple example to illustrate this concept is that a button primitive (the object) on a control panel can be set in the Action Editor so that when it is clicked (the event) during playback, a specific audio file starts to play (the action).

If using Xmedia objects, you will not require the Action Editor since these “smart objects” have actions assigned to events by default. For example, Xstudio knows that when you click the **Bug On** button, it means to trigger the actions that will bring the bug (network logo) on.

However, if your control panel, template, or Xmedia object uses primitives, you will need to access the Action Editor since primitives have no actions attached to them by default. Furthermore, you can also create actions and set them up in playlists to run control panels automatically on a timer.

The following sections provide functional descriptions and instructions for using the Action Editor:

- [Opening the Action Editor](#), on page 212
- [Action Editor interface components](#), on page 213

Using Events

- [Using the object’s default events](#), on page 216
- [Creating custom events](#), on page 218

Using Actions

- [Assigning an action to an object](#), on page 221
- [Creating custom actions](#), on page 222
- [Setting the Target parameter](#), on page 222
- [Using the Expression Builder](#), on page 224
- [Deleting actions from events](#), on page 226
- [Default actions and action parameters](#), on page 227

Opening the Action Editor

The Action Editor is located within a collapsible panel at the bottom of the Xstudio interface when the control side of a template, control panel, or Xmedia object is open.

Since actions cannot be assigned to scene objects, the Animation Editor occupies the bottom collapsible panel when the scene side of a template, control panel, or Xmedia object is active. For an action to affect the scene, it must be assigned to a control object, or to the control panel or template as a whole.

To access the Action Editor:

- 1 Be sure that the control side of the canvas is currently active, by clicking the **Control** or **Template** tab.

If the scene side is active, the Animation Editor will display instead of the Action Editor.

- 2 Click the gripper bar at the bottom of the screen.

Or,

Select **Action Editor** from the **View** menu.

Or,

Use the keyboard shortcut: **Ctrl + E**

Action Editor interface components

The Action Editor allows you to specify actions to be executed in response to an event that is sent to an object that belongs to a control panel, template, or Xmedia object.

The Action Editor is located within a collapsible panel at the bottom of the Xstudio interface when the control side of a template, control panel, or Xmedia object is open.










The following sections will help you become familiar with the function of the Action Editor's interface components:

- [Editing and ordering buttons](#), on page 213
- [Add Actions and Events](#), on page 213
- [Action Table components](#), on page 214

Editing and ordering buttons

A series of buttons are located in the upper left corner of the Action Editor. In general, these buttons allow you to replace, remove, and determine the order the actions that are associated with an event.

The following table provides a description of each button:

| | |
|---|--|
|  Cut | Cuts the currently selected action. |
|  Copy | Copies the currently selected action. |
|  Paste | Pastes the last cut or copied action into the currently selected action row. |
|  Delete | Deletes the currently selected action in the action table. |
|  Delete All | Deletes all actions for the currently selected event from the action table. |
|  Move to Top | Moves the currently selected action to the top of the action table, making it the first action to be triggered by the event. |
|  Move Up | Moves the currently selected action up one place in the action table. |
|  Move Down | Moves the currently selected action down one place in the action table. |
|  Move to Bottom | Moves the currently selected action to the bottom of the action table, making it the last action to be triggered by the event. |


Add Actions and Events

The **Add Actions** and **Add Events** buttons and drop-down lists allow you to add additional actions and custom events to the Action Table for a particular control object.




Fig. 10-1: The Action Editor's Add Actions and Add Events buttons and drop-down lists

The **Actions** drop-down list contains all of the actions that are available to be assigned to an event. Therefore, once you have selected an event in the Actions Table, you can select the desired action from the Actions drop-down list and then click the **Add Actions** button

. The Add Actions button adds the action selected from the Actions drop-down list to the action table. See [Default actions and action parameters](#), on page 227 for a full description of the default actions and their parameters.

The default events listed in the Action Table's Events List are determined by the type of object that is selected. For example, when a button primitive is the selected object, the events that can trigger an action are: **On Load**, **On Unload**, and **On Click**. See [Using the object's default events](#), on page 216 for a full description of each of the default events.

If the default events associated with the object do not meet your needs, you can use the **Events** drop-down list and **Add Event** button  to create custom events. See [Creating custom events](#), on page 218 for a full explanation and instructions for creating custom events.


Action Table components


The **Action Table** is the main area where you create and assign actions and events to the selected control object. As well, the Action Table displays several types of information including:

- A list of the events (default and custom) associated with the selected object (Events List panel)
- Any conditions that are set to guide the action (Condition column)
- The actions assigned to an event (Action List column)
- Any comments or descriptions that explain the purpose or result of executing the action (Comment column)
- The parameter settings associated with an action (Action Parameters panel)

As well, you can use the Action Table to add actions to events and to set the action's parameters.

The following table provides a functional description of each component of the Action Table:

| | |
|------------------|--|
| Events | <p>The Events panel lists all the default events for the selected object. When an action has been added to an event, the event's name appears in bold in the Events panel.</p> <ul style="list-style-type: none">• You can add custom events to this list using the Events drop-down list and the Add Events button (see Creating custom actions, on page 222).• You can change the name and/or edit a custom event by right-clicking on the event's name and selecting the appropriate command from the context menu. |
| Condition | <p>Displays the condition that was set for the action using the Expression Builder.</p> <ul style="list-style-type: none">• Clicking the  button opens the Expression Builder, which provides a list of common building blocks that you can use to set a condition for the action associated to that row in the Action Table. See Using the Expression Builder, on page 224 for more details. |
| Action | <p>Displays the actions assigned to the event selected in the Events List.</p> <ul style="list-style-type: none">• You can replace or add an action to the Action List by selecting an action name from the drop-down list that becomes available when you click the row where you want the action. |

| | |
|--------------------------|--|
| Comment | Displays and allows you to enter comments that provide a description of the purpose and/or result of the action. |
| Action Parameters | <p>Displays and allows you to set the parameters for the action that is currently selected in the Actions List. See Default actions and action parameters, on page 227 for a list of the available actions and their parameters.</p> <ul style="list-style-type: none"> • If Custom is selected action in the Action List, a list of Custom Actions appears instead of the Action Parameters rows. • When certain parameter fields are selected, the  button appears to the right of the field. Clicking this button opens either the Expression Builder and/or the Target/Source Selector dialog boxes, which you can use to help you define or set certain action parameters. See Using the Expression Builder, on page 224 and Setting the Target parameter, on page 222 for more details. |

Using the object's default events

Selecting an object on the canvas displays its default events in the Action Editor. Not all of the default events are available to every type of object though. The type of object and its functionality determines the applicable default events. For example, when a button primitive is the selected object, the default events that can trigger an action are limited to: **On Load**, **On Unload**, and **On Click**.

Note: If you have a special requirement that is not covered by existing default events, you can create custom events that can be triggered in response to another event (User Defined events), or have actions executed whenever certain database variables change (Data Arrival events). See [Creating custom events](#), on page 218 for more details and instructions.

The following table lists and describes each of the events that may appear in the Action Editor.

| | |
|---------------------|---|
| Data Arrival | The Data Arrival event is only available when a Data Arrival custom event has been added to the Events list. This event is triggered by data arriving from the database. See Creating custom events , on page 218 for further details. |
| On Checked | The On Checked event is only available to check box objects. This event is triggered when the check box is selected (check mark). |
| On Clear | The On Clear event is triggered by a graphic being cleared from the output device. This event is only available to text, drop-down, image, and timer objects. |
| On Click | The On Click event is only available to button objects. This event is triggered by the button being clicked. |
| On Cue | The On Cue event is triggered by a graphic being cued as next to be sent to the output device. This event is only available to text, drop-down, text box, image, and timer objects. |
| On Get Data | The On Get Data event is triggered when the user sends a command requesting that data be retrieved from its source. For example, a page could have been programmed to get data every time it is refreshed by the user. This event is available to all objects, except for button and ActiveX objects. |
| On Load | The On Load event is triggered by a graphic entering Production mode or being opened while in Production mode. This event is available to all objects. |
| On Selected | The On selected event is only available to radio button objects. This event is triggered when the radio box is selected (filled in). |
| On Take | The On Take event is triggered by a graphic being sent to the output device. This event is only available to text, text box, image, and timer objects. |

| | |
|----------------------|---|
| On Timer | The On Timer event is triggered when the timer is updated, which is determined by the Timer object's Interval property. |
| On Unchecked | The On Unchecked event is only available to check box objects. This event is triggered when the check box is deselected (cleared). |
| On Unload | The On Unload event is triggered by a graphic leaving Production mode. This event is available to all objects. |
| On Unselected | The On Unselected event is only available to radio button objects. This event is triggered when the radio box is deselected (cleared). |
| On Update | The On Update event is triggered by changes to a check box, drop-box, text, or text box object, followed by the user clicking off the object. The event is signalled not during the change, but after it has been completed and the object is no longer selected. For example, if a text box's contents are being edited, the On Change event is triggered only after the edits have been made, not during. |

Creating custom events

If you have a special requirement that is not covered by the existing default events, you can create and add custom event to the Action Editor. Two types of custom events are made available:

- [User Defined events](#), on page 218
- [Data Arrival events](#), on page 219


User Defined events

User Defined events are used when you have a requirement that cannot be met by any of the default events. Since they cannot be triggered by the system, they are usually triggered by calling a **Send Event** action in response to another event.

A common use for a User Defined event is triggering actions required by more than one object. For example, if you had several buttons on a control panel that used a common set of actions, you could create a User Defined event on one of the objects to contain the common actions. Next, you would use the **Send Event** action for every object requiring these actions in order to send the User Defined event to the object with the common actions. The result is akin to a function call or a subroutine, and it reduces the margin of error when modifications to the common action set are required, since changes only need to be made once.

You can also use User Defined events for external control purposes. For example, if you have an external device controlling Xstudio you can make the text you enter trigger a User Defined event.

To create a User Defined event:

- 1 On the Panel tab, select the object for which you would like the event available.
If the event is for the panel itself, click anywhere where there are no objects.
- 2 In the Action Editor, select **User Defined** from the **Events** drop-down list and click the **Add Event**  button.
The **Add Event** dialog box appears.
- 3 Enter a name for your event in the field provided.
- 4 Click **OK**.
You can now add actions to the event as required.

User Defined events can include parameters to supply more information to be used within the action. Since parametrized User Defined events are only likely to be used in very specific cases, what follows is a generic example with a control panel. It shows how to set text from one text box to the next and change it along the way by clicking a button.

To create a panel using parametrized User Defined events:

- 1 Create a new control panel with three (3) text boxes and a button.
- 2 Select **Textbox2** and then display the **Action Editor (CTRL + E)**.
You will create a new event for Textbox2.
- 3 From the **Events** drop-down list, select **User Defined** and then click the **Add Event** button to the right of the drop-down list.

- 4 In the **Add Event** dialog box, type a name for the event followed immediately by in parentheses into which you will enter a parameter name. For example, `MyEvents(text)`. Then click **OK** and the custom event is added immediately to the Events List.
- 5 You will now assign a **Send Event** action to the button object. This will send the new event to Textbox2.
 - a Select the button object on the control panel.
 - b Select **On Click** from the Event list.
 - c Select the **Send Event** action from the **Actions** drop-down list and click the **Add Actions** button.
The Send Event is immediately added to the Action Table.
- 6 In the **Action Parameters**, enter the name of your event in the **EventType** field followed by a parentheses into which you will enter `Control:Textbox1`. For example, `MyEvent(Control:Textbox1)`.
This means that whatever text is entered in Textbox1 becomes the parameter for the **User Defined** event.
- 7 Enter `Control:Textbox2` in the Action Parameter's **Target** field.
This means that when the button is clicked, it sends the custom event to Textbox2 (which is expecting it) with the contents of Textbox1 as the parameter.
- 8 Select Textbox2 on the control panel and then select your custom event from the Event list.
- 9 Select **Set Text** from the **Actions** drop-down list and click the **Add Actions** button. The **Set Text** action is immediately added to the Action Table.
- 10 In the **Source** field of the **Action Parameters**, type **Event:** and then the event parameter. For example, if your User Defined event is `MyEvent(text)`, you would enter `Event:text`.
 - You can also format the **Source** parameter. For example, to make the parameter text appear uppercase you would put `Event:text` in parentheses and precede it with `ucase`.
 - You could also append text by using `&` and then text in quotation marks. For example, your entry could look like this: `ucase(Event:text) & " is coming up next"`.
- 11 In the **Target** field, type `Control:Textbox3`.
This indicates that Textbox3 will be receiving the text specified by the User Defined event parameter. In this case, the text is coming from Textbox1, and will be formatted and extended by the time it reaches Textbox3.
- 12 Switch to Production mode (**CTRL + P**).
- 13 Type text in Textbox1, click the button, and then watch the text appear in Textbox3.

Data Arrival events

Use the **Data Arrival** event when you want actions to be executed whenever certain database variables change. For each Data Arrival event added, specify the database variables whose changes will trigger the event.

For example, if you have a control panel that displays the time and temperature, you could create a Data Arrival event for the time and another for the temperature. Then for each, you could create an action that only allows the time and temperature to be set when a change has occurred. You could also create a **Send Event** action for the control panel that triggers both actions when the control panel is loaded.

To create a Data Arrival event:

- 1 From the Events drop-down list select **Data Arrival** and click the **Add Event** button



The **Data Arrival Parameters** dialog box appears.

- 2 From the Data Manager, enter your parameters one at a time, or drag data items into the Data Arrival Parameters box.
- 3 When done, click **OK**.

Assigning an action to an object

Actions can only be assigned to objects on the control side.

To assign an action:

- 1 Click the Control tab.

Actions can only be assigned to objects on the control side.

- 2 Select an object on the canvas.

The available events for that object are displayed in the Events panel on the Action Editor table (see [Using the object's default events](#), on page 216 for the description of each event).

- 3 Select the event from the Events panel that you want to add one or more actions to.
- 4 Assign the action to the event by selecting an action from the Action column's drop-down list and clicking the Add Action button.

Selecting an action makes the following things happen in the Action Editor:

- Another row is added below the one currently being worked on. This allows you to easily add other actions to the event.
- If anything other than **Custom** was selected as the action, the Action Parameter fields specific to the type of action you have selected are displayed (if **Custom** was selected, see [Creating custom actions](#), on page 222).
For example, if you have selected **Set Text** as your action you would need to indicate where the text comes from, where it is supposed to display, and whether there are any conditions or comments. These fields remain displayed as long as you are working on the action.

- 5 If a condition exists for the action, enter it in the **Condition** column (your cursor automatically goes there after you have selected an action).

For example, if you were displaying sports scores you could have a box display around the winning team's name by setting conditions that make the box visible only when team X's score is higher than team Y's, and vice versa.

For a listing of conditions and their syntax, see [Using the Expression Builder](#), on page 224.

If including a control or scene object in your condition, be sure to prefix it with Control: or Scene: respectively, e.g. Control:Textbox1 > Control:Textbox2.

Note: To avoid typing out control object or scene object names along with their prefixes, access the Object Tree or the Scene Tree, click on the item you require, and then drag and drop it into the required parameter field.

- 6 When done setting the action, its parameters and any required conditions, press ENTER on your keyboard or click outside the Action Parameters section.

Note: To duplicate an existing action use the **Copy** and **Paste** buttons.

Creating custom actions

If the Action Editor's default actions do not perform the specific functions that you require, you can create custom actions that allow you to use scripts to

To create a custom action:

- 1 Select **Custom** from the Action drop-down list.
The Action Parameters section at right is replaced by the Custom Actions list, which will contain any previous custom actions you or another user may have created.
- 2 Select an existing custom action.
Or,
Edit an existing custom action by right-clicking on the action and then selecting **Edit** from the popup menu.
Or,
Create a new one by right-clicking on a custom action and then selecting **New** from the popup menu. This displays the **Script Editor** (see [page 325](#)). Refer contact our technical support department for information regarding the "Producer API".

Setting the Target parameter


The **Target** parameter is associated with many of the default actions and it allows you to set which object will receive the action, or which object the action will apply to. To help you properly select the target object for this parameter, it is recommended that you use the **Target/Source Selector** dialog box.

The **Target/Source Selector** dialog box is basically divided into three parts: the left panel lists the control panel/template and scene available in the current view; the right panel lists the object contents of the control panel/template or scene that is currently selected in the left panel; and the upper text box where the selected target object is recorded when selected in the right panel.

When the selected object appears in the upper text box, it will be preceded either by `Control:`, `Template:` or `Scene:`, e.g. `Control:Textbox1`. This prefix indicates whether the object comes from the control- or the scene-side. Do NOT delete the prefix.

Click the **Clear** button next to the upper text box to remove the contents of the upper text box.

To use set the Target parameter using the Target/Source Selector:

- 1 Select the **Target** field in the Action Parameters panel and then click the  button.
The **Target/Source Selector** dialog box appears.
- 2 Select the control panel, template, or scene in the box at left.
This displays its object contents in the right panel.
- 3 Double-click an item from the list of contents in the right panel.
The object is automatically added to the upper text box preceded by a prefix that indicates whether the object comes from the control- or the scene-side.
- 4 Click the **OK** button.

The **Target/Source Selector** dialog box closes and the **Target** parameter now displays the object setting.

Using the Expression Builder

The **Expression Builder** is a dialog box that allows you to create expressions and conditions for the action's parameters. Additionally, the Expression Builder allows you to set the source of the data for parameters that bring data in from one object (source) and apply it to another object (target).

The **Expression Builder's** dialog box is divided into four functional sections:

- the left panel, which lists the categories that available to Expression Builder: functions, control panel/template and scene
- the right panel, depending which category is selected in the left panel, the right panel lists either the common building blocks for building functions or the object contents of the control panel/template or scene.
- a set of operator buttons that can be quickly added to the condition statement
- the condition text box, which is where the expression or condition is created or the selected source object is displayed. If the result of the condition is not understood by Xstudio, then it defaults to **True**.

Click the **Clear** button next to the condition text box to remove the contents of the condition text box.

To create an expression or condition:

- 1 Click in the **Condition** row and then click the **drop-down list button**.

Or,

Click in any of the following rows appearing in the **Action Parameters** section (depending on the action selected):

- **Number**
- **Source**
- **SourceLocation**
- **SourceName**
- **TextSource**
- **Time**
- **Value**
- **Visible**

This displays the **Expression Builder** dialog box.

The box at left lists functions and object sources.

- 2 Click **Functions** to display a list of building blocks at right. Click one of the object sources (e.g. Scene, Panel, Template) to display a list of its objects (e.g. text boxes, buttons, images, etc.).

Note: Literal strings (i.e. text that is to be read as is without being interpreted by the application in any way) must be placed in quotation marks.

- 3 Double-click one of the listed functions at right to make it display in the field in the upper portion of the dialog box.



| Function | Description |
|--|--|
| LOOKUP | LOOKUP(<string>,<lookupTable>) A string operator that applies a lookup table to the contents of an object. Replace the expression's generic components as follows: <string> - The object name or literal string. <lookupTable> - The lookup table name (must be in quotation marks). Example: LOOKUP(Control:Textbox1, "Weather") |
| UCASE | UCASE(<string>) A string operator that makes the contents of an object display in uppercase. Replace <string> with the object name or literal string. Examples: UCASE(Control:Textbox2) UCASE("hello") |
| LCASE | LCASE(<string>) A string operator that makes the contents of an object display in lowercase. Replace <string> with the object name or literal string. Examples: LCASE(Control:Textbox3) LCASE("Hello") |
| & | (<string>)&(<string>) A string operator that concatenates two strings. Replace <string> with the object or literal string. Examples: Control:Textbox3 & "%" Control:Textbox3 & Control:Textbox4 |
| SQRT | SQRT(<n>) A math operator that calculates the square root of a number. Replace <n> with the number to be calculated. Example: SQRT(100), SQRT(Control:Textbox4) |
| ABS | ABS(<n>) A math operator that calculates the absolute value of a number. Replace <n> with the number to be calculated. Example: ABS(-10), ABS(Control:Textbox5) |
| AND, OR, NOT, >=, <=, >, <, =, <>, () | Logical operators that determine boolean (true or false) values. These are typically used for setting conditions. |
| +, -, *, / | Math operators. |

- 4 Quickly replace generic expression components (e.g. <string>) by performing the following steps:

- a Select the part of the expression to be replaced.
 - b Select an object source at left to display its contents at right.
 - c Double-click an object to have it replace the selected part of the expression.
- 5 When done, click **OK**.

Deleting actions from events

To delete an action from an event:

- 1 Select the action under its event and click .
Or,
Select the action under its event, right-click, and from the pop-up menu select **Delete Action**.
- 2 To delete all actions for an event, select the event and click .

Default actions and action parameters

The Action Editor allows you to specify actions that are executed in response to an event that is sent to an object that belongs to a control panel, template, or Xmedia object. Once you have added an action to an event, you must set its parameters in the Action Parameters portion of the Action Table.


The following table provides descriptions of the actions and instructions for completing their parameter requirements. Use the following list to quickly navigate to a particular action.

- [Animation command](#)
- [Clip Command](#)
- [Crawl Command](#)
- [Custom](#)
- [Device Command](#)
- [Do Take](#)
- [Exit](#)
- [Flush Batch](#)
- [Load Panel](#)
- [Load Playlist](#)
- [Load Scene](#)
- [Log](#)
- [Pause](#)
- [Play VoiceOver](#)
- [Send Event](#)
- [Set AFD](#)
- [Set Ancillary Source Priority](#)
- [Set Back Color](#)
- [Set Cel Animation](#)
- [Set Data](#)
- [Set Dropbox List](#)
- [Set Fore Color](#)
- [Set Image](#)
- [Set KeyFrame](#)
- [Set Property](#)
- [Set Text](#)
- [Set Visibility](#)
- [Squeeze](#)
- [Ticker Command](#)
- [Timer Command](#)
- [Unload Scene](#)
- [Xprotocol Command](#)

Animation command

Adding this action to an event allows you to control the playout of a scene animation. Animations change the properties of objects in a scene over time to create a visual effect. See [Creating scene animations](#), on page 129 for more information.


Set the following Action Parameters:

| | |
|------------------|--|
| Type | <p>This parameter determines what type of operation will be executed on the animation.</p> <p>Select one of the following from the drop-down list:</p> <ul style="list-style-type: none">• Play: Plays the animation when the event condition is executed.• Stop: Stops the animation when the event condition is executed.• Continue: When the event condition is executed, the animation continues to play from where it was last stopped.• Rewind: When the event condition is executed, the animation is rewinded to the beginning.• End Loop: When the event condition is executed, the animation's loop playout is cancelled. |
| Animation | <p>This parameter identifies the name of the animation being controlled. Note that before you can set this parameter, the animation must already have been added to the scene (see Creating scene animations, on page 129).</p> <p>To set this parameter:</p> <ol style="list-style-type: none">1 Select the parameter's field and click the  button to open the Expression Builder.2 Select the Animation category in the left panel of the Expression Builder. A list of the scene's animation now appears in the right panel of the Expression Builder.3 Double-click the desired animation's name in the right panel. The animation is added to the upper text box surrounded by quotation marks (i.e. "Default In").4 Click OK and the Animation parameter is set. |

Clip Command

When the associated event is executed, this action assigns a clip asset to be displayed in the specified clip object on the scene side. Whether or not it starts playing immediately depends on the output device.

Set the following Action Parameters:

| | |
|-----------------------|---|
| Target | <p>This parameter identifies the clip object on the scene that will display the clip asset.</p> <p>To set this parameter:</p> <ol style="list-style-type: none"> 1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box. 2 Select the Scene category in the left panel of the Target/Source Selector. A list of the scene-side objects now appears in the right panel of the Target/Source Selector. 3 Double-click the clip object in the scene that will display the clip. It is added to the upper text box prefixed by "Scene:". 4 Click OK and the Target parameter is set. |
| SourceName | <p>Enter the name of the clip asset by one of the following methods:</p> <ul style="list-style-type: none"> • Drag in an asset from the Asset Browser or Gallery Browser. • Enter the full name (including location) manually, e.g. XMS:Clips.News:Earthquake. <p>With these methods, both the source name and location are set.</p> |
| SourceLocation | <p>This is set automatically if using either of the two methods explained above for setting the source name. However, if your SourceName is a macro (e.g. a variable), then you can use this row to specify the category, e.g. XMS:Clips.News.</p> |
| AudioProfile | <p>An index (integer) into a set of pre-defined audio profiles. These profiles are defined in the CG control panel and define how the various audio channels are mixed.</p> |
| StartTime | <p>Offset in seconds which indicates the beginning of the clip. To start at the beginning of the clip, this parameter should be set to -1.</p> |
| EndTime | <p>Offset in seconds which indicates the end of the clip. To finish at the end of the clip, this parameter should be set to -1.</p> |

Crawl Command

Adding this action to an event allows you to control the playout of a specific crawl. A crawl makes text and/or graphics “crawl” across the screen from right to left. See [Crawl and Roll objects on the scene canvas](#), on page 104 for more information.

Set the following Action Parameters:

| | |
|---------------|---|
| Type | This parameter determines what type of operation will be executed on the crawl. Select one of the following from the drop-down list: <ul style="list-style-type: none">• Start: Starts the crawl when the event condition is executed.• Stop: Stops the crawl when the event condition is executed.• Clear: Clears the crawl when the event condition is executed.• Reset: Resets the crawl when the event condition is executed. |
| Target | This parameter sets the name of the crawl being controlled. Note that before you can set this parameter, the crawl must already have been added to the scene (see Crawl and Roll objects on the scene canvas , on page 104 for more information). To set this parameter: <ol style="list-style-type: none">1 Select the parameter’s field and click the <input type="button" value="..."/> button to open the Target/Source Selector.2 Select the Scene category in the left panel of the Expression Builder. A list of the scene’s objects now appears in the right panel of the Expression Builder.3 Double-click the desired crawl object’s name in the right panel. The crawl object is added to the upper text box prefixed by “Scene:”.4 Click OK and the Target parameter is set. |

Custom

Adding this action to an event allows you to create your own action by writing a script. For example, you could use one to animate your graphics on take and on clear.

When you add a **Custom** action to the **Action** column, a list of scripts appears in the **Custom Actions** portion of the Action Table (where the Action Parameters are usually located). The Custom Actions lists all of the scripts on your system (also listed within the **Scripts** root category in the Asset Browser).

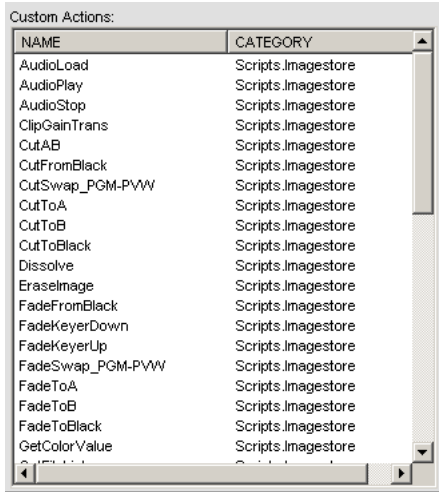


Fig. 10-2: Custom actions list

Select an existing script from the Custom Action list, or access the **Script Editor** to create a new one (see [Accessing the Script Editor](#), on page 326). The required parameters depend on the script selected.

Note that when you select an existing script it will henceforth appear in the **Actions** drop-box for that asset only. See [Creating custom actions](#), on page 222 for further details.

Device Command

Adding this action to an event allows you to use commands specific to the device that are not supported natively by Xstudio.


Set the following Action Parameters:

| | |
|--------------------|---|
| Command | The name of the device specific command to send to the device. |
| Parameter | Enter a comma-separated list of parameters for the command. |
| Device | The device (keyer) this component acts upon. |
| DeviceList | The aliases of the devices this component controls. |
| ChannelList | The channels this component operates on or the Channel Selector it's tied to. |

Do Take

Adding this action to an event allows you to display a control object's associated scene element on the output device if the take number matches. For example, you might want to display a top ten list that uses one take for every list item.

Set the following Action Parameters:

| | |
|---------------|---|
| Number | This parameter allows you to set the take number by typing the number directly in this field. Take 1 (the default) is the principal take that first displays the graphic on the device. Subsequent takes will display additional objects on air. Note that no takes will go to air if Take 1 has not yet been executed. |
| Target | This parameter identifies the object whose associated scene element is sent to air. To set this parameter: <ul style="list-style-type: none">• Type ALL in this parameter's text box to send all scene elements to air. Or, <ul style="list-style-type: none">• If you want some parts of the control panel or template to update and not others.<ol style="list-style-type: none">1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box.2 Select the Scene category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector.3 Double-click the object group you would like updated on the take. It is added to the upper text box prefixed by "Control:".4 Click OK and the Target parameter is set.Note that if nothing is entered in the Target row, then the update starts at the top of the tree and cascades downwards. |

Exit

Adding this action to an event allows you to avoid processing subsequent the actions listed for that event, if a condition is met. This action is useful in cases where you have one action that sets off several others based on a condition. Instead of setting several actions based on a negative outcome, simply create an **Exit** action.

There are no parameters for this action, however, you must set a condition for this action by selecting the **Condition** cell next to the **Exit** action in the Action table. Click the button to display the **Expression Builder** dialog box. Create the condition using the Expression Builder ([page 224](#)) and then click **OK**.

Flush Batch

Adding this action to an event allows you to send a **Flush Batch** command to the device, which instructs it to immediately execute (flush) any pending updates.

There are no parameters for this action.

Load Panel

Adding this action to an event allows you to load a specific control panel.

Set the following Action Parameters:

| | |
|-----------------|--|
| Name | This parameter sets and identifies the exact control panel that is to be loaded. To set this parameter: <ul style="list-style-type: none">• Drag a control panel from the Asset Browser or Gallery Browser into the parameter's field. Or, <ul style="list-style-type: none">• Enter the full name (including location) manually (e.g. XMS:Panels.News:Earthquake). This sets both the source name and category. |
| Location | This is set automatically if you use either of the two methods explained above for setting the source name. However, if using a macro for the Name (e.g. a variable), then you can use this row to specify the category (e.g. XMS:Panels.News). |
| Target | The target of the intended destination. |

Load Playlist

Adding this action to an event allows you to load a specific playlist.

Set the following Action Parameters:

| | |
|-------------------|--|
| Name | <p>This parameter sets and identifies the name of the playlist that will be loaded.</p> <p>To set this parameter:</p> <ul style="list-style-type: none">• Drag a playlist from the Asset Browser or Gallery Browser into the parameter's field. <p>Or,</p> <ul style="list-style-type: none">• Enter the full name (including location) manually (e.g. XMS:Panels.News:Earthquake). <p>This sets both the source name and category.</p> |
| Location | <p>This is set automatically if you use either of the two methods explained above for setting the source name. However, if using a macro for the Name (e.g. a variable), then you can use this row to specify the category (e.g. XMS:Shows.News).</p> |
| Type | <p>This parameter determines how the playlist will be integrated.</p> <ul style="list-style-type: none">• Preempt: Replaces the existing playlist and starts playing immediately.• Cue: Appends the newly loaded playlist to the existing playlist, and then the first playlist is removed. |
| Start Time | <p>Specifies an offset time in the playlist from which the playlist will start to payout.</p> |

Load Scene

Adding this action to an event allows you to load a scene to the output device.

Set the following Action Parameters:

| | |
|-----------------|--|
| Name | <p>How you enter the scene name depends on how you created it.</p> <ul style="list-style-type: none">• If you built the scene in conjunction with a control screen, you must enter the name given to the control side, not the scene side.• If you built the scene independent of a control screen, enter the name you saved the scene to. <p>In either case, enter the name in quotation marks.</p> <p>Or,</p> <ul style="list-style-type: none">• Drag a scene (template/panel) from the Asset Browser or Gallery Browser into the parameter's field. |
| Location | <p>This is set automatically if you drag the scene (template/panel) into the Name parameter. However, if you enter the name manually, then you can use this row to specify the category (e.g. XMS:Templates).</p> |
| Keyer | <p>Specifies the keyer layer on the device.</p> |
| Output | <p>Determines where the scene will be loaded to.</p> <ul style="list-style-type: none">• Preview: Displays on the preview display, not brought to air.• On Air: Displays on the device.• Default: Displays on the view where the action is being run from. |

Log

Adding this action to an event allows you to log a message to the specified log file.


Set the following Action Parameters:

| | |
|----------------|---|
| Log | Enter either the location of the log file on the hard drive (e.g. C:/Temp/Log.txt), or an expression to specify a log file. |
| Message | Enter either specific text to be written to the log file, or an expression (e.g. Control:SponsorImage). |

Pause

Adding this action to an event allows you to pause the execution of the actions for the given number of seconds.

Set the following Action Parameters:

| | |
|-------------|---|
| Time | Either enter the number of seconds, e.g. 2.5. To build an expression, click in the row, and then click the  button to display the Expression Builder . |
|-------------|---|

Play VoiceOver

Adding this action to an event allows you to play a voiceover audio file when the event is executed.

Set the following Action Parameters:

| | |
|----------------------|--|
| Output | Output where audio asset will be played. Valid Values: <ul style="list-style-type: none"> • 0 (Preview output) • 1 (On Air output) |
| SourceName | The name of the audio asset to be set. |
| VoiceOverGain | The audio level at which the audio asset will be played. The valid range of values is from 0-8. Values between 0-1 are will play the audio asset at a lower level than normal (duck) and any values between 1-8 will play the audio asset at a higher level than normal (gain). |
| Duck | The audio level at which all other audio channels will be played. The valid range of values is from 0-8. Values between 0-1 are will play the audio asset at a lower level than normal (duck) and any values between 1-8 will play the audio asset at a higher level than normal (gain). |
| Duration | The duration, in seconds, of the duck transition. |
| Profile | The mix profile number used when executing this command. The mix profiles are defined in the settings of the playout device. |

Send Event

Adding this action to an event allows you to send an event to an object to trigger an additional set of existing actions.

This action is particularly useful in cases where you would like to have all the same actions for one object available for another object without having to program them a second time. For example, if **Button1** has multiple actions defined for it. You want **Button2** to execute the same actions. To avoid having to redefine the same list of actions for Button2 as you did for Button1, you could assign the **Send Event** action to Button2's **On Click** event. By defining the **Event Type** as **On Click** and making the **Target Button1**, you are effectively telling Xstudio to click Button1 when a user clicks Button2.

Set the following Action Parameters:

| | |
|-----------------------|--|
| EventType | Enter an event type from the list that appears at the left side of the Action Editor (e.g. On Load, On Take, On Clear). You can also create user-defined actions for specific requirements. |
| Target | <p>This parameter identifies the object(s) that the event will be sent/applied to.</p> <p>To set this parameter:</p> <ul style="list-style-type: none"> Type ALL in this parameter's text box to send the event to all objects. <p>Or,</p> <ol style="list-style-type: none"> Select the parameter's field and click the <input type="button" value="..."/> button to open the Target/Source Selector dialog box. Select the Control category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector. Double-click the object group you would like the event to be applied to. It is added to the upper text box prefixed by "Control:". Click OK and the Target parameter is set. |
| SendToChildren | Whether the event is only sent to the parent object (False) or to all of the children objects associated to the parent object (True). |

Set AFD

Adding this action to an event allows you to set the Active Format Description (AFD) value of the output video based on either the video input channel, a video clip or by hard coding the AFD value.

Initially, the only parameter that appears is **Source AFD**, but once you select one of the source options other parameters appear:

Set the following Action Parameters:

- **Default:** No additional parameters required as it will use the output device's default value.
- **Video Source:** Displays the **VideoSource** parameter which allows you determine if the AFD value for the output value is based on the AFD either the video source input channel A or channel B.

- **AFD Code:** Displays the **AFDCode** parameter which lets you specify the AFD value of the output video by entering an AFD value (e.g. 8).
- **Clip Object:** Displays the **ClipObjectName** parameter which allows you to specify the name of the clip object on the scene (i.e. Clip1) as the source of the AFD value.

Set Ancillary Source Priority

The **Set Ancillary Source Priority** action allows you to dynamically control what ancillary data is written to the output of the Vertigo XG or Intuition XG.

Set the following Action Parameters:

| | |
|-----------------|---|
| Type | The types of ancillary data that can be controlled are VANC and CDP . VANC represents the all available lines of the vertical ancillary data space. CDP represent caption data packets, which will be written on a specific line of the vertical ancillary data space. |
| Source | Ancillary data can be associated with three inputs: Clip , Input A , and Input B . |
| Priority | The highest priority ancillary data, if it exists, will be written to the output. If the Ancillary Source Priority is not set by the user, the Vertigo XG and Intuition XG defaults to the following ancillary priority assignments: <ul style="list-style-type: none"> • Clip => Priority 3 • Input A => Priority 2 • Input B => Priority 1 Therefore, the clip's ancillary data would be written to the output first. If clip ancillary data does not exist, then Input A's ancillary data will be written to the output. If Input A's ancillary data does not exist, then Input B's ancillary data will be written to the output. When the Ancillary Source Priority is set by the user the Priority field defaults to 0, but it can be set to values between 0 and 100. A Priority value of 0 means do not include ancillary data in the output, while the rest of the values are relative to the other sources. |

Note: Controlling what ancillary data is written to the output of the Vertigo XG or Intuition XG can also be accomplished using two new device commands: `SetVancSourcePriority` and `SetCDPSourcePriority`. These commands are identical, except that one targets VANC sources and the other targets CDP sources. Two parameters are required: `source` and `priority`. The possible values for the `source` parameter are: `clip`, `inputA`, `inputB` and `none`. The `priority` parameter must be a positive integer.

Set Back Color

Adding this action to an event allows you to set the background color (`BackColor` property) of an object on the control side. An example of when this action is used is to indicate a changed state of an object. For example, when an on/off button has been clicked on the control panel its color changes from green to red to visually reinforce its state.

Set the following Action Parameters:

| | |
|---------------|---|
| Target | This parameter identifies the object(s) that the background color will be applied to. To set this parameter: <ol style="list-style-type: none">1 Select the parameter's field and click the <input type="button" value="..."/> button to open the Target/Source Selector dialog box.2 Select the Control category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector.3 Double-click the object you would like the back color to be applied to. It is added to the upper text box prefixed by "Control:".4 Click OK and the Target parameter is set. |
| Color | Sets the color value that determines the color of the object's background. Click in the row, and then click the <input type="button" value="..."/> button to display the Color dialog box. Select the desired color and click OK . |

Set Cel Animation

When the associated event is executed, this action assigns a cel animation asset to be displayed in the specified cel animation object on the scene side. Whether or not it starts playing immediately depends on the output device.

Set the following Action Parameters:

| | |
|-----------------------|--|
| Target | <p>This parameter identifies the cel animation object on the scene that will receive the cel animation asset.</p> <p>To set this parameter:</p> <ol style="list-style-type: none"> 1 Select the parameter's field and click the <input type="button" value="..."/> button to open the Target/Source Selector dialog box. 2 Select the Scene category in the left panel of the Target/Source Selector. A list of the scene-side objects now appears in the right panel of the Target/Source Selector. 3 Double-click the cel animation object in the scene that will display the cel animation. It is added to the upper text box prefixed by "Scene:". 4 Click OK and the Target parameter is set. |
| SourceName | <p>Enter the name of the cel animation asset by one of the following methods:</p> <ul style="list-style-type: none"> • Drag a cel animation asset from the Asset Browser or Gallery Browser into the parameter's text box. • Enter the full name (including location) manually, e.g. XMS:Cels.News:Earthquake. <p>With these methods, both the source name and location are set.</p> |
| SourceLocation | <p>This parameter is set automatically if you use either of the two methods explained above for setting the source name. However, if your SourceName is a macro (e.g. a variable), then you can use this row to specify the category, e.g. XMS:Cels.News.</p> |
| Keyer | <p>The keyer of the scene containing the target scene object</p> |
| DeviceList | <p>The aliases of the devices this component controls.</p> |
| Channellist | <p>The channels this component operates on or the Channel Selector it's tied to.</p> |

Set Data

Adding this action to an event allows you to write a value to the database.



Set the following Action Parameters:

| | |
|---------------|---|
| Target | To specify the name of a the database variable, enter one of the following: <ul style="list-style-type: none">• A local or global variable that will be used by another action or parameter, e.g. L:SelectedStock.• An expression that resolves to the name of a local or global variable. For example, if Control:txtName contained StartTime, the Target would resolve to L:StartTime. |
| Value | This is where you provide the value that will be written to the database variable. It must be the same type as the variable, e.g. if the variable holds a number, this should evaluate to a number. |

Set Dropdown List

Adding this action to an event allows you to set the contents of a drop box object on the control side.



Set the following Action Parameters:

| | |
|-------------------|--|
| Target | This parameter identifies the drop-down list box on the control-side that will display a list of items. To set this parameter: <ol style="list-style-type: none">1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box.2 Select the Control category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector.3 Double-click the dropdown object that will display list of items. It is added to the upper text box prefixed by "Control:".4 Click OK and the Target parameter is set. |
| TextSource | Click in the row, and then click the  button to display the Target/Source Selector . Choose the drop box containing the list items. This could also be a list of names separated by carets (e.g. "Boston^Montreal^Toronto") or a database representing a column of data each. |

Set Fore Color

Adding this action to an event allows you to set the foreground color of an object on the control side.


Set the following Action Parameters:

| | |
|---------------|--|
| Target | <p>This parameter identifies the object(s) that the foreground color will be applied to.</p> <p>To set this parameter:</p> <ol style="list-style-type: none"> 1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box. 2 Select the Control category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector. 3 Double-click the object you would like the foreground color to be applied to. It is added to the upper text box prefixed by "Control:". 4 Click OK and the Target parameter is set. |
| Color | <p>This can be the color from another object, or specific color value in the row in quotation marks, e.g. "127,255,0". If the source is an object whose color is to be used, click in the row, and then click the  button to display the Color dialog box.</p> |

Set Image

Adding this action to an event allows you to assign an image asset to an image, button, or Xmedia object on the control-side.


Set the following Action Parameters:

| | |
|-----------------------|--|
| Target | <p>This parameter identifies the image, button, or Xmedia object on the control-side that will display an image asset.</p> <p>To set this parameter:</p> <ol style="list-style-type: none">1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box.2 Select the Control category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector.3 Double-click the image object that will display the image asset. It is added to the upper text box prefixed by "Control:".4 Click OK and the Target parameter is set. |
| SourceName | <p>Enter the name of the image asset by one of the following methods:</p> <ul style="list-style-type: none">• Drag in an asset from the Asset Browser or Gallery Browser.• Click to select an image from the Image Browser dialog box.• Enter the full name (including location) manually, e.g. XMS:Images.NFLlogos:Redskins. <p>With these methods, both the source name and location are set.</p> |
| SourceLocation | <p>This is set automatically when you use either of the three methods explained above for setting the source name. However, if your SourceName is a macro (e.g. a variable), then you can use this row to specify the category, e.g. XMS:Images.NFLlogos.</p> |
| Keyer | <p>The keyer of the scene containing the target scene object.</p> |
| DeviceList | <p>The aliases of the devices this component controls.</p> |
| ChannelList | <p>The channels this component operates on or the Channel Selector it's tied to.</p> |

Set KeyFrame

Adding this action to an event allows you to set the value of a keyframe in an animation.


Set the following Action Parameters:

| | |
|------------------|--|
| Animation | <p>This parameter identifies the scene animation that the keyframe belongs to.</p> <p>To set this parameter:</p> <ul style="list-style-type: none"> Type in the animation's name directly in the parameter's text box. <p>Or,</p> <ol style="list-style-type: none"> Select the parameter's field and click the  button to open the Expression Builder dialog box. Select the Animation category in the left panel of the Expression Builder. A list of the scene animations now appears in the right panel of the Expression Builder. Double-click the desired animation's name and it is added to the upper text box surrounded by quotation marks (i.e. "DefaultIn"). Click OK and the Target parameter is set. |
| Keyframe | <p>Enter the keyframe name that is specified in the Keyframe Editor (see Keyframe Editor, on page 113).</p> |
| Value | <p>Enter the keyframe value as specified on the Keyframe Editor (see Keyframe Editor, on page 113).</p> |

Set Property

Adding this action to an event allows you to set any property not covered in any of the other actions. For example, you could set the alignment of an object.



Set the following Action Parameters:

| | |
|-----------------|--|
| Target | <p>This parameter identifies the object on the control-side that will be affected by the new property.</p> <p>To set this parameter:</p> <ol style="list-style-type: none">1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box.2 Select the Control category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector.3 Double-click the object that will be affected by the new property. It is added to the upper text box prefixed by "Control:".4 Click OK and the Target parameter is set. |
| Property | <p>Enter the property name, which are displayed in square brackets in the help area at the bottom of the Property Browser when you click on a property in the browser. The bracketed names are also listed in the following section: Control object properties, on page 286.</p> |
| Value | <p>Enter the property value.</p> |

Set Text

Adding this action to an event allows you to set the text that will appear in an object.


Set the following Action Parameters:

| | |
|--------------------|---|
| Source | Point to text from another object, or enter specific text in the row in quotation marks, e.g. "Stay tuned for further updates". If the source is the text from another object, click in the row, and then click the  button to display the Target/Source Selector . |
| Target | This parameter identifies the object that the text will display on. To set this parameter: <ol style="list-style-type: none"> 1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box. 2 Select the Control category in the left panel of the Target/Source Selector. A list of the control-side objects now appears in the right panel of the Target/Source Selector. 3 Double-click the desired object and it is added to the upper text box prefixed by "Control:". 4 Click OK and the Target parameter is set. |
| Format | Enter a formatting command. For details, see Formats for text, numbers, dates and times , on page 314. |
| Render | Not yet documented. |
| Keyer | The keyer this component acts upon. |
| DeviceList | The aliases of the devices this component controls. |
| Channellist | The channels this component operates on or the Channel Selector it's tied to. |

Set Visibility

Adding this action to an event allows you to set whether or not the object is visible on the scene.


Set the following Action Parameters:

| | |
|----------------|---|
| Target | This parameter identifies the object that will either visible or hidden on the scene. To set this parameter: <ol style="list-style-type: none">1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box.2 Select the Scene category in the left panel of the Target/Source Selector. A list of the scene-side objects now appears in the right panel of the Target/Source Selector.3 Double-click the desired object and it is added to the upper text box prefixed by "Scene :".4 Click OK and the Target parameter is set. |
| Visible | Enter TRUE to have the object appear on the scene, or FALSE to have the object hidden from view on the scene. |

Squeeze

Adding this action to an event allows you to control a video squeeze on the output device.

Set the following Action Parameters:

| | |
|-----------------|---|
| Target | <p>This parameter identifies the Video In object that will display the squeezed clip on the scene.</p> <p>To set this parameter:</p> <ol style="list-style-type: none"> 1 Select the parameter's field and click the  button to open the Target/Source Selector dialog box. 2 Select the Scene category in the left panel of the Target/Source Selector. A list of the scene-side objects now appears in the right panel of the Target/Source Selector. 3 Double-click the desired object and it is added to the upper text box prefixed by "Scene: ". <p>Click OK and the Target parameter is set.</p> |
| Type | <p>This parameter determines the sizing of the Video In object on the output.</p> <p>Click in the row, and then select either:</p> <ul style="list-style-type: none"> • Free - Sets squeeze to x, y, width and height values. • Full Screen - Sets squeeze to full screen. • Original - Sets squeeze to original size. |
| Duration | Enter the duration in seconds that it will take the squeeze to expand to full-screen. |
| X | Enter the co-ordinates of the video squeeze from the top of the screen. |
| Y | Enter the co-ordinates of the video squeeze from the left of the screen. |
| Width | Enter the width of the video squeeze. |
| Height | Enter the height of the video squeeze. |

Ticker Command

Adding this action to an event allows you to control the ticker's playout.

Set the following Action Parameters:

| | |
|---------------|---|
| Type | This parameter determines what type of operation will be executed on the ticker. To set this parameter: <ol style="list-style-type: none">1 Click in the row and select the drop-down list button to display a drop-down list of ticker actions.2 Select one of the following actions:<ul style="list-style-type: none">• Start• Stop• Reset• Refresh• Tick• Goto |
| Target | Open the Data Manager , and the drag a database variable and drop it into this row. |

Timer Command

Adding this action to an event allows you to control a timer object. Instructions to create and control a timer are found on [page 176](#).

Set the following Action Parameters:

| | |
|---------------|---|
| Type | This parameter determines what type of operation will be executed on the timer. To set this parameter: <ol style="list-style-type: none">1 Click in the row and select the drop-down list button to display a drop-down list of ticker actions.2 Select one of the following actions:<ul style="list-style-type: none">• Start• Stop• Reset• Set Start• Set End |
| Target | Enter the name of the timer, preceded by timer: , e.g. timer:sports. The word timer must be entered in lowercase letters. |

Unload Scene

Adding this action to an event allows you to clear the scene from the output device.

Set the following Action Parameters:

| | |
|------------------|--|
| SceneName | How you enter the scene name depends on how you created it. <ul style="list-style-type: none"> If you built the scene as a standalone, enter the scene name. If you built the scene in conjunction with a control panel or template, enter the name of the control panel or template. In either case, enter the name in parentheses. |
|------------------|--|

Xprotocol Command

The Vertigo Xprotocol is a TCP protocol that allows external communication with Xplay. An external program may connect to Xplay and control Xplay through Xprotocol commands. Use the Xprotocol Command action within Xstudio projects to issue and execute Xprotocol commands based on operator or payout events.

The following table identifies the Xprotocol Command action's parameters which appear depending upon which Type of Xprotocol command is selected:

| | |
|-----------------------|--|
| Cue | Issues a command to cue the specified item on the specified keyer(s). <ul style="list-style-type: none"> Id: The id of the item being cued. Keyer: The keyer's name that the item will be cued on. |
| Take | Issues a command to take the cued item on the specified keyer(s). <ul style="list-style-type: none"> Keyers: A comma-separated list of keyer names. |
| Take All | Issues a command to take the cued item on all configured keyers. |
| Clear | Issues a command to clear the currently displayed item(s) from the specified keyer(s). <ul style="list-style-type: none"> Keyers: A comma-separated list of keyer names. |
| Clear All | Issues a command to clear the currently displayed item(s) from all of the keyers. |
| Load | <ul style="list-style-type: none"> Id: The id of the item being loaded. Target :Identifies the object(s) that the load command will be sent/applied to. Keyer: The keyer's name that the item will be loaded to. |
| Send Event | Sends an event to an object to trigger an additional set of existing actions. <ul style="list-style-type: none"> Event: Enter an event type from the list that appears at the left side of the Action Editor (e.g. On Load, On Take, On Clear). You can also create user-defined actions for specific requirements. Targets: Identifies the object(s) that the event will be sent/applied to. Send to Children: Whether the event is only sent to the parent object (False) or to all of the children objects associated to the parent object (True). |
| Keyer On | Issues a command to turn the specified keyer(s) on. <ul style="list-style-type: none"> Keyers: A comma-separated list of keyer names. |
| Keyer Off | Issues a command to turn the specified keyer(s) off. <ul style="list-style-type: none"> Keyers: A comma-separated list of keyer names. |
| All Keyers Off | Issues a command to turn all of the active keyers off. |

Saving, publishing and exporting Xstudio projects

In the Vertigo workflow, all assets and projects are centrally stored in the Xmedia Server. Therefore, once you have created a new project or made changes to an existing project, you must save it to the Xmedia Server.

In order for a scene to be played out on an output device, the assets must find their way from the Xmedia Server to the playout device. This is accomplished by publishing the assets to the device.

Xstudio also provides you with a way to transfer assets from the connected Xmedia Server to another server by converting the assets into an export file. The export file can be saved to a folder of your choice and then imported into another Vertigo system.

The following sections describe how to save, publish, and/or export assets created or edited in Xstudio:

- [Saving Xstudio projects and assets](#), on page 252
- [Publishing assets to output devices](#), on page 253
- [Exporting assets to be used on another Vertigo system](#), on page 255

Saving Xstudio projects and assets

When changes have been made to a new or existing project in Xstudio, an asterisks appears next to the project's name in the project window. The asterisks indicates that the changes have not yet been saved to the Xmedia Server. Note that if you attempt to close a project that has unsaved changes, Xstudio will display a message prompting you to save the project.

Xstudio offers two (2) ways to save a project, as well as an overwrite option:

| | |
|------------------|--|
| Save | The File>Save (Ctrl+S) command saves changes made to the currently selected project. If you are saving the asset for the first time, the Save dialog box appears and allows you to specify the name of the asset as well as the categories in which it resides. In addition the following additional data may be specified: <ul style="list-style-type: none">• the name of the person who authored the asset• a short description of the asset• keywords, which are used to search for assets• Reacll ID• links to categories If saving changes to an existing assets, Xstudio simply saves using the current asset name without opening the Save dialog box. |
| Save As | The File>Save As command opens the Save As dialog box, which allows you to save the currently selected project using another name and/or category location. This essentially allows you to create a copy of the project using a different name. |
| Overwrite | The File>Overwrite command allows you replace (overwrite) the content of an existing (saved) project with the content of the currently selected project. This option is provided since Save As always creates a new asset. If the name of an existing asset is specified when doing a Save As, the operation is disallowed. |

Publishing assets to output devices


In the Vertigo workflow, assets are centrally stored in the Xmedia Server. Therefore, in order for a scene to be played out on an output device, you must publish the assets to the device. You can publish after creating your scene and take it to air immediately, or publish it and then take to air later.

When you publish, Xstudio's Publish Progress dialog box keeps track of which assets have been recently published and the status of the publishing process for each.

There are two (2) ways in which assets can be published to a device:

- [Manually publishing assets to devices in Xstudio](#), on page 253
- [Automatically publishing assets to devices in Xstudio](#), on page 254

Manually publishing assets to devices in Xstudio

If the project is open and you want to publish to the device that is currently connected to Xstudio, you simply have to click the Publish icon  on Xstudio's desktop.

If the project is not open, or if you want to manually publish an asset that is saved on the Xmedia Server, the following procedure describes how to manually publishing assets using the Xstudio's **Publish** commands.

To manually publish assets from the XMS to one or several devices:

- 1 Optional - Prior to publishing the asset, verify that the devices are available in Xstudio's Device Configurator (Tools>Devices>Configure Devices). If not, see [Connecting Xstudio to an output device](#), on page 18 for instructions on how to add or activate devices. If they are present, close the Device Configurator and continue.
- 2 In Xstudio's Asset Browser, select the appropriate asset category and navigate to the asset(s) that are to be published to the device(s).
- 3 Select the asset(s) you want to publish:
Select one or more assets that you want to publish. It is also possible to select one or more categories, in which case all assets in the selected category and all sub-categories are published.
Right click and a context menu will appear.
- 4 Select either the **Publish To** command or the **Force Publish To** command.
The Publish command is the most efficient choice, since it only publishes assets which have not previously been published or have changed since last being published. The Force Publish command always publishes all specified assets without performing any checking.
- 5 Select the device(s) to which the asset(s) will be published from the submenu below the **Publish To** or **Force Publish To** command.
The assets are immediately published to the device(s) and its status appears in the Publish Progress window.

Automatically publishing assets to devices in Xstudio

You can set up Xstudio so that when you ingest, import, or save an asset to a category in the Xmedia Server, the asset is automatically published to preselected devices.

Note: Enabling autopublish does not trigger a publish. To force a publish, you must manually publish using the **Publish To** command.

To set up Xstudio for automatic publishing assets to devices:

- 1 Optional - Prior to ingesting, importing, or saving the asset, verify that the devices are available in Xstudio's Device Configurator (Tools>Devices>Configure Devices). If not, see [Connecting Xstudio to an output device](#), on page 18 for instructions on how to add or activate devices. If they are, close the Device Configurator and continue.
- 2 In Xstudio's Asset Browser, select the asset category that matches the asset type of the asset to be published.
- 3 Expand the category and navigate to the sub-category where the asset(s) will be ingested, imported, or saved to.
- 4 Right-click on the sub-category and select the **Properties** command from the context menu. The **Category Properties** dialog box appears.
- 5 Enable the check boxes next to the devices that you want to publish the asset to when it is received by the Xmedia Server.
- 6 Click **OK**.

From this point on, every time an asset is added to this asset sub-category, the asset will be automatically published to the selected devices.

Note: You can set auto-publishing at the category and root category levels. As a result, any asset that is added to any of the sub-categories with the parent category will be published to the selected devices.

To deactivate the automatic publishing of a category:

- 1 Right-click on the category and select the **Remove From Devices** command. The **Publish Remove** window appears and lists the assets that are being deleted from the device. Once finished close the Publish Remove window.
- 2 Right-click on the category once again and select the **Properties** command from the context menu. The **Category Properties** dialog box appears.
- 3 Disable the check boxes next to the devices that you no longer want to automatically publish to.
- 4 Click **OK**.

Exporting assets to be used on another Vertigo system

Exporting consists of saving Xstudio assets as files. Exporting is most commonly used for copying objects and assets from one server to another. For example, a broadcaster with several stations can replicate their look and feel at each station by exporting assets from one and importing them to the others.

Xstudio offers maximum flexibility by letting you export anything -- from an individual asset (e.g. an image file) to the entire contents of an asset type's root category (e.g. Panels). Moreover, anything you export will automatically be accompanied by the assets used to build it, any proxy images, etc. For example, if exporting a control panel, all assets that were used to create it will be exported at the same time (e.g. Xmedia objects, image files, clip files, etc.).

To export one or more assets or asset categories:

- 1 From the **File** menu, select **Export**. This displays the **Export Assets** dialog box.
- 2 From either the **Asset Browser** (CTRL + A) or the **Gallery Browser** (CTRL + J), drag and drop the assets and/or asset categories you require onto the **Export Assets** dialog box.


Note: Selecting an item for export more than once will not create multiple copies.

- 3 When done, click **OK** and the **Create Export Destination Directory** dialog box appears.
- 4 Note that the **File name** field will contain a default name for the export which can be edited if desired. Browse to the folder where you wish to save the export and click **Save**. This creates a new folder containing a set of files which comprise the export.

Exporting data sources within a schema

You can export data just as you would an asset or asset category, except that you access the Export Assets dialog box from the Data Manager.

To access the Export Assets dialog box to export a schema:

- 1 Display the **Data Manager** by clicking .
- 2 Right-click on the schema to be exported and select the **Export** command. This displays the **Export Assets** dialog box.
- 3 When done, click **OK** and the **Create Export Destination Directory** dialog box appears.
- 4 Browse to the folder where you wish to save the export and then click **Save**. This exports the schema to the folder.

12

Playing out a scene using Xstudio's Production mode

While Xplay is the Vertigo Suite's full featured playout application, you can use Xstudio's Production mode to preview the playout of a project's scene on the device that is actively connected to Xstudio.

Unlike the Xstudio's Preview mode, which provides a static preview of the scene's "look", Production mode previews the scene's actual playout. Therefore, you can preview the scene's transitions, scene animations, and content. If the scene is associated with a template, you can use the template's editable fields to edit the scene's content (e.g. change text in text boxes and/or make new selections from drop-down lists) and then preview the new content. If the scene is associated with a panel, you can operate a control panel to determine the scene's content and playout behavior.

The following sections describe how to preview the playout of the various project types using Xstudio's Production mode:

- [Playing out a scene associated with a template or page](#), on page 258
- [Playing out a scene controlled by a control panel](#), on page 259
- [Playing out a standalone scene](#), on page 260


Playing out a scene associated with a template or page

Once you have finished authoring the Template+Scene project in Design mode, you can switch to Production mode to view the template's interface, as well as preview the scene's payout on an actual payout device. Similarly, when you create a new page from an existing template, you can also use Production mode to preview the new page's payout.

When you load the project into Production mode, the template/page appears in the **Next To Air** window. When a template is visible in the next to air window, it is not visible on the device. If the template/page contains any editable fields, you can use them to make changes to the scene's content (e.g. change text in text boxes and/or make new selections from drop-down lists).

When you are ready to preview the scene's payout, you can payout the scene by clicking the **Take** button. Since Production mode is a true payout of the scene, you will immediately see the scene's payout including its currently programmed transitions, animations, and content. To preview the scene's exiting transition and clear the device, simply click the **Clear** button.

The following instructions guide you through the exact procedure for playing out a Template+Scene or a page in Xstudio's Production mode.

- 1 Ensure that the Template+Scene or Page project is open and currently selected in Xstudio's Design mode.
- 2 Ensure that Xstudio is actively connected to the payout device (see [page 18](#)).
- 3 Save and publish the current project to the active output device (see [page 252](#) and [page 253](#)).
- 4 Switch Xstudio to Production Mode by double-clicking the **Design Mode button**  or selecting the **View>Production Mode (CTRL+P)**.

Note: If the Template's AutoLoadScene property is set to false the scene is not automatically cued on the device and may not appear on the payout device when the Take button is pressed.

- 5 Use the template's editable fields (i.e. text boxes, drop-down lists, buttons...etc.) to make changes to the scene's content.
- 6 Click the Take button to bring the scene to air on the device.
Watch the scene's entry transition, its content and animations.
- 7 After viewing enough of the scene's payout, click the **Clear** button to execute the scene's exit and clear the device. Watch the scene's exit transition.
- 8 Optional: To immediately preview the scene again, click the **Cue** button and repeat steps 5 to 7 again.


Playing out a scene controlled by a control panel

Once you have finished authoring the Panel+Scene project in Design mode, you can switch to Production mode to view the control panel's interface, as well as preview the scene's playout on an actual playout device.

When you load the project into Production mode, the control panel is automatically loaded in Xstudio's Production mode window and the scene is taken to air. You can now operate the control panel to determine the scene's content and playout behavior, which will be immediately reflected on the output device.

To stop the playout of the scene and clear the device, select the **Tasks>Clear Device** menu command or press Ctrl+F4. Note that the **Cue**, **Refresh**, **Clear** and **Take** buttons are not enabled when previewing a control panel and scene.

The following instructions guide you through the exact procedure for playing out a Panel+Scene in Xstudio's Production mode.

- 1 Ensure that the Panel+Scene project is open and currently selected in Xstudio's Design mode.
- 2 Ensure that Xstudio is actively connected to the playout device (see [page 18](#)).
- 3 Save and publish the current project to the active output device (see [page 252](#) and [page 253](#)).
- 4 Switch Xstudio to Production Mode by double-clicking the **Design Mode button**  or selecting the **View>Production Mode (CTRL+P)**.
Note that if the panel's **autoLoadScene** property is false, the scene may not appear on the playout device.
- 5 Use the panel's elements (i.e. text boxes, drop-down lists, buttons...etc.) to control the scene's content and behavior.
- 6 After viewing enough of the scene's playout, select the **Tasks>Clear Device** menu command or press Ctrl+F4 to execute the scene's exit and clear the device.


Playing out a standalone scene

Once you have finished authoring a standalone Scene in Design mode, you can switch to Production mode to view the scene's elements in the Production mode window and preview its playout on an actual playout device.

When you load the scene into Production mode, the scene is automatically loaded in Xstudio's Production mode window and the scene is displayed on the device.

Since Production mode is a true playout of the scene, you will immediately see the scene's playout including its currently programmed transitions, animations, and content. After viewing enough of the scene's playout, select the **Tasks>Clear Device** menu command or press **Ctrl+F4** to execute the scene's exit and clear the device.

The following instructions guide you through the exact procedure for playing out a standalone Scene in Xstudio's Production mode.

- 1 Ensure that the standalone scene is open and currently selected in Xstudio's Design mode.
- 2 Ensure that Xstudio is actively connected to the playout device (see [page 18](#)).
- 3 Save and publish the current scene to the active output device (see [page 252](#) and [page 253](#)).
- 4 Switch Xstudio to Production Mode by double-clicking the **Design Mode button**  or selecting the **View>Production Mode (CTRL+P)**.

Watch the scene's entry transition, its content and animations.

- 5 After viewing enough of the scene's playout, select the **Tasks>Clear Device** menu command or press **Ctrl+F4** to execute the scene's exit and clear the device.

Asset Browser

Assets are the building blocks that you use for creating and controlling your broadcast graphics. As such, Xstudio provides you with convenient access to these assets in Design mode through the Asset Browser. The Asset Browser is a collapsible panel on the left-side of the Xstudio interface that allows you to view and access assets according to type and category.

The practical purpose of the Asset Browser allows you to drag-and-drop assets into your new creations, or quickly open existing assets.

The following topics describe how to perform various tasks using the Asset Browser:

- [Opening, closing, and resizing the Asset Browser](#), on page 262
- [Structure of the Asset Browser](#), on page 263
- [Creating Asset Browser categories](#), on page 266
- [Deleting Asset Browser categories](#), on page 267
- [Renaming an Asset Browser category](#), on page 268
- [Publishing a category's assets to one or more output devices](#), on page 268
- [Moving or copying assets to different categories](#), on page 269
- [Moving or copying assets to different categories](#), on page 269
- [Opening and editing assets using the Asset Browser](#), on page 271
- [Deleting assets using the Asset Browser](#), on page 272
- [Viewing and editing asset properties using the Asset Browser](#), on page 273

Note: The Asset Browser and the Gallery Browser are two Xstudio browsers that compliment each other, and as such, some of their functionality regarding managing and providing access to assets overlaps. See [Asset Browser and Gallery Browser](#), on page 38 for descriptions and instructions of the tasks that can be performed using the Gallery Browser.

Opening, closing, and resizing the Asset Browser

The Asset Browser is a collapsible browser that is available in Xstudio's Design mode. By collapsible, we mean that you open, hide, and resize the browser's display allowing you to setup the layout of your Xstudio interface to your liking. The Asset Browser is located and anchored to the left side of the Xstudio interface.

To display or hide the Asset Browser:

- From the **View** menu, select **Asset Browser**.
Or,
- Press **CTRL + A**.
Or,
- Click the gripper bar at the left-hand side of the screen.
Or,
- While holding down the **SHIFT** key, move your mouse over to the blue bar on the left side of the screen.

To resize the Asset Browser:

- Drag the blue boundary on the right side of the browser until it is the desired width.

Structure of the Asset Browser

The Asset Browser organizes assets within a set of root categories according to the type of asset (e.g., scenes, panels, templates, images, clips, scripts, etc.). Therefore, a root category contains all of the assets of the indicated type. For example, all control panels are found under the **Panels** root category. This logical organization provides you with quick access to the assets.

To further organize your assets, you can create sub-categories under a root category. Each of these categories, in turn, may contain assets and further sub-categories (see [Creating Asset Browser categories](#), on page 266).

When saving an asset, you can choose one or more categories in which to save it. The Pages root category may contain a Headlines category as well as a Health category. A new page that contains graphics relating to a major community health warning could be saved to both the Headlines and Health categories. Note that only one copy of the page is stored, but it is accessible through both categories. Taking this into consideration, any edits to the asset accessed from one location will apply to the asset in all other locations too.

Root categories are pre-set within the system and cannot be moved or deleted, nor can new ones be created. However, you can create as many subcategories within root categories as required, as well as edit and delete them (see [Creating Asset Browser categories](#) and [Deleting Asset Browser categories](#), on page 267).

The following table lists all root categories and describes the asset types they contain:

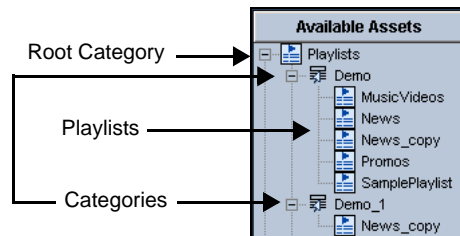
| | |
|------------------|--|
| Playlists | Contains playlists which are an ordered sequence of pages and other elements that determine what is brought to air, in what order, and for how long. |
| Rundowns | Contains Rundowns which are a form of playlist that contains a collection of Segments. Note that Rundowns cannot be opened in Xstudio. |
| Segments | Contains Segments which are sets or groupings of related playlist elements. |
| Pages | Pages are copies of a template that are then used for entering specific data and corresponding assets such as images, clips, animations, etc. Once created, pages can be organized into playlists. Page assets can be directly opened in Xstudio. |
| Templates | Contains templates, either with or without an associated scene. A template is form that determines the functionality of a scene of visual and/or audio elements that is to be played out sequentially. When the template is used, the original template is not changed but copies of the template itself are made to accommodate specific text and/or images. These copies are called pages. Pages are then arranged within playlists for sequential payout. This asset type does not include on-air graphics. To build on-air graphics you need a scene. Control panels can be opened directly from the Asset Browser. |

| | |
|----------------------|--|
| Panels | <p>Contains control panels, either with or without an associated scene. A control panel is a form that allows visual and/or audio assets to be taken to air as they are required using manual intervention or triggered events, rather than in a planned sequential order.</p> <p>Control panels can be opened directly from the Asset Browser.</p> |
| Scenes | <p>Contains Scenes that were created as standalones, meaning that they are not associated to a template or control panel. The scenes contain the actual visual and audio elements that make up the final broadcast output.</p> <p>Scenes can be opened directly from the Asset Browser.</p> |
| XmediaObjects | <p>Contains Xmedia objects which are pre assembled objects that contain graphic elements, objects to control the graphic elements, and links to data. You can bring Xmedia objects into control panels, templates, scenes, and even use them as the basis for other Xmedia objects.</p> <p>Xmedia objects can be opened directly from the Asset Browser and/or dragged onto the Xstudio canvas.</p> |
| Images | <p>Contains common image files in the following formats: .tga, .tif, .jpg, .bmp, and .oxt files.</p> <p>Image assets can be dragged directly from the Asset Browser onto the Xstudio canvas.</p> |
| CelAnimations | <p>Contains common image and cel animation files in the following formats: .tga and .oxi files.</p> <p>Cel animation assets can be dragged directly from the Asset Browser on to the Xstudio canvas.</p> |
| Clips | <p>Contains common video files in the following formats: .MPG or VAF (Vertigo Animation Format).</p> <p>Clip assets can be dragged directly from the Asset Browser onto the Xstudio canvas.</p> |
| Audio | <p>Contains common audio files in the following formats: Wave and MP3 files.</p> <p>Audio files can be added to animations using the Play Audio channel command in the Animation Editor, or using the Play Audio action in the Action Editor.</p> |
| Scripts | <p>Contains script files that are a series of commands in any standard scripting language such as VB Script, JavaScript, or Perl.</p> <p>Opening a script from the Xstudio Asset Browser launches the Script Editor.</p> |
| LookupTables | <p>Contains Lookup tables which are substitution tables that are used to define replacement values for incoming data (e.g. abbreviations, file names, etc.).</p> <p>Lookup tables are applied to specific assets (e.g. Control Panel + Scene). For example, if "sunny" is an incoming data item that will be used in your weather display, you can use a lookup table to automatically replace the word with the file name of an image depicting the sun.</p> <p>Opening a lookup table from the Xstudio Asset Browser launches and populates the lookup table dialog box.</p> |
| Work Orders | <p>Contains Work Orders which are requests to perform one or more tasks. Each task is a Job, so the Work Order is essentially a list of Jobs.</p> <p>Note that the Work Order assets listed in this root category cannot be opened in Xstudio.</p> |

| | |
|----------------|---|
| Jobs | <p>Contains Jobs which are requests to perform a task. A Job passes through a series of user-defined states from creation to completion, which allows jobs to be assigned, submitted and approved before they are considered complete.</p> <p>Note that the Jobs assets listed in this root category cannot be opened in Xstudio.</p> |
| Generic | <p>Contains assets that do not appropriate for any other root category, but the asset needs to be stored on the Xmedia Server. For example, Word documents, or special files for third party device drivers can be added to the Asset tree on the Xstudio canvas with the intention of publishing it to a device.</p> <p>Note that the Generic assets listed in this root category cannot be opened in Xstudio.</p> |
| Fonts | <p>TrueType font files (.TTF) can be stored in this root category and then be added to the Asset tree on the Xstudio canvas with the intention of publishing it to a device.</p> <p>Note that the Fonts listed in this root category cannot be opened in Xstudio.</p> |

Creating Asset Browser categories

The Asset Browser organizes assets into root categories according to the type of asset. You can further organize your assets by creating further categories within a root category.



To create a category within a root category:

- 1 In the Asset Browser, expand the root category in which you want to add a category by clicking the + next to its heading name.
- 2 Right-click on the root category heading in which you would like to create a new category (e.g. Pages).
A context menu appears.
- 3 Select **New Category** from the context menu.
The **New Category** dialog box appears and asks you to enter a name.
- 4 Type the name that you would like to give to the new category in the text box and click **OK**.

The new category is displayed below the root category heading in alphabetical order.

Note: You cannot add or delete root categories from the Asset Browser. However, you can create as many categories within the root categories as you would like.
Also note, that you cannot create or delete categories within the Layouts root category.

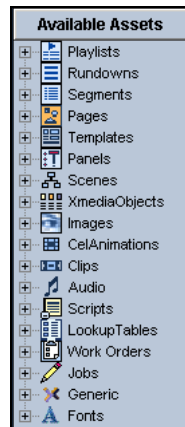
To create a category within a category:

- 1 In the Asset Browser, navigate through the root category down to the category into which you want to add a category.
- 2 Right-click on the category heading in which you would like to create a new category.
A context menu appears.
- 3 Select **New Category** from the context menu.
The **New Category** dialog box appears and asks you to enter a name.
- 4 Type the name that you would like to give to the new category in the text box and click **OK**.

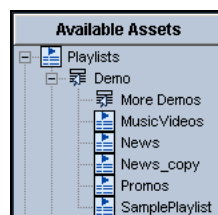
The new category is displayed below the category heading in alphabetical order.

Deleting Asset Browser categories

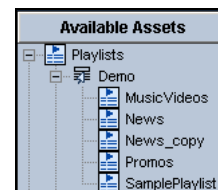
Root categories and categories containing subcategories cannot be deleted. In other words, only categories which contain no other categories can be deleted from the Asset Browser.



Root categories cannot be deleted



Categories containing subcategories cannot be deleted



Categories containing no subcategories can be deleted, but so will the asset links

If a category contains only assets, but no sub-categories, then the category can be deleted. However, be aware that the assets contained in the category will also be deleted, unless a copy of the asset is located in another category.

To delete a category:

- 1 In the Asset Browser, right-click on the category heading.
A context menu appears.
- 2 Select **Delete** from the context menu.
- 3 The **Confirm Delete** dialog box appears and verifies whether you really want to delete this category. The dialog box even gives you a summary of the contents of the category.
- 4 If you would still like to delete the category, click **Yes** and the category is immediately removed from the Asset Browser.
If, however, you no longer want to delete the category, click **No** and no changes are made to the category.

Renaming an Asset Browser category

While root category heading names cannot be changed, you can change the names of the categories. When changing category names, keep in mind that the purpose of the Asset Browser is to create a organized structure where assets can be located quickly. Therefore, it is highly recommended that you choose category names that are concise, yet meaningful, so that you can easily locate the necessary assets.

To rename a category:

- 1 In the Asset Browser, right-click on the category heading.
A context menu appears.
- 2 Select **Rename Category** from the context menu.
The **New Category** dialog box appears displaying the current name in the text box. It also asks you to enter a new name for the category.
- 3 Type the name that you would like to give to the category in the text box and click **OK**.
- 4 The category's new name is immediately assigned to the category and the category's location may change to maintain the alphabetical order.

Publishing a category's assets to one or more output devices

If you would like to display a category's assets on one or several output devices, you will have to use the **Category Properties** dialog box to publish the asset category to the selected devices.

To have a category automatically publish assets to one or more specified output device(s):

- 1 In the Asset Browser, right-click on a category.
A context menu appears.
- 2 Select **Properties** from the context menu.
- 3 The **Category Properties** dialog box appears displaying the name of the category and listing the available output devices configured to your system.
- 4 Select the desired output devices and then click the **OK** button to apply the selection(s) and close the dialog box. Or, click **Apply** to apply the selection(s) and keep the **Category Properties** dialog box open.

Moving or copying assets to different categories

The Asset Browser allows you to have easy access to an organized, central listing of assets. You may, however, want to adjust location of the assets in your Asset Browser to a logic that better fits your needs. It is for this reason that Xstudio allows you to move assets within categories containing assets of the same type.

For example, an image asset of the Police Chief that was originally saved to the **Politics** category during nominations, might now be more appropriate in the **Police** category. You can either move the asset to the Police category, or create an asset link, which adds a placeholder in the Police category and leaves the original asset in the Politics category. Note that the asset link means that the asset is linked to the original asset, so saved changes to one will also be applied to the other.

Note: You cannot move or copy assets from one root category to another because all of the assets in a root category must be of the same type (i.e. images root category only contains images).

To move an asset within the root category:

- 1 In the Asset Browser, navigate to and select the category where you would like to move the asset to.
- 2 Navigate to and select the asset that you would like to move.
- 3 Drag the asset and drop it onto the desired category.

To copy an asset link to another category:

- 1 In the Asset Browser, navigate to and select the category where you would like to copy the asset to.
- 2 Navigate to and select the asset that you would like to copy.
- 3 Click and hold down the **CTRL** key while dragging the asset, and then drop the asset over the root category or desired category within the root before releasing the **CTRL** key. An asset link is now placed in the new location and the asset still remains in its original location.

Note: You can verify which categories an asset is linked to by right-clicking on the asset heading in the Asset Browser and choosing Properties from the context menu. The **Properties** dialog box appears and lists the location(s) of the asset in the **Categories** field.

Removing assets from categories

An asset can exist within one or many categories in the Asset Browser as linked assets, meaning that each instance of the asset is linked together. Therefore, saved changes to one instance of the asset are applied to all of the same linked assets.

Using the **Remove from Category** command, Xstudio allows you to remove an instance of an asset from a category, without deleting it from the system. Note, however, that the **Remove from Category** command is only available if another instance of the asset exists in another category. This is a safety feature to prevent you from mistakenly deleting the only remaining instance of the asset.

If your intention is to completely remove the asset from the system, you will use the **Delete** command, instead of the **Remove from Category** command (see [page 272](#)).

To remove an asset from a category:

- 1 In the Asset Browser, right-click on the asset that is to be deleted.
A context menu appears. If the asset has other linked locations, the **Remove From Category** command will be available. However, if the asset does not exist in any other category, then the command will not be available.
- 2 Select **Remove From Category** from the context menu.
The **Confirm Remove from Category** dialog box appears and asks you to confirm that you want to remove the specified asset from the specified category.
- 3 Click **Yes** to remove the asset from the category, or click **No** to cancel the operation.

Opening and editing assets using the Asset Browser

You can use the **Asset Browser** to open assets so that they can be edited. Assets that were created in Xstudio can be edited within the application. Those that were created in a third-party product must be edited outside Xstudio and then re-ingested (see [page 59](#)).

The following table provides a list of the assets that can be edited using Xstudio, and which require a third-party application.

| Open and edit within Xstudio | Open and edit in third-party application |
|--|--|
| <ul style="list-style-type: none"> • Pages • Templates • Control panels • Scenes • Xmedia objects • Scripts • Lookup Tables | <ul style="list-style-type: none"> • Images • Clips • Cel animations • Audio |

Note: Opening and editing assets can also be done from within the Gallery Browser using the same procedure.

To open an asset for editing using the Asset Browser:

- 1 In the Asset Browser, double-click the desired asset, e.g. Panel1 under the Panels category.

The asset opens in its appropriate window in the center of the Xstudio interface.

- 2 Edit as required using Xstudio's commands, toolbars, and browsers.

For example, you can drag and drop another asset onto the scene, reposition a text box or change the font size or style, add a new segment to a playlist, create a new control element, along with a number of other operations that affect the content, appearance, or functioning of the asset.

To edit an asset for which a third-party application is required:

- 1 In the Asset Browser, right-click on the name of the asset to be edited.

A context menu appears.

- 2 Select **Save Local Copy** from the context menu.

The **Save Local Copy** dialog box appears, which allows you to navigate to the directory location where the asset will be exported to.

- 3 Either select the default export folder, create a new folder in the default location, or navigate to another location.

- 4 Click the **Save** button.

The asset has been exported to the selected folder.

- 5 Open and edit the asset in the appropriate third-party application. Be sure to save all of the desired changes.

- 6 Ingest the asset into Xstudio using the same name as the version you are replacing. This will overwrite the old one. For instructions on ingesting, see [Ingesting files created using third-party applications](#), on page 59.

Deleting assets using the Asset Browser

The Asset Browser's **Delete** command allows you can delete assets completely from the system, regardless of whether the asset is linked to other categories.

If your intention is only to remove a linked asset from a category, not to remove the asset from the system, use the **Remove from Category** command, instead of the **Delete** command (see [Removing assets from categories](#), on page 270).

To delete an asset:

- 1 In the Asset Browser, right-click on the asset that is to be deleted.
A context menu appears.
- 2 Select **Delete** from the context menu.
The **Confirm Delete** dialog box appears and asks you to confirm that you want to remove the specified asset.
- 3 Click **Yes** to remove the asset, or click **No** to cancel the operation.

Viewing and editing asset properties using the Asset Browser

When you save an asset in Xstudio, you are prompted to provide information that helps to characterize and archive the asset, such as the author's name, a description of the asset, category and keyword assignments, the creation date... etc. We refer to this information as the asset's Properties.

Once the asset has been saved, its properties can be viewed and some property fields can be edited. There is a set of property fields that are common to all assets, but there are also property fields that are specified to the type of asset. The following table identifies the asset property fields that are common to all assets:

| | |
|-------------------------|--|
| Name | Displays asset's name. <ul style="list-style-type: none">• This is a read-only field. However, if you would like to change the name of the asset, you can by using the Asset Browser's Rename command.• It is recommended that you give assets names that reflect their content or use, so that they are easily recognizable and distinguish from similar assets. |
| Type | Displays the asset type. This is a read-only field that cannot be changed. |
| Author | Displays the name of the creator/author of the asset. You can edit this field by clicking in the text box and typing the desired name. |
| Description | Displays a textual description of the asset. <ul style="list-style-type: none">• You can edit this field by clicking in the text box and typing the desired description.• It is recommended that you provide a concise description for each asset that identifies its content or use, so that it is easily recognizable and distinguish from similar assets. |
| Created Time | Displays the date and time of when the asset was originally created. This is a read-only field that cannot be changed. |
| Modified Time | Displays the time of the asset's most recent modification. This is a read-only field that cannot be changed. |
| Expiry Date | Displays the calendar date when the asset will be deleted from the XmediaServer. You can edit this field by clicking in the field's text box, which activates a button on the right side of the field. Click the button to display a calendar from which you can select the desired date, or select the None button at the bottom of the calendar to set no expiration date. |
| Publish Lifespan | Displays the specific number of days before the asset is deleted from the XmediaServer. You can edit this field by clicking in the text box and typing a number corresponding to the number of days the asset will be stored it is deleted from the XmediaServer. |

| | |
|-------------------|--|
| Categories | Displays the category, or categories, where the asset is accessible from in the Asset Browser. This is a read-only field. However, if you would like to move, add, or remove the asset to/from a category, you can do so from within the Asset Browser (see Moving or copying assets to different categories , on page 269 and Removing assets from categories , on page 270. |
| Keywords | Displays the keyword(s) that have been assigned to the asset when it was created and saved. Keywords are used as a searchable property by Xstudio's Search XMS feature to help you locate assets within the system. |

Not only do Image assets have their own additional property fields, but they are also the only asset type to have access to the Advanced button. When clicked, the Advanced button opens an Image Properties window for the asset that allows you to document further details about the contents, characteristics, and history of the image asset.

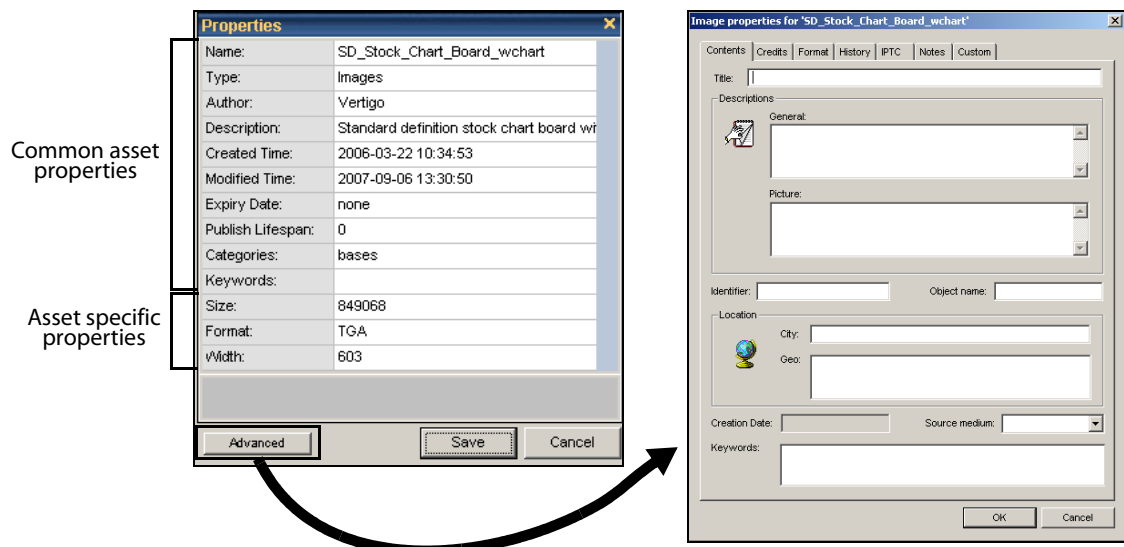


Fig. A-1: Image assets have access to more advanced property fields

To view and/or modify a saved asset's properties:

- 1 In the Asset Browser, right-click on the desired asset.
A context menu appears.
- 2 Select **Properties** from the context menu.
The **Properties** dialog box appears and displays a list of the asset's properties.
- 3 Edit the appropriate property fields.
- 4 When done, click the **Save** button to register modifications, or **Cancel** to close the dialog box without saving changes.

B Gallery Browser

While the Asset Browser provides you with access to all of the assets stored on the system, the Gallery Browser helps you to better determine which assets to drag and drop onto your canvas. The Gallery Browser is a collapsible browser in Design mode, which displays the contents of an Asset Browser category as either thumbnail images or as a detailed list. Using the Gallery Browser allows you to select and use assets quickly and with confidence.

The default state of the Gallery Browser displays the root category of Xmedia objects, which are pre-built broadcast graphic elements (e.g. headlines, tickers, lower-thirds, over-the-shoulder graphics) that can be dragged and dropped into control panels or templates. However, the Gallery Browser can also be made to display any other asset type, and in greater detail than when it is listed in the Asset Browser.

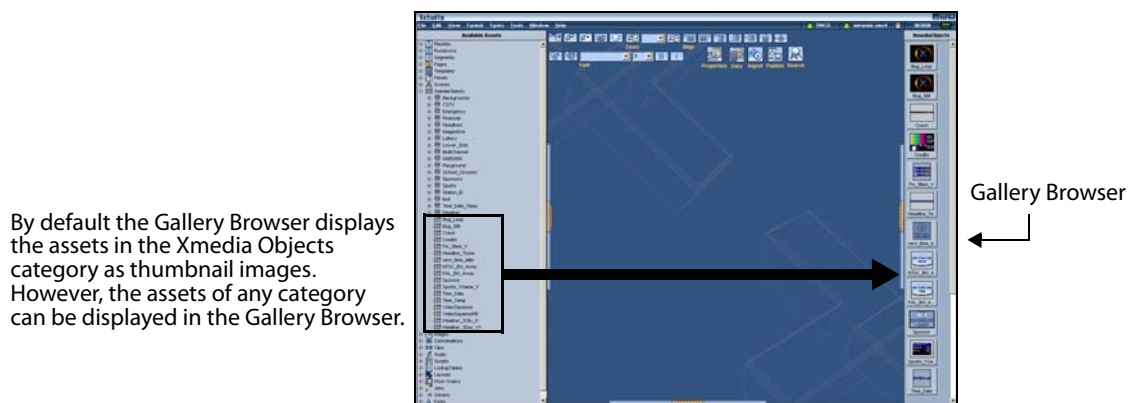


Fig. B-1: The Gallery Browser displays the contents of the Asset Browser categories

The following topics describe how to perform various tasks using the Gallery Browser:

- [Opening, closing, and resizing the Gallery Browser](#), on page 276
- [Switching between Proxy and Detail view in the Gallery Browser](#), on page 277
- [Viewing assets in the Gallery Browser](#), on page 279
- [Adding and deleting asset categories to the Gallery Browser](#), on page 280
- [Refreshing the Gallery Browser](#), on page 281
- [Opening and editing assets using the Gallery Browser](#), on page 281
- [Deleting assets using the Gallery Browser](#), on page 282
- [Viewing and editing asset properties using the Gallery Browser](#), on page 282

Note: The Gallery Browser and the Asset Browser are two Xstudio browsers that compliment each other, and as such, some of their functionality regarding managing assets overlaps. See [Asset Browser and Gallery Browser](#), on page 38 for descriptions and instructions of the tasks that can be performed using the Asset Browser.

Opening, closing, and resizing the Gallery Browser

The Gallery Browser is a collapsible browser that is available in Xstudio's Design mode. By collapsible, we mean that you open, hide, and resize the browser's display allowing you to setup the layout of your Xstudio interface to your liking. The Gallery Browser is located and anchored to the right side of the Xstudio interface.

To display or hide the Gallery Browser:

- From the **View** menu, select **Gallery**.
Or,
- Press **CTRL + G**.
Or,
- Click the gripper bar on the right-hand side of the screen.
Or,
- While holding down the **SHIFT** key, move your mouse over to the blue bar on the right side of the screen.

To resize the Gallery Browser:

- Drag the blue boundary on the left side of the browser until it is the desired width.

Switching between Proxy and Detail view in the Gallery Browser

You can choose to display assets in the Gallery Browser in two formats: as proxies (thumbnails) or as detailed asset information.

The following sections describe each format:

- [Proxy View](#)
- [Detail View](#)
- [Switching between Proxy and Detail views](#)

Proxy View

Displaying assets in Proxy view provides that advantage of previewing an image of the asset for identification purposes. This can be helpful for quickly retrieving an asset by simply recognizing the look of the asset.

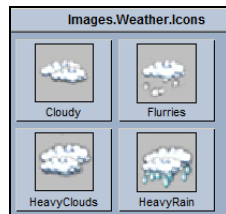


Fig. B-2: Image asset category displayed in proxy view

Not all assets can be viewed in proxy mode. Since proxy view displays the assets using thumbnail images to preview the asset, the asset must have an image or visual component associated to it. Some assets, like playlists or audio, don't have any helpful visual elements to it and therefore cannot be displayed in proxy mode. The following table lists the asset types that can and those that cannot be viewed in proxy view:

| Asset types that can be displayed in Proxy view | Asset types that cannot be displayed in Proxy view |
|---|--|
| • Pages | • Playlists |
| • Templates | • Rundowns |
| • Panels | • Segments |
| • Scenes | • Audio |
| • Xmedia Objects | • Scripts |
| • Images | • Lookup Tables |
| • Cel Animations | • Work Orders |
| • Clips | • Jobs |
| • Generic | |
| • Fonts | |

An asset whose type is usually able to display a thumbnail in proxy mode, but does not have an associated image displays the following thumbnail in the Gallery Browser:



Detail View

Displaying assets in Detail view provides that advantage of being able to see some of the asset's property information in the listing (i.e. name, description, modified timestamp, and file size). Displaying assets in detail view can be helpful for quickly identifying and retrieving an asset.

| Images.sports.logos.nhl | | | | |
|-------------------------|-------------|---------------------|--------|--|
| Name | Description | Modified Time | Size | |
| AllStarGame2007_Logo | | 2006-03-21 21:58:28 | 126 KB | |
| Anaheim_MightyDucks | | 2006-03-21 21:58:29 | 126 KB | |
| Atlanta_Thrashers | | 2006-03-21 21:58:30 | 126 KB | |
| Boston_Bruins | | 2006-03-21 21:58:31 | 126 KB | |
| Buffalo_Sabres | | 2006-03-21 21:58:32 | 126 KB | |
| Calgary_Flames | | 2007-07-25 17:31:41 | 126 KB | |
| Carolina_Hurricanes | | 2006-03-21 21:58:33 | 126 KB | |
| Chicago_Blackhawks | | 2007-07-25 17:31:41 | 126 KB | |
| Colorado_Avalanche | | 2006-03-21 21:58:35 | 126 KB | |
| Columbus_BlueJackets | | 2006-03-21 21:58:36 | 126 KB | |
| Dallas_Stars | | 2006-03-21 21:58:37 | 126 KB | |

Fig. B-3: Image asset category displayed in detail view

Unlike proxy view, assets of any type can be displayed in the Gallery Browser's detail view. In fact, as the table on [page 277](#) shows, certain asset types can only be displayed in the Gallery Browser using the detail view because they do not have an image to use in proxy view.

Switching between Proxy and Detail views

Each asset category has its own tab in the Gallery Browser that is used to access and identify the asset category. The view setting (proxy/detail) for each asset category is independently set. In other words, some of the tabs can be set to display its assets in proxy view, while others are in detail view.

To switch between Proxy and Detail view:

- 1 In the Gallery Browser, right-click on category tab.

A context menu appears.

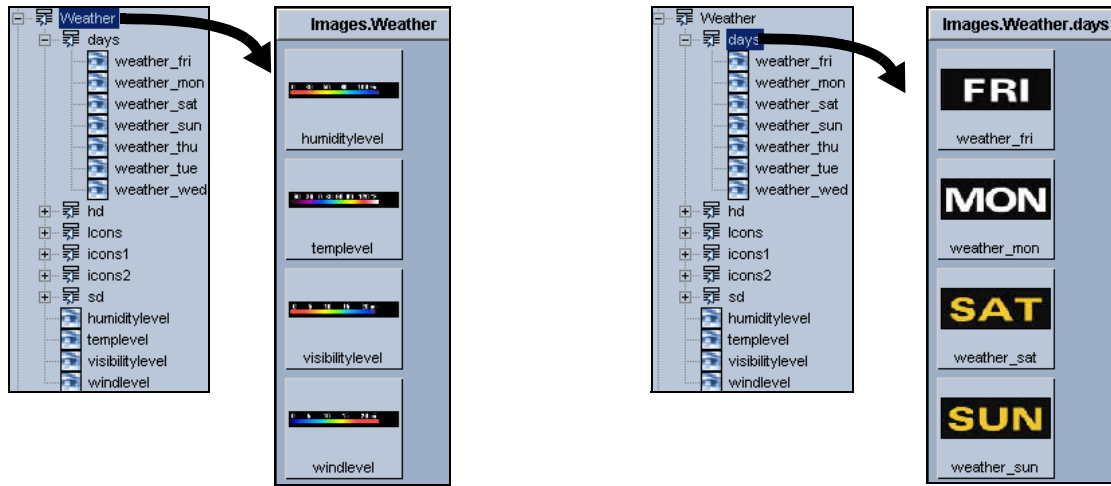
- 2 Select the **Proxy View** or **Detail View** command from the context menu.

The Gallery Browser automatically changes to the new view format.

If you were not able to select a command from the context menu because it was disabled, it probably means that the asset category's view is already set in the only view setting (detail) that is compatible with that asset type (see the table on [page 277](#) for a list of asset types that can only be set to detail).

Viewing assets in the Gallery Browser

By default, the Gallery Browser displays the root category of Xmedia objects. However, the Gallery Browser can also be made to display the assets contained in any other asset category from the Asset Browser. Be aware however, that the Gallery Browser displays only assets directly in the category, so if the category contain subcategories, you will not be able to view the sub-categories or their contents in the Gallery Browser. If you want to see the assets in the sub-categories, then you would open the sub-categories in the Gallery Browser separately.



The Gallery Browser only displays the assets immediately in the category. It does not display the sub-categories, nor the assets in those sub-categories.

To display the assets contained in a sub-category, open the sub-category in the Gallery Browser as a separate tab.

To open/view the assets of a category in the Gallery Browser:

- 1 In the Asset Browser, right-click on category name.
A context menu appears.
- 2 Select the **Open** command from the context menu.

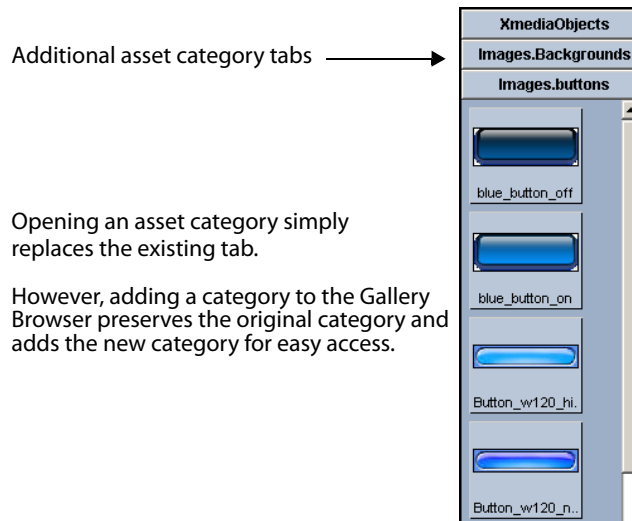
The Gallery Browser automatically displays the category's assets.

Upon closer inspection you may notice that the Gallery Browser has actually replaced the previous Gallery Browser contents with the new category's assets. This is evident by the Gallery Browser's single tab, which now has the new category's name. If you would rather add the category's assets to the Gallery Browser, while maintaining the previous/existing category, see [Adding and deleting asset categories to the Gallery Browser](#), on page 280.

Note: Categories displayed in the Gallery Browser display their full name. For example, the Sports category in the Scenes root category will display as Scenes.Sports.

Adding and deleting asset categories to the Gallery Browser

If there is a root category or asset category that you would like to have available for display in the Gallery Browser at all times -- not just when you select it from the Asset Browser -- you can set up a permanent tab for it. For example, you may want all background images and all button images available for quick and easy viewing. Rather than opening these categories in the Gallery Browser, which would just replace each other, you can add a tab for each in the Gallery Browser. Once added to the Gallery Browser, you can quickly access the assets by simply clicking on the category's tab in the Gallery Browser.



Additional asset category tabs →

Opening an asset category simply replaces the existing tab.

However, adding a category to the Gallery Browser preserves the original category and adds the new category for easy access.

Fig. B-4: The Gallery Browser can manage multiple asset categories for practical access

To add a category to the Gallery Browser:

- Click on the category's name in the Asset Browser and drag it into main part of the Gallery Browser (not on the tabs).

A new tab is created, displaying the full name of the category, and the assets of the category are displayed in the Gallery Browser.

Once you no longer have an immediate need for the category to occupy a permanent tab in the Gallery Browser, you can remove it from the Gallery Browser.

Note: Adding these category tabs to the Gallery Browser are user-specific and only appear on the machine where they were created.

To delete a category's tab from the Gallery Browser:

- 1 Right-click on the tab.

A context menu appears.

- 2 Select the **Remove** command from the context menu.

The category's tab and its contents are immediately removed from the Gallery Browser.

Note: Although you can remove categories from the Gallery Browser, there must always remain at least one category tab in the Gallery Browser.

Refreshing the Gallery Browser

If you are using Xstudio on a network, you might be working while other users are making changes or additions to the assets available on the MediaServer. To ensure that what you are viewing in the Gallery Browser is as up-to-date as possible, we encourage you to refresh the Gallery Browser periodically.

To refresh the Gallery Browser:

- 1 Right-click on one of the Gallery Browser's tabs.
A context menu appears.
- 2 Select the **Refresh** command from the context menu.
The Gallery Browser quickly ensures that all assets are updated.

Opening and editing assets using the Gallery Browser

You can use the Gallery Browser to open assets so that they can be edited. Assets that were created in Xstudio can be edited within the application. Those that were created in a third-party product must be edited outside Xstudio and then re-ingested (see [page 59](#)).

The following table provides a list of the assets that can be edited using Xstudio, and which require a third-party application.

| Open and edit within Xstudio | Open and edit in third-party application |
|--|--|
| <ul style="list-style-type: none"> • Pages • Templates • Control panels • Scenes • Xmedia objects • Scripts • Lookup Tables | <ul style="list-style-type: none"> • Images • Clips • Cel animations • Audio |

Note: Opening and editing assets can also be done from within the Asset Browser using the same procedure.

To open an asset for editing using the Gallery:

- 1 In the Gallery Browser, double-click the desired asset.
The asset opens in its appropriate window in the center of the Xstudio interface.
- 2 Edit as required using Xstudio's commands, toolbars, and browsers.
For example, you can drag and drop another asset onto the scene, reposition a text box or change the font size or style, add a new segment to a playlist, create a new control element, along with a number of other operations that affect the content, appearance, or functioning of the asset.

To edit an asset for which a third-party application is required:

- 1 In the Gallery Browser, right-click on the name of the asset to be edited.
A context menu appears.
- 2 Select **Save Local Copy** from the context menu.

The **Save Local Copy** dialog box appears, which allows you to navigate to the directory location where the asset will be exported to.

- 3 Either select the default export folder, create a new folder in the default location, or navigate to another location.
- 4 Click the **Save** button.
The asset has been exported to the selected folder.
- 5 Open and edit the asset in the appropriate third-party application. Be sure to save all of the desired changes.
- 6 Ingest the asset into Xstudio using the same name as the version you are replacing. This will overwrite the old one. For instructions on ingesting, see [page 59](#).

Deleting assets using the Gallery Browser

The Gallery Browser’s **Delete** command allows you can delete assets completely from the system, regardless of whether the asset is linked to other categories.

To delete an asset:

- 1 In the Gallery Browser, right-click on the asset that is to be deleted.
A context menu appears.
- 2 Select **Delete** from the context menu.
The **Confirm Delete** dialog box appears and asks you to confirm that you want to remove the specified asset.
- 3 Click **Yes** to remove the asset, or click **No** to cancel the operation.

Viewing and editing asset properties using the Gallery Browser

When you save an asset in Xstudio, you are prompted to provide information that helps to characterize and archive the asset, such as the author’s name, a description of the asset, category and keyword assignments, the creation date... etc. We refer to this information as the asset’s Properties.

Once the asset has been saved, its properties can be viewed and some property fields can be edited. There is a set of property fields that are common to all assets, but there are also property fields that are specified to the type of asset. The following table identifies the asset property fields that are common to all assets:

| | |
|---------------|--|
| Name | Displays asset’s name. <ul style="list-style-type: none"> • This is a read-only field. However, if you would like to change the name of the asset, you can by using the Asset Browser’s Rename command. • It is recommended that you give assets names that reflect their content or use, so that they are easily recognizable and distinguish from similar assets. |
| Type | Displays the asset type. <ul style="list-style-type: none"> • This is a read-only field that cannot be changed. |
| Author | Displays the name of the creator/author of the asset. You can edit this field by clicking in the text box and typing the desired name. |

| | |
|-------------------------|---|
| Description | <p>Displays a textual description of the asset.</p> <ul style="list-style-type: none"> You can edit this field by clicking in the text box and typing the desired description. It is recommended that you provide a concise description for each asset that identifies its content or use, so that it is easily recognizable and distinguish from similar assets. |
| Created Time | <p>Displays the date and time of when the asset was originally created. This is a read-only field that cannot be changed.</p> |
| Modified Time | <p>Displays the time of the asset's most recent modification. This is a read-only field that cannot be changed.</p> |
| Expiry Date | <p>Displays the calendar date when the asset will be deleted from the XmediaServer.</p> <p>You can edit this field by clicking in the field's text box, which activates a button on the right side of the field. Click the button to display a calendar from which you can select the desired date, or select the None button at the bottom of the calendar to set no expiration date.</p> |
| Publish Lifespan | <p>Displays the specific number of days before the asset is deleted from the XmediaServer.</p> <p>You can edit this field by clicking in the text box and typing a number corresponding to the number of days the asset will be stored it is deleted from the XmediaServer.</p> |
| Categories | <p>Displays the category, or categories, where the asset is accessible from in the Asset Browser.</p> <p>This is a read-only field. However, if you would like to move, add, or remove the asset to/from a category, you can do so from within the Asset Browser.</p> |
| Keywords | <p>Displays the keyword(s) that have been assigned to the asset when it was created and saved. Keywords are used as a searchable property by Xstudio's Search XMS feature to help you locate assets within the system.</p> |

Not only do Image assets have their own additional property fields, but they are also the only asset type to have access to the **Advanced** button. When clicked, the Advanced button opens an Image Properties window for the asset that allows you to document further details about the contents, characteristics, and history of the image asset.

To view and/or modify a saved asset's properties:

- In the Gallery Browser, right-click on the desired asset.
A context menu appears.
- Select **Properties** from the context menu.
The **Properties** dialog box appears and displays a list of the asset's properties.
- Edit the appropriate property fields.
- When done, click the **Save** button to register modifications, or **Cancel** to close the dialog box without saving changes.

Xstudio properties



Each object that is added to the canvas (scene or control side) is associated with a set of properties, which determines that object's appearance and behavior. The **Properties Browser** is an Xstudio tool that lists the current property settings for the object that is selected on the canvas, or for the canvas itself. This chapter describes how editing the properties allows you to change the appearance and behavior of the objects and/or canvas.

Since the properties are specific to the type of object that is currently selected, the Properties Browser lists only those properties that are relevant to that object type. For this reason, this chapter describes the

Similarly, when the canvas is selected, the properties that are listed in the Properties Browser depend upon whether the control-side (template or panel) canvas or the scene-side canvas is selected.

Note: If an object has both a control side and scene side, you can only access the properties for one side at a time. For example, a bug object might have an image (i.e. logo) on the scene and buttons on the control-side to control the bug on air. When you select the image on the scene-side, the Properties Browser will only display the properties belonging to the image object. Similarly, selecting a button on the control-side only displays the properties of the button.

- [Control object properties](#), on page 286
- [Scene object properties](#), on page 293
- [Group object properties](#), on page 311
- [Formats for text, numbers, dates and times](#), on page 314


Control object properties




To help you customize the look and functionality of your objects, a series of properties are available to you when you are building, or working with, control panels or templates. When an object, or the canvas, is selected, the Properties Browser displays the current property settings for that item. Properties are organized thematically within the Properties Browser. Selecting the text box beside an individual property allows you to edit the property's value or setting. Most changes to property settings take immediate effect, but only they become permanent when the asset is saved.


Note: The actual title for the each property appears in square brackets [] beneath the name used in the Property Browser. You will use the bracketed name when creating a **Set Property** action (although the brackets are not required when creating the action).


The following table lists (alphabetically) and describes the properties that are available for the control-side canvas and primitives. See [Scene object properties](#), on page 293 for a similar table describing the scene-side properties.

| Control-side property | Description |
|--|--|
| Alignment [Align] | <ul style="list-style-type: none"> • Applies to: Image and button objects. • Description: Indicates how the asset should be placed on the object. • Accepted values: Select from the drop-down list: • Top Left - Places asset in the top left section of the object. • Top Center - Centers asset along top of the object. • Top Right - Places asset in the top right section of the object. • Mid Left - Places asset in the middle left section of the object. • Mid Center - Centers asset in the middle of the object. • Mid Right - Places asset in the middle right section of the object. • Bottom Left - Places asset in the lower left section of the object. • Bottom Center - Centers asset at the lower edge of the object. • Bottom Right - Places asset in the lower right section of object. |
| Allow Clicks [Clickable] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: Indicates whether the object responds to mouse clicks in Production Mode. • Accepted values: Select from the drop-down list: True or False. |
| Allow Edits [Enterable] (for Text) [Editable] (for Image) | <ul style="list-style-type: none"> • Applies to: Text and image objects. • Description: Determines whether or not text displayed in a text or image object can be edited in Production mode. • Accepted values: Select from the drop-down list: True or False. |
| Allow In Place Pan/Zoom [AllowInPlaceEdits] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: Indicates whether to allow interactive panning and zooming to display a specific part of an image. • Accepted values: Select from the drop-down list: True or False. |
| Area of Interest Height [AreaofInterestHeight] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: The default height of the area of interest in the image editor. See Cropping an image, on page 321 in the Image Editor chapter. • Accepted values: Enter the desired height, in pixels, of the area of interest. |
| Area of Interest Width [AreaofInterestWidth] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: The default width of the area of interest in the image editor. See Cropping an image, on page 321 in the Image Editor chapter. • Accepted values: Enter the desired width, in pixels, of the area of interest. |

| Control-side property | Description |
|---|--|
| Auto Create Job [AutoCreateJob] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: Determines if a job and corresponding placeholder are added to the currently open work order when a page is created. • Accepted values: Select from the drop-down list: True or False. |
| AutoStart [AutoStart] | <ul style="list-style-type: none"> • Applies to: Timer objects. • Description: Determines whether or not the timer starts automatically upon entering Production mode. An alternative methods for starting the timer object's count involve associating a Timer Command action to an event using the Action Editor (see Timer Command, on page 248). • Accepted values: Select from the drop-down list: True or False. |
| BackColor [BackColor] | <ul style="list-style-type: none"> • Applies to: Text, text box, drop box and button objects. • Description: Sets the background color for the object. • Accepted values: Click in the property's text box and then click the  button to display the Color dialog box. Select the desired RGB color definition, e.g. R: 0 G: 0 B: 0. |
| Basic Rendering [BasicRender] | <ul style="list-style-type: none"> • Applies to: Text and text box objects. • Description: Use an alternative simplified rendering. This property sets the text object's basic render attribute (uses GDI to render the text images). When set to False, advanced text properties appear for the object. • Accepted values: Select from the drop-down list: True or False. |
| Checked [Checked] | <ul style="list-style-type: none"> • Applies to: Check box objects. • Description: Indicates whether or not the box is checked. • Accepted values: Select from the drop-down list: True or False. |
| Data Auto Write [AutoWriteData] | <ul style="list-style-type: none"> • Applies to: Text, text box, drop box, radio button and check box objects. • Description: Determines whether the value supplied in the Data Target property will be triggered automatically or manually. • Accepted values: Select from the drop-down list: Auto or Manual. |
| Data Source [DataSource] | <ul style="list-style-type: none"> • Applies to: Text, text box, check box, drop box, radio button and image objects. • Description: Indicates the data source populating the object. • Accepted values: Drag and drop data items from the Data Manager. See Linking objects to a data source, on page 177. |
| Data Target [DataTarget] | <ul style="list-style-type: none"> • Applies to: Text, text box, drop box, radio button and check box objects. • Description: This property writes a value to the database. • Accepted values: If writing to a local variable (available to your panel/template only), prefix the variable with L: (e.g. L:SelectedStock). If writing to a global variable (available to all panels/templates), prefix the variable with G: (e.g. G:SelectedStock). See Prefixes (Scene;, Control;, Data;, L;, G:), on page 190. |
| Data Update [AutoReadData] | <ul style="list-style-type: none"> • Applies to: Text, text box, check box, drop box, radio button and image objects. • Description: Indicates whether or not data will automatically update when the source is checked by the Data Server. See Linking objects to a data source, on page 177. • Accepted values: Select from the drop-down list: Automatic or Manual |
| EndTime [EndTime] | <ul style="list-style-type: none"> • Applies to: Timer objects. • Description: Sets the precise time at which the timer object will stop counting. Be sure to set the timer's Infinite property to False, otherwise the timer will disregard the EndTime setting and continue counting. • Accepted values: Enter a value that follows the same time format as the Format property, e.g. 0:30. To have the timer count up, set the EndTime value to a higher value than the StartTime value. To have the timer count down, set the EndTime value to a lower value than the StartTime value. |

| Control-side property | Description |
|---------------------------------|--|
| FaceColor [FaceColor] | <ul style="list-style-type: none"> • Applies to: Text and text box objects. • Description: Sets the precise color of the text characters face, excluding the characters' edge. • Accepted values: Click in the property's text box and then click the  button to display the Color dialog box. Select an RGB color definition (e.g. R: 0 G: 0 B: 0) and then click OK. |
| Font [Font] | <ul style="list-style-type: none"> • Applies to: Text, text box, drop box, check box, image, radio button, ActiveX, timer and button objects. • Description: Sets the font for the text characters. • Accepted values: Select the property's text box and then click the  button to display the Font dialog box. Select a font name, style, and size (e.g. Arial - 18) and then click OK. |
| ForeColor [ForeColor] | <ul style="list-style-type: none"> • Applies to: Check box, radio button, timer and button objects. • Description: Sets the foreground color for the object. • Accepted values: Click in the property's text box and then click the  button to display the Color dialog box. Select an RGB color definition (e.g. R: 0 G: 0 B: 0) and then click OK. |
| Format [Format] | <ul style="list-style-type: none"> • Used in: Text, text box and timer objects. • Description: Modifies the format of the text displayed in the object. Further options for text formatting are provided in Formats for text, numbers, dates and times, on page 314. • Accepted values: Select from the drop-down list: • Upper Case - Displays text as upper case. • Lower Case - Displays text as lower case. • Title Case - Capitalizes the first letter of every word. • Currency - Displays number with currency symbol, thousand separator (if appropriate) and displays two digits to the right of the decimal separator. • Stock Change - Displays number as positive (+) unless specified negative (-), and displays three digits to the right of the decimal separator. • Trim - Deletes additional blank spaces surrounding the text. • Time - AM/PM - Displays the hour, minutes and AM/PM (hh:mm AM/PM). Defaults to AM if no time of day is entered. • Time - 12 hour - Displays the hour and minutes. • Time - 24 hour - Displays the hour and minutes according to the twenty-four hour clock (hh:mm). • Short Date - Uses the Short Date format specified by the user's system settings (mm/dd/yy or dd/mm/y). • Long Date - Uses the Long Date format specified by the user's system settings (e.g. mmmm dd, yyyy). • Celsius - Adds the Celsius symbol to the number entered. • Fahrenheit - Adds the Fahrenheit symbol to the number entered. |
| Height [Height] | <ul style="list-style-type: none"> • Applies to: Text, text box, button, image, ActiveX and timer objects. • Description: Sets the height of the object in pixels. • Accepted values: Enter the number of pixels directly in the property's text box, or resize the object by dragging the horizontal object zone lines (the top and bottom blue lines surrounding the object) and the object's height is automatically recorded in the property's text box. |

| Control-side property | Description |
|---|---|
| ImageLocation [ImageLocation] | <ul style="list-style-type: none"> • Applies to: Image and button objects. • Description: Indicates the image's location within Xstudio. • Accepted values: This property's setting is automatically populated when you drag and drop an image either directly onto the image/button object, or onto the ImageName row. Or, If you would prefer to enter the location manually, you must enter XMS: to indicate the Xmedia Server, followed by the name(s) of the category or categories to which it belongs, e.g. XMS:Images.News to indicate that the image is in the News category under Images. |
| ImageName [ImageName] | <ul style="list-style-type: none"> • Applies to: Image and button objects. • Description: The name of the image asset that is applied to the image or button object. • Accepted values: Click in the property's text box and then click the  button to display the select an image dialog box. Select the desired image asset and then click OK. Or, Drag and drop an image from the Asset Browser or Gallery Browser either directly onto the image/button object, or onto this property's text box. Or, If you prefer to enter the name manually, you can enter the name and then the location in the ImageLocation property, or you can enter both the name and location all in this row, e.g. XMS:Images.News:Earthquake. |
| Infinite [Infinite] | <ul style="list-style-type: none"> • Applies to: Timer objects. • Description: Sets whether the timer runs indefinitely or not. • Accepted values: Select from the drop-down list: True or False. When this property is set to True, the EndTime property setting is ignored and the timer's count continues. Therefore, if you want to specify an exact ending point for the timer, the Infinite property must be set to False. |
| Interval [Interval] | <ul style="list-style-type: none"> • Applies to: Timer objects. • Description: Interval at which the timer ticks. • Accepted values: Enter the Number of seconds. The smallest possible interval is 0.05 seconds. |
| Justification [Justification] | <ul style="list-style-type: none"> • Applies to: Text and text box objects. • Description: Indicates how text should be positioned within the object. • Accepted values: Select from the drop-down list: <ul style="list-style-type: none"> • Top Left - Left-justifies text along top of object. • Top Center - Centers text along top of object. • Top Right - Right-justifies text along top of object. • Mid Left - Left-justifies text in middle of object. • Mid Center - Centers text in middle of object. • Mid Right - Right-justifies text in middle of object. • Bottom Left - Left-justifies text along bottom of object. • Bottom Center - Centers text in middle of object. • Bottom Right - Right-justifies text along bottom of object. |
| Limit Width [LimitWidth] | <ul style="list-style-type: none"> • Applies to: Text and text box objects. • Description: Limits the amount of text entered in Production mode to the width of the text box. • Accepted values: Select from the drop-down list: True or False. |

| Control-side property | Description |
|---|---|
| Locked [Locked] | <ul style="list-style-type: none"> Applies to: Text, text box, drop box, check box, image, radio button, ActiveX, timer and button objects. Description: When the lock is enabled (True), then you can no longer select, move, or change the object's properties from the canvas. This is a safeguard to help prevent accidental edits. To unlock or make edits to a locked object, right-click on the object in the Control Tree and choose the Properties command. Make the appropriate edit and/or unlock the object by setting the Locked property to False. Accepted values: Select from the drop-down list: True or False. |
| Lookup Location [StringMapLocation] | <ul style="list-style-type: none"> Applies to: Text, text box, image and drop box objects. Description: Indicates the lookup table's location within Xstudio. A lookup table substitutes incoming data with new values. See Creating and applying lookup tables, on page 191 for more information. Accepted values: This property's setting is automatically populated when you drag and drop a lookup table from the Asset Browser or Gallery Browser onto the Lookup Table property's text box in the Properties Browser. Or, If you would prefer to enter the location manually, you must enter XMS: to indicate the Xmedia Server, followed by the category name, e.g. XMS:LookupTables. |
| Lookup Table [StringMapName] | <ul style="list-style-type: none"> Applies to: Text, text box, image and drop box objects. Description: Applies a lookup table to the incoming data. A lookup table substitutes incoming data with new values. See Creating and applying lookup tables, on page 191 for more information. Accepted values: Click in the property's text box and then click the  button to display the Select a lookup table dialog box. Select the desired lookup table asset and then click OK. Or, Drag and drop a lookup table from the Asset Browser or Gallery Browser either directly onto the object, or onto this property's text box. Or, If you prefer to enter the name manually, you must enter the name in the property's text box, and the location in the Lookup Location property's text box. |
| Max Length [MaxLength] | <ul style="list-style-type: none"> Applies to: Text and text box objects. Description: Limits the number of characters that can be entered while in Production mode. Accepted values: Enter the number of characters that will be the maximum number of characters allowed. |
| Name [Name] | <ul style="list-style-type: none"> Applies to: Text, text box, drop box, check box, image, radio button, ActiveX, timer and button objects. Description: The default name given to objects is usually the name of the object type followed by an incrementing number (i.e. Text1, Button3). To edit or rename the canvas or object's name, simply type the new name in this properties text box. Accepted values: To edit or rename the canvas or object's name, simply type the new name (text) in this properties text box. |
| ProgID [ClassName] | <ul style="list-style-type: none"> Applies to: ActiveX objects. Description: The registered classname of the active X object. You can use any active X component available on your PC. Accepted values: Enter the classname in text. For example, if adding the Microsoft Media Player, you would enter MediaPlayer.MediaPlayer.1. |

| Control-side property | Description |
|--|---|
| Restricted [Restricted] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: Restricts the images that you can use to replace the current image in Production mode to those in the selected category. • Accepted values: Enter a category's name. You must first enter XMS: (this stands for the Xmedia Server, where all assets reside), followed by the name of the image category. For example, XMS:Images.Sports indicates that the user is restricted to the Sports category under Images. |
| Scene Element [SceneElement] | <ul style="list-style-type: none"> • Applies to: Text, text box, drop box and image objects. • Description: Indicates the object in the scene that the currently selected object on the control side will link to and control. • Accepted values: Drag the scene element from the Scene tree and drop it onto the control object. |
| Scene Update [AutoUpdateScene] | <ul style="list-style-type: none"> • Applies to: Text, text box, drop box, and image objects. • Description: Determines whether the object is updated in the scene automatically (i.e. as soon as the value changes) or manually (i.e. when a manual action is performed). • Accepted values: Select from the drop-down list: Auto or Manual. |
| Selected [Selected] | <ul style="list-style-type: none"> • Applies to: Radio button objects. • Description: Indicates whether or not the radio button is selected. • Accepted values: Select from the drop-down list: True or False. |
| Selection [Selection] | <ul style="list-style-type: none"> • Applies to: Drop box objects. • Description: The ordering (index) of the selected item in the list. For example, is the item first, second, or third in a list of three items. • Accepted values: Enter the placement value of 1, up to the maximum number of items listed in the drop-down list. |
| StartTime [StartTime] | <ul style="list-style-type: none"> • Applies to: Timer objects. • Description: Sets the precise time (numeric value) at which the timer object will start counting. Be sure to set the timer's Interval and EndTime properties. • Accepted values: Enter a value that follows the same time format as the Format property, e.g. 0:30. To have the timer count up, set the StartTime value to a lower value than the EndTime value. To have the timer count down, set the StartTime value to a lower value than the EndTime value. |
| Style [Style] | <ul style="list-style-type: none"> • Applies to: Button and image objects. • Description: Determines how the asset will display once it has been laid onto the image, cel animation, clip, or button object. • Accepted values: Select from the drop-down list: <ul style="list-style-type: none"> • Keep Aspect - Asset is resized to fit the object while maintaining the aspect ratio. • Natural Size - Asset maintains its original dimensions and aspect ratio, although it may be clipped if its container object is resized. • Free - Asset is resized to fit the object without maintaining the aspect ratio. |
| Tab Index [TabIndex] | <ul style="list-style-type: none"> • Applies to: Text, text box, drop box, check box, radio button, active X and button objects. • Description: This property allows you to set the order for tabbing between text boxes while in Production mode. The tab index follows the order of creation. Changing the tab index becomes necessary when you move text boxes around and do not want the tab order to jump around (see Selecting, grouping, and/or layering objects on the canvas, on page 75). • Accepted values: Enter a value of 1 (first), up to the maximum number of objects on the canvas, to specify the tabbing order. |

| Control-side property | Description |
|--|--|
| Take Number [TakeNumber] | <ul style="list-style-type: none"> Applies to: Text, text box, drop box and image objects. This property only applies to entities in a multi-take template that has an associated scene element and the Scene Update property set to Manual. Description: The Take Number determines when the entity initializes its associated scene element. Accepted values: When the Take Number is 1, the associated scene element is updated during the cue of the template. When the Take Number is greater than 1, the associated scene element is initialized when the specified take is executed. |
| Text [Text] | <ul style="list-style-type: none"> Applies to: Text, text box, check box, radio button and button objects. Description: The text that displays in the object (e.g. on a button, in a text box). Accepted values: Enter the text exactly as you want it to appear in the object. |
| Text Direction [Direction] | <ul style="list-style-type: none"> Applies to: Text and text box objects. Description: The default direction of the text. This property is useful for accommodating international languages. Accepted values: Select from the drop-down list: Left to right, or right to left. |
| Transparent [Transparent] | <ul style="list-style-type: none"> Applies to: ActiveX objects. Description: Determines whether or not the control has a transparent background. Accepted values: Select from the drop-down list: True or False. |
| Trigger Events [AlwaysTrigger] | <ul style="list-style-type: none"> Applies to: Text, text box, and drop box objects. Description: Indicates whether to trigger events and scene updates only when the text changes. Accepted values: Select from the drop-down list: Always or On Change. |
| Visible [Visible] | <ul style="list-style-type: none"> Applies to: Text, text box, drop box, check box, image, radio button, ActiveX, timer and button objects. Description: Determines whether or not the object is visible on the control side's canvas. Accepted values: Select from the drop-down list: True or False. |
| Width [Width] | <ul style="list-style-type: none"> Applies to: Text, text box, drop box, image, ActiveX, timer and button objects. Description: Sets the width of the object in pixels. Accepted values: Enter the number of pixels directly in the property's text box, or resize the object by dragging the vertical object zone lines (the right and left blue lines surrounding the object) and the object's width is automatically recorded in the property's text box. |
| Word Wrap [MultiLine] | <ul style="list-style-type: none"> Applies to: Text and text box objects. Description: Determines whether text can wrap to the next line or whether it cuts off at the end of the box. Accepted values: Select from the drop-down list: True or False. |
| X [X] | <ul style="list-style-type: none"> Applies to: All objects. Description: Sets the position of the object's lower-left corner on the scene's X axis. See page 144 for more information. Accepted values: Enter a value (number of pixels from the point of origin) in the property's text box, or move the object on the scene to the desired location (the property text box is automatically updated). |
| Y [Y] | <ul style="list-style-type: none"> Applies to: All objects. Description: Sets the position of the object's lower-left corner on the scene's Y axis. See page 144 for more information. Accepted values: Enter a value (number of pixels from the point of origin) in the property's text box, or move the object on the scene to the desired location (the property text box is automatically updated). |


Scene object properties

To help you customize the look and functionality of your objects, a series of properties are available to you when you are building, or working with, the scene side of control panels, templates, or Xmedia objects. The following table lists (alphabetically) and describes the properties that are available for the scene-side canvas and primitives. See [Control object properties](#), on page 286 for a similar table describing the control-side properties.



Note: Each property's actual title appears in square brackets [] beneath the name used in the Property Browser. The bracketed name is what you need to use when creating a **Set Property** action in the Action Editor (although the brackets are not required when creating the action). See [Adding logic to control objects](#), on page 211 for more information.



| Scene-side properties | Descriptions |
|---------------------------------------|--|
| Alignment [Align] | <ul style="list-style-type: none"> Applies to: Clip, image, and cel animation objects. Description: Indicates how the asset should be placed on the object. Accepted values: Select from the drop-down list: Top Left - Places asset in the top left section of the object. Top Center - Centers asset along top of the object. Top Right - Places asset in the top right section of the object. Mid Left - Places asset in the middle left section of the object. Mid Center - Centers asset in the middle of the object. Mid Right - Places asset in the middle right section of the object. Bottom Left - Places asset in the lower left section of the object. Bottom Center - Centers asset at the lower edge of the object. Bottom Right - Places asset in the lower right section of object. |
| Allow Word Break [] | <ul style="list-style-type: none"> Applies to: Text2D objects Description: Determines whether line breaks can occur within a word or must always occur between words. Accepted values: Select from the drop-down list: True or False. |
| AntiAliasing [AntiAliasing] | <ul style="list-style-type: none"> Applies to: Clip, image, cel animation and video in objects. Description: Determines whether antialiasing (which smooths out the jagged edges of curves) is activated or not. Accepted values: Select from the drop-down list: True or False. |
| AudioLevel [AudioLevel] | <ul style="list-style-type: none"> Applies to: Clip and video in objects. Description: Sets the audio level for the object. Accepted values: The valid range of values is from 0-8. Values between 0-1 play the audio at a lower level than normal (duck) and values between 1-8 play the audio asset at a higher level than normal (gain). |
| AudioProfile [AudioProfile] | <ul style="list-style-type: none"> Applies to: Clip objects. Description: Sets the audio profile to use to mix the input audio tracks of the clip. Accepted values: An index (integer) that represents a set of pre-defined audio profiles. |
| AutoPlay [AutoPlay] | <ul style="list-style-type: none"> Applies to: Clip and cel animation objects. Description: Determines if the clip or cel animation begins playing as soon as the scene is taken. Accepted values: Select from the drop-down list: True or False. |

| Scene-side properties | Descriptions |
|---|---|
| Autostart [Autostart] | <ul style="list-style-type: none"> • Applies to: Crawl and roll objects. • Description: Sets whether the crawl or roll will automatically start upon scene loading. • Accepted values: Select from the drop-down list: True or false |
| Basic Rendering [BasicRender] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Use an alternative simplified rendering. This property sets the text object's basic render attribute (uses GDI to render the text images). • Accepted values: Select from the drop-down list: True or False. |
| Cellocation [Cellocation] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: Indicates the cel animation's location within Xstudio and the Xmedia Server. • Accepted values: This property is automatically populated when you drag and drop a cel animation from the Asset Browser or Gallery Browser either directly onto the cel animation object, or onto the CelName property's text box. If you would prefer to enter the location manually, you must enter XMS: to indicate the Xmedia Server, followed by the name(s) of the category or categories to which it belongs, e.g. XMS:CellAnimations.Sports to indicate that the cel animation is in the Sports category under Cel Animations. |
| CelName [CelName] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: The name of the cel animation. Click in the property's text box and then click the <input type="button" value="..."/> button to display the Select a cel animation window. • Accepted values: Click in the property's text box and then click the <input type="button" value="..."/> button to display the Select a cel animation dialog box. Select the cel animation asset and click OK. Or, This property's text box is automatically populated when you drag and drop a cel animation from the Asset Browser or Gallery Browser either directly onto the cel animation object, or onto this property's text box. Or, If you would prefer to enter the name manually, you must enter the name in this property's text box, and the location in the Cellocation property's text box. |
| ClipLocation [ClipLocation] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Indicates the clip's location within Xstudio. • Accepted values: This property's text box is automatically populated when you drag and drop a clip either directly onto the clip object, or onto the ClipName property's text box. If you would prefer to enter the location manually, you must enter XMS: to indicate the Xmedia Server, followed by the name(s) of the category or categories to which it belongs, e.g. XMS:Clips.Entertainment to indicate that the cel animation is in the Entertainment category under Clips. |
| ClipName [ClipName] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: The name of the clip asset that belongs to the clip object. • Accepted values: Click in the property's text box and then click the <input type="button" value="..."/> button to display the Select a video clip dialog box. Select the Clip asset and click OK. Or, This property's text box is automatically populated when you drag and drop a clip from the Asset Browser or Gallery Browser either directly onto the clip object, or onto this property's text box. Or, If you would prefer to enter the name manually, you can enter the name in this property's text box, and the location in the ClipLocation property's text box. |

| Scene-side properties | Descriptions |
|--|--|
| ClosedCaption [ClosedCaption] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Indicates whether or not the clip will show closed captioning. Only for MPEG2 clips. • Accepted values: Select from the drop-down list: True or false |
| CropBottom [CropBottom] | <ul style="list-style-type: none"> • Applies to: Video In objects. • Description: Sets the number of pixels to crop from the bottom edge of the Video In window. • Accepted values: Enter the Number of pixels to crop from the bottom edge of the Video In window. |
| CropLeft [CropLeft] | <ul style="list-style-type: none"> • Applies to: Video In objects. • Description: Sets the number of pixels to crop from the left edge of the Video In window. • Accepted values: Enter the Number of pixels to crop from the left edge of the Video In window. |
| CropRight [CropRight] | <ul style="list-style-type: none"> • Applies to: Video In objects. • Description: Sets the number of pixels to crop from the right edge of the Video In window. • Accepted values: Enter the Number of pixels to crop from the right edge of the Video In window. |
| CropTop [CropTop] | <ul style="list-style-type: none"> • Applies to: Video In objects. • Description: Sets the number of pixels to crop from the top edge of the Video In window. • Accepted values: Enter the Number of pixels to crop from the top edge of the Video In window. |
| Direction [Direction] | <ul style="list-style-type: none"> • Applies to: Crawl and roll objects. • Description: The direction in which text flows for crawls and rolls. • Accepted values: Select from the drop-down list: Left or Right for crawls; Up or Down for rolls. |
| Dynamic [Dynamic] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Indicates whether the object is dynamic (i.e. changing) or static. Fewer resources are used by the output device for non-dynamic objects. • Accepted values: Select from the drop-down list: True or False. |
| EdgeBlur [EdgeBlur] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Blurs the edges that surround each character in the Text2D object. Effects display only on the output device. This property does not affect the blur of the character's face (see FaceBlur, on page 297). • Accepted values: Enter the number of pixels that will define the thickness of the blurred portion of the edge. |
| EdgeColor [EdgeColor] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the color of the edge that surrounds each character in the Text2D object. This property does not affect the color of the character's face (see FaceColor, on page 297). Note that when the EdgeType property is set to Gradient, an additional EdgeColor property appears in the object's Properties Browser. • Accepted values: Click in the property's text box and then click the  button to display the Color dialog box. Select an RGB color definition, e.g. R: 0 G: 0 B: 0 |
| EdgeGradient [EdgeGradientStyle] | <ul style="list-style-type: none"> • Applies to: Text2D objects, when the EdgeMaterialType property is set to Gradient. • Description: Sets the type (direction) of gradient used to draw the edge that surrounds each character in the Text2D object. • Accepted values: Select from the drop-down list: Vertical or Horizontal. |


| Scene-side properties | Descriptions |
|---|--|
| EdgeMaterialType [EdgeMaterialType] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the type of material used to draw the edge that surrounds each character in the Text2D object. Note that when this property is set to Gradient, an additional EdgeColor property appears in the object's Properties Browser. • Accepted values: Select from the drop-down list: Color (solid) or Gradient (gradual transition in color). |
| EdgeOpacity [EdgeOpacity] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the opacity of the edge that surrounds each character in the Text2D object. This property does not affect the opacity of the character's face (see FaceOpacity, on page 297). • Accepted values: Enter a value greater than or equal to 0 (0% opacity or completely transparent) and less than or equal to 1 (100% opaque). |
| EdgeShape [EdgeShape] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the shape of the edge that surrounds each character in the Text2D object. Effects display only on the output device. • Accepted values: Select from the drop-down list: • None - No edge. • Round - Rounds off the text edge's corners. • Square - Squares off the text edge's corners. |
| EdgeSize [EdgeSize] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the size (thickness) of the edge that surrounds each character in the Text2D object. Effects display only on the output device. • Accepted values: Enter the number of pixels of the desired width of the edge. |
| EdgeType [EdgeType] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the style (type) of the edge that surrounds each character in the Text2D object as either filled or bordered. Effects display only on the output device. • Accepted values: Select from the drop-down list: • Filled - Fills the text's edge. • Border - Creates a border around the text's edge, with no fill. |
| EndFrame [EndFrame] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: Sets the frame at which the cel animation will stop playing. • Accepted values: Enter the value that corresponds to the last frame in the cel animation that you want to be played/displayed. Note: You can determine the total number of frames the cel animation consists of by right-clicking on the cel animation asset in the Asset Browser and selecting the Properties command. The total number of frames is displayed within the #Frames property. |
| EndEvent [EndEvent] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: Specifies the event that will be launched when the cel animation ends. See page 216 for more information about using events. • Accepted values: Type the event's name exactly as it appears in the Action Editor's Events list (see page 216). |

| Scene-side properties | Descriptions |
|---|---|
| EndLoop [EndLoop] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: Sets the last frame at which the cel animation will loop to. • Accepted values: Enter the value that corresponds to the last frame in the cel animation that you want the loop to play to before looping back to the starting point. Note: You can determine the total number of frames the cel animation consists of by right-clicking on the cel animation asset in the Asset Browser and selecting the Properties command. The total number of frames is displayed within the #Frames property. |
| EndTime (sec) [EndTime] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Sets the frame at which the clip will stop playing. • Accepted values: Enter the value that corresponds to the last frame in the clip that you want to be played/displayed. |
| ExtractVBI [ExtractVBI] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Indicates whether or not VBI will be extracted from the clip. • Accepted values: Select from the drop-down list: True or False. |
| FaceBlur [FaceBlur] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: The FaceBlur property is a factor that blurs the text face only, not the edges (see EdgeBlur, on page 295). • Accepted values: Enter the number of pixels. A value of 0 means no blur. Increasing the factor increases the amount of blurring. |
| FaceColor [FaceColor] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the foreground color for the characters of the Text2D object. This property does not affect the color of the character edges (see EdgeColor, on page 295). • Accepted values: Click in the property's text box and then click the  button to display the Color dialog box. Select an RGB color definition, e.g. R: 0 G: 0 B: 0. |
| FaceGradient [FaceGradientStyle] | <ul style="list-style-type: none"> • Applies to: Text2D objects, when the FaceGradient property is set to Gradient. • Description: Sets the type (direction) of gradient used to fill in the face of the characters in the Text2D object. • Accepted values: Select from the drop-down list: Vertical or Horizontal. |
| FaceMaterialType [FaceMaterialType] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the type of material used to fill in the face of the characters in the Text2D object. Note that when this property is set to Gradient, an additional EdgeColor property appears in the object's Properties Browser. • Accepted values: Select from the drop-down list: Color or Gradient. |
| FaceOpacity [FaceOpacity] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the opacity (transparent versus opaque) of the face of the text characters in the Text2D object. This property does not affect the opacity of the character edges (see EdgeOpacity, on page 296). • Accepted values: Enter a value greater than or equal to 0 (0% opaque or completely transparent) and less than or equal to 1 (100% opaque). |
| Font [Font] | <ul style="list-style-type: none"> • Applies to: The canvas and Text2D objects. • Description: Sets the font for the text within the object. • Accepted values: Click in the property's text box and then click the  button to display the Font dialog box. Then select the Font, Font Style, and Size, e.g. Arial - 18 - Bold. |

| Scene-side properties | Descriptions |
|---|--|
| Format [TimeFormat] | <ul style="list-style-type: none"> • Applies to: Text2D objects when the Time property is enabled. • Description: Sets the format in which the time will be displayed. See Creating simple clocks and timers using Text2D scene elements, on page 92 for more information. • Accepted values: Selecting the Format property exposes the  button to open the Time Format dialog box. You can select whether hours, minutes, and/or seconds should be displayed and/or if leading zeroes should be displayed. Hours are always displayed in 24 hour format. |
| FrameColor [FrameColor] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the color of the object's frame, which is a background that appears directly behind the characters of the Text2D object. Note that when the FrameMaterialType property is set to Gradient, an additional FrameColor property appears in the object's Properties Browser. • Accepted values: Click in the property's text box and then click the  button to display the Color dialog box. Select an RGB color definition, e.g. R: 0 G: 0 B: 0. |
| FrameGradient [FrameGradientStyle] | <ul style="list-style-type: none"> • Applies to: Text2D objects, when the FrameMaterialType property is set to Gradient. • Description: Sets the type (direction) of gradient used to draw the frame, which is a background that appears directly behind the characters of the Text2D object. • Accepted values: Select from the drop-down list: Vertical or Horizontal. |
| FrameMaterialType [FrameMaterialType] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the type of material used to draw the frame. Note that when this property is set to Gradient, an additional FrameColor property appears in the object's Properties Browser. • Accepted values: Select from the drop-down list: Color or Gradient. |
| FrameOpacity [FrameOpacity] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the opacity (transparent versus opaque) of the frame behind the text characters in the Text2D object. This property does not affect the opacity of the character face or edges (see FaceOpacity, on page 297 and EdgeOpacity, on page 296). • Accepted values: Enter 0 for opaque or completely transparent; a value of 1 for completely opaque; or a value greater than 0, but less than 1 for various grades of translucency. |
| FrameShape [FrameShape] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Determines the shape of the frame. Effects display only on the output device. • Accepted values: Select from the drop-down list: Round, Square or None. |
| HingePosition [TransitionHingePosition] | <ul style="list-style-type: none"> • Applies to: All primitives when TransitionType property is set to one of the following: <ul style="list-style-type: none"> • Hinge Swing In • Hinge Swing Out • Hinge Slide In • Hinge Slide Out • Description: Determines the position of the fixed hinge of the object, or stationary pivot point, that the transition rotates around. • Accepted values: Select from the drop-down list: Left, Right, Top, or Bottom. |

| Scene-side properties | Descriptions |
|---|--|
| HoldLastFrame [HoldLastFrame] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Indicates whether or not the clip will hold the last frame when the clip completes. • Accepted values: Select from the drop-down list: True or False. When True is selected, the last frame remains displayed. When False is selected, the clip completes and is removed from sight. |
| Justification [Justification] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Indicates how text will be positioned within the Text2D object. • Accepted values: Select from the drop-down list: • Top Left - Left-justifies text along top of object. • Top Center - Centers text along top of object. • Top Right - Right-justifies text along top of object. • Mid Left - Left-justifies text in middle of object. • Mid Center - Centers text in middle of object. • Mid Right - Right-justifies text in middle of object. • Bottom Left - Left-justifies text along bottom of object. • Bottom Center - Centers text in middle of object. • Bottom Right - Right-justifies text along bottom of object. |
| ImageLocation [ImageLocation] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: Indicates the image's location within Xstudio. • Accepted values: This property's text box is automatically populated when you drag and drop an image either directly onto the image object, or onto the ImageName property's text box. If you would prefer to enter the location manually, you must enter XMS: to indicate the Xmedia Server, followed by the name(s) of the category or categories to which it belongs, e.g. XMS:Images.News to indicate that the image is in the News category under Images. |
| ImageName [ImageName] | <ul style="list-style-type: none"> • Applies to: Image objects. • Description: The name of the image asset that belongs to the selected image object on the scene. • Accepted values: Click in the property's text box and then click the 3-dot button to display the Select an Image dialog box. Select the image asset and click OK. Or, The property's text box can be automatically populated when you drag and drop an image from the Asset Browser or Gallery Browser either directly onto the image object, or onto this property's text box. Or, If you would prefer to enter the name manually, you can enter the name in this property's text box, and the location in the ImageLocation property's text box. Otherwise, you can enter both the name and location all in this property's text box, e.g. XMS:Images.News:Earthquake. |
| InTransitionDirection [TransitionDirection] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the InTransitionType is set to one of the following: • Reveal • Slide • Valve • Roll In • Roll out • Description: Determines the direction of the transition. • Accepted values: Select from the drop-down list: Left, Right, Up, or Down. |


| Scene-side properties | Descriptions | | | | | | | | | | | | |
|---|--|------------------|-------|----------|---------|-------|---------|------------------|-------------------|------------------|-------------------|-----------|------------|
| InTransitionDuration [TransitionDuration] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the InTransitionType is set to anything, except Cut. • Description: Determines the length of time (in seconds) that the transition will take to complete. • Accepted values: Enter the number of seconds that the transition will take to complete. | | | | | | | | | | | | |
| InTransitionFeathering [TransitionFeathering] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the InTransitionType is set to one of the following: <ul style="list-style-type: none"> • Reveal • Slide • Box • Description: Determines the transparency gradient that is applied to the leading edge of the object as it transitions. • Accepted values: Assigning a positive value allows you to set this property as a percentage of the image. For example, 0 results in a zero percent or no gradient, while 1.0 is 100% transparent. Using a negative value allows you to specify the value in pixels, rather than a percentage of the object. | | | | | | | | | | | | |
| InTransitionHingePosition [TransitionHingePosition] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the InTransitionType is set to one of the following: <ul style="list-style-type: none"> • Hinge Swing In • Hinge Swing Out • Hinge Slide In • Hinge Slide Out • Description: Determines the position of the fixed hinge of the object. • Accepted values: Select from the drop-down list: Left, Right, Top, or Bottom. | | | | | | | | | | | | |
| InTransitionOffset [TransitionOffset] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite. • Description: Determines the length of time (in seconds) that the “in transition” will take to begin executing. • Accepted values: Enter the number of seconds that the in transition will take to begin executing. -1 means start immediately after the out transition completes. 0 value has them simultaneously enter/exit. | | | | | | | | | | | | |
| InTransitionType [TransitionType] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite. • Description: Creates a transition between two objects, where one object is replaced by another object of the same type. Depending on the transition type selected, the transition can be stylized (i.e. involve rotations, gradual revealing...etc.), or an immediate replacement of one by another (Cut). • Accepted values: Select one of the following TransitionType styles from the drop-down list. <table border="0" data-bbox="748 1570 1414 1738"> <tbody> <tr> <td>• FADE</td> <td>• CUT</td> <td>• REVEAL</td> </tr> <tr> <td>• SLIDE</td> <td>• BOX</td> <td>• VALVE</td> </tr> <tr> <td>• HINGE SWING IN</td> <td>• HINGE SWING OUT</td> <td>• HINGE SLIDE IN</td> </tr> <tr> <td>• HINGE SLIDE OUT</td> <td>• ROLL IN</td> <td>• ROLL OUT</td> </tr> </tbody> </table> | • FADE | • CUT | • REVEAL | • SLIDE | • BOX | • VALVE | • HINGE SWING IN | • HINGE SWING OUT | • HINGE SLIDE IN | • HINGE SLIDE OUT | • ROLL IN | • ROLL OUT |
| • FADE | • CUT | • REVEAL | | | | | | | | | | | |
| • SLIDE | • BOX | • VALVE | | | | | | | | | | | |
| • HINGE SWING IN | • HINGE SWING OUT | • HINGE SLIDE IN | | | | | | | | | | | |
| • HINGE SLIDE OUT | • ROLL IN | • ROLL OUT | | | | | | | | | | | |


| Scene-side properties | Descriptions |
|---|--|
| ItemStartEvent [ItemStartEvent] | <ul style="list-style-type: none"> • Applies to: Crawl and roll objects. • Description: The event that will be executed every time a new crawl or roll item is displayed. See Using the object's default events, on page 216. • Accepted values: Enter an event name exactly as it appears in the Action Editor's Events list. |
| Leading [Leading] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the spacing between lines. • Accepted values: Enter a pixel value. |
| Locked [Locked] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Determines if the object is locked for editing. When the lock is enabled (True), then you can no longer select, move, or change the object's properties from the canvas. This is a safeguard to help prevent accidental edits. To unlock or make edits to a locked object, right-click on the object in the Scene Tree and choose the Properties command. Make the appropriate edit and/or unlock the object by setting the Locked property to False. • Accepted values: Select from the drop-down list: True or False. |
| Loop [Loop] | <ul style="list-style-type: none"> • Applies to: Cel animation and clip objects. • Description: Determines whether or not the cel animation or clip starts playing again after completing. • Accepted values: Select from the drop-down list: True or False. |
| LoopTransition [LoopTransition] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Indicates whether or not the clip plays a transition when the clip loops. If you set the LoopTransition property to True, then you must also set the TransitionType property and its associated properties to define the transition style. • Accepted values: Select from the drop-down list: True or False |
| Marker [TimeMarkerType] | <ul style="list-style-type: none"> • Applies to: Text2D objects when the Time property is enabled. • Description: Determines whether or not an AM/PM marker is displayed and in what format. See Creating simple clocks and timers using Text2D scene elements, on page 92 for more information. • Accepted values: Select from the drop-down list: am / pm, AM / PM or None. |
| Name [Name] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: The name that is used to identify the canvas or the selected object on the scene and in the Scene Tree. The default name given to objects is usually the name of the object type followed by an incrementing number (i.e. Text2D1, CelAnimation3). To edit or rename the canvas or object's name, simply type the new name in this properties text box. • Accepted values: To edit or rename the canvas or object's name, simply type the new name (text) in this properties text box. |
| Offset [ClockOffset] | <ul style="list-style-type: none"> • Applies to: Text2D objects when the Time property is enabled. • Description: Determines if an offset is added to the clock's time. See Creating simple clocks and timers using Text2D scene elements, on page 92 for more information. • Accepted values: Selecting the Offset property displays the  button to open the Time Offset dialog box. Use the up/down arrows to set the offset value for hour, minute, and/or second relative to the current system time. Values can be positive or negative. |


| Scene-side properties | Descriptions |
|--|---|
| Opacity [Opacity] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Sets the degrees of opacity (transparent versus opaque) of the object. This property is sometimes referred to as the “master opacity”, since Text 2D objects also have opacity properties for the text’s face, edge, shadow, and frame. If the other opacity settings are already set, the master opacity setting is multiplied by these settings. For example, if the Opacity property is set to 0.5 and the FaceOpacity is set to 0.5, then the object’s FaceOpacity becomes 0.25, while the rest of the object has an opacity of 0.5. • Accepted values: Enter a value greater than or equal to 0 (0% opaque or completely transparent) and less than or equal to 1 (100% opaque). |
| OutTransitionDirection [TransitionDirection] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the OutTransitionType is set to one of the following: <ul style="list-style-type: none"> • Reveal • Slide • Valve • Roll In • Roll out • Description: Determines the direction of the transition. • Accepted values: Select from the drop-down list: Left, Right, Up, or Down. |
| OutTransitionDuration [TransitionDuration] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the OutTransitionType is set to anything, except Cut. • Description: Determines the length of time in seconds that the transition will take to complete. • Accepted values: Enter the number of seconds that the transition will take to complete. |
| OutTransitionFeathering [TransitionFeathering] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the OutTransitionType is set to one of the following: <ul style="list-style-type: none"> • Reveal • Slide • Box • Description: Determines the transparency gradient that is applied to the leading edge of the object as it transitions. • Accepted values: Assigning a positive value allows you to set this property as a percentage of the image. For example, 0 results in a zero percent or no gradient, while 1.0 is 100% transparent. Using a negative value allows you to specify the value in pixels, rather than a percentage of the object. |
| OutTransitionHingePosition [TransitionHingePosition] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Composite, and the OutTransitionType is set to one of the following: <ul style="list-style-type: none"> • Hinge Swing In • Hinge Swing Out • Hinge Slide In • Hinge Slide Out • Description: Determines the position of the fixed hinge of the object. • Accepted values: Select from the drop-down list: Left, Right, Top, or Bottom. |

| Scene-side properties | Descriptions |
|--|--|
| OutTransitionOffset [OutTransitionOffset] | <ul style="list-style-type: none"> Applies to: All primitives when the TransitionType property is set to Composite. Description: Determines the length of time in seconds that the out transition will take to begin executing. Accepted values: Enter the number of seconds that the out transition will take to begin executing. -1 means start immediately after the in transition completes |
| OutTransitionOrientation [TransitionDirection] | <ul style="list-style-type: none"> Applies to: All primitives when the TransitionType property is set to Composite, and the OutTransitionType is set to Box. Description: Determines the direction of the transition. Accepted values: Select from the drop-down list: Horizontal, Vertical, or Center. |
| OutTransitionType [TransitionType] | <ul style="list-style-type: none"> Applies to: All primitives when the TransitionType property is set to Composite. Description: Creates a transition between two objects, where one object is replaced by another object of the same type. Depending on the transition type selected, the transition can be stylized (i.e. involve rotations, gradual revealing...etc.), or an immediate replacement of one by another (Cut). Accepted values: Select one of the following TransitionType styles from the drop-down list. <div style="display: flex; flex-wrap: wrap; justify-content: space-around;"> <ul style="list-style-type: none"> • FADE • SLIDE • HINGE SWING IN • HINGE SLIDE OUT <ul style="list-style-type: none"> • CUT • BOX • HINGE SWING OUT • ROLL IN <ul style="list-style-type: none"> • REVEAL • VALVE • HINGE SLIDE IN • ROLL OUT </div> |
| PingPong [PingPong] | <ul style="list-style-type: none"> Applies to: Cel animations objects. Description: Indicates whether or not the cel animation will start playing again in reverse after completing. Accepted values: Select from the drop-down list: True or False. When True is selected, the cel animation plays forward and then returns in reverse. When False is selected, the cel animation plays forward, completes, then returns to the beginning immediately and plays forward. |
| Playback State On Invisible [PlaybackStateOnInvisible] | <ul style="list-style-type: none"> Applies to: Clip objects Description: Sets the state of playback of the clip when the clip is invisible. Accepted values: Select from the drop-down list: Play, Pause or Recue. |
| Rate [Rate] | <ul style="list-style-type: none"> Applies to: Cel animation, crawl and roll objects. Description: Sets the number of frames per second for the cel animation, crawl or roll. Accepted values: Enter the number of frames per second. |
| Repeat [Repeat] | <ul style="list-style-type: none"> Applies to: Crawl and roll objects. Description: Determines whether or not the crawl or roll starts playing again after completing. Accepted values: Select from the drop-down list: True or False. |
| Reverse [Reverse] | <ul style="list-style-type: none"> Applies to: Cel animation objects. Description: Indicates whether or not the cel will start playing in reverse. Accepted values: Select from the drop-down list: True or False |

| Scene-side properties | Descriptions |
|---|--|
| RotateX [RotateX] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Sets the object's rotation angle/direction around the horizontal axis. • Accepted values: The Rotate X property value is set in units of angular degrees, referenced from the point of origin (0.0) up to 360 degrees which provides a full rotation around the X axis. Entering a positive value rotates the object from front-to-back, while negative values rotate the object back-to-front. |
| RotateY [RotateY] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Sets the object's rotation angle/direction around the vertical axis. • Accepted values: The Rotate Y property value is set in units of angular degrees, referenced from the point of origin (0.0) up to 360 degrees which provides a full rotation around the Y axis. Entering a positive value rotates the object from right-to-left, while negative values rotate the object left-to-right. |
| RotateZ [RotateZ] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Sets the object's rotation angle/direction around the z axis. • Accepted values: The Rotate Z property value is set in units of angular degrees, referenced from the point of origin (0.0) up to 360 degrees which provides a full rotation around the Z axis. Entering a positive value rotates the object counter-clockwise, while negative values rotate the object clockwise. |
| ScaleX [ScaleX] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Without changing the asset's actual measurements, this property allows you to scale up or down the displayed width of the object. See Scale X and Scale Y, on page 144 for more information. • Accepted values: Select the property's text box and click the <input type="button" value="..."/> button to display the Width dialog box. See Scale X and Scale Y, on page 144 for specific instruction for using the Width dialog box for scaling the width of an object. |
| ScaleY [ScaleY] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Without changing the asset's actual measurements, this property allows you to scale up or down the displayed height of the object. See Scale X and Scale Y, on page 144 for more information. • Accepted values: Select the property's text box and click the <input type="button" value="..."/> button to display the Height dialog box. See Scale X and Scale Y, on page 144 for specific instruction for using the Height dialog box for scaling the width of an object. |
| SendEvents [SendEvents] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Determines if user-defined clip events (name_swap, name_stop, or name_fail) are sent to the template from the device. • Accepted values: Select from the drop-down list: True or false. |
| SendPlaylistEvents [SendPlaylistEvents] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Determines if clip events are sent to the playlist from the device. • Accepted values: Select from the drop-down list: True or false. |
| ShadowBlur [ShadowBlur] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Blurs the shadow that displays behind the characters of Text 2D objects. Effects display only on the output device. • Accepted values: Enter the number of pixels that will define the thickness of the blurred portion of the shadow. |

| Scene-side properties | Descriptions |
|---|--|
| ShadowColor [ShadowColor] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the color for the shadow that displays behind the characters of Text 2D objects. Effects display only on the output device. • Accepted values: Click in property's text box and then click the  button to display the Color dialog box. Select an RGB color definition, e.g. R: 0 G: 0 B: 0. |
| ShadowDirection [ShadowDirection] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the direction of the shadow that displays behind the characters of Text 2D objects. Effects display only on the output device. • Accepted values: Enter an angular degree value within the range of 0 to 359 degrees, where: <ul style="list-style-type: none"> • 0 - 90 projects the shadow to the upper-right side of the characters. • 91 - 180 projects the shadow to the lower-right side the characters. • 181 - 270 projects the shadow to the lower-left side of the characters. • 271 - 359 projects the shadow to the upper-left side of the characters. |
| ShadowDistance [ShadowDistance] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the distance of the shadow from the center of the characters of the Text 2D object. Effects display only on the output device. • Accepted values: Enter a value that represents the number of pixels from the center of the text character. Depending on the setting of the ShadowDirection property, a positive ShadowDistance value advances the shadow in the specified direction, while a negative value retracts the shadow. |
| ShadowGradient [ShadowGradientStyle] | <ul style="list-style-type: none"> • Applies to: Text2D objects, when the ShadowType property is set to Gradient. • Description: Sets the type (direction) of gradient used to draw the shadow that displays behind the characters of Text 2D objects. • Accepted values: Select from the drop-down list: Vertical or Horizontal. |
| ShadowMaterialType [ShadowMaterialType] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the type of material used to draw the shadow. • Accepted values: Select from the drop-down list: Color or Gradient. |
| ShadowOpacity [ShadowOpacity] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the opacity (transparent versus opaque) of the shadow that appears behind the text characters in the Text2D object. • Accepted values: Enter a value greater than or equal to 0 (0% opaque or completely transparent) and less than or equal to 1 (100% opaque). |

| Scene-side properties | Descriptions |
|---------------------------------------|---|
| ShadowType [ShadowType] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Sets the type of material used to draw the shadow that is displayed behind the characters of a Text2D object. Note that when this property is set to Gradient, an additional ShadowColor property appears in the object's Properties Browser. • Accepted values: Select from the drop-down list: • None - No shadow. • Drop - The shadow is separated from the text. • Deep - The shadow is attached to the text (giving the text more of a 3D look). |
| Shrink To Fit [ShrinkToFit] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Determines if the text is scaled down to fit within its bounding box when necessary. • Accepted values: Select from the drop-down list: True or false |
| Source [Source] | <ul style="list-style-type: none"> • Applies to: Video in objects. • Description: Source of the video. • Accepted values: The availability of the following options depends on whether the device has the necessary hardware. The source of a video-in object can be: • A or B: represents the A or B source channel of the device hardware. • Capture: the video capture device specified in the device settings. • an empty string: uses the default video source of the device. • The name of a clip object on the same scene. This is useful to show multiple instances of the same clip. |
| Spacing [Spacing] | <ul style="list-style-type: none"> • Applies to: Crawl and roll objects. • Description: The number of pixels separating the elements being displayed in a crawl or roll object. • Accepted values: Enter a value to represent the number of pixels that will be present between two elements in a crawl or roll object. |
| Start [CounterStart] | <ul style="list-style-type: none"> • Applies to: Text2D objects when the Time property is enabled. • Description: Sets the time from which the counter will start. See Creating simple clocks and timers using Text2D scene elements, on page 92 for more information. • Accepted values: Selecting the Start property exposes the  button to open the Counter Start dialog box. Use the up/down arrows to select the exact hours, minutes, and seconds values to be set as the displayed start time. Values can only be positive. |
| StartFrame [StartFrame] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: Sets the frame at which the cel animation starts playing. • Accepted values: Enter the frame number at which the cel animation will start playing. |
| StartLoop [StartLoop] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: Sets the frame at which the cel animation will start looping. • Accepted values: Enter the frame number at which the cel animation will start looping. |
| StartTime (sec) [StartTime] | <ul style="list-style-type: none"> • Applies to: Clip objects. • Description: Sets the time at which the clip starts playing. • Accepted values: Enter the time (in seconds) at which the clip will start playing. |

| Scene-side properties | Descriptions |
|---|--|
| Style [Style] | <ul style="list-style-type: none"> • Applies to: Clip, cel animation and image objects. • Description: Determines how the asset displays once it has been laid onto the image, cel animation, clip, or button object. • Accepted values: Select from the drop-down list: • Keep Aspect - Asset is resized to fit the object while maintaining the aspect ratio. • Natural Size - Asset maintains its original dimensions and aspect ratio, although it may be clipped if its container object is resized. • Free - Asset is resized to fit the object without maintaining the aspect ratio. |
| Target [CounterTarget] | <ul style="list-style-type: none"> • Applies to: Text2D objects when the Time property is enabled. • Description: The target time to which the timer will count down to. When the current time passes the target time, the count time will stop at zero when the target time has been reached. See Creating simple clocks and timers using Text2D scene elements, on page 92 for more information. • Accepted values: Selecting the Target property exposes the  button to open the Counter Target dialog box. Use the up/down arrows to select the exact hours, minutes, and seconds values to be set as the used as the target time. Values can only be positive. |
| Text [Text] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: The exact text that displays in the Text2D object. • Accepted values: Type text characters exactly as you want it to display on the object. |
| Text Direction [Direction] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: The default direction of the text. This property is useful for accommodating international languages. • Accepted values: Select from the drop-down list: Left to right, or right to left. |
| Tracking [Tracking] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Determines whether or not tracking is used to appropriately space characters. Tracking generally is used to increase letter spacing as type gets smaller and decrease it as type gets larger. • Accepted values: Enter the pixel value. A positive value spreads the font, whereas a negative value squeezes it in. |
| TransitionDirection [TransitionDirection] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to one of the following: <ul style="list-style-type: none"> • WIPE • SLIDE • ROLL IN: • CUBE ROLL OUT • PUSH • FLIP • ROLL OUT • CUBE SLIDE IN • REVEAL • VALVE • CUBE ROLL IN • CUBE SLIDE OUT • Description: Determines the direction of the transition. • Accepted values: Select from the drop-down list: Left, Right, Up, or Down. |
| TransitionDuration [TransitionDuration] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to anything, except Cut. • Description: Determines the length of time that the transition will take to complete. • Accepted values: Enter the number of seconds that the transition will take to complete. |

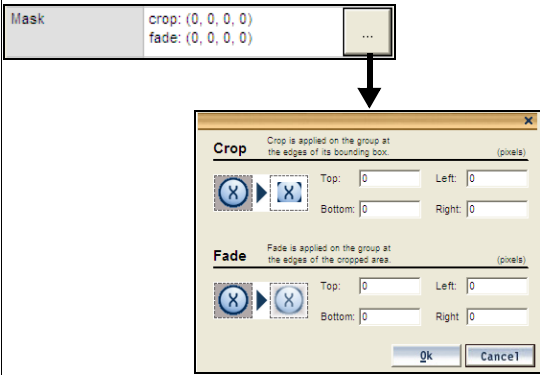
| Scene-side properties | Descriptions |
|---|--|
| TransitionFeathering [TransitionFeathering] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to one of the following: • Wipe • Push • Reveal • Slide • Box • Flip • Valve • Description: Determines the transparency gradient that is applied to the leading edge of the object as it transitions. • Accepted values: Assigning a positive value allows you to set this property as a percentage of the image. For example, 0 results in a zero percent or no gradient, while 1.0 is 100% transparent. Using a negative value allows you to specify the value in pixels, rather than a percentage of the object. |
| TransitionOrientation [TransitionDirection] | <ul style="list-style-type: none"> • Applies to: All primitives when the TransitionType property is set to Box. • Description: Determines the direction of the transition. • Accepted values: Select from the drop-down list: Horizontal, Vertical, or Center. |
| TransitionType [TransitionType] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Creates a transition between two objects by replacing the existing object with another object of the same type. For example, if you select Fade for a text object being updated from a data source, the text currently on air will fade out, and the latest text from the data source will fade in. • Accepted values: Select one of the following transition types from the drop-down list. Also see page 147 for a description of each transition type. • Fade • Dissolve • Cut • Wipe • Push • Reveal • Slide • Box • Flip • Valve • Hinge Swing In • Hinge Swing Out • Hinge slide In • Hinge slide Out • Roll in • Roll out • Cube Roll In • Cube Roll Out • Cube Slide In • Cube Slide Out • Composite |
| TriggerEvent [TriggerEvent] | <ul style="list-style-type: none"> • Applies to: Crawl and roll objects. • Description: The user defined event that will be executed when the crawl or roll passes the trigger point. • Accepted values: Enter an event name exactly as it is listed in the Action Editor's Events list (see page 213). |



| Scene-side properties | Descriptions |
|--|---|
| TriggerPoint [TriggerPoint] | <ul style="list-style-type: none"> • Applies to: Crawl and roll objects. • Description: Once the current crawl or roll object passes this pixel point, the TriggerEvent is activated. This event usually sets off an action to retrieve the next data item, thereby ensuring that only the most current data is used (ideal for rapidly-changing data such as stock quotes). • Accepted values: Enter the pixel value of the point on the screen where the crawl/roll will have to reach/pass over to trigger. |
| Trigger Transition [TriggerTransition] | <ul style="list-style-type: none"> • Applies to: All primitives • Description: Indicates whether to trigger a transition only if the scene element has changed. • Accepted values: Select from the drop-down list: Always or On Change. |
| Type [TimeType] | <ul style="list-style-type: none"> • Applies to: Text2D objects • Description: Sets the mode in which the time is displayed within a text2D object. See Creating simple clocks and timers using Text2D scene elements, on page 92 for more information. • Accepted values: Select from the drop-down list: None, Clock, Count Up Timer, Count Down Timer or Count Down Timer To Target Time |
| Use As Mask [UseAsMask] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Determines if the group is used as a mask to apply to the item behind it. See Using scene objects as a mask, on page 109 for more information. • Accepted values: Select from the drop-down list: None, Positive or Negative. |
| Use OXI Default Position [UseOXIDefaultPosition] | <ul style="list-style-type: none"> • Applies to: Cel animation objects. • Description: Sets the cel animation object's X, Y, Width and Height properties to the OXI default values. If enabled, you will not be able to override the OXI defaults by setting the individual properties. • Accepted values: Select from the drop-down list: True or False. |
| Visible [Visible] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Determines whether or not the object is visible on the scene. • Accepted values: Select from the drop-down list: True or False. |
| WordWrap [WordWrap] | <ul style="list-style-type: none"> • Applies to: Text2D objects. • Description: Determines whether text can wrap to the next line, or whether it cuts off at the end of the bounding box. • Accepted values: Select from the drop-down list: True or false. |
| X [X] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Sets the position of the object's lower-left corner on the scene's X axis. See page 144 for more information. • Accepted values: Enter a value (number of pixels from the point of origin) in the property's text box, or move the object on the scene to the desired location (the property text box is automatically updated). |

| Scene-side properties | Descriptions |
|-----------------------|---|
| Y [Y] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Sets the position of the object's lower-left corner on the scene's Y axis. See page 144 for more information. • Accepted values: Enter a value (number of pixels from the point of origin) in the property's text box, or move the object on the scene to the desired location (the property text box is automatically updated). |
| Z [Z] | <ul style="list-style-type: none"> • Applies to: All primitives. • Description: Sets the position of the object's lower-left corner on the scene's Z axis. See page 144 for more information. • Accepted values: Enter a value (number of pixels from the point of origin) in the property's text box, or move the object on the scene to the desired location (the property text box is automatically updated). |

Group object properties

When objects (scene or control) are grouped (see [page 77](#) for instructions), the group object has a set of properties that can be edited to determine the group's positioning or enable the group as a mask. The following table identifies the properties that are available to grouped objects:

| Group Properties | Description |
|--------------------------------|--|
| Height [Height] | <ul style="list-style-type: none"> • Applies to: Control-side • Description: Sets the height of the object. • Accepted values: Enter a value in pixels. |
| Locked [Locked] | <ul style="list-style-type: none"> • Applies to: Control-side or scene-side • Description: Determines if the object is locked for editing. • Accepted values: Select from the drop-down list: True or False |
| Mask [Mask] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Displays the defined values for crop and fade for the mask. A mask is defined by the following properties: <ul style="list-style-type: none"> • Crop (left, top, right, bottom): offset from the edge of the group's bounding box where opacity starts. • Fade (left, top, right, bottom): width of the opacity gradient bevel, starting from the mask's outer edge. <div style="text-align: center;">  <p>The image shows a 'Mask' property editor with 'crop: (0, 0, 0, 0)' and 'fade: (0, 0, 0, 0)'. A button with three dots opens a dialog box titled 'Crop' and 'Fade'. The dialog has sections for 'Crop' and 'Fade', each with 'Top', 'Left', 'Bottom', and 'Right' input fields, all currently set to '0'. There are also 'Ok' and 'Cancel' buttons at the bottom of the dialog.</p> </div> <ul style="list-style-type: none"> • Accepted values: Click the button to open the window that allows you to set the mask's crop and fade values. |
| Mask Type [MaskType] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Determines the type of mask to apply. See Mask above for more information • Accepted values: Select from the drop-down list: None or Positive |
| Name [Name] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: The name that is used to identify the group object on the scene and in the Scene Tree. The default name given to group objects is usually Group followed by an incrementing number (i.e. Group1, Group2). To edit or rename the group object's name, simply type the new name in this properties text box. • Accepted values: To edit or rename the group object's name, simply type the new name (text) in this property's text box. |
| Opacity [Opacity] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Sets the degree of opacity for the object. Note that the effects of setting the opacity property are only displayed on the output device. • Accepted values: Specify a value between 0.0 (transparent) and 1.0 (opaque). |

| Group Properties | Description |
|--|--|
| RotateX [RotateX] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Sets the group object's rotation angle/direction around the horizontal axis. • Accepted values: The Rotate X property value is set in units of angular degrees, referenced from the point of origin (0.0) up to 360 degrees which provides a full rotation around the X axis. Entering a positive value rotates the object from front-to-back, while negative values rotate the object back-to-front. |
| RotateY [RotateY] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Sets the group object's rotation angle/direction around the vertical axis. • Accepted values: The Rotate Y property value is set in units of angular degrees, referenced from the point of origin (0.0) up to 360 degrees which provides a full rotation around the Y axis. Entering a positive value rotates the object from right-to-left, while negative values rotate the object left-to-right. |
| RotateZ [RotateZ] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Sets the group object's rotation angle/direction around the z axis. • Accepted values: The Rotate Z property value is set in units of angular degrees, referenced from the point of origin (0.0) up to 360 degrees which provides a full rotation around the Z axis. Entering a positive value rotates the object counter-clockwise, while negative values rotate the object clockwise. |
| ScaleX [ScaleX] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Without changing the assets' actual measurements, this property allows you to scale up or down the displayed width of the group object. See Scale X and Scale Y, on page 144 for more information. • Accepted values: Select the property's text box and click the  button to display the Width dialog box. See Scale X and Scale Y, on page 144 for specific instruction for using the Width dialog box for scaling the width of an object. |
| ScaleY [ScaleY] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Without changing the assets' actual measurements, this property allows you to scale up or down the displayed height of the group object. See Scale X and Scale Y, on page 144 for more information. • Accepted values: Select the property's text box and click the  button to display the Height dialog box. See Scale X and Scale Y, on page 144 for specific instruction for using the Height dialog box for scaling the width of an object. |
| Tab Index [TabIndex] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Allows you to set the order for tabbing between the objects while in Production mode. • Accepted values: Enter a numeric index number. |
| Use As Mask [UseAsMask] | <ul style="list-style-type: none"> • Applies to: Scene-side • Description: Determines if the group is used as a mask to apply to the item behind it. See Using scene objects as a mask, on page 109 for more information. • Accepted values: Select from the drop-down list: None, Positive or Negative. |
| Visible [Visible] | <ul style="list-style-type: none"> • Applies to: Control-side or scene-side • Description: Determines whether or not the group object is visible. • Accepted values: Select from the drop-down list: True or False |
| Width [Width] | <ul style="list-style-type: none"> • Applies to: Control-side or scene-side • Description: Sets the width of the group object. • Accepted values: Enter a value in pixels. |

| Group Properties | Description |
|------------------|---|
| X [X] | <ul style="list-style-type: none">• Applies to: Control-side or scene-side• Description: Sets the position of the object's lower-left corner on the scene's X axis.• Accepted values: Enter a value (number of pixels from the point of origin), or move the object on the scene to the desired location (the property text box is automatically updated). |
| Y [Y] | <ul style="list-style-type: none">• Applies to: Control-side or scene-side.• Description: Sets the position of the object's lower-left corner on the scene's Y axis.• Accepted values: Enter a value (number of pixels from the point of origin), or move the object on the scene to the desired location (the property text box is automatically updated). |
| Z [Z] | <ul style="list-style-type: none">• Applies to: Scene-side.• Description: Sets the position of the group object's lower-left corner on the scene's Z axis.• Accepted values: Enter a value (number of pixels from the point of origin), or move the object on the scene to the desired location (the property text box is automatically updated). |

Formats for text, numbers, dates and times

Xstudio allows you to format text and numbers in various ways using the **Format** property (page 288). The table below identifies and describes the formats that are available from the Format property's drop-down list. Once you set the object's format property, you can immediately view the results of your formatting by switching to **Production** mode.

Note:

- **For number formats:** If anything other than a numeric value is entered, the format will be ignored and whatever the user entered will display as-is.
- **For date and/or time formats:** If text is entered that is not a date and/or time, the format will be ignored and the text will display as is.

While the default formats are adequate for most circumstances, you are not limited to these default formats. Many others can be entered directly into the Format row. The following sections identify and describe alternative symbols and syntax that can be used for formatting numbers, dates and time values:

- [Formatting numbers](#), on page 315
- [Formatting dates and times](#), on page 316

| Format | Result | Example |
|-----------------------|--|--|
| Upper Case | Displays text as upper case. | Stayed tuned becomes STAY TUNED . |
| Lower Case | Displays text as lower case. | Stayed tuned becomes stay tuned . |
| Title Case | Capitalizes the first letter of each word. | stay tuned becomes Stay Tuned . |
| Currency | Displays number with currency symbol, thousand separator (if appropriate) and displays two digits to the right of the decimal separator. Further digits to the right round the number up or down. Output is based on user's system settings. | 52384.99 becomes \$52,382.99 . 52384.998 becomes \$52, 383.00 . |
| Stock Change | Displays numbers as positive (+) unless specified negative (-), and displays two digits to the right of the decimal separator. Further digits to the right round the number up or down. | 3.912778 becomes +3.91 . |
| Trim | Deletes additional blank spaces surrounding the text. | " On Cue " becomes "On Cue" . |
| Time - AM/PM | Displays the hour, minutes, and AM/PM (hh:mm:AM/PM). Defaults to AM if no time of day is entered. | 4:00 becomes 04:00 AM . 4.00 pm becomes 04:00 PM . |
| Time - 12 hour | Displays the hour and minutes. | 4:00 becomes 04:00 . |
| Time - 24 hour | Displays the hour and minutes according to the twenty-four hour clock (hh:mm). | 4:00 pm becomes 16:00 . |

| Format | Result | Example |
|-------------------|--|--|
| Short Date | Uses the Short Date format specified by the user's system settings (mm/dd/yy or dd/mm/yy). | May 9 2002 becomes (depending on your system's settings) 5/09/11 . |
| Long Date | Uses the Long Date format specified by the user's system settings (e.g. mmmm dd, yyyy). | May 9 2002 becomes (depending on your system's settings) May 9, 2011 . |
| Celsius | Adds the Celsius symbol to the number entered. | 28 becomes 28°C . |
| Fahrenheit | Adds the Fahrenheit symbol to the number entered. | 82 becomes 82°F . |

Formatting numbers

The following table contains symbols that you can use to format numbers.

| Symbol | Result | Example |
|----------------------------------|---|---|
| 0 | Digit placeholder that prints a trailing or leading zero, if appropriate. | If you entered 000.00 in the Format row, 14 would become 014.00 . |
| # | Digit placeholder; never prints trailing or leading zero. | If you entered ###.## in the Format row, 14 would become 14. in the output (the decimal after 14 will be displayed). If you entered ###.00 , 14 would become 14.00 . |
| . | Decimal placeholder. | If you entered #.00 in the Format row, 14 would become 14.00 . |
| , | Thousands separator. | If you entered #,### in the Format row, 35987 would become 35,987 . |
| - + \$, ¥ () space (etc.) | Literal characters. These are displayed exactly as entered in the Format row. | If you entered \$###.00 in the Format row, 14 would become \$14.00 . |

Formatting dates and times

Dates and times can also be formatted using a format syntax.

| Format syntax | Result | Example |
|----------------------------|--|---|
| m/d/yy | Displays the date in short date format. | January 9 2002 becomes 1/9/02 . |
| dddd, mmmm dd, yyyy | Adds the day of the week to a long date format. | January 9 2002 becomes Wednesday, January 9, 2002 . |
| d-mmm | Abbreviates the date to day and abbreviated month. | January 9 2002 becomes 9-Jan . |
| mmmm-yy | Displays the full month and year. | January 9 2002 becomes January-02 . |
| h:mm:ss a/p | Displays the hour, minutes, seconds and a/p. Defaults to a if no time of day is entered. | 4:00 becomes 4:00:00 a . 4:00 pm becomes 4:00:00 p . |
| d-mmmm h:mm | Displays the day, month, hour and minutes. | January 9 2002 4:00 becomes 9-January 4:00 . |

D Image Editor

The Image Editor allows you to perform some basic image editing tasks on image assets like, adjusting the image's color and sharpness, cropping the image, among others.

The following sections provide functional descriptions and instructions for how to use the Image Editor to edit image asset:

- [Opening the Image Editor](#), on page 318
- [Zooming in/out and repositioning the image](#), on page 319
- [Cropping an image](#), on page 321
- [Making color adjustments](#), on page 323
- [Adjusting the sharpness of an image](#), on page 324
- [Using the Image Modification buttons to flip an image](#), on page 324

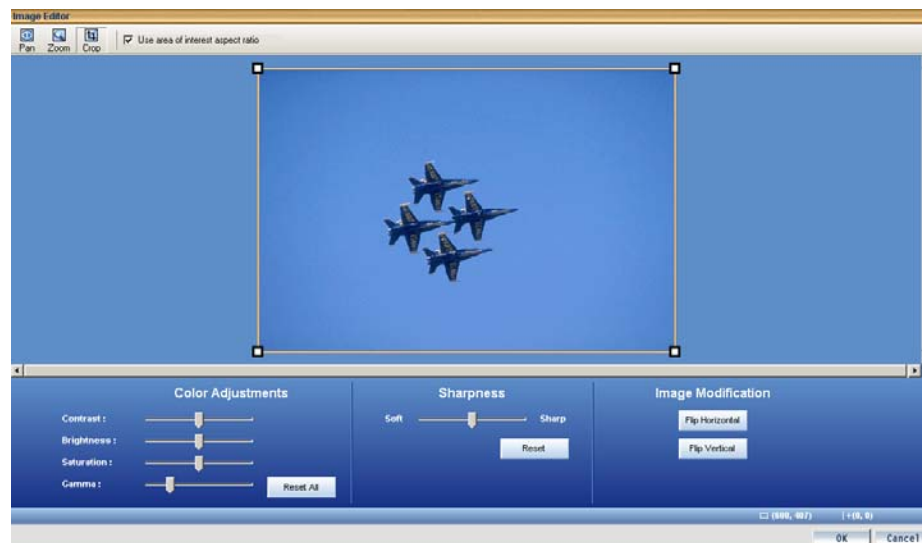


Fig. D-1: Xstudio's Image Editor allows you to perform some basic image editing tasks on image assets

Opening the Image Editor

There are two (2) ways to open the Image Editor:

- 1 Add an image asset to the scene's canvas by dragging it onto the canvas from the Asset Browser or Gallery Browser.
- 2 Right-click on the image and select the **Modify Image** command from the context menu.

The **Image Editor** window appears.

Or,

- 1 Add an image asset to the scene's canvas by dragging it onto the canvas from the Asset Browser or Gallery Browser.
- 2 Double-click the image on the canvas.
The **Select an Image** dialog box appears.
- 3 Navigate to and select the image that is to be edited.
- 4 Once the image is selected, press the **Edit** button.

The **Image Editor** window appears.

Zooming in/out and repositioning the image

When editing an image, it is often necessary to view the image, or a portion of the image, in greater detail. It is for this reason that the Image Editor allows you to magnify, or zoom in on the image, as well as zoom out to get an expanded or distanced view of the image.

When an image is opened in the Image Editor it is placed directly in the center of Editor's canvas. However, when you zoom in on the image, you may not see certain areas of the image. To reposition the image within the display, while maintaining the zoom scale, you can pan across the image to the area that is otherwise outside of the window's display.

Instructions for zooming in/out and panning across the image are provided below.

- [Zooming in/out](#), on page 319
- [Reposition the image by panning](#), on page 320

Zooming in/out

There are two (2) methods that you can use to zoom in on an image:

- 1 The Image Editor's **Zoom** tool allows you to zoom in on a particular area on the image.

- Click the **Zoom** button  on the Image Editor's toolbar.

Your pointer transforms into a magnifying glass.

- Click the part of the image that you want to zoom in on.

The Image Editor zooms in on the image in increments.

- 2 You can zoom in on the image using your mouse or keyboard. Note, however, that this type of zoom operation zooms in on the center of the portion of the image that is currently displayed. As a result, you may need to pan across the image to view a particular area of interest.

- Roll the mouse wheel forward.

Or,

- Press the + (plus) key on your keyboard's number pad.

To zoom out of the image:

You can zoom out of the image using your mouse or keyboard. Note, however, that this type of zoom operation zooms out on the center of the portion of the image that is currently displayed.

- Roll the mouse wheel backward.

Or,

- Press the - (minus) key on your keyboard's number pad.

Or,

- Press the **Space bar** on your keyboard and the zoom is reset to the image's original scale (1:1).

Reposition the image by panning

There are three (3) methods that you can use to reposition an image by panning:

Panning is only required, and therefore only available, when a portion of the zoomed image is not visible in the Image Editor's canvas. In such a case, the canvas' horizontal and/or vertical scroll bars become active.

- 1 The Image Editor's **Pan** tool allows you to reposition an image using your mouse to drag the image around the Editor's canvas.

- Click the **Pan** button  on the Image Editor's toolbar.

Your pointer transforms into a hand.

- Drag the image in the direction (up/down, and/or right/left) required to reposition and display the desired portion of the image in the Editor's canvas.

- 2 You can reposition an image using the Editor's canvas scroll bars. When an upper or lower portion of the image is not visible in the Editor's canvas, the canvas' vertical scroll bar becomes active. Similarly, when the right or left portion of the image is not visible in the Editor's canvas, the canvas' horizontal scroll bar becomes active.

- Drag the scroll box up/down and/or back/forth to reposition the image on the Editor's canvas.

- 3 You can also use the keyboard's direction keys (up, down, left, right, page up, and page down) to reposition the image in small increments.

Cropping an image

The Image Editor's cropping function allows you to conserve a selected portion of the image, while removing the remaining portions surrounding the selected area. Cropping is a popular image processing technique that is commonly used to focus the image on a particular subject or area on the image.

Note: Since cropping permanently alters the image, the Image Editor allows you to save a copy of the changed image before the cropping is actually performed. As a result you can preserve the original state of the image asset.


A bounding outline surrounds the image when it is opened in the Image Editor. The bounding outline is a frame that is used for defining the area of interest when using the cropping feature. In other words, the content inside of the bounding outline will be preserved, while whatever is outside the bounding outline will be permanently removed. When the Image Editor is in crop mode, dragging the bounding outline's corner handles (squares) allow you use to resize the bounding outline around the area of interest on the image. If the **"Use area of interest aspect ratio"** check box is selected, then the bounding outline can only be resized to dimensions that maintain the aspect ratio of the image's placeholder on the scene, template, or control panel. Clearing this check box allows you to resize the bounding outline freely without any aspect ratio restrictions.

When resizing or repositioning the bounding outline, its dimensions and location is display in the lower right corner of the Image Editor window. The first set of number displays the X/Y dimensions of the bounding outline in pixels, while the second set represents the location coordinates of the bounding outline's upper left corner.



Fig. D-2: The dimensions and location of the bounding outline

To crop an image:

- 1 Open the original image in the Image Editor (see [Opening the Image Editor](#), on page 318).
- 2 With the image completely visible in the Image Editor, press the **Crop** button .
- 3 Resize and/or reposition the bounding area to include only the content to be preserved.
 - a Select the **Use area of interest aspect ratio** check box if you want to resize the bounding outline, but still maintain the aspect ratio of the image's placeholder on the scene, template, or control panel. Clear the check box is you want to freely resize the bounding area.
 - b Drag the bounding outline's sizing handles to resize the bounding outline around the area of interest.
 - c If necessary, you can reposition the bounding outline within the image by clicking inside the bounding outline and dragging it over the area of interest in the image.
- 4 Once the bounding outline contains only the area of interest, press **OK**.
The **Name new image** dialog box appears.
- 5 Save the cropped image:

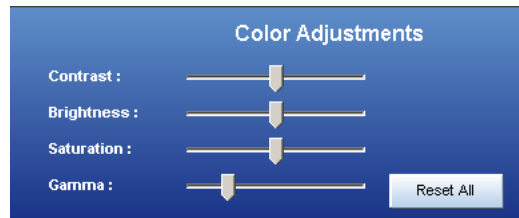
- Select the **Stretch image to fill placeholder** check box if you want the Image Editor to enforce scaling so that the resulting image is the exact size of the placeholder.
- Leave the **Stretch image to fill placeholder** check box clear (unchecked) you want to defer any scaling between the cropped image and its final space until playout time.
- If you want to preserve the original image asset, you can save the cropped image as a new image by typing in the name of the new image asset in the text box and then pressing **OK** on the **Name new image** dialog box.
- To replace the original image with the cropped image, leave the original image's name in the text box and press **OK**.

The Image Editor closes automatically and the cropped image appears on Xstudio's canvas.

Making color adjustments

The Image Editor allows you to adjust the color properties of an image asset. Color adjustments are applied immediately to the entire image, not just a specific area of interest. You can experiment with different color settings, however there is no way to undo your changes except to press the **Reset All** button, which returns the color settings to the state they were in when the image was last saved, or by pressing the **Cancel** button, which closes the Image Editor and the image asset remains unchanged.

The following table demonstrate that you adjust the image's color properties by moving the property's slider to the right or left.



| | |
|-------------------|---|
| Contrast | <p>The difference in the luminance of the brightest and the darkest areas of an image. A high contrast will display an obvious difference (i.e. black versus white) in the luminance between two areas, while a low contrast makes the difference less obvious or more gradual (i.e. shades of grey).</p> <ul style="list-style-type: none"> • Move the slider to the right to increase the image's contrast ratio. • Move the slider to the left to decrease the image's contrast ratio. |
| Brightness | <p>Generally, the brightness setting determines how light or dark a the colors of the image are based on how close the colors are to white. The more white a color has, the brighter it is.</p> <ul style="list-style-type: none"> • Move the slider to the right to increase the image's brightness (until all white) • Move the slider to the left to decrease the image's brightness (until all black) |
| Saturation | <p>Saturation refers to the intensity of a specific hue. It is based on the color's purity; a highly saturated hue has a vivid, intense color, while a less saturated hue appears more muted and grey. With no saturation at all, the hue becomes a shade of grey.</p> <ul style="list-style-type: none"> • Move the slider to the right to increase the image's saturation. • Move the slider to the left to decrease the image's saturation. |
| Gamma | <p>Alters the contrast of the mid-tone colors with little effect on pure white or black, which is the function of the Contrast property above. Increasing the gamma ratio increases the image's highlight to shadow contrast in terms of mid-tone colors' density values. Decreasing the gamma setting intensifies the shadows amongst the mid-tone colors.</p> <ul style="list-style-type: none"> • Move the slider to the right to increase the image's gamma setting. • Move the slider to the left to decrease the image's gamma setting. |

Adjusting the sharpness of an image

The Image Editor allows you to adjust the sharpness setting of an image asset. The sharpness adjustment is applied immediately to the entire image, not just a specific area of interest. You can experiment with different settings, however there is no way to undo your changes except to press the **Reset** button, which returns the sharpness setting to the state it was in when the image was last saved, or by pressing the **Cancel** button, which closes the Image Editor and the image asset remains unchanged.

You adjust the image's sharpness property by moving the property's slider to the right to increase the focus and to the left to blur the image.

Using the Image Modification buttons to flip an image

The Image Editor allows you to quickly change the orientation of an image by flipping it horizontally or vertically. Figure demonstrates that the **Image Modification** section of the Image Editor's interface features two buttons: **Flip Horizontal** and **Flip Vertical**. Press these buttons to immediately flip the image, and press the button again to return to the original orientation.

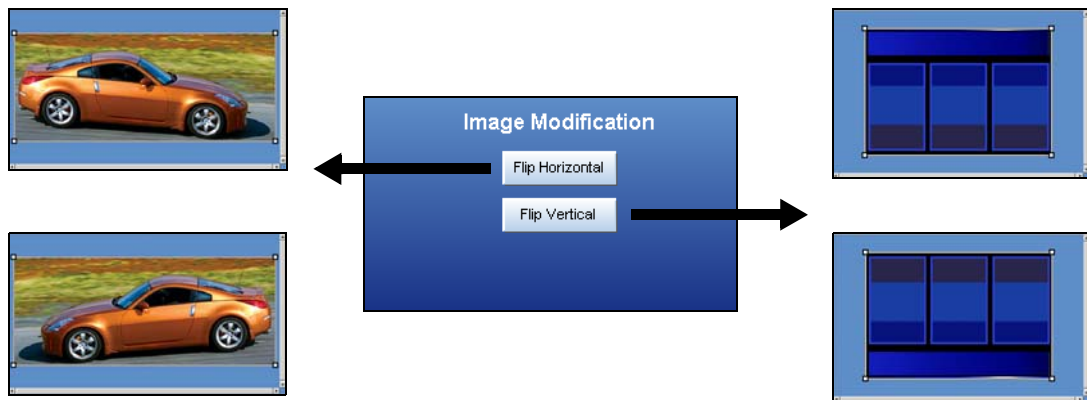


Fig. D-3: The Image Editor's Image Modification buttons allow you to quickly flip image assets

E Script Editor

Xstudio streamlines the creation and control of data-driven graphics by automating tasks as much as possible. This means that the average user can make a lot of things happen in Xstudio without ever writing a line of code.

For advanced users with specific requirements not addressed by the application, however, the Script Editor has been included.

Use the Script Editor to create custom actions that can be called in response to any event. Commands can be written in any of the following common scripting languages:

- VB Script
- JavaScript
- Perl

Once scripts have been written, they can be linked to control panels, pages, playlists, etc., using the Action Editor.

Our technical support department can also provide you with information regarding the "Producer API" for using scripts to trigger events, and access and manipulate other Producer objects.

Accessing the Script Editor

To access the Script Editor:

- From the **File** menu select **New**, and when the **Create New** dialog box displays, select **Script**.
- Or,
- From within the **Action Editor** add a **Custom** action, then right-click in the **Custom Actions** list at right to display a pop-up menu. If you have right-clicked on an existing script you can either select **Edit** or **New**. Otherwise, **New** is your only choice.








Your selection opens the Script Editor.









Script Editor screen items

The Script Editor window consists of:

- A toolbar along the top of the window. It contains the buttons used for performing basic functions such as opening new or existing scripts, editing, saving, adding comments, etc.
- Two window panes below. The one at right lists all currently open scripts, allowing you to click on one to make it display in the left window pane. Work on the script is done in the left window pane.

The Script Editor has the following screen items:

| Screen item | What it does |
|--|--|
| New button  | Clears the right window pane so you can start working on a new script. The name of the new script appears in the left window pane. If another script was open when you clicked this button, it has not been lost. You can toggle between both scripts by clicking the titles in the left window pane. |
| Open button  | Allows you to open an existing script in the right window pane. The name of the newly opened script is added to the left window pane. If another script was open when you clicked this button, it has not been lost. You can toggle between both scripts by clicking the titles in the left window pane. |
| Save button  | Saves the script currently open in the right window pane. |
| Save As button  | Saves the script currently open in the right window pane under a new name. |
| Close button  | Closes the Script Editor. |
| Cut button  | Cuts the text currently selected in the script editing window onto the clipboard. |
| Copy button  | Copies the text currently selected in the script editing window onto the clipboard. |

| Screen item | What it does |
|--|---|
| Paste button  | Pastes the contents of the clipboard into the script editing window. |
| Search button  | Allows you to perform a search within the script. |
| Replace button  | Allows you to perform a search and replace within the script. |
| Comment button  | Allows you to comment out selected lines in the script. |
| Uncomment button  | Uncomments selected lines in the script. |
| Indent button  | Indents selected line(s) or whichever line the cursor is currently in. |
| Outdent button  | Outdents selected line(s) or whichever line the cursor is currently in. |
| Line numbers button  | Displays incremental numbers next to each line in the script providing easy identification. |
| Left window pane | Displays the names of all currently open scripts. An asterisk appearing after a name indicates the script contains unsaved changes. |
| Right window pane | Displays the code for the script currently selected in the left window pane. |

Creating scripts

To create a new script if the **Script Editor** is not yet open:

- In the main **Xstudio** window, select **New** from the **File** menu and when the **Create New** dialog box displays select **Script** from the **Other** tab. This opens the **Script Editor** window with a new, blank script.

To create a new script with the **Script Editor** already open:


- Click the **New** button  in the **Script Editor** window.

Editing commands


- Use the **Cut**, **Copy** and **Paste** buttons to edit the script.

Performing a search within the script

To perform a search in the script:

- 1 Click the **Search** button  in the **Script Editor** window. This displays the **Find** dialog box.
- 2 Enter the item to find in the **What** field.
- 3 Customize the search by doing the following:
 - Select the **Match whole word only** check box to match only a complete word. For example, if you enter **Width** and select this check box you will not find **Widths**. Keep the check box deselected to broaden the search.
 - Select the **Match case** check box to find matching items that also have identical casing. For example, if you enter **WIDTH** and select this check box, you will not find **Width** or **width**.
 - Select the **Regular expression** check box to search using a regular expression. Supports standard GNU regular expressions.
 - Select the **Up** or **Down** radio button to specify whether you would like the search performed forwards or backwards through the script from the cursor's current location.
- 4 Click the **Find** button to locate the next match.
Or,
Click the **Mark All** button to place an arrowhead mark beside all matches.
Or,
Click the **Close** button to close the dialog box without performing the search.


To perform a search and replace in the script:

- 1 Click the **Replace** button  in the **Script Editor** window. This displays the **Find & Replace** dialog box.
- 2 Enter the item to find in the **Find** field, and the item to replace it in the **Replace** field.
- 3 The search customizations are the same here as they are for the **Find** dialog box. The only new specification is the **Preserve case** check box, which ensures that the casing of the item being replaced is preserved. For example, if the word **width** is being replaced by **Height** and you have selected this check box, the replacement will appear as **height**.


- 4 Select the **Selection** radio button to search and replace only within the selected portion of the script, or the **Entire buffer** button to search the entire script file.
- 5 Click the **Find Next** button to locate the next match.
Or,
Click the **Replace** button to replace the currently selected match or the **Replace All** button to replace all matches at once.
Or,
Click the **Close** button.

Adding comments to scripts

To add a comment to the script:

- Select a block and click the **Comment** button  in the Script Editor window. Anything typed into the line will appear as a comment.

Or,


- Place your cursor in an existing line and click the **Comment** button  to make the line a comment.

To change commented text to script text:

- Select a block and click the **Uncomment** button  in the Script Editor window.

Opening existing scripts

To open an existing script:

- 1 Click the **Open** button  in the Script Editor. This displays the **Open Asset** dialog box, with the available scripts listed in the right-hand window pane.
- 2 Either double-click your selection, or single-click and press the **OK** button.

Saving scripts

When saving you have the option of providing the following additional metadata (information about the asset):

- the name of the person who authored the file
- a short description of the file
- keywords
- links to categories

To save a newly-created script:

- 1 From the **File** menu select **Save**. This displays the **Save Script** dialog box.
- 2 If desired, enter a new name for the asset in the **Name** field.
- 3 If desired, enter your name in the **Author** field.
- 4 If desired, enter a short description for the file in the **Description** field.
- 5 If desired, link the file to one or more categories by selecting check boxes in the **Categories** box. To create a new category, right-click in the **Categories** box and select **Add Category** from the pop-up menu.

- 6 If desired, assign the script one or more keywords by selecting check boxes in the **Keywords** box. (Keywords are used by Xstudio's Search XMS feature for searching and retrieving assets.) To create a new keyword, right-click in the **Keywords** box and select **Add Keyword** from the pop-up menu.
- 7 Click the **OK** button to save the file and exit the dialog box.
Or,
Click the **Cancel** button to discard all entered information and return to the unsaved file.

External Controls

The following topics are covered in this chapter:

- [About external controls](#), on page 332
- [Serial tab](#), on page 332
- [GPI tab](#), on page 332
- [Settings tab](#), on page 333

About external controls

You can control Xstudio using one of the two following external control systems:

- **Serial** - Commands received on the serial port are mapped to Xstudio commands. Available functions include: menu items, standard actions, user-defined actions.
- **III** - Allows you to specify a numeric range of commands. All numbers outside that range will be interpreted by Xstudio as being associated with pages.
- **GPI** - Similar to external control on serial port, but specifically allows you to link discreet GPI input signals to Xstudio actions.

Once you have set up your external controls on either the Serial or GPI tab, use the Settings tab to configure your external control device to function with Xstudio.

To access the external control setup:

- From the **Tools** menu select **External Control**. This displays the **External Control Mapping Manager** dialog box.

Serial tab

Use the Serial tab to map commands from Serial, TCP/IP and III external control systems to Xstudio commands.

To map commands from Serial external control systems to Xstudio commands:

- 1 From the **Tools** menu select **External Control**. This displays the **External Control Mapping Manager** dialog box.
- 2 On the **Serial** tab, enter your external control command in the **Commands** column.
- 3 Click in the **Functions** row to display a drop-down list of Xstudio commands.
Most functions do not require any parameters.
The exceptions are:
 - **Run Script** - Prompts you to select from the list of user-defined scripts.
 - **Send Event To Panel** - Prompts you to select an event from a drop-down list that will be sent to the panel. If you opt for **User Defined Event**, a dialog box will display. Use it to enter the name of the event.
- 4 Select the Xstudio command that you would like triggered when your external control command is performed.
- 5 Click the **OK** button.
- 6 Repeat steps 2 to 5 until all external control commands have been mapped to equivalent Xstudio commands, then click the **Close** button.

GPI tab

Use the GPI tab to map commands from a GPI control system to Xstudio commands.

To map commands from a GPI external control system to Xstudio commands:

- 1 From the **Tools** menu select **External Control**. This displays the **External Control Mapping Manager** dialog box.
- 2 On the GPI tab, find the desired number in the **Commands** column and then click in the **Functions** row to display a drop-down list of Xstudio commands.

- 3 Select the Xstudio command that you would like triggered when your external control command is performed.

Most functions do not require any parameters. The exceptions are:

- **Run Script** - Prompts you to select from the list of user-defined scripts.
- **Send Event To Panel** - Prompts you to select an event from a drop-down list that will be sent to the panel. If you opt for User Defined Event, a dialog box will display. Use it to enter the name of the event.

- 4 Click the **OK** button.
- 5 Repeat steps 2 to 4 until all external control commands have been mapped to equivalent Xstudio commands, then click the **Close** button.

Settings tab

Use the Settings tab to configure your external control device to function with Xstudio.

To configure the external control device:

- 1 From the **Tools** menu select **External Control**. This displays the **External Control Mapping Manager** dialog box.
- 2 On the **Settings** tab, enter the COM Port information for the computer to which you will be connecting the external control device. The fields under **Serial 1** are used for setting up Serial or TCP/IP external control systems, whereas the fields under **Serial 2** are for configuring GPI control system settings.
- 3 If setting up a GPI control system, select one of the common GPI prefixes from the **GPI Prefix** drop-down list.
- 4 Click the **OK** button.



Grass Valley Technical Support

For technical assistance, contact our international support center, at 1-800-547-8949 (US and Canada) or +1 530 478 4148.

To obtain a local phone number for the support center nearest you, please consult the Contact Us section of Grass Valley's website (www.grassvalley.com).

An online form for e-mail contact is also available from the website.

Corporate Head Office

Grass Valley
3499 Douglas-B.-Floreani
St-Laurent, Quebec H4S 2C6
Canada
Telephone: +1 514 333 1772
Fax: +1 514 333 9828
www.grassvalley.com