

# Creating and Managing ArcestrA Graphics User's Guide

**Invensys Systems, Inc.**

Revision B

Last Revision: 11/3/08



## **Copyright**

© 2008 Invensys Systems, Inc. All Rights Reserved.

All rights reserved. No part of this documentation shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Invensys Systems, Inc. No copyright or patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this documentation, the publisher and the author assume no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

The information in this documentation is subject to change without notice and does not represent a commitment on the part of Invensys Systems, Inc. The software described in this documentation is furnished under a license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of these agreements.

Invensys Systems, Inc.  
26561 Rancho Parkway South  
Lake Forest, CA 92630 U.S.A.  
(949) 727-3200

<http://www.wonderware.com>

For comments or suggestions about the product documentation, send an e-mail message to [productdocs@wonderware.com](mailto:productdocs@wonderware.com).

## **Trademarks**

All terms mentioned in this documentation that are known to be trademarks or service marks have been appropriately capitalized. Invensys Systems, Inc. cannot attest to the accuracy of this information. Use of a term in this documentation should not be regarded as affecting the validity of any trademark or service mark.

Alarm Logger, ActiveFactory, ArchestrA, Avantis, DBDump, DBLoad, DT Analyst, Factelligence, FactoryFocus, FactoryOffice, FactorySuite, FactorySuite A<sup>2</sup>, InBatch, InControl, IndustrialRAD, IndustrialSQL Server, InTouch, MaintenanceSuite, MuniSuite, QI Analyst, SCADAAlarm, SCADASuite, SuiteLink, SuiteVoyager, WindowMaker, WindowViewer, Wonderware, Wonderware Factelligence, and Wonderware Logger are trademarks of Invensys plc, its subsidiaries and affiliates. All other brands may be trademarks of their respective owners.

---

# Contents

	<b>Welcome.....</b>	<b>15</b>
	Documentation Conventions.....	15
	Technical Support .....	16
<b>Chapter 1</b>	<b>About Creating and Managing ArcestrA Symbols .....</b>	<b>17</b>
	Managing ArcestrA Symbols .....	18
	Managing Symbols in the Graphic Toolbox .....	18
	Managing Symbols in Automation Objects .....	19
	Re-using ArcestrA Symbols .....	20
	ArcestrA Symbols and Instantiation .....	22
	ArcestrA Symbol Creation: The ArcestrA Symbol Editor .....	22
	The ArcestrA Symbol Editor.....	23
	Elements .....	26
	Properties.....	32
	Animations.....	37
	Embedded Symbols .....	44
	Appearance of Embedded Symbols.....	44
	Changing Embedded Symbols .....	44
	Embedding and Instantiation.....	45
	Symbol Change Propagation.....	46
	Size Propagation and Anchor Points .....	47
	Showing Quality and Status.....	49
	Showing Quality and Status with the Status Element .....	49
	Showing Quality and Status by Overriding.....	49

<b>Chapter 2</b>	<b>Comparing WindowMaker and ArcestrA Symbol Editor.....</b>	<b>51</b>
	Differences between WindowMaker and the ArcestrA Symbol Editor.....	51
	Elements .....	52
	Appearance .....	52
	Enhanced Functionality.....	52
	Procedures for Common WindowMaker Tasks and Techniques.....	54
	Using Graphics .....	54
	Using Animations.....	57
	Using Scripts .....	62
<b>Chapter 3</b>	<b>Managing Symbols .....</b>	<b>65</b>
	About Symbols .....	65
	Creating a New Symbol .....	66
	Creating Symbols in the Graphic Toolbox .....	66
	Creating Symbols in AutomationObject Templates .....	68
	Creating Symbols in AutomationObject Instances .....	70
	Opening Symbols for Editing.....	71
	Organizing Symbols in the Graphic Toolbox .....	73
	Creating Graphic Toolsets in the Graphic Toolbox .....	73
	Moving Symbols between Graphic Toolsets.....	74
	Renaming Symbols.....	74
	Copying Symbols .....	75
	Renaming Graphic Toolsets.....	75
	Deleting Graphic Toolsets.....	75
	Moving Graphic Toolsets .....	76
	Customizing Graphic Toolsets.....	76
	Importing and Exporting Symbols as ArcestrA Object Files .....	77
	Importing Symbols .....	77
	Exporting Symbols .....	78
	Deleting a Symbol .....	79
	Configuring Security for Symbols .....	80
	Viewing a Symbol in Read-Only Mode.....	82
<b>Chapter 4</b>	<b>Using the ArcestrA Symbol Editor .....</b>	<b>83</b>
	Showing, Hiding, and Adjusting Panels .....	84
	Panning and Zooming the Canvas .....	84
	Panning.....	84
	Zooming.....	87

Configuring the Grid .....	90
Configuring the Grid Appearance .....	90
Aligning Elements with the Grid .....	91
<b>Chapter 5 Working with Graphic Elements .....</b>	<b>93</b>
About Graphic Elements.....	93
Drawing and Dragging Elements .....	94
Drawing Rectangles, Rounded Rectangles, Ellipses, and Lines .....	94
Drawing Polylines, Polygons, Curves, and Closed Curves.....	95
Drawing 2-Point Arcs, 2-Point Pies and 2-Point Chords.....	95
Drawing 3-Point Arcs, 3-Point Pies, and 3-Point Chords.....	96
Placing and Importing Images .....	96
Drawing Buttons .....	96
Placing Text .....	97
Drawing Text Boxes .....	97
Drawing Status Elements.....	97
Drawing Windows Controls .....	98
Dragging Elements.....	98
Editing Element Properties .....	98
Selecting Elements .....	100
Selecting Elements by Mouse Click.....	101
Selecting Elements by Lasso .....	102
Selecting All Elements .....	102
Selecting Elements Using the Elements List .....	103
Unselecting Elements .....	103
Inline Editing.....	104
Copying, Cutting, and Pasting Elements .....	106
Copying Elements.....	106
Cutting or Deleting Elements.....	107
Duplicating Elements.....	108
Moving Elements .....	109
Aligning Elements .....	110
Aligning Elements Horizontally .....	111
Aligning Elements Vertically.....	112
Aligning Elements by their Center Points.....	114
Aligning Elements by their Points of Origin .....	114
Adjusting the Spacing between Elements .....	115
Distributing Elements.....	116

Making Space between Elements Equal.....	117
Increasing Space between Elements .....	118
Decreasing Space between Elements .....	119
Removing All Space between Elements .....	120
Resizing Elements .....	121
Resizing a Single Element with the Mouse .....	121
Resizing Elements by Changing Size Properties.....	122
Resizing Elements Proportionally .....	122
Making Elements the Same Width, Height, or Size ..	123
Adjusting the z-Order of Elements .....	124
Rotating Elements .....	126
Rotating Elements with the Mouse.....	126
Rotating Elements by Changing the Angle Property.	127
Rotating Elements by 90 Degrees .....	128
Moving the Origin of an Element .....	129
Changing Points of Origin with the Mouse .....	129
Changing Points of Origin in the Properties Editor...	130
Flipping Elements .....	131
Locking and Unlocking Elements.....	132
Making Changes Using Undo and Redo .....	133
Working with Groups of Elements .....	134
Creating a Group of Elements .....	134
Ungrouping .....	135
Adding Elements to Existing Groups.....	136
Removing Elements from Groups.....	136
Editing Components within a Group .....	137
Using Path Graphics .....	138
Creating a Path Graphic .....	139
Breaking the Path of a Path Graphic .....	140
Changing a Path Graphic .....	140
Adding Elements to an Existing Path Graphic .....	148
Removing Elements from a Path Graphic .....	149

## Chapter 6 Editing Common Properties of Elements and Symbols..... 151

Editing the Name of an Element.....	152
Editing the Fill Properties of an Element.....	152
Setting Fill Style .....	153
Setting Unfilled Style.....	154
Setting Fill Orientation .....	155
Setting Fill Behavior .....	155

Setting Horizontal Fill Direction and Percentage .....	156
Setting Vertical Fill Direction and Percentage.....	156
Editing the Line Properties of an Element .....	157
Setting Start or End Points of a Line .....	157
Setting the Line Weight .....	158
Setting the Line Pattern .....	158
Setting the Line Style .....	159
Setting the Text Properties of an Element .....	160
Setting the Displayed Text .....	160
Setting the Text Display Format.....	160
Setting the Text Font .....	161
Setting the Text Color .....	161
Setting the Text Alignment .....	162
Substituting Strings.....	164
Setting Style .....	166
Setting a Solid Color.....	166
Setting a Gradient.....	171
Setting a Pattern .....	181
Setting a Texture.....	183
Setting the Style to No Fill .....	184
Setting the Transparency of a Style.....	185
Setting the Transparency Level of an Element .....	185
Tweaking the Colors and Transparency of a Gradient.....	186
Enabling and Disabling Elements for Run-Time Interaction .....	187
Changing the Visibility of Elements .....	187
Editing the Tab Order of an Element.....	188
Using the Format Painter to Format Elements .....	189
Editing the General Properties of a Symbol .....	190

## **Chapter 7 Setting Symbol and Element-Specific Properties ..... 193**

Setting the Radius of Rounded Rectangles .....	194
Setting Line End Shape and Size .....	195
Setting Auto Scaling and Word Wrapping for a Text Box .....	196
Using Images .....	197
Placing an Image on the Canvas .....	197
Setting the Image Display Mode .....	198
Setting the Image Alignment .....	199
Setting the Image Color Transparency .....	200

Editing the Image.....	200
Setting the Image Editing Application .....	201
Selecting a Different Image .....	202
Using Buttons.....	202
Automatically Scaling Text in Buttons .....	202
Wrapping Text in Buttons .....	203
Configuring Buttons with Images .....	203
Editing Control Points .....	204
Moving Control Points .....	204
Adding and Removing Control Points.....	205
Changing the Tension of Curves and Closed Curves ....	206
Changing Angles of Arcs, Pies and Chords.....	207
Monitoring and Showing Quality and Status.....	209
Using Status Elements .....	209
Overriding Element Appearance Depending on Quality and Status of its Attributes .....	212
Using Windows Common Controls.....	219
Changing Background Color and Text Color of Windows Common Controls .....	220
Reading and Writing the Selected Value at Run Time .....	221
Configuring Radio Button Group Controls .....	222
Configuring Check Box Controls .....	224
Configuring Edit Box Controls .....	225
Configuring Combo Box Controls .....	227
Configuring Calendar Controls .....	230
Configuring DateTime Picker Controls.....	234
Configuring List Box Controls.....	236

## **Chapter 8 Using Custom Properties..... 239**

About Custom Properties.....	239
Managing Custom Properties .....	240
Adding and Deleting Custom Properties .....	241
Configuring Custom Properties.....	242
Validating Custom Properties .....	244
Clearing the Configuration of Custom Properties.....	244
Renaming Custom Properties.....	244
Linking Custom Properties to External Sources.....	245
Overriding Custom Properties.....	245
Reverting to Original Custom Property Values .....	246
Examples of Using Custom Properties .....	246



Using Binding in Custom Properties .....	246
Changing the Expression or Reference of a Custom Property at Run Time .....	248
<b>Chapter 9 Animating Graphic Elements .....</b>	<b>249</b>
Adding an Animation to an Element.....	250
Reviewing which Animations are Assigned to an Element .....	251
Showing and Hiding the Animation List .....	252
Removing Animations from an Element.....	253
Enabling and Disabling Animations .....	254
Validating the Configuration of an Animation .....	255
Clearing the Configuration from an Animation .....	256
Connecting Animations with Data Sources .....	256
Connecting Animations with ArcestrA Attributes ...	256
Connecting Animations with Element Properties .....	257
Connecting Animations with Custom Properties .....	258
Connecting Animations with InTouch Tags .....	259
Setting the Input Mode .....	261
Managing Animations.....	262
Organizing the Animation List.....	262
Switching between Animations .....	263
Configuring Common Types of Animations .....	263
Configuring a Visibility Animation .....	264
Configuring a Fill Style Animation .....	264
Configuring a Line Style Animation .....	269
Configuring a Text Style Animation .....	273
Configuring a Blink Animation .....	277
Configuring a Percent Fill Horizontal Animation .....	279
Configuring a Percent Fill Vertical Animation.....	281
Configuring a Horizontal Location Animation .....	284
Configuring a Vertical Location Animation.....	285
Configuring a Width Animation .....	286
Configuring a Height Animation .....	287
Configuring an Orientation Animation.....	288
Configuring a Value Display Animation .....	291
Configuring a Tooltip Animation.....	297
Configuring a Disable Animation.....	298
Configuring a User Input Animation .....	299
Configuring a Horizontal Slider Animation.....	307
Configuring a Vertical Slider Animation .....	309

Configuring a Pushbutton Animation.....	311
Configuring an Action Script Animation .....	316
Configuring a Show Symbol Animation.....	319
Configuring a Hide Symbol Animation.....	327
Configuring Element-Specific Animations .....	328
Configuring Animation for a Status Element.....	328
Configuring a Radio Button Group Animation.....	330
Configuring a Check Box Animation.....	335
Configuring an Edit Box Animation.....	336
Configuring a Combo Box Animation.....	337
Configuring a Calendar Control Animation .....	342
Configuring a DateTime Picker Animation .....	343
Configuring a List Box Animation .....	345
Submitting the Value Changes .....	349
Cutting, Copying and Pasting Animations .....	350
Substituting References in Elements.....	351

## Chapter 10 Adding and Maintaining Symbol Scripts..... 353

About Symbol Scripts.....	353
Predefined and Named Scripts.....	354
Execution Order of Symbol Scripts .....	354
Security in Symbol Scripts.....	354
Symbol Script Timeouts.....	355
Error Handling .....	355
Configuring the Predefined Scripts of a Symbol .....	356
Adding Named Scripts to a Symbol .....	358
Editing Symbol Scripts .....	360
Renaming Scripts in a Symbol .....	360
Removing Scripts from a Symbol .....	361
Substituting Attribute References in Scripts .....	361
Example of Changing Element Properties using Scripts .....	362
Using Methods in Scripting .....	363
Configuring Edit Box Methods .....	364
Configuring Combo Box and List Box Methods.....	365

## Chapter 11 Using Client Controls..... 371

About Client Controls .....	372
Importing Client Controls.....	373
Importing Client Controls.....	373
Importing Previously Exported Client Controls.....	376

Organizing Client Controls.....	376
Embedding Client Controls.....	377
Example of Embedding the ActiveFactory TagPicker Client Control.....	378
Viewing and Changing the Properties of Client Controls .....	379
Example of Changing a Property of the ActiveFactory TagPicker Control.....	380
Binding Client Control Properties to Attributes or Element References.....	381
Example of Data Binding in the ActiveFactory TagPicker Control.....	383
Configuring Client Control Event Scripts.....	384
Example of Configuring an Event Script for the ActiveFactory TagPicker Control.....	385
Animating Client Controls.....	387
Exporting Client Controls.....	388
Securing Client Controls.....	388
Viewing Additional Client Control Information .....	388
Viewing the Client Control Assemblies.....	389
Viewing Class Name, Vendor, and Version of a Client Control.....	389
Viewing Objects and Symbols Referencing Client Controls .....	390
<b>Chapter 12 Embedding Symbols within Symbols .....</b>	<b>391</b>
Embedding Symbols .....	392
Renaming Source Symbols and Hosting AutomationObjects.....	395
Editing the Embedded Symbol .....	395
Overriding Custom Properties of the Source Symbol....	396
Restoring an Embedded Symbol to the Original Size of its Source Symbol .....	397
Converting an Embedded Symbol to a Group.....	398
Detecting the Source Symbol of an Embedded Symbol .....	398
Editing the Source of an Embedded Symbol.....	399
Controlling Size Propagation of Embedded Symbols ....	399
Setting the Anchor Point of a Source Symbol .....	400
Showing or Hiding the Anchor Points of Embedded Symbols.....	401
Enabling or Disabling Dynamic Size Change of Embedded Symbols .....	401

Selecting Alternate Symbols and Instances .....	403
Selecting Alternate Symbols.....	403
Selecting Alternate Instances.....	404
Detecting and Editing the Containing AutomationObject Instance .....	404
Creating a New Instance of the Containing AutomationObject .....	405
<b>Chapter 13 Migrating InTouch SmartSymbols .....</b>	<b>407</b>
Importing InTouch SmartSymbols into an Arcestra Symbol .....	408
Restrictions for SmartSymbol Import .....	411
Importing InTouch Graphics .....	411
Importing Graphical Animation .....	414
Importing Action Scripts.....	415
Importing References .....	416
<b>Chapter 14 Switching Languages for Graphic Elements.....</b>	<b>419</b>
About Language Switching for Arcestra Graphics .....	420
Graphic Elements that Support Translation.....	420
Animations that Support Translation.....	421
Selecting the Language for a Symbol.....	422
Removing a Language for a Symbol.....	423
Creating Elements When Multiple Languages are Defined for a Galaxy .....	423
Moving Symbols to Galaxies with Different Language Settings.....	424
How Fonts are Applied at Design Time .....	425
Language Switching for Embedded Symbols.....	425
String Substitutions and Language Switching .....	426
Translating String Custom Properties.....	428
Translating Custom Properties for a Base Symbol ...	428
Translating Custom Properties for an Embedded Symbol .....	429
Translation Support for Client Controls with Satellite Assemblies .....	429
Translation Support for Arcestra Client Controls.....	430
Importing InTouch SmartSymbols that Have Translated Data .....	431
Support for Empty Strings .....	431
Language Switching Example .....	432

Overriding Translated Strings for ArchestrA	
Symbols in WindowMaker .....	435
Overriding Translated String Substitutions.....	435
Overriding Translated Custom Properties .....	435
Language Switching at Run Time.....	435
How Languages are Shown in WindowViewer .....	436
Precedence Rules for Showing the Language	
and Font .....	437
Default Language Fonts at Run Time.....	437
Switching Languages for Custom Properties at	
Run Time .....	438
Switching Languages and String Substitutions	
at Run Time.....	439
Language Settings for Popup Symbols.....	440
Language Settings and Data Types .....	441
<b>Appendix A List of Element Properties .....</b>	<b>443</b>
Alphabetical List of Properties .....	444
List by Functional Area .....	467
Graphic Category Properties .....	468
Appearance Category Propeties.....	469
Fill Style Group Properties .....	481
Line Style Group Properties .....	484
Text Style Group Properties .....	486
Runtime Behavior Group Properties.....	488
Custom Properties Group Properties .....	491
<b>Appendix B Windows Common Control List</b>	
<b>Methods .....</b>	<b>493</b>
Overview of Windows Common Control List Methods..	494
<b>Index .....</b>	<b>499</b>



# Welcome

You can use the ArcestrA Integrated Development Environment (IDE) to create symbols with the ArcestrA Symbol Editor that appear in managed InTouch applications. This document explains how to use the Symbol Editor to create and manage ArcestrA graphics.

You can view this document online or you can print it, in part or whole, by using the print feature in Adobe Reader.

This documentation assumes you know how to use Microsoft Windows, including navigating menus, moving from application to application, and moving objects on the screen. If you need help with these tasks, see the Microsoft documentation.

## Documentation Conventions

This documentation uses the following conventions:

Convention	Used for
Initial Capitals	Paths and file names.
<b>Bold</b>	Menus, commands, dialog box names, and dialog box options.
Monospace	Code samples and display text.

## Technical Support

Wonderware Technical Support offers a variety of support options to answer any questions on Wonderware products and their implementation.

Before you contact Technical Support, refer to the relevant section(s) in this documentation for a possible solution to the problem. If you need to contact technical support for help, have the following information ready:

- The type and version of the operating system you are using.
- Details of how to recreate the problem.
- The exact wording of the error messages you saw.
- Any relevant output listing from the Log Viewer or any other diagnostic applications.
- Details of what you did to try to solve the problem(s) and your results.
- If known, the Wonderware Technical Support case number assigned to your problem, if this is an ongoing problem.



# Chapter 1

## About Creating and Managing ArcestrA Symbols

ArcestrA symbols are graphics you can create to visualize data in an InTouch HMI system.

You use the ArcestrA Symbol Editor to create ArcestrA symbols from basic elements, such as rectangles, lines, and text elements.

After you create an ArcestrA symbol, you can embed it into another symbol or an InTouch window and use it at run time.

You can embed an ArcestrA symbol in a template or instance of an ArcestrA Object, providing a way to visualize object-specific information quickly and easily. Embedding a symbol in a template means that you can update one symbol and cascade the changes throughout your application.

The ArcestrA Symbol Editor is a powerful addition to the standard InTouch editor, called WindowMaker.

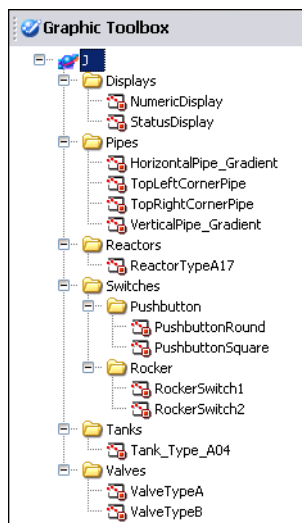
## Managing ArcestrA Symbols

Depending on your development requirements, you can select where and how to store ArcestrA symbols.

- Store symbols in the Graphic Toolbox if you want to define them as a standard that you can re-use, such as a generic valve symbol. You can store ArcestrA symbols here if you only want to use them in the InTouch HMI.
- Store symbols as AutomationObject templates if you want to use the symbols in multiple instances at run time. For example, you can create a valve symbol contained in an AutomationObject template that represents the functionality of a valve type on your plant floor.
- Store symbols as AutomationObject instances if you want to use the symbols in only one specific object instance. For example, an AutomationObject instance that can be assigned a very specific piece of machinery as a symbol.

### Managing Symbols in the Graphic Toolbox

The Graphic Toolbox lets you organize your symbols in special folders called toolsets. You can create a hierarchy of toolsets. You can also move symbols between toolsets.



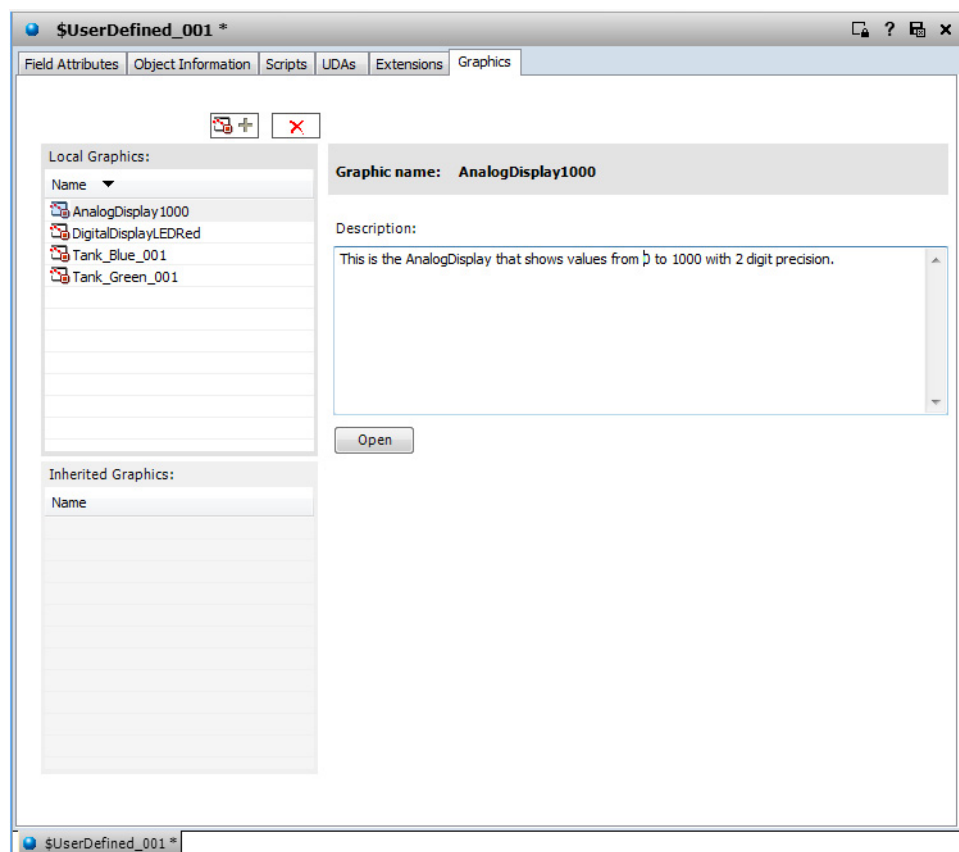
**Note** Symbol names must be unique within the Graphic Toolbox.

## Managing Symbols in Automation Objects

You can create ArcestrA symbols in AutomationObjects. Each AutomationObject has a **Graphics** tab that lets you create, edit, rename, and delete ArcestrA symbols. These symbols appear in the **Local Graphics** list.

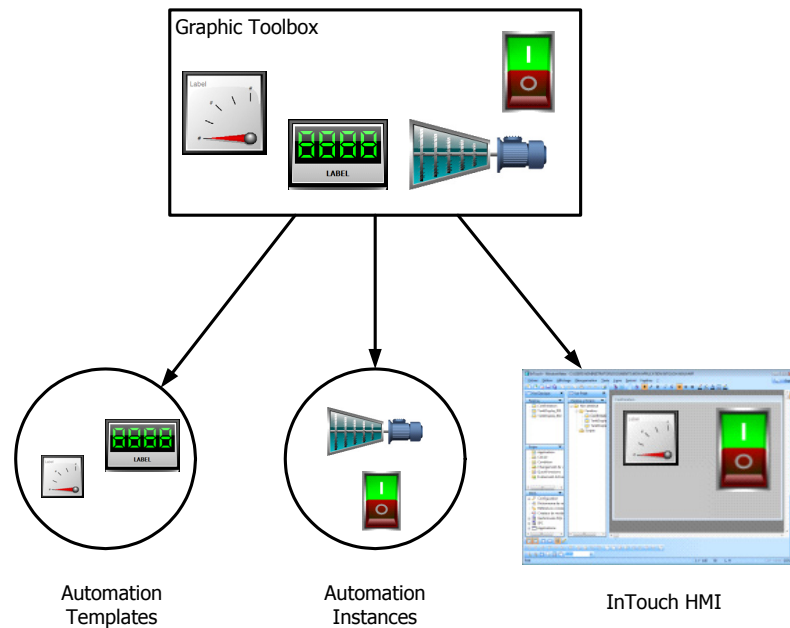
When you derive an AutomationObject from a parent AutomationObject that contains symbols, all the symbols are inherited. Inherited symbols appear in the **Inherited Graphics** list.

**Note** You can only open inherited symbols in the ArcestrA Symbol Editor in read-only mode.



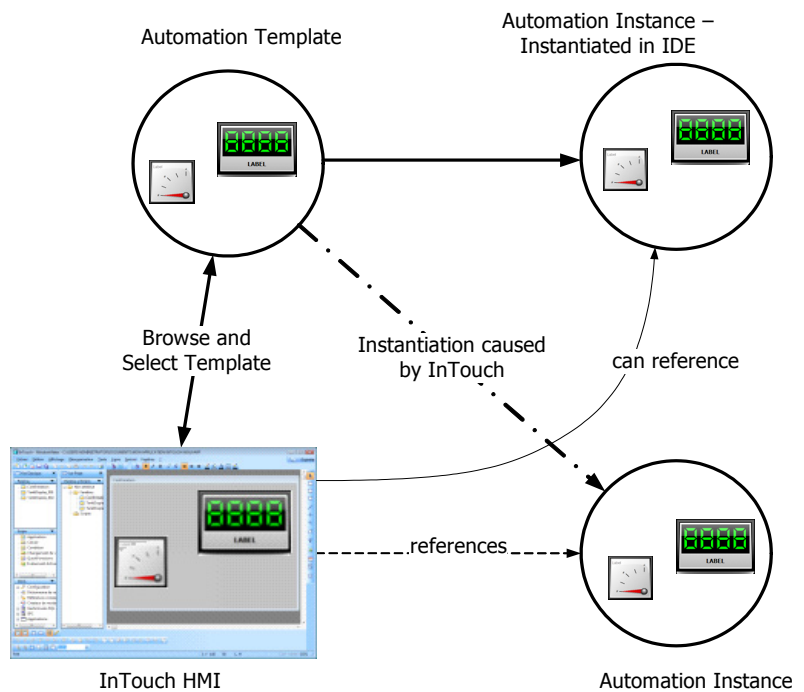
## Re-using ArcestrA Symbols

You can re-use ArcestrA symbols that you create in the Graphic Toolbox in AutomationObject templates, AutomationObject instances, or in InTouch windows. This is called embedding.



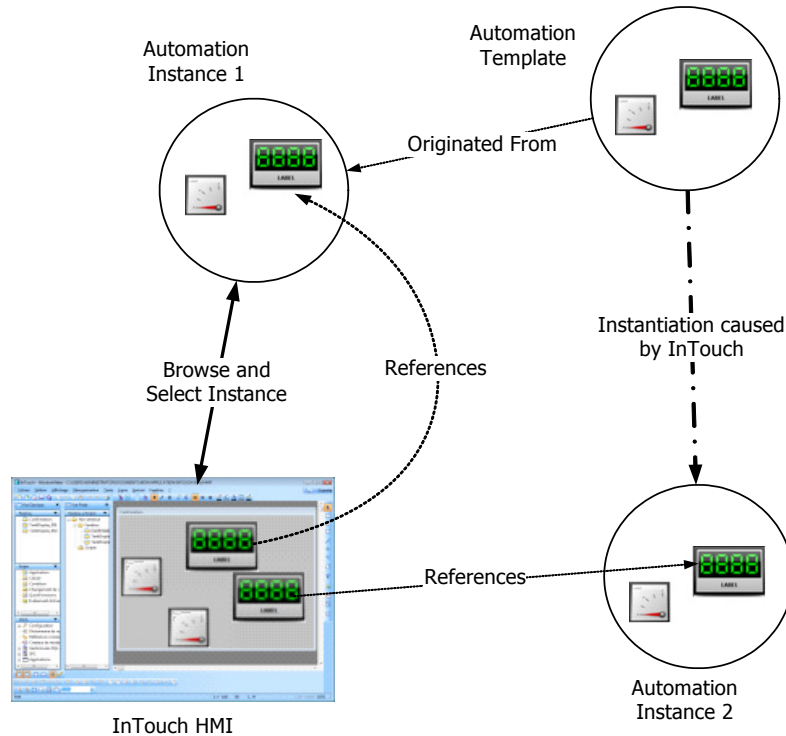
When you derive an AutomationObject template, its ArchestrA symbols are inherited by the new instance. This can be caused by:

- Deriving an instance of the template in the IDE. When you derive an instance of an AutomationObject template that contains symbols, the created instance contains inherited symbols.
- Embedding a new ArchestrA symbol in WindowMaker. A new AutomationObject instance is derived to which the symbol in InTouch WindowMaker then points.



## ArcestrA Symbols and Instantiation

When you embed an ArcestrA symbol into an InTouch window and the symbol is contained in an AutomationObject template, you can easily create a new instance of the AutomationObject. The embedded ArcestrA symbol automatically references the new object.



## ArcestrA Symbol Creation: The ArcestrA Symbol Editor

The ArcestrA Symbol Editor is the tool you use to create an ArcestrA symbol.

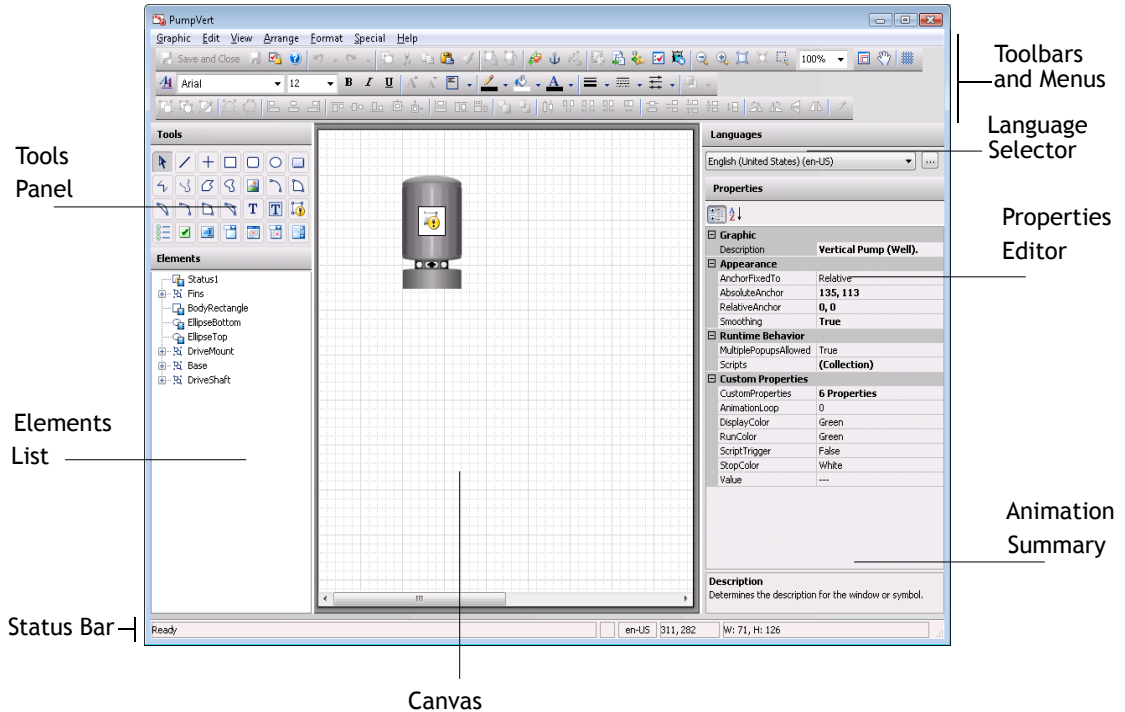
First, you select a basic graphical object, called an element, from a tools panel and place it on the drawing area, called the canvas. Typical elements are lines, rectangles, ellipses, curves, and so on.

You can then change the appearance of your drawn elements by accessing their properties directly or by graphically manipulating them.

Finally, you can configure animations for the elements or the symbol.

## The ArchestrA Symbol Editor

After you open the ArchestrA Symbol Editor, you see the various tools and palettes that you use to create and customize symbols.



The ArchestrA Symbol Editor includes the following areas:

- **Tools Panel.** This is a collection of elements you use to create your symbol.
- **Canvas.** This is the area in which you place and edit elements to create a symbol.
- **Elements List.** This list shows you named elements on the canvas in a hierarchical view.
- **Language Selector.** This list shows the configured languages for the symbol. For more information, see Chapter 14, Switching Languages for Graphic Elements.
- **Properties Editor.** This editor shows the properties belonging to one or more currently selected elements.
- **Animation Summary.** This area shows you a list of animations belonging to the currently selected element. It is only visible if an element is selected.

## Tools Panel

The Tools panel contains elements you can select to create your symbol on the canvas.



The Tools panel includes:

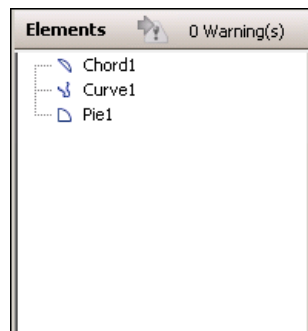
- Basic objects, such as lines, rectangles, polygons, arcs, and so on.
- A pointer tool to select and move elements on the canvas.
- Windows controls, such as combo boxes, calendar controls, radio button groups, and so on.
- A status element that you can use to show quality and status of selected ArcestrA attributes.

For more conceptual information, see [Elements](#) on page 26.

For more information on how to use elements, see [Working with Graphic Elements](#) on page 93.

## Elements List

The Elements List is a list of all elements on the canvas.



The Elements List is particularly useful for selecting one or more elements that are visually hidden by other elements on the canvas. You can use the Elements List to:

- See a list of all elements, groups of elements, and embedded symbols on the canvas.
- Select elements or groups of elements to work with them.
- Rename an element or a group of elements.

---

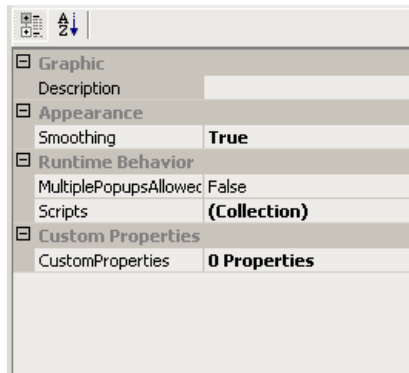
**Caution** If you rename an element or a group, the animation references to it do not automatically update. You must manually change all animation links referencing the old name. For more information, see [Substituting References in Elements](#) on page 351.

---



## Properties Editor

You can use the Properties Editor to view and set properties for the selected element or group of elements.

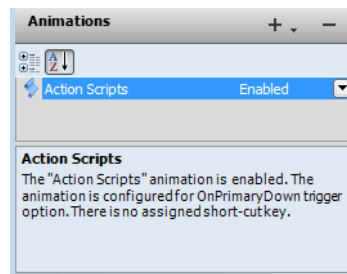


For more conceptual information about element properties, see Properties on page 32.

For more information on how to use element properties, see Editing Common Properties of Elements and Symbols on page 151.

## Animations Summary

You can use the Animations summary to review, select, and configure the animation behavior of a selected element.



For an overview of the different animation types, see Animation Types on page 37.

For more information on how to use the animations, see Animating Graphic Elements on page 249.

## Canvas

The canvas is your drawing area. You use it as you would in other image editing software by drawing elements and changing them to your requirements.

## Elements

You use elements to create a symbol. The ArcestrA Symbol Editor provides the following:

- Basic elements such as lines, rectangles, ellipses, arcs, and so on
- Status element to show a quality status icon
- Windows controls, such as combo boxes, calendar controls, radio button groups, and so on

You can create the following from existing elements on the canvas:

- Groups
- Path graphics

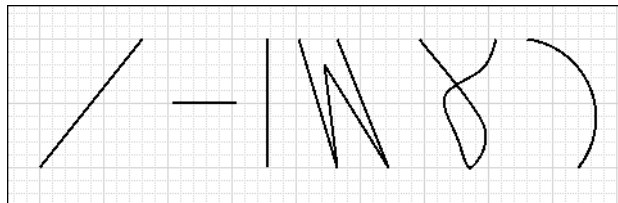
You can embed the following on the canvas:

- Imported Client Controls
- Other symbols

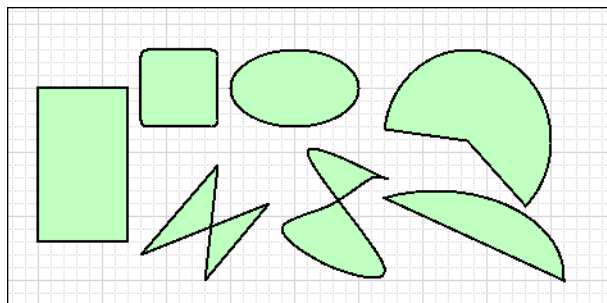
### Basic Elements

You can use the following basic elements to create a symbol:

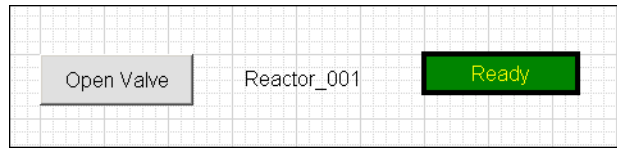
- Open elements, such as lines, H/V lines, polylines, curves, and arcs



- Closed elements, such as rectangle, rounded rectangle, ellipse, polygon, closed curve, pie, and chord. You can draw arcs, pies, and chords from two points or from three points.



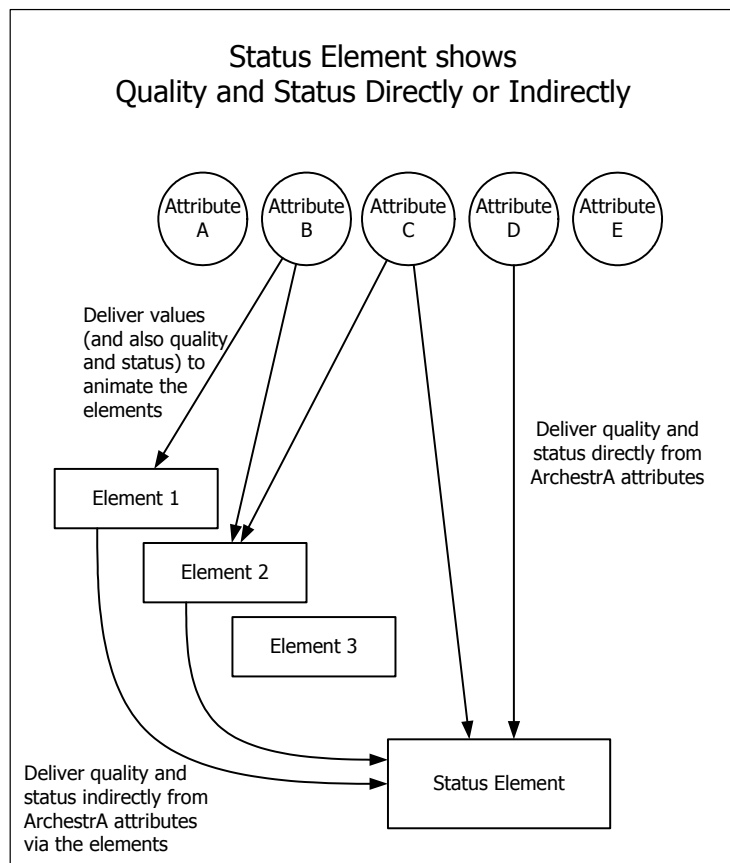
- Text elements, such as buttons, text, and text boxes



## Status Element

You can use a status element to monitor and indicate the status or quality of:

- All ArchestrA attributes used in one or more specified animated elements in the same hierarchical level.
- One or more specified ArchestrA attributes.



Abnormal quality and status can be:











- An error status due to communication, configuration, operational, security, or software error.
- Bad, initializing, pending, uncertain, or warning quality.

The status element lets you monitor:

- One or more elements on the canvas that contain animations.
- One or more ArcestrA attributes, values, or expressions.

As a status element can monitor multiple attributes at the same time, but only show one icon, it needs to select the more severe quality or status.

The element prioritizes according to an order, as shown in the following table:

	Communication Error (highest priority)
	Configuration Error
	Bad
	Uncertain
	Initializing
	Operational Error
	Software Error
	Security Error
	Warning
	Pending (lowest priority)

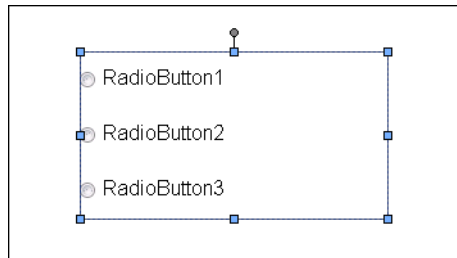
For more information, see [Configuring Animation for a Status Element](#) on page 328.

You can also change the icons that are shown for each status or quality. For more information, see [Showing Quality and Status](#) on page 49.

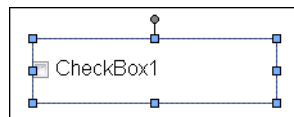
## Windows Common Controls

Using Windows common controls, you can add extended user interaction to your symbol. You can use:

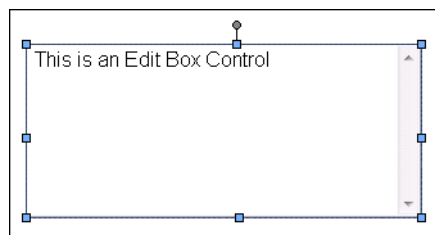
- A radio button group to select an option from a mutually exclusive group of options.



- A check box to add a selectable option.



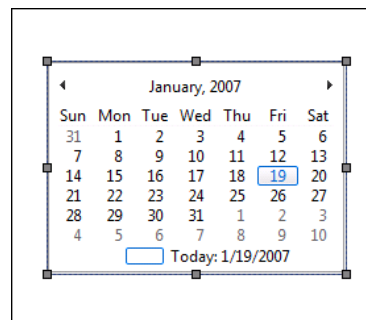
- An edit box to add an entry box for text.



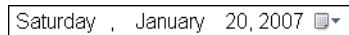
- A combo box to select an option from a drop-down list.



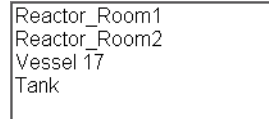
- A calendar to use a date selection control.



- A date and time picker to select a date and time in a compact format .



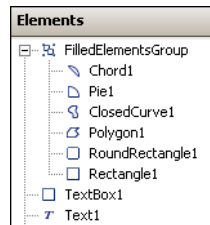
- A list box to select one or more options from a list.



## Groups

Grouping lets you relate elements together as a unit. Groups can contain elements and other groups.

Groups are shown in the Elements List with a default name, such as Group1. They are shown as a branch in the element hierarchy.



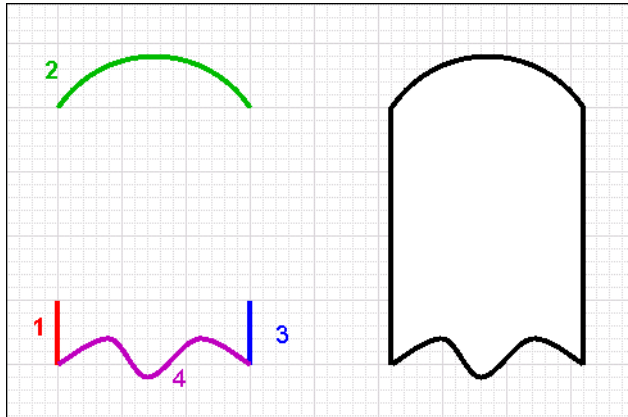
For example, you can create a series of elements that model a valve in your facility. When the valve has all the properties and animations you want, you can group the elements together.

You can then work with the elements as one set of elements or, by selecting the elements in the Elements List, you can work with the individual elements in the group without having to break the group. This is called inline editing.

Another advantage of inline editing is that you can easily select an individual element graphically without having to know its element name.

## Path Graphics

Path graphics are elements that combine selected open elements, such as lines, H/V lines, polylines, curves, and arcs, into a single closed graphic element.



A path graphic depends on the:

- Order in which you drew the elements. Each element is linked to the next element by z-order. The z-order of the elements is the order shown in the Elements List.
- Direction in which you drew its elements. The ending point of one element is connected to the starting point of the next element.

The properties of the elements contained within a path graphic are retained. When you break a path graphic, the elements it contains appear as they did before you created the path graphic.

A path graphic has the same properties as a rectangle, ellipse, or polygon. These properties are lost when you break the path.

## Windows Client Controls

Windows client controls are .NET-based controls you can use in an ArcestrA symbol to extend its functionality.

After you embed a client control into a symbol, you can:

- Connect the native properties of the client control to ArcestrA attributes and element references.
- Configure scripts for each of its events.
- Edit the native properties directly with the Properties Editor.
- Configure and override animations.

You can embed a symbol that contains an embedded client control into a managed InTouch application and use the functionality of the client control directly in the InTouch HMI.

For more information, see [Using Client Controls](#) on page 371.

## Properties

Properties determine the appearance and behavior of an element or the symbol. For example, the width property determines the width in pixels of the selected element.

There are two types of properties:

- Predefined properties
- Custom properties

When you configure an element to reference one of its **own** properties in a configuration field or a script, you can just use its property name. For ArcestrA symbols, there are no self-referencing keywords such as “me.” as used for AutomationObjects.

You can, however, use the “me.” keyword to reference attributes of the AutomationObject that is hosting the ArcestrA symbol you are currently configuring.



## Predefined Properties

Properties are specific to the selected element and can vary between elements of different types. All elements have the following property categories:

- Graphic - the name of the element (or group)
- Appearance - element dimension, location, rotation, transparency, and locked status

You can view specific properties for a specific kind of element or group by clicking a drawing tool and drawing an element.

You set properties at design time. Some properties can be read or written to at run time, such as X, Y, Width, Height, Visible, and so on. The element type determines which properties are available and can be read or written at run time.

## Custom Properties

You can use custom properties to extend the functionality of a symbol. A custom property can contain:

- A value that can be read and written to.
- An expression that can be read.
- An ArchestrA attribute that can be read and written to if the attributes allows being written to.
- A property of an element or symbol.
- A custom property of a symbol.
- A reference to an InTouch tag.

For example, for a tank symbol called TankSym you can create a custom property called TankLevel that is calculated from an attribute reference to Tank\_001.PV. You can then reference the tank level by TankSym.TankLevel.

Custom properties appear in the Properties Editor when no elements are selected. You can edit default initial values of custom properties in the editor directly or use the **Edit Custom Properties** dialog box to do so.

For more information, see Using Custom Properties on page 239.

## Properties of Groups

Groups have their own properties you can view and set in the Properties Editor. For most properties, changing group properties indirectly affects the properties of its contained elements.

You can change the following group properties:

- Name (Name)
- Position (X, Y)
- Size (Width, Height)
- Orientation (Angle)
- Point of Origin (AbsoluteOrigin, RelativeOrigin)
- Transparency (Transparency)
- Locked (Locked)
- Enablement (Enabled)
- Tab Order (TabOrder)
- Tab Stop (TabStop)
- Single Object Treatment (TreatAsIcon)
- Visibility (Visible)

### Changing a Group Name

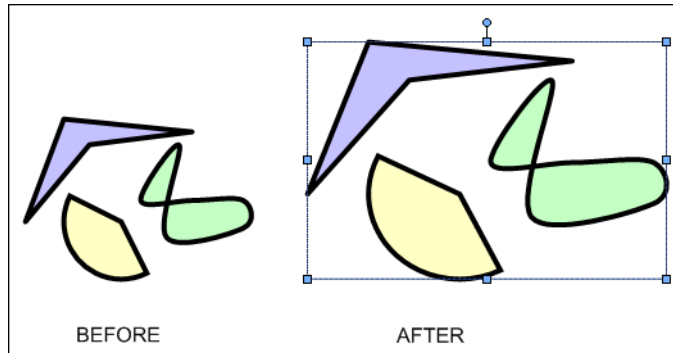
If you change the group name, it has no affect on the contained elements. The contained elements keep their name.

### Changing the Position of a Group

If you change the position of the group, all contained objects are moved with the group. They maintain the relative position to each other, but their absolute positions change.

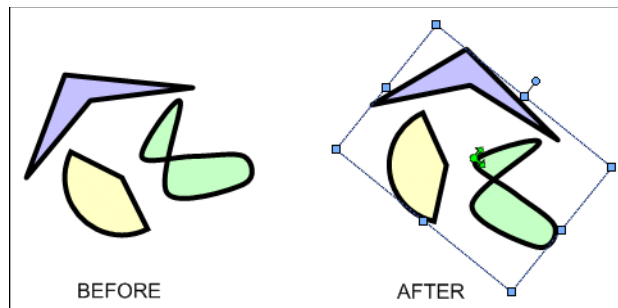
### Changing the Size of a Group

If you change the size of the group, all contained objects are resized proportionally.



### Changing the Orientation of a Group

If you change the angle of the group, all contained objects are rotated with the group around the origin of the group, so that the group remains visually intact.



### Changing the Transparency of a Group

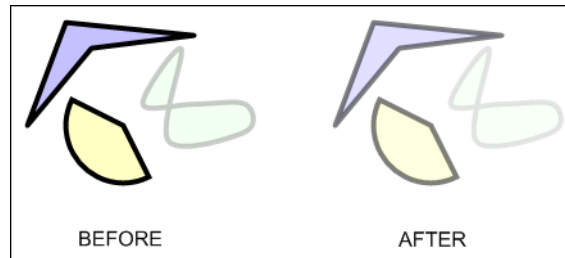
If you increase the transparency of the group, all contained objects appear more transparent, but their own transparency property values do not change. If you change their transparency values, it is in relation to the transparency level of the group.

For example, if you add an element with 80 percent transparency to a group, and then apply 50 percent transparency to the group, the element appears to have 90 percent transparency.

This is calculated as follows:

$$1 - (1 - 0.8) * (1 - 0.5) = 0.9$$

The transparency property values, however, stay unchanged at 80 percent for the element and 50 percent for the group.



### Locking the Group

If you lock the group, it has no affect on the contained elements. You can still edit the contained elements in inline editing mode. You cannot move, resize ,or rotate the group.

### Run-Time Properties of a Group

If you change the run-time properties of a group, the elements do not inherit the property value of the group, but they do inherit the behavior of the group.

For example, if you create a group from elements, some of which have their visibility set to true and some to false, then set the group visibility to false, ALL elements in that group are invisible.

However the Visible property values of the contained elements still maintain their original values (true or false).

### Renaming a Group or its Elements

If you rename an element or a group, the animation references to it are not automatically updated. You must manually change all animation links referencing the old name. For more information, see [Substituting References in Elements](#) on page 351.

## Animations

You can use animations to bind the run-time behavior and appearance of elements to ArchestrA attributes, InTouch tags, custom properties, and other element's properties.

For example, you can bind the vertical fill of a rectangle to an ArchestrA attribute that contains the current level of a tank.

Animations are specific to the selected element and vary between elements of different types.

### Animation Types

There are two types of animations:

- Visualization animations determine the element's appearance, such as blinking, fill style, percent fill horizontal, value display, and so on.
- Interaction animations determine the element's behavior, such as horizontal sliders, user input, and so on.

There are visualization and interaction animations that are specific to certain elements. For example, the `DataStatus` animation is specific to the `Status` element. Element-specific animations also determine element behavior and appearance.

You can configure the following common animation types:

Animation Type	Description
<b>Visibility</b>	Shows or hides the element depending on a value or an expression.
<b>Fill Style</b>	Specifies the interior fill style depending on a discrete or analog expression or one or more conditions.
<b>Line Style</b>	Specifies the style and pattern of the element line depending on a discrete or analog expression or one or more conditions.
<b>Text Style</b>	Specifies the style of the element text depending on a discrete or analog expression or one or more conditions.
<b>Blink</b>	Sets the element to blink invisibly or with specified colors depending on a discrete value or expression.

<b>Animation Type</b>	<b>Description</b>
<b>% Fill Horizontal</b>	Fills the element with color partially from left to right or vice versa, depending on an analog value or expression.
<b>% Fill Vertical</b>	Fills the element with color partially from top to bottom or vice versa, depending on an analog value or expression.
<b>Location Horizontal</b>	Positions the element with a horizontal offset depending on an analog value or expression.
<b>Location Vertical</b>	Positions the element with a vertical offset depending on an analog value or expression.
<b>Width</b>	Increases or decreases the element width depending on an analog value or expression.
<b>Height</b>	Increases or decreases the element height depending on an analog value or expression.
<b>Orientation</b>	Rotates the element by an angle around its center point or any other point depending on an analog value or expression.
<b>Value Display</b>	Shows a discrete, analog, string value, time value, name or expression.
<b>Tooltip</b>	Shows a value or expression as a tooltip when the mouse is moved over the element.
<b>Disable</b>	Disables the element's animation depending on a Boolean value or expression.
<b>User Input</b>	Allows the run-time user to type a Boolean, analog, string, time or elapsed time value that is then assigned to an attribute.

---

<b>Animation Type</b>	<b>Description</b>
<b>Slider Horizontal</b>	Allows the run-time user to drag the element left or right and write back the offset to an analog attribute.
<b>Slider Vertical</b>	Allows the run-time user to drag the element up or down and write back the offset to an analog attribute.
<b>Pushbutton</b>	Writes predetermined values to Boolean or analog references when the user clicks on the element.
<b>Action Scripts</b>	Runs an action script when the run-time user clicks on the element.
<b>Show Symbol</b>	Shows a specified symbol at a specified position when the run-time user clicks on the element.
<b>Hide Symbol</b>	Hides a specified symbol when the run-time user clicks on the element.

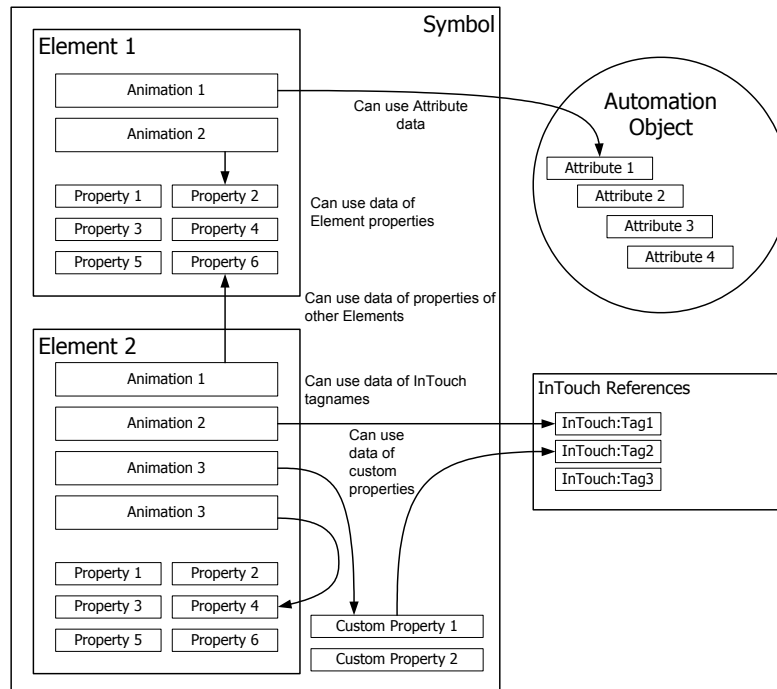
---

## Data Sources for Animations

The data used for animations can come from various sources. You can configure the animation to point at these sources. Animation data can come from:

- Attributes of AutomationObjects.
- Predefined properties of an element or symbol.
- Custom properties of a symbol.
- InTouch tags.

To use the data of an InTouch tag in an ArcestrA symbol, you need to use the special InTouch keyword in the configuration field. The syntax is: `InTouch:TagName`, where *TagName* is a tag that you intend using in the InTouch application into which you are embedding the ArcestrA symbol.





## Animation Capabilities of Groups

By default, a group of elements has limited animation capabilities of its own. For a group you can configure the following animations:

- Blinking
- Enabling/disabling
- Vertical and horizontal location
- Orientation
- Height and width
- Visibility

However, you can set the `TreatAsIcon` property value to `True`. The group is then treated as a single object and you can configure more animations. These animations take precedence over animations defined for the elements within the group.

## Animation States

Some animations have multiple configuration panels.

A state selection panel appears, where you can select the animation state. Depending on what you select, the configuration panel is populated differently. The animation state can be a:

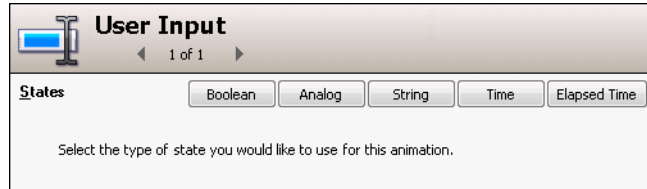
- Data type, where the animation is tied to a specific data type.
- Truth table, where the animation is tied to a set of Boolean conditions.

### Data Type Animation State

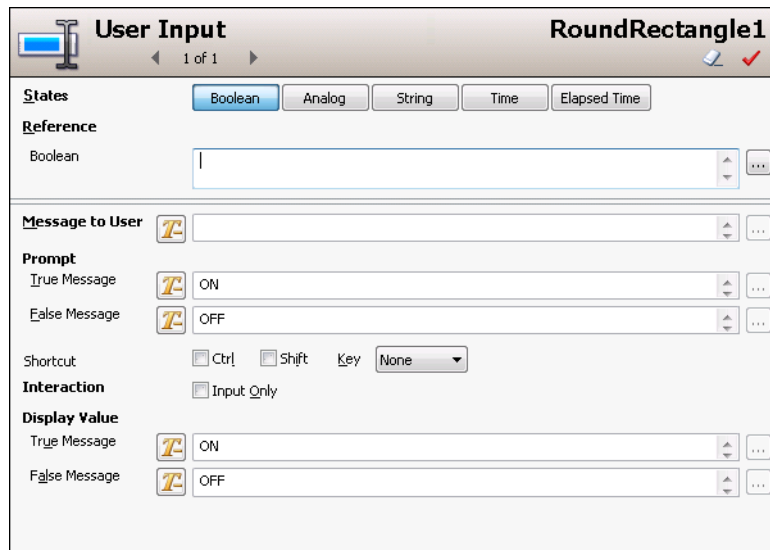
Certain animations support configuration of one or more data types. In the configuration panel of an animation, you can select the data type you want to configure, such as:

- Boolean
- Analog
- String
- Time
- Elapsed Time
- Name

For example, if you select the **User Input** animation link, the **User Input** state selection page appears on the right in the **Edit Animations** dialog box.



A configuration panel appears below the **States** buttons. For example, a configuration panel that is specific to the user input of a Boolean value.



### Truth Table Animation State

Certain animations support the configuration of a truth table. The truth table is a collection of up to 100 Boolean conditions you can configure to determine the output.

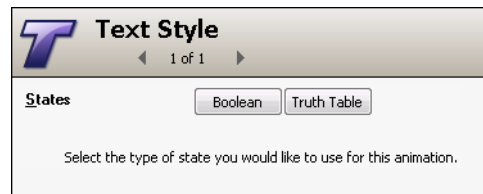
You can configure the default appearance for the case that none of the conditions are fulfilled.

The conditions are evaluated from top to bottom of the list. When the first true condition is met, its assigned appearance is the one used and the condition evaluation stops.

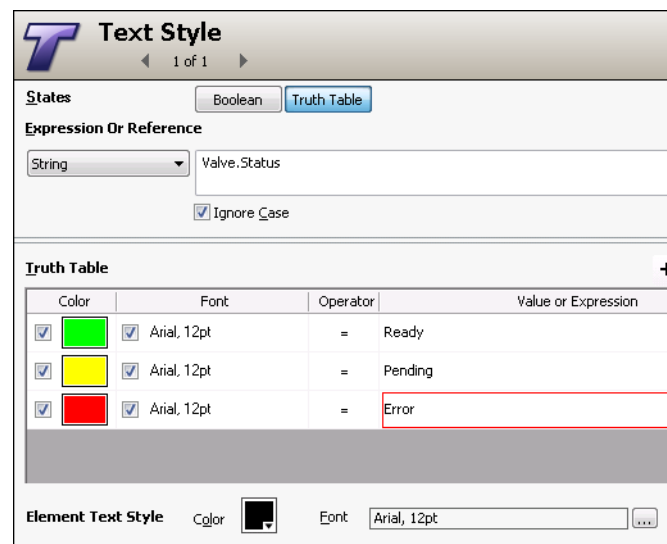
For example, you want a text animation to use a different text color depending on the value of a string attribute, such as a status indicator.

Status indicator	Text color
Ready	Green
Pending	Yellow
Error	Red

If you select the **Text Style** animation link, the **Text Style** state selection page appears on the **Edit Animations** dialog box.



You can click the **Truth Table** button to configure conditions for the appearance of the text style.



By default the text color is black if none of the conditions are fulfilled at run time.

## Embedded Symbols

You can embed symbols from the Graphic Toolbox, AutomationObject templates, and instances into other symbols. This lets you rapidly develop more complex symbols with common components.

For example, you can create a single tank symbol, then embed the tank symbol multiple times in another symbol to create a symbol representing a collection of tanks.

There is no limit to the number of levels of embedding.

## Appearance of Embedded Symbols

Embedded symbols appear in the Elements List. The default name is the same as the source symbol, followed by a numeric sequence.

## Changing Embedded Symbols

After you embed a symbol, you can change its size, orientation or transparency. You can add a limited set of animations to the symbol, such as:

- Visibility
- Blink
- Horizontal and vertical location
- Width and height
- Orientation
- Disable
- Touch Pushbuttons (Discrete Value, Action, Show Window, and Hide Window)

You can configure its public custom properties, if any exist.

You cannot:

- Change the graphic definition of the embedded symbol from within the hosting symbol.
- Embed a symbol contained in an AutomationObject into a symbol that is contained in the Graphic Toolbox.
- Create circular references. A circular reference occurs when one symbol (Symbol A) has embedded within it another symbol (Symbol B) that has embedded within it a symbol that directly or indirect references back to the first symbol (Symbol A).

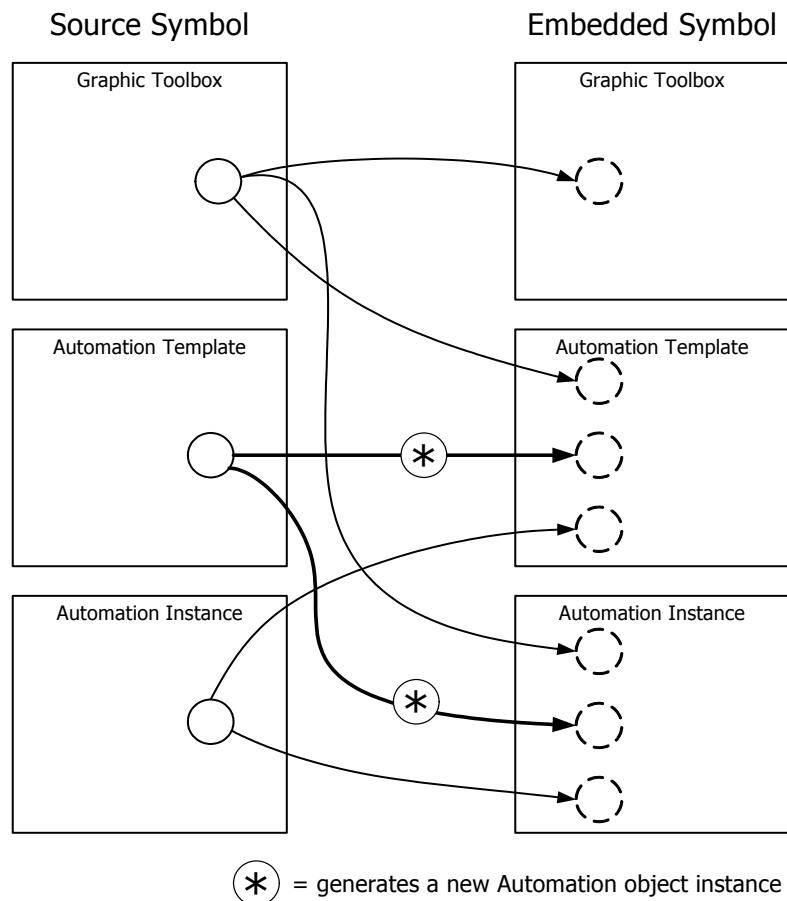
You can, however, change the embedded symbol by changing its source symbol. The changes you make propagate to the embedded symbol.

## Embedding and Instantiation

You can embed Symbols contained in the Graphic Toolbox into symbols contained in the Graphic Toolbox, an AutomationObject template, or an AutomationObject instance.

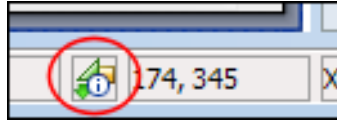
You can embed symbols contained in an AutomationObject template into symbols contained only in other AutomationObjects. When you do so, a new AutomationObject instance is created. You can give it a name, and the new instance inherits the symbol, but does not contain it.

You can only embed symbols contained in an AutomationObject instance into symbols contained in other AutomationObjects. The template or instance inherits the symbol, but does not contain it.



## Symbol Change Propagation

When you make changes and save a symbol, the changes are propagated to any other symbols that embed that symbol. The ArcestrA Symbol Editor shows an icon in the status bar below the canvas that indicates that the source of an embedded symbol changed.



You can accept the change immediately or when you open the symbol again.

---

**Important** You cannot undo/redo the modifications done for a symbol whose source has changed and the propagated change is accepted.

---

When a symbol is changed, its external size can also be changed. ArcestrA symbols support dynamic size propagation and anchor points that let you determine how and if size changes are propagated. For more information about size propagation, see [Size Propagation and Anchor Points](#) on page 47.

If the symbol is hosted by the Graphic Toolbox and edited:

- All symbols hosted by AutomationObject templates and instances that contain embedded instances of this symbol are also updated.
- All embedded instances of this symbol in InTouch WindowMaker are also marked for an update.

If the symbol is hosted by an AutomationObject and edited:

- All symbols hosted by derived AutomationObjects are also updated.
- All embedded instances of this symbol in InTouch WindowMaker hosted by derived AutomationObjects are marked for an update.

## Size Propagation and Anchor Points

To control how size changes are propagated to the embedded instances, the symbol uses an anchor point. By default, the anchor point of the symbol is the center point of all elements on the canvas.

This can be done graphically on the canvas, or by setting anchor position properties in the Properties Editor.

There are two types of anchors:

- Use the `AbsoluteAnchor` property to specify its position as absolute coordinates.
- Use the `RelativeAnchor` property to specify its position as coordinates relative to the symbol center.

When you embed a symbol, the embedded symbol inherits the anchor point in relation to its own center point.

You can also set the `AnchorFixedTo` property. When you make changes to the symbol that affects its size, the `AnchorFixedTo` property determines if the absolute position or relative position of the anchor point is recalculated. This property can have following values:

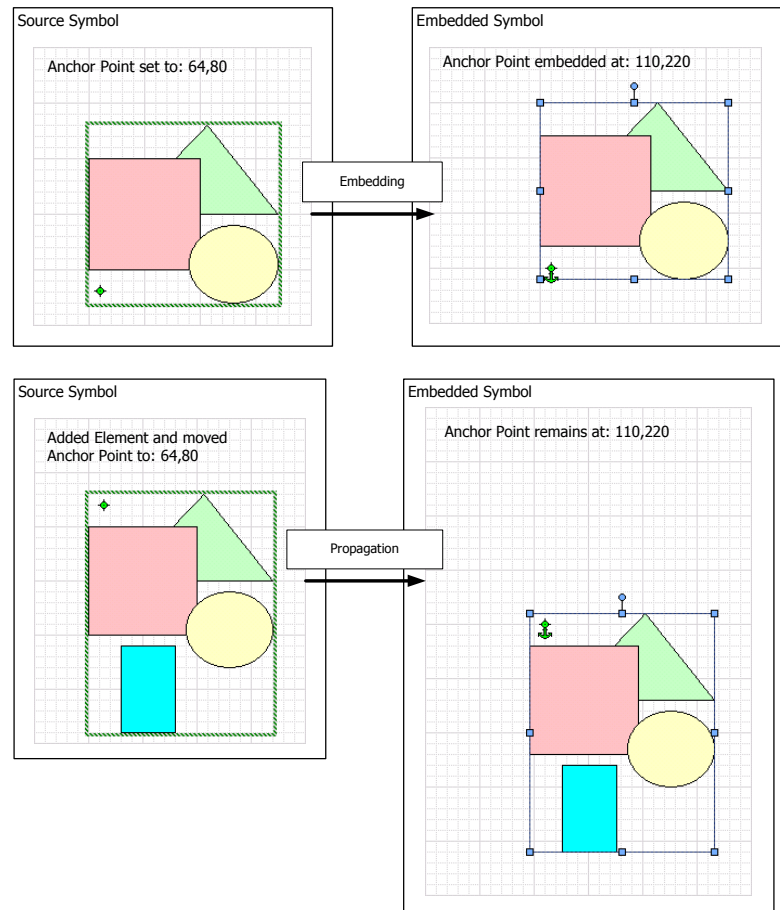
- **Absolute:** The absolute anchor point position is unchanged, and the relative anchor point position is recalculated.
- **Relative:** The absolute anchor point position is recalculated, and the relative anchor point position is unchanged.

---

**Note** When you change the `AbsoluteAnchor` property, the `AnchorFixedTo` property is set to the value `Absolute`. When you change the `RelativeAnchor` property, the `AnchorFixedTo` property is set to the value `Relative`.

---

You can change the position of the anchor point of the symbol. This affects the position of the embedded instances. The anchor points of the embedded instances, however, remain unchanged.



You can change the anchor point of an embedded symbol. This moves the embedded symbol. It does not change the anchor point position in relation to the symbol.

You can resize or rotate the embedded symbol. The anchor point moves in relation to the embedded symbol.

---

**Note** You can also use the AnchorPoint property in the Properties Editor to change the position.

---



## Showing Quality and Status

To show a specified status or quality at run time, you can:

- Use a Status element that shows you an icon. It indicates the status or quality of specified ArcestrA attributes directly or those used indirectly in elements.
- Change the appearance of animated elements based on the status and quality of ArcestrA attributes they use.

### Showing Quality and Status with the Status Element

The Status element cannot monitor attributes of:

- Elements that are not in the same hierarchy level in the Elements List.
- Elements that use the attributes in scripts.
- Elements that are invisible at run time.

For more information on how to configure status on an element, see [Configuring Animation for a Status Element](#) on page 328.

For more information on how to configure the appearance of a status element, see [Setting the Appearance of a Status Element](#) on page 210.

### Showing Quality and Status by Overriding

You can override the appearance of animations depending on its configured attributes by:

- Overriding the animation or changing the appearance of the element.
- Drawing an outline around the element.

This also applies to:

- Elements contained in groups.
- Elements in symbols embedded in other symbols.

This does not apply to:

- Elements that use the monitored attribute in scripts.
- Elements that are invisible at run time.

For more information, see [Overriding Element Appearance Depending on Quality and Status of its Attributes](#) on page 212.



# Chapter 2

## Comparing WindowMaker and ArcestrA Symbol Editor

You can use the ArcestrA Symbol Editor to do most of the tasks you do in InTouch WindowMaker. You can also use many of the same shortcut keys.

### Differences between WindowMaker and the ArcestrA Symbol Editor

The ArcestrA Symbol Editor has features that are not available in InTouch WindowMaker, such as:

- Additional elements.
- Additional and enhanced appearance of the elements.
- Additional and enhanced design-time functionality.

## Elements

Elements are the graphical objects you use to create an ArcestrA Symbol. The ArcestrA Symbol Editor provides elements that are not available in InTouch WindowMaker, such as:

- Curves and closed curves.
- Arcs, pies, and chords defined by two or three points.
- Status elements to conditionally show an icon depending on quality and status of attribute data.
- Path graphics that you create by joining line-based elements together to form a new closed element.
- Windows common controls, such as the Calendar control and Date Time Picker control.

## Appearance

The ArcestrA Symbol Editor extends the InTouch graphic configuration. For example, you can use:

- Gradients for line, fill, and text color.
- Patterns for line, fill, and text color.
- Textures for line, fill, and text color.
- Partial transparency.
- Fill behavior in relation to a symbol or screen.

## Enhanced Functionality

The ArcestrA Symbol Editor provides a entire array of enhancements to make your life easier when creating visualization for your manufacturing environment.

### Usability Enhancements

The ArcestrA Symbol Editor makes it easy to select and configure elements. You can:

- Select elements from a list as well as from the canvas. This lets you select elements beneath others without having to move them.
- View and change properties and animation (links) of an element by simply selecting it on the canvas.
- Edit elements contained in groups and path graphics without having to break the group or path graphic. This is called inline editing.

### Style Replication

Using the Format Painter, you can simply apply the style of one element with one click to another element, even to an element of a different type.

### Animation Replication

Using the ArcestrA Symbol Editor you can copy, cut, and paste animations from one element to another element, even to an element of a different type.

### Element Positioning

The ArcestrA Symbol Editor extends the positioning feature of InTouch WindowMaker and lets you:

- Distribute elements equally in horizontal or vertical direction.
- Make elements same horizontal and/or vertical size.
- Increase or decrease horizontal or vertical space.
- Remove horizontal or vertical space between elements.
- Lock an element so that you do not accidentally move or edit it.
- Rotate any element at design time by any angle around a center of rotation.
- Apply resizing and rotating to multiple elements at the same time.
- Move the z-order of an element one level backward or forward.
- Align text within text boxes and buttons.

### Group Functionality

The ArcestrA Symbol Editor uses the concept of groups instead of the cell and symbol concepts of InTouch WindowMaker. You can:

- Embed groups within groups.
- Edit individual elements within a group (or an embedded group) without breaking up the group.
- Easily remove elements from or add elements to existing groups.

### Extensibility with Custom Properties

You can add custom properties to a symbol or embedded symbol. You can connect custom properties to AutomationObject attributes, element properties, and even InTouch tags. You can use the custom properties as you would with any pre-defined property at design time and run time.

### Miscellaneous Enhancements

Using the Arcestra Symbol Editor, you can:

- Access the properties of the elements and custom properties of the symbol through scripting.
- Set the tab order of the elements.
- Use line end styles, such as arrows.
- Dynamically disable specific animations from elements without losing the configuration information.
- Use image meta files and other image formats.
- Use anti-aliasing to improve how the symbol is shown.

## Procedures for Common WindowMaker Tasks and Techniques

Most of the configuration that you do in InTouch WindowMaker can be easily done in the Arcestra Symbol Editor. There are some differences between and similarities of graphics, animations, and scripts.

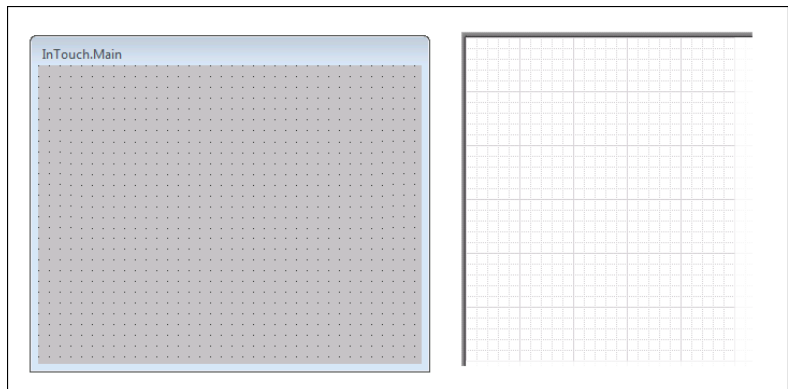
### Using Graphics

You can use the Arcestra Symbol Editor in basically the same way as you use InTouch WindowMaker. The Arcestra Symbol Editor includes a drawing area on which you can place graphical objects to construct a visual representation of production processes and to provide an interface between a human and a machine.

Some objects you use in InTouch do not exist in the Arcestra Symbol Editor, such as ActiveX controls and some Wizards. Their functionality is replaced other controls that are more powerful and integrate better into the Arcestra environment.

## Using the Drawing Area

The drawing area of the ArcestrA Symbol Editor is called the canvas. You use it like an InTouch window. Its maximum size is 2,000 by 2,000 pixels.



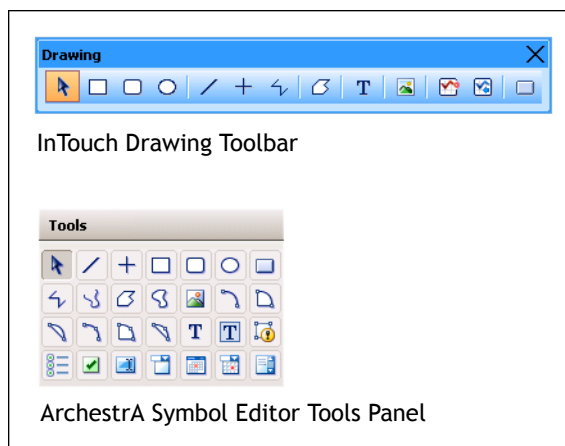
## Setting the Drawing Area Color

You cannot set the drawing area color in the ArcestrA Symbol Editor. The drawing area color is transparent and inherits the color of the environment that the symbol is embedded into.

If you embed an ArcestrA Symbol into an InTouch window, the area between the elements adopts the color of the InTouch window.

## Using Basic Objects

InTouch basic objects such as rectangles, ellipses, and polylines can be drawn in very similar way in the ArcestrA Symbol Editor. The basic objects are called elements in the ArcestrA Symbol Editor.



### Using Complex Objects

InTouch objects such as ActiveX controls, Wizards, cells, and symbols do not exist in the ArcestrA Symbol Editor.

You can, however, import SmartSymbols into an ArcestrA symbol. When you import a SmartSymbol, the elements and animations of a SmartSymbol are converted.

In the ArcestrA Symbol Editor, you can create groups of elements. Groups maintain the properties of the contained individual elements. You can set the TreatAsIcon property of a group to change the behavior of the group.

### Using Wizards

You cannot import InTouch Wizards to an ArcestrA Symbol or into the Graphic Toolbox. Instead, use:

- The ArcestrA Symbol Library, which you can import into the Graphic Toolbox.
- Windows controls that are part of the Toolbox. You can use:
  - Radio button groups
  - Check boxes
  - Edit boxes
  - Combo boxes
  - Calendar control
  - DateTime picker
  - List boxes



## Using Animations

You can use animations in the ArcestrA Symbol Editor to set run-time behavior of the symbols as you would in InTouch WindowMaker. You can configure one or more animations for an element or symbol. The data can come from various sources.

### Configuring Data Sources

In InTouch WindowMaker, you use the Tagname Dictionary to define variables that hold values. In the ArcestrA Symbol Editor, data sources can be:

- ArcestrA AutomationObject attributes.
- Custom properties and inherited properties of the symbol.
- InTouch tags themselves. The ArcestrA Symbol Editor uses a special InTouch reference you can use to directly connect to InTouch tags.

### Using Data Types

ArcestrA Symbols use the ArcestrA data types, which are different than InTouch data types.

The following table shows you the data types of both and how they correspond to each other:

InTouch	ArcestrA	Description
Discrete	Boolean	Boolean value. For example: 0 or 1
Integer	Integer	Integer value. For example: -4, 7, or 22
Real	Float or Double	Float or double value with different precision. For example: 3.141, -5.332, or 1.343e+17
Message	String	String value. For example: "Hello World"
n/a	DateTime	Datetime value. For example: "04/13/2006 04:03:22.222 AM"

InTouch	ArcestrA	Description
n/a	ElapsedTime	<p>Float value that represents a time that has elapsed in seconds. It is shown often in the following format, but is stored as a float value.</p> <p>[–][DDDDDD] [HH:MM:]SS[.ffffff]</p> <p>Values are as follows:</p> <ul style="list-style-type: none"> <li>• DDDDDD is from 0 to 999999</li> <li>• HH is from 0 to 23</li> <li>• MM is from 0 to 59</li> <li>• SS is from 0 to 59</li> <li>• fffffff is fractional seconds to right of the decimal</li> </ul> <p>Elapsed time can be positive or negative.</p>
n/a	InternationalizedString	<p>A special string data type that can store special characters.</p>

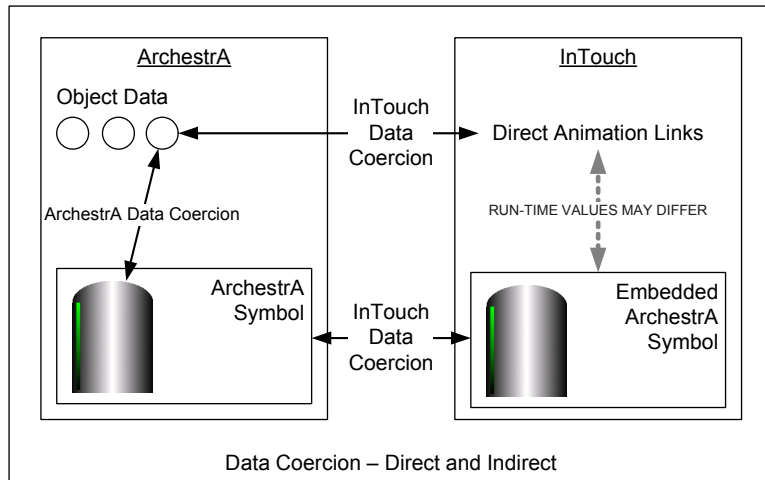
You can configure ArcestrA Symbols to retrieve data from the Galaxy.

When the source data type is different than the data type it is used for, the data is coerced according to the rules of ArcestrA data coercion and a string value of “-10” is coerced to “True” in the animation.

If you embed this ArcestrA Symbol into an InTouch window, the data type of the animation link is coerced according to the InTouch data coercion. The embedded ArcestrA Symbol shows “True” in the InTouch HMI.

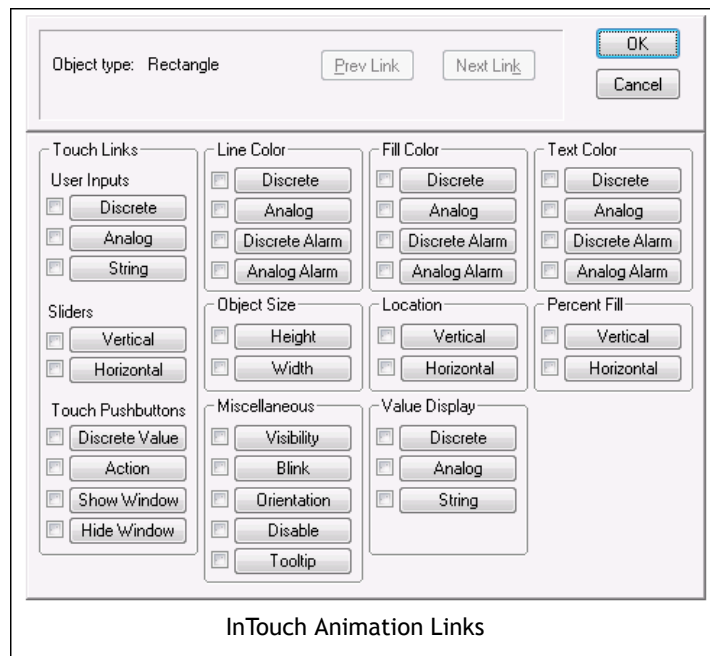
However, if you directly create an discrete animation display link in the InTouch HMI that points at the original data source, the resulting value can be different.

In this example the string value “-10” is shown as “False” in the InTouch HMI.

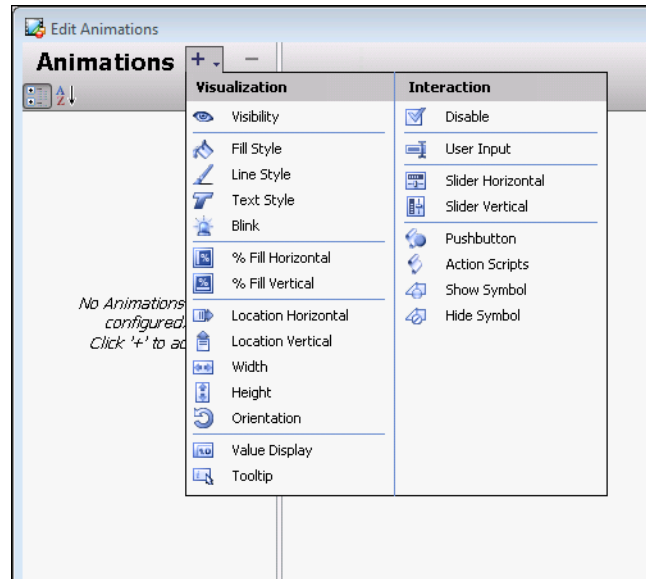


## Using Animations

You configure InTouch animations using the **Animation Links** dialog box. You can open this dialog box by double-clicking an InTouch object.



You configure animations in the ArcestraA Symbol Editor using the **Edit Animations** dialog box, which is normally opened by double-clicking an element.



Animation Links in ArcestraA Symbol Editor

Some of the animation types are different and others have been grouped to simplify configuration. Use the following table to find the equivalent animation type in the ArcestraA Symbol Editor:

InTouch Animation	ArcestraA Symbol Editor Animation
User Inputs - Discrete	User Input - Boolean
User Inputs - Analog	User Input - Analog
User Inputs - String	User Input - String
Sliders - Vertical	Slider Vertical
Sliders - Horizontal	Slider Horizontal
Touch Pushbuttons - Discrete Value	Pushbutton - Boolean
Action	Action Scripts
Show Window	(not supported)
Hide Window	(not supported)
Line Color - Discrete	Line Style - Boolean
Line Color - Analog	Line Style - Truth Table
Line Color - Discrete Alarm	converted to Line Style

<b>InTouch Animation</b>	<b>Archestra Symbol Editor Animation</b>
Line Color - Analog Alarm	converted to Line Style
Fill Color - Discrete	Fill Style - Boolean
Fill Color - Analog	Fill Style - Truth Table
Fill Color - Discrete Alarm	converted to Fill Style
Fill Color - Analog Alarm	converted to Fill Style
Text Color - Discrete	Text Style - Boolean
Text Color - Analog	Text Style - Truth Table
Text Color - Discrete Alarm	converted to Text Style
Text Color - Analog Alarm	converted to Text Style
Object Size - Height	Height
Object Size - Width	Width
Location - Vertical	Location Vertical
Location - Horizontal	Location Horizontal
Percent Fill - Vertical	% Fill Vertical
Percent Fill - Horizontal	% Fill Horizontal
Miscellaneous - Visibility	Visibility
Miscellaneous - Blink	Blink
Miscellaneous - Orientation	Orientation
Miscellaneous - Disable	Disable
Miscellaneous - Tooltip	Tooltip
Value Display - Discrete	Value Display - Boolean
Value Display - Analog	Value Display - Analog
Value Display - String	Value Display - String

## Using Scripts

You can configure scripts in the ArcestrA Symbol Editor the same way as you do in InTouch WindowMaker. There are, however, some small differences:

InTouch Script	ArcestrA Symbol Editor Script
Application Script	(not available)
Window Script	Symbol Predefined Script
Key Script	Action Script animation with a key trigger
Condition Script	Symbol Named Script with an OnTrue, OnFalse, WhileTrue or WhileFalse trigger
Data Change Script	Symbol Named Script with a DataChange trigger
QuickFunction	(not available)
ActiveX Event Script	(not available)
Action Script	Action Script animation

### Using Application Scripts

In the InTouch HMI, application scripts are triggered:

- One time when the application starts up in WindowViewer.
- Periodically when the application runs in WindowViewer.
- One time when the application shuts down in WindowViewer.

ArcestrA Symbols correspond to InTouch applications and let you configure predefined scripts directly associated with symbol events. Such events are:

- On Show
- While Showing
- On Hide

### Using Key Scripts

You cannot use key scripts in the ArcestrA Symbol Editor, but you can associate an element with an action script that is activated by a key combination.

## Using Condition Scripts

You can configure a script that is run when a condition is fulfilled by using the Symbol Scripts feature. It lets you define triggers that run a script when a value or expression:

- Is fulfilled. (WhileTrue)
- Becomes fulfilled. (OnTrue)
- Is not fulfilled. (WhileFalse)
- Becomes no longer fulfilled. (OnFalse)

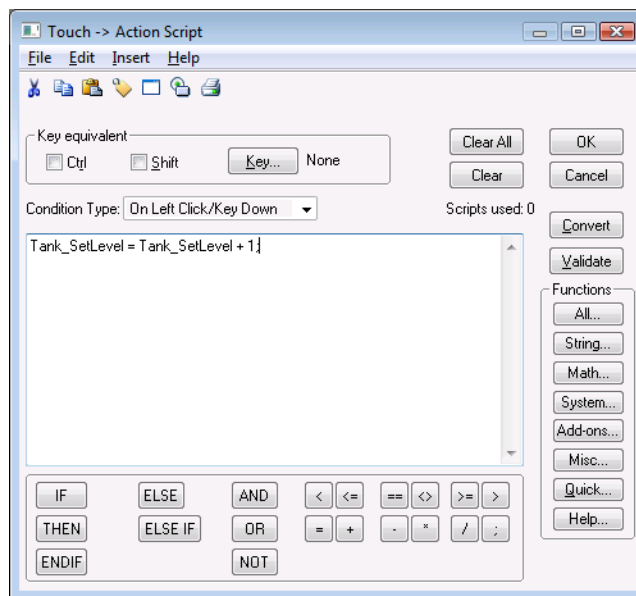
## Using Data Change Scripts

You can configure a script that is run when a value or expression changes by using the Symbol Scripts feature. It lets you define a trigger that runs a script when a value or expression changes.

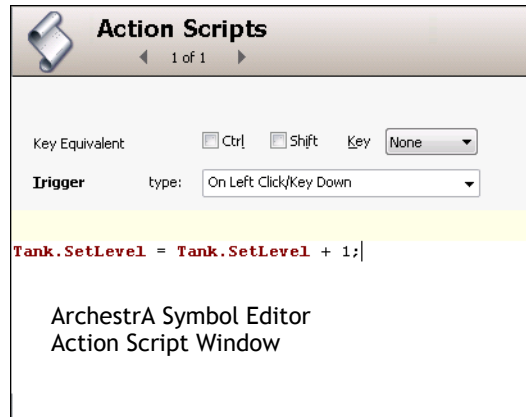
## Using Action Scripts

You can configure Action Scripts in the ArchestraA Symbol Editor the same way as you would in InTouch WindowMaker. When the run-time user interacts with an element, such as with the mouse or by pressing a key, an action script can run.

You use the InTouch action script window to create action scripts.



You use the ArcestrA Symbol Editor action script window to create action scripts.



You can configure action scripts for individual elements or for the entire symbol.

You can use many of the predefined functions of InTouch WindowMaker in the ArcestrA Symbol Editor. For a complete list of InTouch predefined functions that can be used with ArcestrA Symbols, see [Importing Action Scripts](#) on page 415.

Other InTouch script types, such as application scripts and key scripts, can be configured with ArcestrA AutomationObjects.



# Chapter 3

## Managing Symbols

This section describes ArcestrA symbols, how they are stored in the ArcestrA environment, and how they are managed from the IDE.

### About Symbols

ArcestrA symbols are graphical symbols you use to visualize data in an InTouch application.

You manage ArcestrA symbols from the IDE to::

- Create a new symbol.
- Edit a symbol with the ArcestrA Symbol Editor.
- Organize symbols within the Graphic Toolbox.
- Duplicate symbols.
- Import and export symbols.
- Delete a symbol.
- Configure security for a symbol's operations.
- Open the symbol in read-only mode with the ArcestrA Symbol Editor.

## Creating a New Symbol

You can create a new symbol in:

- The Graphic Toolbox for generic symbols that you frequently use in different situations. For example, a valve symbol.
- The **Graphics** tab of an AutomationObject template. Do this if you want to re-use the symbol in combination with the object functionality. An example is a symbol representing a specific tank and your production facility has multiple tanks.
- The **Graphics** tab of an AutomationObject instance. Do this if you are unlikely to re-use the symbol in any other situation.

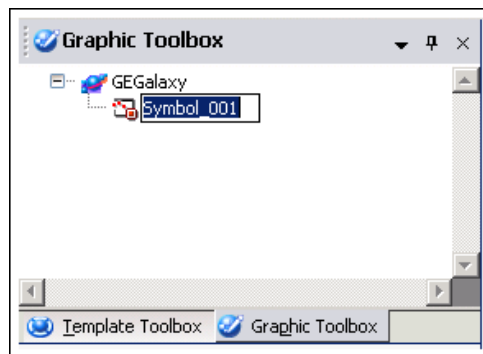
## Creating Symbols in the Graphic Toolbox

You can create a new symbol in the Graphic Toolbox. It is then listed in the Graphic Toolbox with a proposed default name. You can:

- Rename the symbol.
- Move the symbol.
- Edit the symbol with the ArchestrA Symbol Editor.

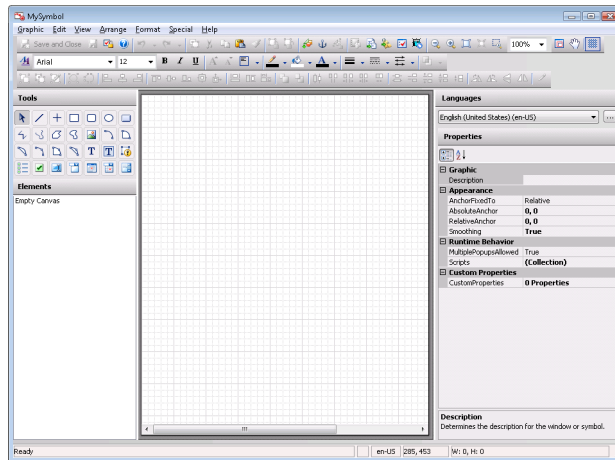
### To create a new symbol from the IDE

- 1 On the **Galaxy** menu, point to **New**, and then click **Symbol**. The Graphic Toolbox appears and a new symbol is listed.



**Note** You can also press **CTRL + SHIFT + S** to create a new ArchestrA Symbol.

- 2 Rename the symbol. Names must be unique within the entire Graphic Toolbox hierarchy. Valid characters for symbol names include alphanumeric characters, #, and \_ (underscore). Symbol names cannot contain spaces and the symbol name cannot begin with the \$ character.
- 3 Double-click the symbol name. The ArchestrA Symbol Editor appears.



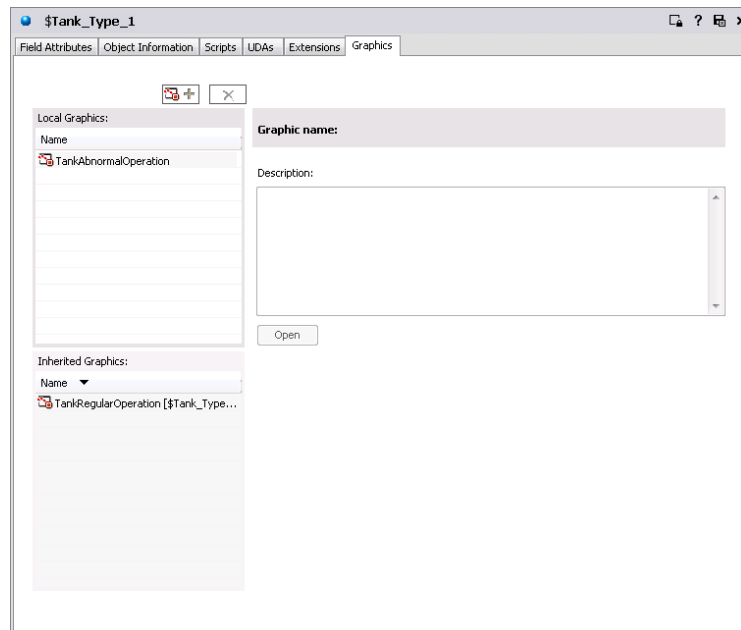
- 4 Draw the symbol. For specific information about using the drawing tools, see Working with Graphic Elements on page 93.

## Creating Symbols in AutomationObject Templates

You can create a symbol from the **Graphics** tab in an AutomationObject template. Creating a symbol this way automatically associates the new symbol with the AutomationObject.

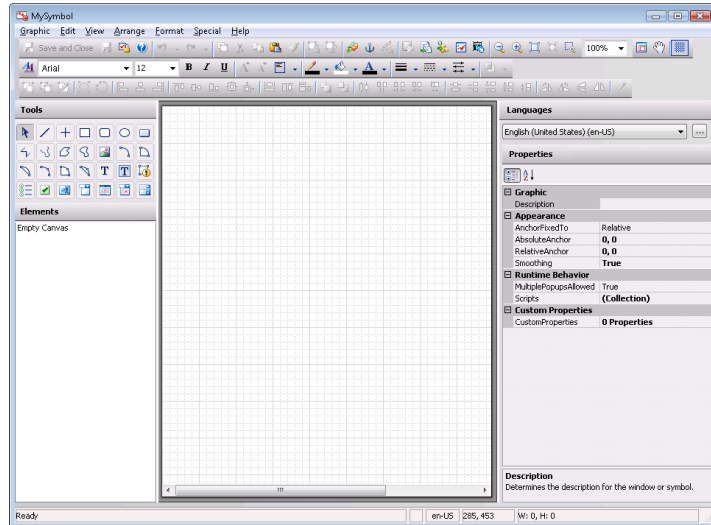
**To create a new symbol for an AutomationObject template**

- 1 Open the AutomationObject template. Click the **Graphics** tab. Any local and inherited symbols are listed.



- 2 Click the **New Symbol** icon. Give the new symbol a name. Names must be unique. Valid characters for symbol names include alphanumeric characters, \$, #, and \_ (underscore). Symbol names cannot include spaces and the symbol name cannot begin with the \$ character.

- 3 If needed, type the description of the symbol in the **Description** box.
- 4 Click the symbol name and click **Open**. The ArchestraA Symbol Editor appears.



- 5 Draw the symbol. For specific information about using the drawing tools, see Working with Graphic Elements on page 93.

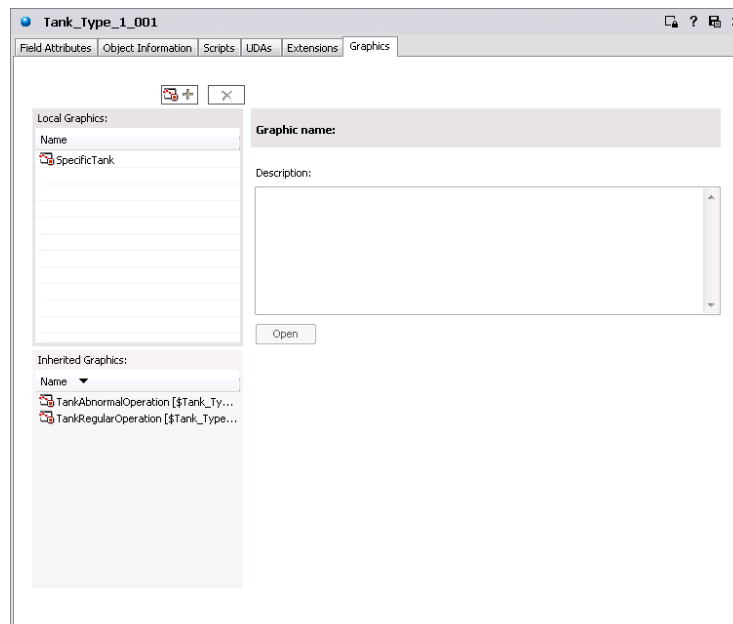
## Creating Symbols in AutomationObject Instances

You can create a symbol from the **Graphics** tab in an AutomationObject instance. Creating a symbol this way automatically associates the new symbol with the AutomationObject instance.

**Note** AutomationObjects can also inherit symbols from their parent template. You can only view an inherited graphic in read-only mode. Inherited graphics cannot be removed or edited.

### To create a new symbol for an AutomationObject instance

- 1 Open the AutomationObject instance. Click the **Graphics** tab. Any local and inherited symbols are listed.



- 2 Click the **New Symbol** icon. Give the new symbol a name. Names must be unique. Valid characters for symbol names include alphanumeric characters, \$, #, and \_ (underscore). Symbol names cannot include spaces and the symbol name cannot begin with the \$ character.
- 3 If needed, type the description of the symbol in the **Description** box.

- 4 Select the symbol name, and then click **Open**. The ArcestrA Symbol Editor appears.
- 5 Draw the symbol. For specific information about using the drawing tools, see *Working with Graphic Elements* on page 93.

## Opening Symbols for Editing

You can start the ArcestrA Symbol Editor from a symbol:

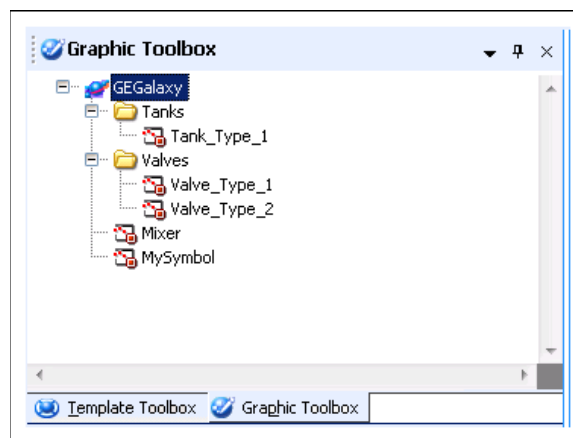
- Contained in the Graphic Toolbox.
- Contained in an AutomationObject template or instance.
- Embedded in an InTouch window.

You check out a symbol when you open it for editing with the ArcestrA Symbol Editor. No other user can edit the symbol while you have it checked out.

You can open multiple instances of the ArcestrA Symbol Editor at the same time. However, you cannot edit the same symbol in multiple instances of the ArcestrA Symbol Editor.

### To edit a symbol in the Graphic Toolbox

- 1 Open the Graphic Toolbox.
- 2 Browse to the symbol you want to edit.



- 3 Double-click the symbol. The ArcestrA Symbol Editor appears.
- 4 Edit the symbol. For specific information about using the drawing tools, see *Working with Graphic Elements* on page 93.
- 5 Click **Save and Close**. The ArcestrA Symbol Editor closes and the updated symbol is checked in.

**To edit a symbol contained in an AutomationObject**

- 1 Open the AutomationObject.
- 2 Click the **Graphics** tab.
- 3 Select the symbol to edit and click **Open**. The ArcestrA Symbol Editor appears.
- 4 Edit the symbol. For specific information about using the drawing tools, see Working with Graphic Elements on page 93.
- 5 Click **Save and Close**. The ArcestrA Symbol Editor closes and the updated symbol is checked in. Depending on the AutomationObject, a confirmation message may appear. Click **Yes** to save.

**To edit a symbol that is embedded into an InTouch window**

- 1 In WindowMaker, open the InTouch window that contains the embedded symbol.
- 2 Right-click the embedded symbol to edit, point to **ArcestrA Graphic “Symbolname”**, and then click **Edit Symbol**. The ArcestrA Symbol Editor appears.
- 3 Edit the symbol. For specific information about using the drawing tools, see Working with Graphic Elements on page 93.
- 4 Click **Save and Close**. The ArcestrA Symbol Editor is closed and the updated symbol is checked in.

---

**Note** To leave the symbol checked out, click **Keep Checked Out** in the ArcestrA Symbol Editor. This ensures that no other user can check out your symbol for editing.

---



## Organizing Symbols in the Graphic Toolbox

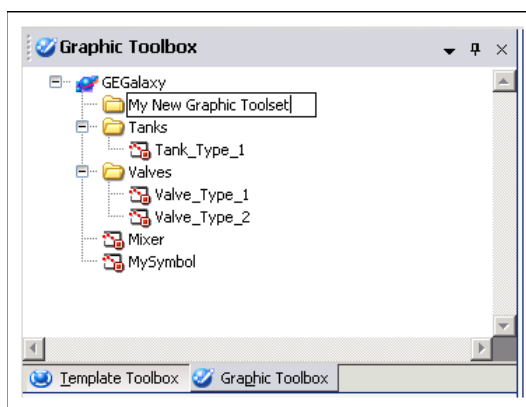
You use the Graphic Toolbox to organize your symbols by creating a folder hierarchy as you would with files and folders in Windows Explorer. You can move symbols around within the folder hierarchy. These folders are called Graphic Toolsets.

### Creating Graphic Toolsets in the Graphic Toolbox

You can create Graphic Toolsets in the Graphic Toolbox to organize your symbols. For example you can create a Graphic Toolset called “Valves” to store different valve symbols.

#### To create a Graphic Toolset in the Toolbox

- 1 Open the Graphic Toolbox.
- 2 Select the Graphic Toolset under which you want to create a new Graphic Toolset. Select the Galaxy name if there are currently no Graphic Toolsets.
- 3 On the **Galaxy** menu, point to **New**, and then click **Graphic Toolset**. A new Graphic Toolset is created with a default name.
- 4 Rename the new Graphic Toolset as needed.



## Moving Symbols between Graphic Toolsets

You can move symbols from one Graphic Toolset in the Graphic Toolbox to another. Moving symbols between Graphic Toolsets does not affect its functionality.

### To move symbols between Graphic Toolsets in the Graphic Toolbox

- 1 Open the Graphic Toolbox.
- 2 Locate the symbol you want to move.
- 3 Do either of the following:
  - Drag the symbol to the Graphic Toolset you want to place it in. The symbol moves to the new Graphic Toolset.
  - To move the symbol to the top level in the Graphic Toolset hierarchy, drag the symbol to the Galaxy name icon.

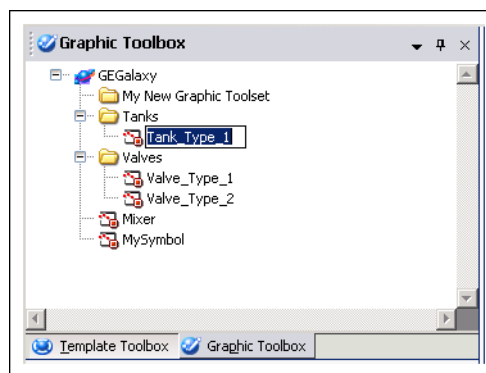
## Renaming Symbols

You can rename a symbol at any time. Renaming a symbol does not affect its functionality.

Symbol names must be unique within the entire hierarchy of the Graphic Toolbox.

### To rename a symbol in the Graphic Toolbox

- 1 Open the Graphic Toolbox.
- 2 Select the symbol you want to rename.
- 3 On the **Edit** menu, click **Rename**. The symbol name is in edit mode.



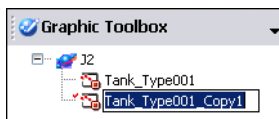
- 4 Type a new unique name for the symbol and click **ENTER**.

## Copying Symbols

You can create copies of symbols in the Graphic Toolbox. The copies are suffixed with “\_Copy1”, “\_Copy2”, and so on.

### To create a copy of a symbol

- 1 Select the symbol you want to copy.
- 2 On the **Edit** menu, click **Duplicate**. A copy of the symbol is created.



- 3 If needed, type a new name for the symbol.

## Renaming Graphic Toolsets

You can rename a Graphic Toolset at any time. Renaming a Graphic Toolset does not affect the functionality of any symbols it contains.

### To rename a Graphic Toolset in the Graphic Toolbox

- 1 Open the Graphic Toolbox.
- 2 Select the Graphic Toolset you want to rename.
- 3 On the **Edit** menu, click **Rename**. The Graphic Toolset name is in edit mode.
- 4 Type a new unique name for the Graphic Toolset and click **ENTER**.

## Deleting Graphic Toolsets

You can delete a Graphic Toolset in the Graphic Toolbox at any time.

You can only delete Graphic Toolsets that do not contain any symbols. Move the symbols to another Graphic Toolset or delete them before deleting the Graphic Toolset.

### To delete a Graphic Toolset in the Graphic Toolbox

- 1 Open the Graphic Toolbox.
- 2 Select the Graphic Toolset you want to delete.
- 3 On the **Edit** menu, click **Delete**. When a message appears, click **Yes**.

## Moving Graphic Toolsets

You can move a Graphic Toolset within the Graphic Toolset hierarchy of the Graphic Toolbox. If you move a Graphic Toolsets, all symbols and Graphic Toolsets it contains are also moved.

### To move a Graphic Toolset in the Graphic Toolbox

- 1 Open the Graphic Toolbox.
- 2 Locate the Graphic Toolset you want to move.
- 3 Do either of the following:
  - Drag it to the Graphic Toolset you want to place it in. The Graphic Toolset is moved to the new Graphic Toolset.
  - To move the Graphic Toolset to the top level in the Graphic Toolset hierarchy, drag the Graphic Toolset to the Galaxy name icon.

## Customizing Graphic Toolsets

You can hide or show Graphic Toolsets after creating them. You can do this for selected Graphic Toolsets or for all Graphic Toolsets.

### To customize Graphic Toolsets

- 1 Open the Graphic Toolbox.
- 2 Right-click the Galaxy icon in the Graphic Toolbox, and then click **Customize Toolsets**. The **Customize Toolsets** dialog box appears.
- 3 Do one or more of the following:
  - To hide Graphic Toolsets, clear the check box next to the Toolsets you want to hide.
  - To show Graphic Toolsets, check the check box next to the Toolsets you want to show.
  - To select or clear all Toolsets, click **Check All** or **Uncheck All**.
- 4 Click **Close**. The selected Graphic Toolsets are shown or hidden, depending on your settings.

## Importing and Exporting Symbols as ArcestrA Object Files

You can import and export symbols as .aaPKG ArcestrA AutomationObject files.

### Importing Symbols

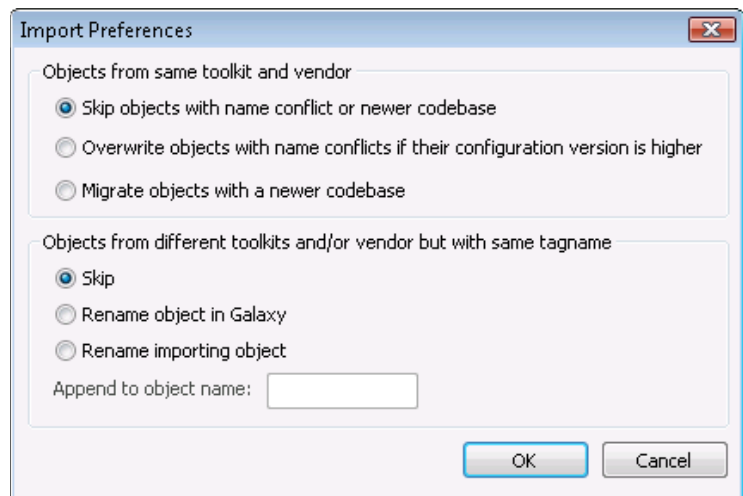
You can import symbols and graphic toolsets from a symbol .aaPKG file.

When you import templates or instances that contain symbols, the symbols are imported with the template or instance.

When you import all AutomationObjects, the contained symbols and the symbols in the graphic toolsets are also imported.

#### To import symbols from a symbol .aaPKG file

- 1 On the **Galaxy** menu, point to **Import**, and then click **Object(s)**. The **Import AutomationObject(s)** dialog box appears.
- 2 Select one or more symbol .aaPKG files you want to import and click **Open**. The **Import Preferences** dialog box appears.



- 3 Select the appropriate options for the import and click **OK**. The symbols and graphic toolsets are imported.
- 4 Click **Close**.

## Exporting Symbols

You can export symbols to a symbol .aaPKG file.

When you export templates or instances that contain symbols, the symbols are exported with the template or instance.

When you export all AutomationObjects, the contained symbols and the symbols in the graphic toolsets are also exported.

When you export an AutomationObject that contains symbols, if these symbols contain embedded ArcestrA Symbols from the Graphic Toolbox, they are exported along with the symbols associated with the AutomationObject.

If the symbols contain other AutomationObjects, the symbols, including any embedded symbols from the Graphic Toolbox contained in them, and their parent AutomationObject are not exported with the AutomationObject. They remain as references. Upon import, the system tries to reestablish the connection with those exact references in the target Galaxy.

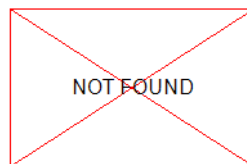
### To export symbols to a symbol .aaPKG file

- 1 In the Graphic Toolbox, select the symbols that you want to export.
- 2 On the **Galaxy** menu, point to **Export**, and then click **Object(s)**. The **Export Automation Object(s)** dialog box appears.
- 3 Browse to the save location and type a name for the symbol .aaPKG file.
- 4 Click **Save**. The symbols and their toolset folders are exported.
- 5 Click **Close**.

## Deleting a Symbol

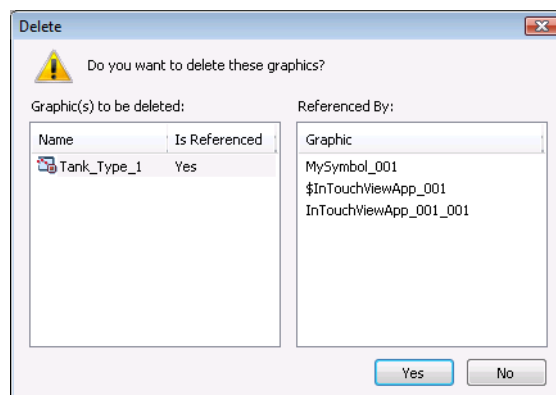
You can delete a symbol that you no longer need. Deleting a symbol removes it completely from the Application Server. You can delete a symbol from the Graphic Toolbox or from an AutomationObject in the ArchestrA Symbol Editor.

- When you delete a symbol, you are shown where the symbol is used. This lets understand the impact of deleting the symbol before you actually delete it.
- You cannot delete symbols that someone else has open for editing or left checked out.
- If you delete a symbol from an AutomationObject, the symbol still appears to other users until you check in the AutomationObject.
- If you delete a symbol that is embedded in another symbol or in an InTouch window, it shows a Not Found message.



### To delete a symbol

- 1 Do one of the following:
  - Open the Graphic Toolbox.
  - Open the AutomationObject with the symbols you want to delete. Click the **Graphics** tab.
- 2 Select the symbol you want to delete and click **Delete**. The **Delete** dialog box appears.



- 3 Review the places the symbol is being used, and then click **Yes**.

## Configuring Security for Symbols

You can set IDE security permissions so that at design time, the user cannot:

- Import or export symbols.
- Create, modify, or delete symbols in the Graphic Toolbox.
- Create, modify, or delete symbols in any AutomationObject template.
- Create, modify, or delete symbols in any AutomationObject instance.
- Create, modify, or delete View Applications, such as InTouchView Applications.
- Deploy or undeploy View Applications, such as InTouchView Applications.
- Edit the configuration of the quality and status display.

To restrict a user, the user must be assigned to a role and the permissions must be assigned to that role, and security must be enabled. For more information on how to configure security, users, and roles, see the *Application Server User's Guide*.

If the user attempts to export a symbol without appropriate permissions, the message “User doesn't have permission to export graphics object” appears.

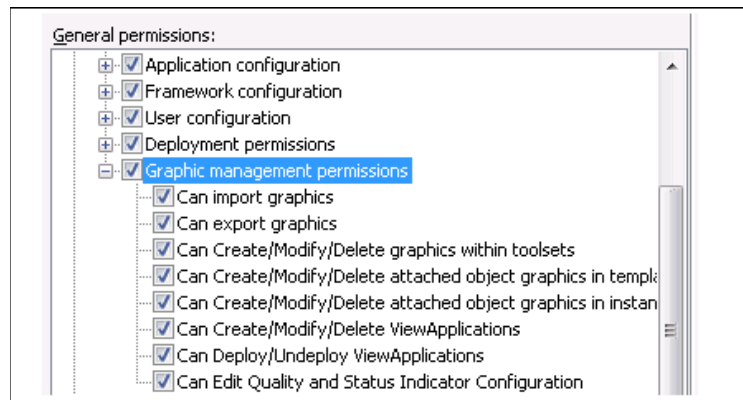
If the user attempts to import a symbol without appropriate permissions, the message “User doesn't have permission to import graphics object” appears.

### To configure security for symbols

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Security**. The **Configure Security** dialog box appears.
- 2 Click the **Roles** tab.
- 3 In the **Roles available** list, click the role you want to assign the permissions to.



- 4 In the **General Permissions** list, expand the **Graphic management permissions**.



- 5 If you want to restrict the user from:
- Importing symbols, clear the **Can import graphics** check box.
  - Exporting symbols, clear the **Can export graphics** check box.
  - Creating, modifying, or deleting symbols within toolsets, clear the **Can Create/Modify/Delete graphics within toolsets** check box.
  - Creating, modifying, or deleting symbols attached to a template, clear the **Can Create/Modify/Delete attached object graphics in template** check box.
  - Creating, modifying, or deleting symbols attached to an instance, clear the **Can Create/Modify/Delete attached object graphics in instance** check box.
  - Creating, modifying, or deleting view applications, clear the **Can Create/Modify/Delete ViewApplications** check box.
  - Deploying and undeploying view applications, clear the **Can Deploy/Undeploy ViewApplications** check box.
  - Accessing the edit quality and status display configuration, clear the **Can Edit Quality and Status Indicator Configuration** check box.
- 6 Click **OK**.

## Viewing a Symbol in Read-Only Mode

You can view a symbol in read-only mode if you don't want to edit it, or if it is checked out by somebody else.

If you open a symbol in read-only mode, you have access to all functions in the ArcestrA Symbol Editor that do not change the symbol.

### To view a symbol in read-only mode

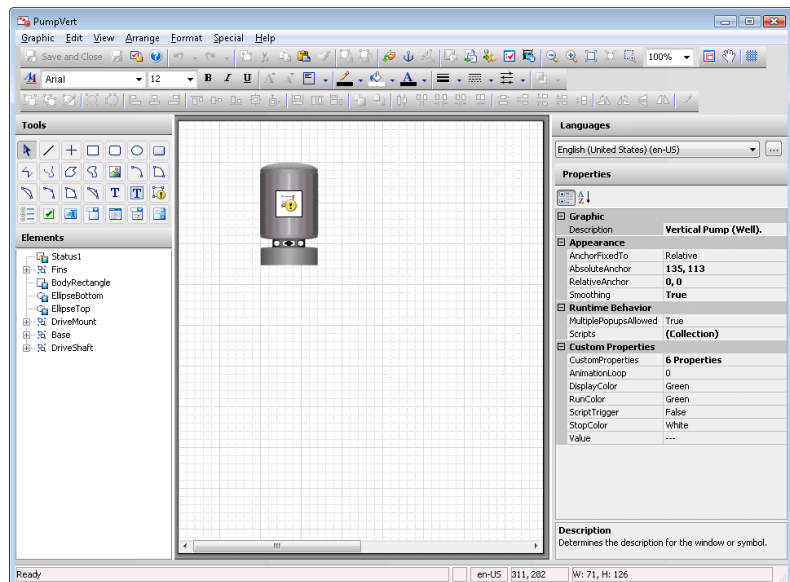
- 1 In the Graphic Toolbox, select the symbol that you want to view in read-only mode.
- 2 On the **Galaxy** menu, click **Open Read-Only**. The selected symbol opens in the ArcestrA Symbol Editor.

# Chapter 4

## Using the ArcestrA Symbol Editor

You can edit ArcestrA Symbols using the ArcestrA Symbol Editor. Depending on where the ArcestrA Symbol is contained, you can start the ArcestrA Symbol Editor from:

- The Graphic Toolbox.
- The **Graphics** tab of an AutomationObject template.
- The **Graphics** tab of an AutomationObject instance.
- An embedded ArcestrA Symbol in an InTouch window.



You can:

- Show and hide ArcestrA Symbol Editor panels to allocate more space on the canvas.
- Pan and zoom the canvas to make finer or more granular adjustments to elements.
- Place a grid on the canvas surface to align elements more precisely.

## Showing, Hiding, and Adjusting Panels

You can hide the Properties Editor and Animation Summary to allocate more more space on the canvas.

### To show or hide the Properties Editor and Animation Summary panels

- ◆ Do either of the following:
  - Press **ALT + ENTER**.
  - On the **View** menu, click **Properties**.

You can also adjust the size of the Elements List and Properties Editor.

### To adjust the size of panels

- 1 Drag the dividing line between the panels to specify the new panel size.
- 2 Release the mouse button and the panels are resized.

## Panning and Zooming the Canvas

You can pan and zoom the canvas to make finer visual adjustments to the elements or to get a better overview of a large symbol.

Use the Pan and Zoom toolbar to pan and zoom.



### Panning

You can use the Pan functions of the Pan and Zoom toolbar to do the following:

- Use the **Pan and Zoom** window to select which part of the canvas appears on the screen.
- Grab the canvas with the Hand tool and move it (Pan).

You can also use the scroll wheel of the mouse to pan up and down in the current canvas display.

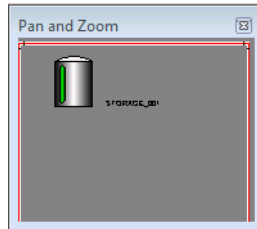
## Using the Pan and Zoom Window to Pan

You can use the **Pan and Zoom** window to pan the canvas area.

### To use the Pan and Zoom window for panning



- 1 On the Pan and Zoom toolbar, click the **Pan and Zoom** window icon. The **Pan and Zoom** window appears.



- 2 In the **Pan and Zoom** window, move the mouse within the red rectangle. The pointer hand icon appears.
- 3 Click and hold the left mouse button down.
- 4 Drag the mouse. The red rectangle moves with the mouse.
- 5 Release the mouse button. The area shown in the canvas is changed accordingly.

## Using the Hand Tool to Pan

You can use the Hand tool to pan the canvas area. This is equivalent to picking up the canvas and moving it so that the visible canvas area changes.

### To use the Hand tool to pan



- 1 On the **Pan and Zoom** toolbar, click the **Pan** icon.
- 2 Move the mouse over the canvas. The Hand tool pointer appears.
- 3 Click the canvas to grab the canvas and keep the mouse button down.
- 4 Move the mouse to change the area of canvas that is shown.
- 5 Release the mouse button.

## Using the Mouse Scroll Wheel to Pan

You can use the mouse scroll wheel to:

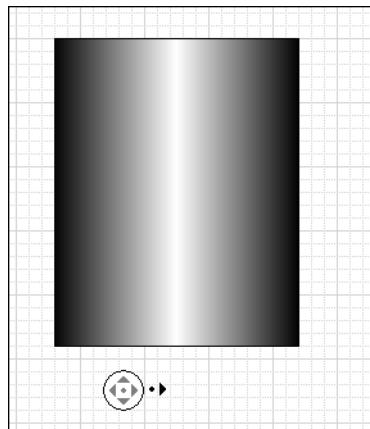
- Pan up or down.
- Pan 360 degrees.

### To use the mouse scroll wheel to pan up or down

- 1 Click the canvas so that no elements are selected.
- 2 Move the mouse scroll wheel:
  - Forward to pan up.
  - Backward to pan down.

### To use the mouse scroll wheel to pan in any direction

- 1 Click the canvas so that no elements are selected.
- 2 Click the mouse scroll wheel. The pointer appears in 360 degrees scroll mode.
- 3 Move the mouse. The visible area of the canvas is panned accordingly.



- 4 When you are done, click the canvas.

## Zooming

You can use the Pan and Zoom toolbar to:

- Zoom in on a specified point to magnify the current elements.
- Zoom out from a specified point.
- Zoom to the default zoom factor (100 percent).
- Zoom so that the currently selected element is shown across the available canvas area or zoomed to the maximum value of 500 percent.
- Zoom in on an area of the canvas using a “rubber band” selection with your mouse.
- Specify or select a zoom factor.

You can also use the CTRL key and the scroll wheel of the mouse to zoom in and zoom out the current canvas view.

### Zooming In to a Specified Point

You can zoom in by 25 percent of the default scale to any specified point on the canvas.

#### To zoom in to a specified point



- 1 Click the Zoom In icon in the toolbar.
- 2 Move the mouse over the canvas. The Zoom In pointer appears.
- 3 Click the canvas to where you want to zoom in. The canvas is zoomed in at the specified point.

### Zooming Out from a Specified Point

You can zoom out by 25 percent of the default scale from any specified point on the canvas.

#### To zoom out to a specified point



- 1 Click the Zoom Out icon in the toolbar.
- 2 Move the mouse over the canvas. The Zoom Out pointer appears.
- 3 Click the canvas from where you want to zoom out. The canvas is zoomed out from the specified point.

### Zooming to the Default Zoom Value

You can reset the zoom to the default zoom value 100 percent.

#### To reset the zoom to the default zoom value



- ◆ Click the Zoom to Normal icon in the toolbar. The canvas zoom is reset to its default.

### Zooming a Selected Element

You can zoom one or more selected elements so that they appear as large as possible in the allocated canvas area. This is useful when you want to make fine adjustments to one or more elements.

#### To zoom a selected element



- 1 Select the elements you want to zoom.
- 2 Click the Zoom To Selection icon in the toolbar. The visible canvas is zoomed so that the selected elements appear as large as possible.

### Zooming a Specified Area

You can zoom a specified area by using the “rubber band” selection method.

#### To zoom a specified area



- 1 Click the Rubber Band Zoom icon.
- 2 Move the mouse over the canvas. The Rubber Band pointer appears.
- 3 Move the mouse to the top left corner of the area you want to zoom.
- 4 Hold the left mouse button down and then drag the mouse to the bottom right corner of the area you want to zoom.
- 5 Release the mouse button. The area is zoomed to the entire canvas area.

### Selecting or Specifying a Zoom Value

You can select a defined zoom value or type a zoom value. Valid values are 25 percent to 500 percent.

#### To select or specify a zoom value

- ◆ On the Zoom and Pan toolbar, do one of the following:
  - Click the zoom value list and select a zoom value.
  - Click the zoom value in the zoom value list, type a valid value, and then click **ENTER**.



## Using the Pan and Zoom Window to Change the Zoom

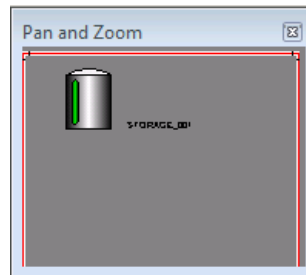
You can use the **Pan and Zoom** window to change the zoom of the canvas.

**Note** You can also use the **Pan and Zoom** window to “scroll” to a different part of the canvas. This is called panning. For more information, see [Panning](#) on page 84.

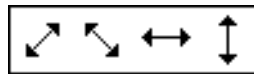
### To use the Pan and Zoom window for zooming



- 1 On the Zoom and Pan toolbar, click the **Pan and Zoom** window icon. The **Pan and Zoom** window appears.



- 2 In the **Pan and Zoom** window, move the mouse over a corner or an edge of the red rectangle.
- 3 Click and hold the left mouse button down. The corresponding resize pointer appears.



- 4 Drag the mouse. The red rectangle changes size proportionally.
- 5 Release the mouse button. The zoom of the canvas is changed accordingly.

## Using the Mouse Scroll Wheel for Zooming

You can use the mouse scroll wheel to zoom the canvas area. The canvas is then zoomed on the midpoint of all selected elements or, if none are selected, on the midpoint of the canvas.

### To use the mouse scroll wheel for zooming

- ◆ Press and hold the CTRL key and move the scroll wheel:
  - Forward to zoom in by a factor of 25 percent of the default zoom value.
  - Backward to zoom out by a factor of 25 percent of the default zoom value.

## Configuring the Grid

Use the grid to help you align elements on the canvas. You can configure the appearance and visibility of the grid.

Grid settings, such as color, grid size, number of subdivisions, visibility, and snap to grid, are saved with the symbol.

### Configuring the Grid Appearance

You can configure the following options for the grid:

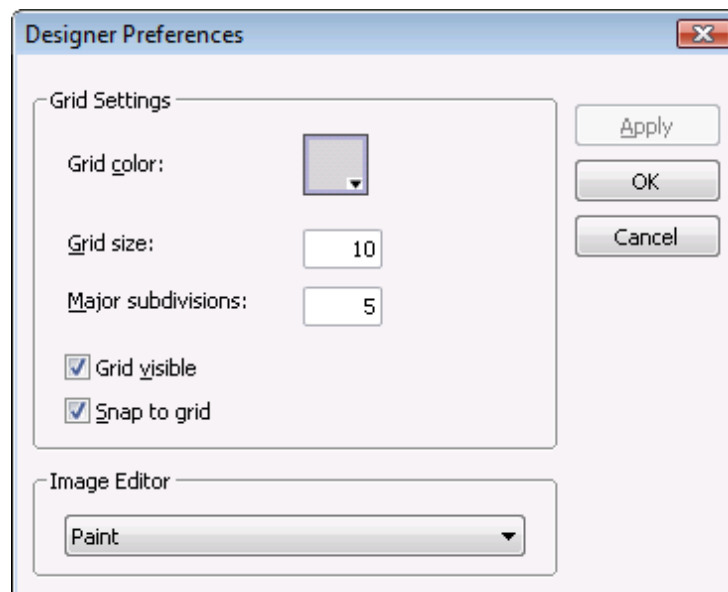
- Color
- Size
- Major subdivision
- Visibility
- Default snap-to-grid option

#### Configuring the Grid Color

You can change the color of the grid to any solid color.

To configure the grid color

- 1 On the **Special** menu, click **Preferences**. The **Designer Preferences** dialog box appears.



- 2 Click the box next to the **Grid color** label. The **Style Selection** dialog box appears. For more information, see [Setting a Solid Color](#) on page 166.

## Changing the Grid Size

You can change the size of the grid. The size number specifies how far apart in pixels the individual lines of the grid are.

### To change the grid size

- 1 On the **Special** menu, click **Preferences**. The **Designer Preferences** dialog box appears.
- 2 In the **Grid size** box, type a value from 1 to 100 for the grid size in pixels.
- 3 Click **OK**.

## Changing the Grid Major Subdivisions

You can change the number of major subdivisions of the grid. Major subdivisions are emphasized lines that visually create larger grid cell blocks. You can use the grid to precisely place and move elements on the canvas.

### To change the grid major subdivisions

- 1 On the **Special** menu, click **Preferences**. The **Designer Preferences** dialog box appears.
- 2 In the **Major subdivisions** box, type a value from 1 to 10.
- 3 Click **OK**.

## Hiding or Showing the Grid

You can hide or show the grid.

### To hide or show the grid

- 1 On the **Special** menu, click **Preferences**. The **Designer Preferences** dialog box appears.
- 2 Select or clear **Grid visible** check box to hide or show the grid.
- 3 Click **OK**.

## Aligning Elements with the Grid

You can set the snap-to-grid option so that when you move elements or groups on the canvas they are moved to the closest grid intersection. If this option is not set, you can move the elements freely to any location on the canvas.

### To switch on or switch off the snap-to-grid option

- 1 On the **Special** menu, click **Preferences**. The **Designer Preferences** dialog box appears.
- 2 Select or clear the **Snap to grid** check box.



# Chapter 5

## Working with Graphic Elements

This section shows you how to work with the common features of graphic elements. For information about features specific to certain elements such as element properties, see [Setting Symbol and Element-Specific Properties](#) on page 193.

### About Graphic Elements

Graphic elements are basic shapes and controls you can use to create a symbol to your specifications. You can:

- Draw an element by selecting an element from the Tools panel, placing it on the canvas, and then configuring its properties.
- Select one or more elements on the canvas with the mouse or from the **Element** list.
- Edit certain elements in a special way called inline editing.
- Copy, cut, paste, and duplicate elements.
- Move elements around on the canvas.
- Align elements to each other.
- Change the spacing between elements.
- Resize elements.
- Change the z-order of elements to change which elements appear on top of others when they overlap.
- Rotate elements.

- Change the origin of elements to specify around which point the elements are rotated.
- Flip elements on their horizontal or vertical axis.
- Lock elements to stop them being moved or changed.
- Undo and redo any number of changes made previously to the symbol.
- Create groups of elements to bind them together.
- Create a path graphic from multiple open line elements.

## Drawing and Dragging Elements

You can create elements such as lines, curves, circles, squares, and so on. You can combine these elements to create complex drawings of all the equipment in your manufacturing environment.

After you draw an element, you can modify its properties. For more information about modifying properties, see [Editing Element Properties](#) on page 98.

Regardless of the kind of element you are drawing, drawing each kind of element is very similar.

After you draw an element, the pointer tool is selected again by default. To draw multiple elements of the same type, double-click the element in the Tools panel. It remains selected after you draw your first element of that type. You can press the ESC key to return to the pointer tool again.

If you draw or drag an element outside of the visible canvas area to the right or bottom, horizontal and/or vertical scroll bars appear but the visible area does not follow the mouse. You can later use the scroll bars to scroll the canvas and see the element you drew or moved.

## Drawing Rectangles, Rounded Rectangles, Ellipses, and Lines

You can draw rectangles, rounded rectangles, ellipses, and lines on the canvas.



### To draw a rectangle, rounded rectangle, ellipse, or line

- 1 Click the appropriate icon in the Tools panel.
- 2 Click the canvas and drag the shape of the element on the canvas.
- 3 When you are done, release the mouse button.

## Drawing Polylines, Polygons, Curves, and Closed Curves

You can draw polylines, polygons, curves, and closed curves on the canvas.

If you are drawing a closed element, the element automatically closes when you are done drawing.



### To draw a polyline, polygon, curve, or closed curve

- 1 Click the appropriate icon in the Tools panel.
- 2 Click the canvas where you want to start the element.
- 3 Click the next point for the element.
- 4 Continue clicking until you have all the points you require.
- 5 When you are done, right-click.
- 6 You can change the shape of these elements anytime by editing their control points. For more information, see [Editing Control Points](#) on page 204.

## Drawing 2-Point Arcs, 2-Point Pies and 2-Point Chords

You can draw 2-point arcs, 2-point pies, and 2-point chords on the canvas.

If you are drawing a closed element, the element automatically closes when you are done drawing.



### To draw a 2-point arc, 2-point pie, or 2-point chord

- 1 Click the appropriate icon in the Tools panel.
- 2 Click the canvas where you want to start the element and hold the mouse button.
- 3 Drag the mouse to where you want the element to end.
- 4 When you are done, release the mouse button.
- 5 You can change the shape of these elements anytime by editing their control points. For more information, see [Editing Control Points](#) on page 204.

## Drawing 3-Point Arcs, 3-Point Pies, and 3-Point Chords

You can draw 3-point arcs, 3-point pies and 3-point chords on the canvas.

If you are drawing a closed element, the element automatically closes when you are done drawing.



### To draw a 3-point arc, 3-point pie, or 3-point chord

- 1 Click the appropriate icon in the Tools panel.
- 2 Click the canvas where you want to start the element.
- 3 Click the canvas in two other places to define the element.
- 4 You can change the shape of these elements anytime by editing their control points. For more information, see [Editing Control Points](#) on page 204.

## Placing and Importing Images

You can place an image element on the canvas and import an image into it.



### To draw an image

- 1 Click the image icon in the Tools panel.
- 2 Click the canvas and drag the shape of the image element.
- 3 Release the mouse button. The **Open** dialog box appears.
- 4 Browse to the image file, select it, and then click **Open**. The image file is loaded into the image element.

## Drawing Buttons

You can draw a button on the canvas. You can configure a button with a text label or an image.

For more information on how to configure a button with an image after drawing it on the canvas, see [Configuring Buttons with Images](#) on page 203.



### To draw a button

- 1 Click the button icon in the Tools panel.
- 2 Click the canvas and drag the shape of the button element.
- 3 Release the mouse button. The button text appears in edit mode.
- 4 Type a text label for the button and click ENTER.



## Placing Text

You can place text on the canvas.

The text element has no border and no background fill. The text does not wrap. When you type the text, the size of the Text element expands.

You can also drag the handles of the Text element to resize it.

### To place text



- 1 Click the text icon in the Tools panel.
- 2 Click the canvas where you want to place the text.
- 3 Type the single line of text you want.
- 4 When you are done, do one of the following:
  - Click ENTER to type a new line of text. This new line is a new element.
  - Click the canvas outside the text element.

## Drawing Text Boxes

You can draw text boxes on the canvas. Text boxes can have borders and background fill.

You can also configure the text to wrap in the text box. For more information, see [Wrapping Text in Buttons](#) on page 203.

### To draw a text box



- 1 Click the text box icon in the Tools panel.
- 2 Click the canvas where you want to place the text box.
- 3 Drag a rectangle on the canvas.
- 4 Release the mouse button. The text appears in edit mode.
- 5 Type a text label for the text box, and then click ENTER.

## Drawing Status Elements

You can use the status element to indicate specific quality and status conditions of attributes.

### To draw status elements



- 1 Click the status icon in the Tools panel.
- 2 Click the canvas where you want to place the status element.
- 3 Drag a rectangle on the canvas.
- 4 Release the mouse button.

## Drawing Windows Controls

You can draw Windows controls on the canvas to add additional functionality to your symbol. Each of the Windows controls has specific behavior when it is drawn. For example, you can change the width of a combo box, but not the height.



### To draw a windows control

- 1 Click the appropriate Windows control icon in the Tools panel.
- 2 Click the canvas where you want to place the Windows control.
- 3 Drag a rectangle on the canvas.
- 4 Release the mouse button.

## Dragging Elements

After you draw elements on the canvas, you can drag them to a new position.

### To drag elements on the canvas

- 1 Select one or more elements.
- 2 Click one of them and hold the mouse button down.
- 3 Drag the mouse to the new position.
- 4 Release the mouse button.

## Editing Element Properties

You can control the appearance of an element, a group of elements, or multiple elements with functions on the toolbar and/or properties in the Properties Editor.

Often you can edit an element by changing the values of its properties instead of using the mouse to perform the same function. This is useful when you want very exact editing, such as when you want to resize an element to a specific width.

The Properties Editor shows the properties common to all selected elements.

- Read-only properties appear in grey.
- Non-default values appear in bold.

---

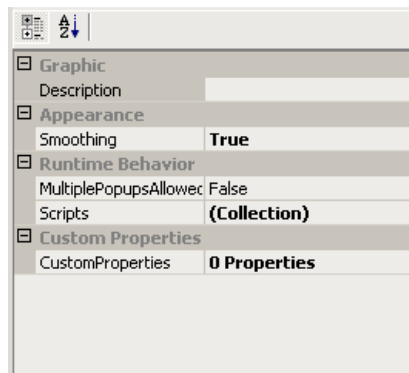
**Note** The Properties Editor not only supports values, but also allows input of color, font, and file information in the respective dialog boxes.

---

Properties are organized in categories so you can find them more easily. The following table shows the categories:

Property Category	Purpose
<b>Graphic</b>	Element name or other describing identifiers
<b>Appearance</b>	Element location, size, orientation, offset, transparency and locked status
<b>Fill Style</b>	Any parameters related to the fill appearance of the element
<b>Line Style</b>	Any parameters related to the line appearance of the element
<b>Text Style</b>	Any parameters related to the text appearance of the element
<b>Runtime Behavior</b>	Element visibility, tab order and any other element behavior at run time
<b>Custom Properties</b>	Additional user-defined properties you can associate with any element

For more information on the individual properties for each element, see [Alphabetical List of Properties](#) on page 444.

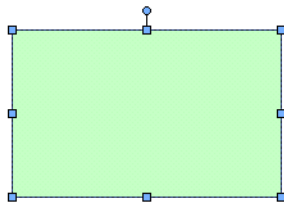


## Selecting Elements

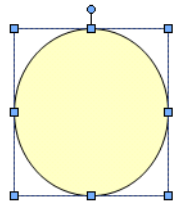
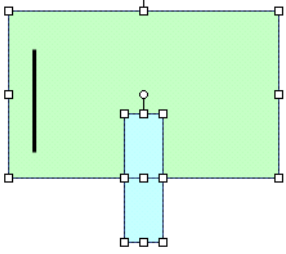
You can select one or more elements by:

- Clicking on them with the mouse.
- Dragging the lasso around them with the mouse.
- Selecting them with a menu option or with a shortcut key.
- Selecting them in the Elements List.

When you select an element, it appears with handles that give you control over its size and orientation.



When you select multiple elements, the last selected element is the primary element. All other previously selected elements are secondary elements.

Selected Element	Description
Primary Element 	Appears with color-filled handles. Behaves as an active selected element. Is the point of reference for all operations, such as aligning or spacing multiple selected elements.
Secondary Elements 	Appear with white handles. Behave as inactive selected elements. Follow the edits made to the primary element.

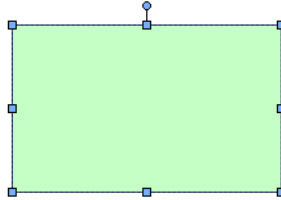
To select a group, you must click one of the elements contained in the group.

## Selecting Elements by Mouse Click

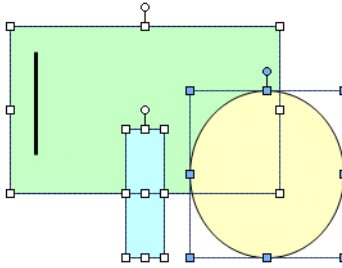
You can select one or more elements by pressing SHIFT + clicking. This is particularly useful for selecting multiple elements that are not necessarily all included in a specified rectangular area on the canvas.

**To select an element or multiple elements by mouse click**

- 1 On the canvas, click an element. It becomes selected.



- 2 To select further elements, press SHIFT + click. The other elements become selected.



---

**Note** You can see in the Elements List which elements are selected.

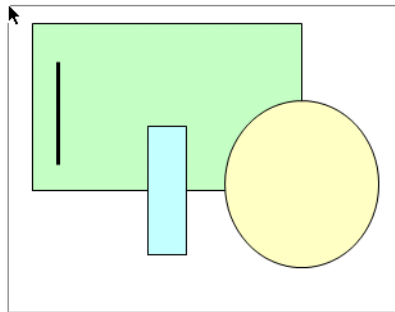
---

## Selecting Elements by Lasso

You can select one or more elements by lassoing them with your mouse. This is useful for selecting multiple elements within a specified rectangular area on the canvas.

### To select elements by lasso

- 1 On the canvas, click outside any element and hold the mouse button down.
- 2 Drag the mouse so that the lasso wraps around all elements that you want to select.



- 3 When you are done, release the mouse button. The elements that are fully enclosed within the lasso are selected.

## Selecting All Elements

You can select all elements using the Select All function.

### To select all elements

- ◆ On the **Edit** menu, click **Select All**. All elements on the canvas are selected.

---

**Note** You can also press the F2 key to select all elements.

---

## Selecting Elements Using the Elements List

You can use the Elements List to select any elements on the canvas. The Elements List is particularly useful for selecting elements behind other elements.

The Elements List shows which elements are currently selected. The primary selected element appears by default in dark blue, the secondary selected elements appear by default in light blue.

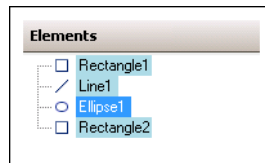
---

**Note** The color setting of the Elements List depends on the setting for the **Selected Items** option in the operating system's **Display Properties Appearance** panel.

---

### To select elements using the Elements List

- 1 In the Elements List, select the element name.
- 2 To select multiple elements, CTRL + click the other elements.



## Unselecting Elements

You can unselect one or more selected elements. You can do this by clicking on them individually on the canvas or in the Elements List.

If you want to remove the selected elements in a specified rectangular area, you can use the lasso.

### To unselect elements individually

- 1 Do one of the following:
  - SHIFT + click the selected element on the canvas.
  - CTRL + click the selected element name in the Elements List.
- 2 Repeat the previous step for all elements you want to unselect.

**To unselect elements from a specified rectangular area**

- 1 SHIFT + click the canvas outside of any element.
- 2 Drag the mouse so that the lasso surrounds the elements that you want to unselect.
- 3 Release the mouse button. The selected elements within the lasso are unselected, and the selected elements outside the lasso remain selected.

## Inline Editing

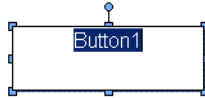
After you place certain elements on the canvas, you can edit them by selecting them and clicking on them again. This is called inline editing. The following elements can be edited in this way:

Element	Use inline editing to
Button, text, text box	Edit the text.
Polyline, polygon, curve, closed curve	Edit the control points.
2-point arc, 2-point pie, 2-point chord, 3-point arc, 3-point pie, 3-point chord	Edit the start and sweep angles.
Group	Edit the individual elements and groups contained in the group.
Path	Edit the control points.

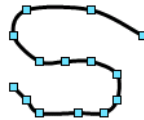


**To edit elements with inline editing**

- 1 Select an element. The element handles appear.
- 2 Click the element again to begin inline editing.
  - For buttons, text, and text boxes, the text is selected and you can type new text.

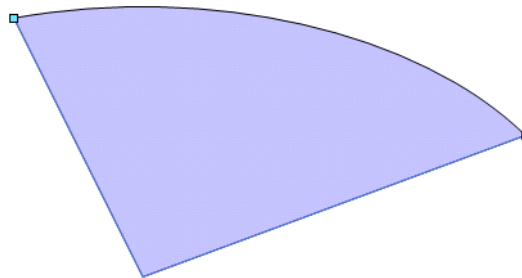


- For polylines, polygons, curves, and closed curves, the control points of the element appear. Use these to change the shape of the element.

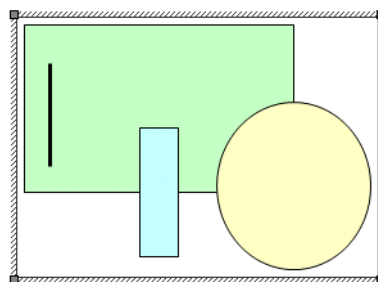


You can also add and delete control points. For more information, see [Adding and Removing Control Points](#) on page 205.

- For arcs, pies, and chords, the handles for the start angle and sweep angle appear. Use these to change the start angle and sweep angle.



- For groups, the group handle is replaced with a shaded outline. You can select individual elements and groups within the group to edit and move them.



- 3 Click the canvas outside the element.

## Copying, Cutting, and Pasting Elements

After you draw elements, you have the same cut, copy, and paste options available to you as in any other Windows application. However, some of these options behave differently in the ArcestrA Symbol Editor.

You can also duplicate elements. Duplicating elements lets you quickly make copies of existing selected elements without first copying or cutting. You can duplicate one or more selected elements at the same time.

When you copy or duplicate elements, all of its properties are copied with the element. If you do not want the properties to be identical, you must change the properties after you copy.

Locked grouped elements and the path element behave differently when you copy or duplicate them.

If you copy or duplicate:

- A set of elements that are locked, the copy is not locked.
- Grouped elements, the copy is still grouped.
- A path element, the copy is also a path.

### Copying Elements

After you select an element, you can copy it by using menu options or you can CTRL + click.

#### To copy one or more elements



- ◆ Do any of the following:
  - Select one or more elements to be copied on the canvas. On the **Edit** menu, click **Copy**. On the **Edit** menu, click **Paste**. The paste pointer appears. Click the canvas where you want to place the copy.
  - CTRL + click an element.
  - Select one or more elements to be copied on the canvas. Press CTRL + C. Press CTRL + V. The paste pointer appears. Click the canvas where you want to place the copy.

## Cutting or Deleting Elements

You can cut elements or groups or you can delete them. Cutting lets you select elements or groups and remove them from the canvas. You can paste the removed elements or groups.

Deleting elements or groups deletes them from the canvas. You cannot paste deleted elements or groups.

### To cut one or more elements

- ◆ Select one or more elements, and then do one of the following:
  - On the **Edit** menu, click **Cut**.
  - Press CTRL + X.

### To cut and paste elements on the canvas

- 1 Select the element or group.
- 2 On the **Edit** menu, click **Cut**.
- 3 Do one of the following:
  - Click **Paste** on the **Edit** menu.
  - Press CTRL + V.
- 4 Click the canvas location where you want the element or group to be placed.

### To delete an element or a group

- 1 To remove the element or group and **not** use it in the future, select the element or group.
- 2 Do one of the following:
  - Click **Delete** on the **Edit** menu.
  - Press Delete on your keyboard.

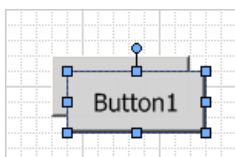
## Duplicating Elements

Duplicating elements allows you to select an element or elements and quickly make copies of them.

You can also specify the amount of overlap when you duplicate.

### To duplicate elements

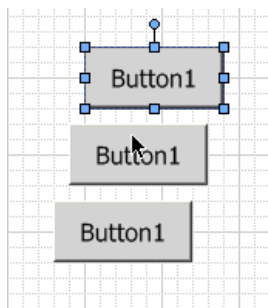
- 1 Select one or more elements.
- 2 Do one of the following:
  - a Click **Duplicate** on the **Edit** menu. The selected element is duplicated and appears offset to the original element.
  - b Press **CTRL + D**. The selected element is duplicated and appears offset to the original element.



- c **CTRL + click** one of the selected elements to duplicate all selected elements. You can keep the mouse button down and drag them to the new position on the canvas.

### To set the overlap when you duplicate

- 1 Duplicate an element or elements. The element is copied overlapping the original.
- 2 Move the duplicated element to the location relative to the original. For example, move the duplicated element five grid spaces above the original element.
- 3 Duplicate the element again. The new duplicate is placed in the same offset you specified in the preceding step. For example, five grid spaces above the original element.



## Moving Elements

After you create elements, you can move them to the location you want on the canvas.

You can move elements or groups by dragging them to the new location or you can open the properties for the element or group and change the X and Y properties.

If you turned on snap to grid, moving an element or group with the mouse snaps the element or group to the grid. For more information about using the grid, see [Configuring the Grid](#) on page 90.

If you move an element or group by specifying X and Y coordinates, it does not snap to the grid.

You can move an element or group vertically or horizontally using the keyboard.

### To move an element or group using the mouse

- 1 Select the element or group you want to move.
- 2 Drag the elements or group to the new location.

### To move an element or group by specifying the X and Y properties

- 1 Select the element or group you want to move.
- 2 In the Properties Editor, expand **Appearance**.
- 3 Do the following:
  - In the X box, type the new X location.
  - In the Y box, type the new Y location.
- 4 Click in the canvas or click ENTER.

### To move an element or group vertically or horizontally using the mouse

- 1 SHIFT + click to select the element or group you want to move.
- 2 Drag the elements or group to the new location.

**To move an element or group vertically or horizontally using the keyboard**

- 1 Select the element or group you want to move.
- 2 Do one of the following:
  - Press the Up or Down arrow keys to move the element or group vertically by one unit in the grid.
  - Press the Left or Right arrow keys to move the element or group horizontally by one unit in the grid.

---

**Note** You can move the element or group by two units in the grid by additionally pressing the SHIFT key, by four units by additionally pressing the CTRL key, and by 10 units by additionally pressing both keys.

---

**To move multiple elements or groups**

- 1 Select the elements and/or groups.
- 2 Move them as you would with one single element. The elements are moved together and maintain their spacial relationship when moving.

## Aligning Elements

After you draw elements, you can align them:

- Horizontally so that their top or bottom sides or their center points are horizontally aligned.
- Vertically so that their left, right, or center points are vertically aligned.
- So that their center points are on top of each other.
- So that their points of origin are on top of each other.

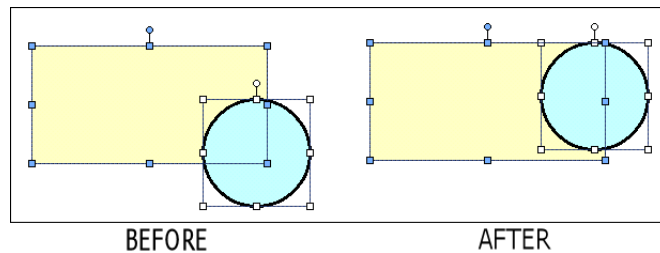
When you align elements, the secondary elements are moved so that they align with the primary element. For more information about primary and secondary elements, see [Selecting Elements](#) on page 100.

## Aligning Elements Horizontally

You can align multiple elements by their top or bottom sides or horizontally on their middle points.

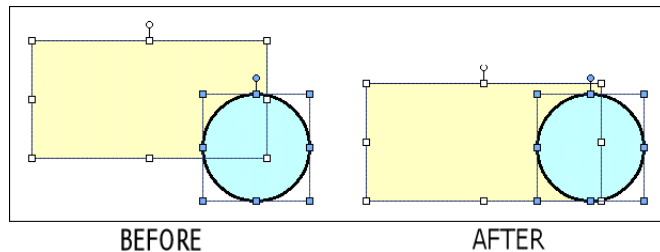
### To align elements by their top sides

- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Top**. The secondary elements are moved so that their top sides are aligned with the top side of the primary element.



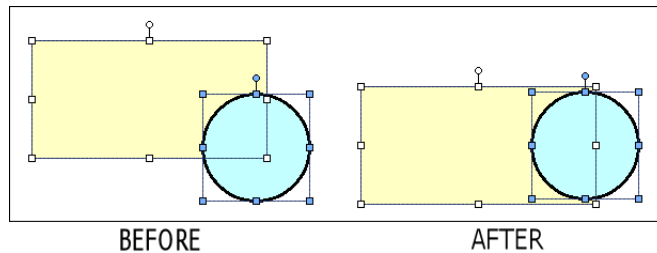
### To align elements by their bottom sides

- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Bottom**. The secondary elements are moved so that their bottom sides are aligned with the bottom side of the primary element.



**To align elements horizontally by their center points**

- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Middle**. The secondary elements are moved vertically so that their center points are aligned with the center point of the primary element.

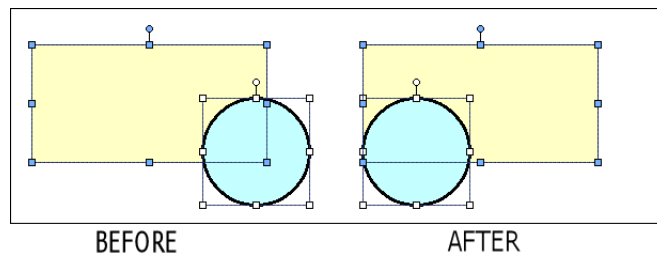


## Aligning Elements Vertically

You can vertically align multiple elements on the left, right, or their center points.

**To align elements by their left sides**

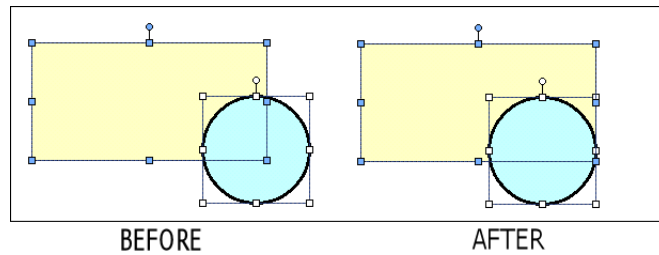
- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Left**. The secondary elements are moved so that their left sides are aligned with the left side of the primary element.





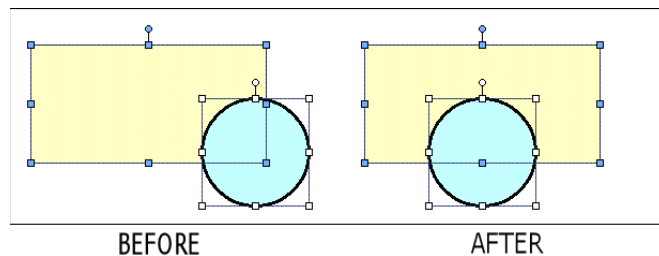
### To align elements by their right sides

- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Right**. The secondary elements are moved so that their right sides are aligned with the right side of the primary element.



### To align elements vertically by their centers

- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Center**. The secondary elements are moved horizontally so that their center points are aligned with the center point of the primary element.

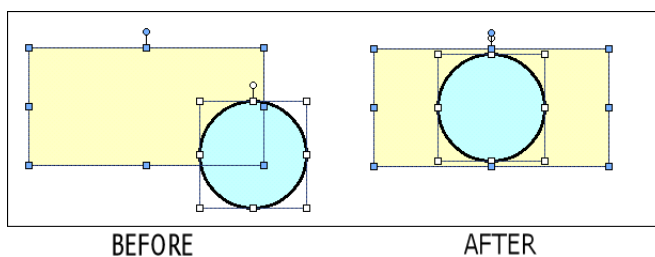


## Aligning Elements by their Center Points

You can align elements by their center points. The center point of one or more elements is the point halfway between the horizontal and vertical boundaries.

### To align elements on their center points

- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Centers**. The secondary elements are moved so that their center points are placed on top of the center point of the primary element.

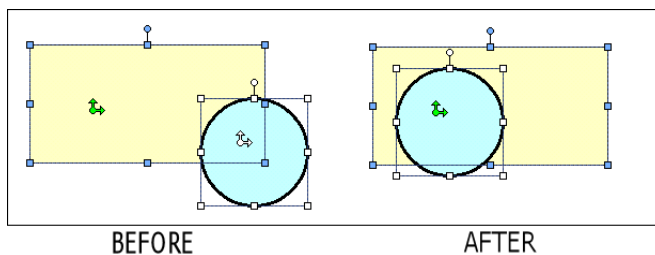


## Aligning Elements by their Points of Origin

You can align elements by their points of origin. By default, the element's center point is the point of origin. But, an element's center point can be changed. The center point is the anchor point of an element to the canvas.

### To align elements on their points of origin

- 1 Select all elements that you want to align. Make sure the element you want to align all other elements to is the primary element.
- 2 On the **Arrange** menu, point to **Align**, and then click **Align Origins**. The secondary elements are moved so that their points of origins are placed on top of the point of origin of the primary element.



## Adjusting the Spacing between Elements

You can adjust the space between elements according to specific rules.

You can adjust the spacing between elements in the following ways:

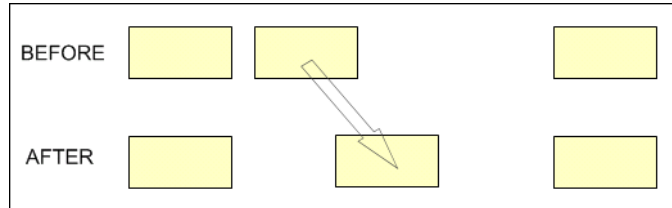
- Horizontally - moves the selected elements left or right without changing the vertical positions.
- Vertically - moves the selected elements up or down without changing the horizontal positions.
- Distribution - moves the selected elements so that their center points are distributed in equal distance to each other.
- Equal spacing - moves the selected elements so that the distance between their edges is equal.
- Increase spacing - moves all selected elements one pixel further away from each other. The primary element does not move.
- Decrease spacing - moves all selected elements one pixel closer toward each other. The primary element does not move.
- Remove spacing - removes all space between selected elements so that their edges touch.

## Distributing Elements

You can distribute elements so that their center points are distributed in equal distance to each other.

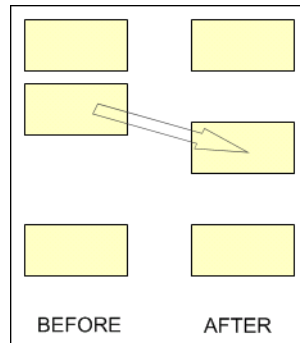
### To distribute elements horizontally

- 1 Select at least three elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Distribute Horizontal**. The selected elements are distributed horizontally.



### To distribute elements vertically

- 1 Select at least three elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Distribute Vertical**. The selected elements are distributed vertically.



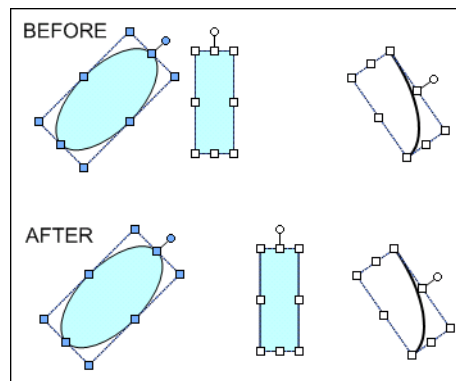
## Making Space between Elements Equal

You can space elements so that the distances between their boundaries are equal.

The difference between making space between elements equal and distributing them is that making space equal uses the boundaries of the elements, whereas distributing uses the center points. Both do not necessarily lead to the same result.

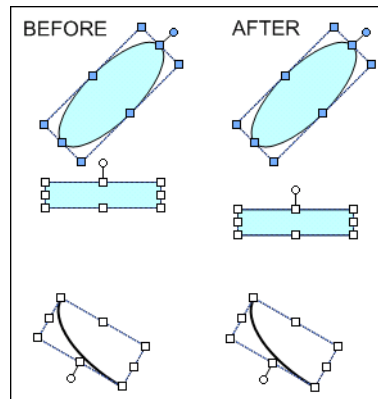
### To make the horizontal space between elements equal

- 1 Select at least three elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Make Horizontal Space Equal**. The selected elements are moved so that the horizontal spaces between their boundaries are equal.



### To make the vertical space between elements equal

- 1 Select at least three elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Make Vertical Space Equal**. The selected elements are moved so that the vertical spaces between their boundaries are equal.



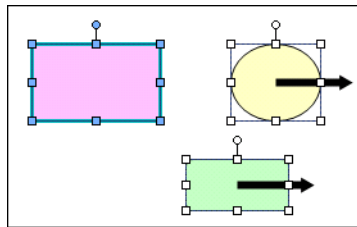
## Increasing Space between Elements

You can increase space between elements equally.

The primary element does not move. All secondary elements are moved away from the primary element.

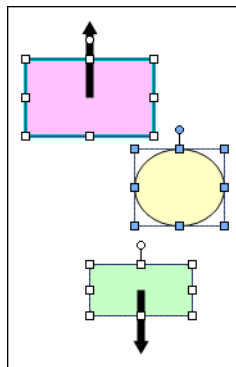
### To increase the horizontal space between elements

- 1 Select at least two elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Increase Horizontal Spacing**. The selected elements are moved so that the horizontal space between them is increased by one pixel.
- 3 Repeat the previous step to move the selected elements further away from each other.



### To increase the vertical space between elements

- 1 Select at least two elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Increase Vertical Spacing**. The selected elements are moved so that the vertical space between them is increased by one pixel.



- 3 Repeat the previous step to move the selected elements further away from each other.

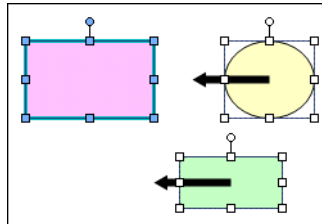
## Decreasing Space between Elements

You can decrease space between elements equally.

The primary element does not move. All secondary elements move toward the primary element. You can move them until the left sides of all elements are aligned.

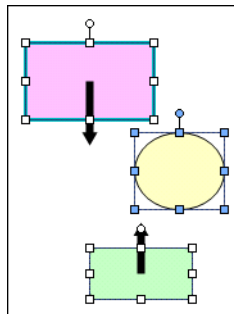
### To decrease the horizontal space between elements

- 1 Select at least two elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Decrease Horizontal Spacing**. The selected elements are moved so that the horizontal space between them is decreased by one pixel.
- 3 Repeat the previous step to move the selected elements closer toward each other.



### To decrease the vertical space between elements

- 1 Select at least two elements.
- 2 On the **Arrange** menu, point to **Space**, and then click **Decrease Vertical Spacing**. The selected elements are moved so that the vertical space between them is decreased by one pixel.
- 3 Repeat the previous step to move the selected elements closer toward each other.



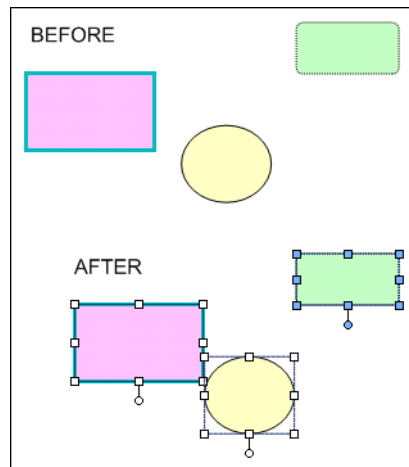
## Removing All Space between Elements

You can remove all space between selected elements so that their boundaries touch.

The primary element does not move. All secondary elements move toward the primary element. You can move them until the left and right sides of all secondary elements are aligned.

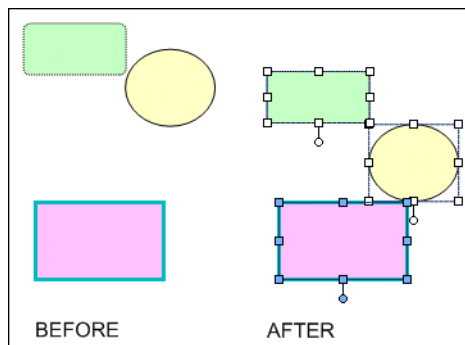
### To remove all horizontal space between elements

- 1 Select all elements between which you want to remove the space.
- 2 On the **Arrange** menu, point to **Space**, and then click **Remove Horizontal Spacing**. The horizontal space between all selected elements is removed, so that their boundaries touch.



### To remove all vertical space between elements

- 1 Select all elements between which you want to remove the space.
- 2 On the **Arrange** menu, point to **Space**, and then click **Remove Vertical Spacing**. The vertical space between all selected elements is removed, so that their boundaries touch.





## Resizing Elements

You can resize selected elements by:

- Dragging the handles of a single element to increase or decrease its horizontal or vertical size.
- Changing the Width and Height properties of one or more elements using the Properties Editor.
- Proportionally resizing multiple elements.
- Making multiple objects the same width and/or height.

Some elements cannot be resized or can only be resized in certain directions, such as the Calendar control or DateTime Picker. If the primary element has such restrictions, then any secondary elements resize proportional to the change in primary element's size and do not resize independently.

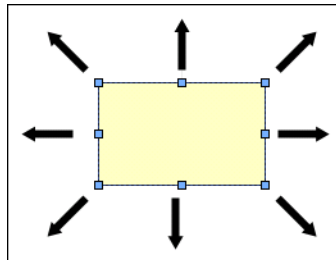
### Resizing a Single Element with the Mouse

You can resize a single selected element with the mouse.

You can resize most elements to any given width and height, or to a fixed width to height ratio.

#### To resize a single selected element with the mouse

- 1 Select an element. The handles of the selected element appear.



- 2 Drag one of the handles. The object is resized while you drag.
- 3 Release the mouse button.

#### To resize a single selected element with the mouse and keeping a fixed width/height ratio

- 1 Select an element. The handles of the selected element appear.
- 2 Press and hold the SHIFT key.
- 3 Drag one of the handles. The object is resized while you drag, the width/height ratio stays unchanged.
- 4 Release the mouse button and SHIFT key.

## Resizing Elements by Changing Size Properties

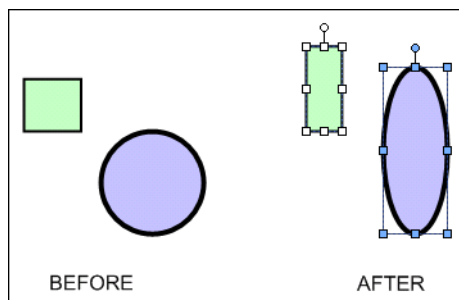
You can resize one or more elements by changing the width and/or height property of the selected elements.

### To resize elements by changing their size properties

- 1 Select one or more elements.
- 2 In the Properties Editor, type a value for **Width** and for **Height**. The selected elements are resized accordingly.

## Resizing Elements Proportionally

You can resize multiple elements proportionally on the canvas. One element is the primary element you can use to resize. The secondary elements resize proportionally to the change of the primary element.



### To resize elements proportionally

- 1 Select multiple elements.
- 2 Drag one of the handles of the primary element. The secondary elements are resized accordingly by the same percentage.
- 3 Release the mouse button.

For example, assume the primary element is 100 pixels wide and 50 pixels high. A secondary element is 200 pixels wide and 20 pixels high.

You drag the handle of the primary element so that it is 120 pixels wide (20 percent increase) and 100 pixels high (100 percent increase).

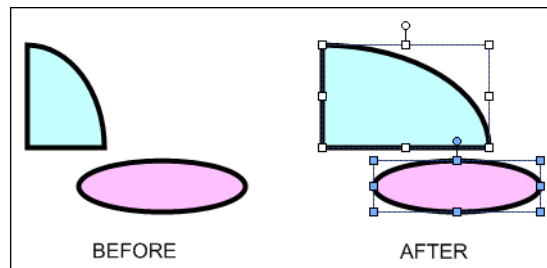
Then the secondary element is resized to 240 pixels wide (20 percent increase of the original width of 200 pixels) and 40 pixels high (100 percent increase of the original width of 20 pixels).

## Making Elements the Same Width, Height, or Size

You can make elements the same width, height, or size.

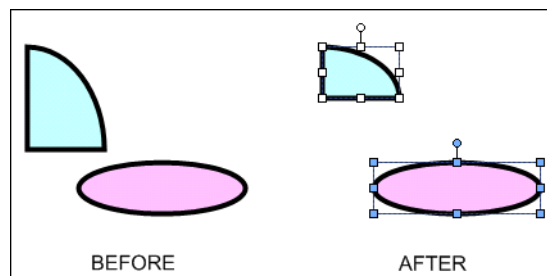
### To make elements the same width

- 1 Select at least two elements. Make sure the primary element is the element with the target width for all elements.
- 2 On the **Arrange** menu, point to **Size**, and then click **Make Same Width**. The width of the secondary elements are resized to the same width as the primary element.



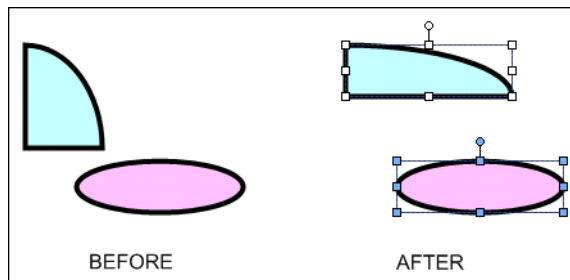
### To make elements the same height

- 1 Select at least two elements. Make sure the primary element is the element with the target height for all elements.
- 2 On the **Arrange** menu, point to **Size**, and then click **Make Same Height**. The height of the secondary elements are resized to the same height as the primary element.



**To make elements the same size**

- 1 Select at least two elements. Make sure the primary element is the element with the target size for all elements.
- 2 On the **Arrange** menu, point to **Size**, and then click **Make Same Size**. The size of the secondary elements are resized to the same size as the primary element.



## Adjusting the z-Order of Elements

The z-order of elements specifies which element appears on top of other elements when the elements overlap on the canvas. The z-order also determines how the elements of a path graphic connect.

When you place new elements on the canvas, they are placed at the top and can cover all other elements.

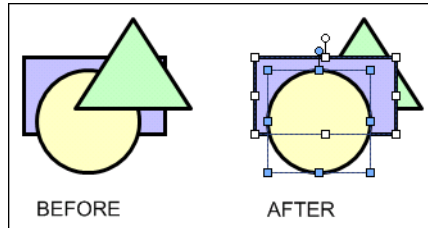
However, you might want to bring certain elements forward so that they are always visible or overlap certain other elements. Or you may want to use a large background element behind all other elements. You can:

- Bring one or more elements to the very front.
- Send one or more elements to the very back.
- Bring one or more elements one level forward.
- Send one or more elements one level backward.

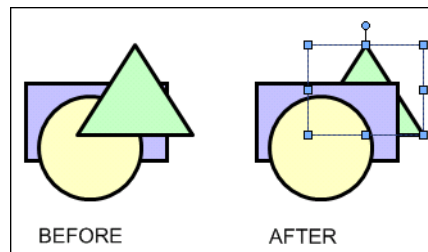
You can use the Elements List to see or change the z-order of the elements.

**To bring selected elements to the front**

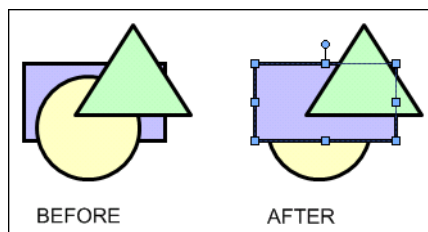
- ◆ On the **Arrange** menu, point to **Order**, and then click **Bring To Front**. The selected elements are brought to the front. They do not change their relative z-order.

**To send selected elements to the back**

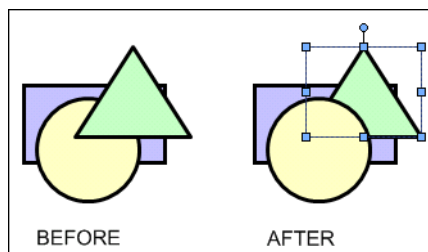
- ◆ On the **Arrange** menu, point to **Order**, and then click **Send To Back**. The selected elements are sent to the back. They do not change their relative z-order.

**To bring selected elements one level forward**

- ◆ On the **Arrange** menu, point to **Order**, and then click **Bring Forward**.

**To send selected elements one level backward**

- ◆ On the **Arrange** menu, point to **Order**, and then click **Bring Backward**.



## Rotating Elements

You can rotate elements to any orientation (0 - 359 degrees):

- Graphically with the rotation handle.
- Numerically by typing the orientation angle in the Properties Editor.
- By rotating in 90 degree increments in a clockwise or counter-clockwise direction.

The element rotates around its point of origin. By default, the point of origin is in the center of the element. You can move the point of origin to any other location, even outside of the object itself. To change the point of origin, see [Moving the Origin of an Element](#) on page 129.

## Rotating Elements with the Mouse

You can rotate one or more elements with the mouse. If you select multiple elements, you can rotate the primary element. The secondary elements rotate in unison with the primary element.

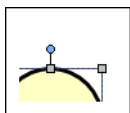
You can rotate elements:

- Freely in the range 0 to 359 in integer degrees.
- In multiples of 15 degrees.
- In multiples of 45 degrees.

You can rotate an element with the rotation handle. The rotation handle is a light-blue circle at the top of a selected element.

### To rotate elements freely with the mouse

- 1 Select one or more elements.
- 2 Grab the rotation handle of the primary element.
- 3 Drag the mouse across the screen. All selected elements are rotated around their own points of origin as you move the mouse.
- 4 Release the mouse button.



**To rotate elements by multiple of 15 degrees with the mouse**

- 1 Select one or more elements.
- 2 Grab the rotation handle of the primary element.
- 3 Press and hold the SHIFT key.
- 4 Drag the mouse across the screen. All selected elements are rotated in multiples of 15 degrees around their own points or origin as you move the mouse.
- 5 Release the mouse button and SHIFT key.

**To rotate elements by multiple of 45 degrees with the mouse**

- 1 Select one or more elements.
- 2 Grab the rotation handle of the primary element.
- 3 Press and hold the CTRL key.
- 4 Drag the mouse across the screen. All selected elements are rotated in multiples of 45 degrees around their own points or origin as you move the mouse.
- 5 Release the mouse button and CTRL key.

## Rotating Elements by Changing the Angle Property

You can change the angle property of one or more selected elements.

**To rotate elements by changing the angle property**

- 1 Select one or more elements.
- 2 In the Properties Editor, type a value in the **Angle** box.
- 3 Click ENTER. The selected elements rotate to the specified angle.

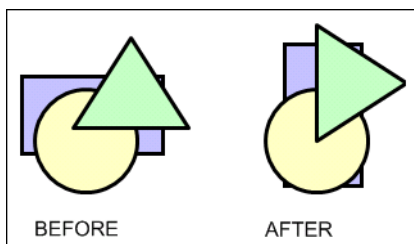
## Rotating Elements by 90 Degrees

You can rotate elements in 90 degrees clockwise or counter-clockwise increments.

To rotate elements by multiples of 15 and 45 degrees, see [Rotating Elements with the Mouse](#) on page 126.

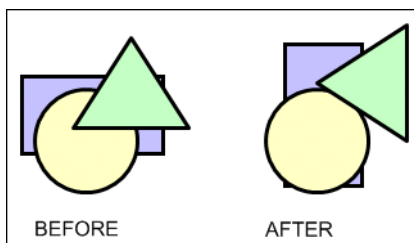
### To rotate elements by 90 degrees clockwise

- 1 Select one or more elements.
- 2 On the **Arrange** menu, point to **Transform**, and then click **Rotate Clockwise**. The selected elements rotate by 90 degrees clockwise.



### To rotate elements by 90 degrees counter-clockwise

- 1 Select one or more elements.
- 2 On the **Arrange** menu, point to **Transform**, and then click **Rotate Counter Clockwise**. The selected elements rotate by 90 degrees counter-clockwise.





## Moving the Origin of an Element

You can change the point of origin of any element. The point of origin specifies around which point the element rotates or flips. By default the point of origin is in the center of the element.

You can change the point of origin:

- With the mouse on the canvas.
- By specifying the absolute origin in the Properties Editor.
- By specifying the relative origin in the Properties Editor.

### Changing Points of Origin with the Mouse

You can change the point of origin for an element with the mouse.

**To change the point of origin for an element with the mouse**

- 1 Select an element on the canvas.
- 2 Move the mouse over the rotation handle of the element. The point of origin icon for the element appears.
- 3 Drag the Point of Origin icon to where you want to place the new point of origin for the element.
- 4 Release the mouse button.



## Changing Points of Origin in the Properties Editor

You can change the absolute or relative point of origin in the Properties Editor.

The absolute point of origin shows the position of the point of origin in relation to the canvas. The absolute point of origin changes when the element moves.

The relative point of origin shows the position of the point of origin in relation to the center of the element. The relative point of origin does not change when the element moves.

### To change the point of origin in the Properties Editor

- 1 Select one or more elements on the canvas.
- 2 In the Properties Editor, do one of the following:
  - Type the absolute coordinates in the x, y format for the point of origin.
  - Type the relative coordinates in the x, y format for the point of origin.
- 3 Click ENTER. The points of origin move to the specified absolute position or to the specified position in relation to the center points of the selected elements.

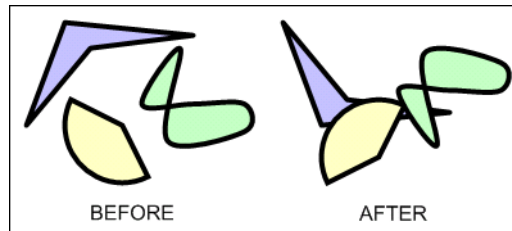
For example, if you have two elements, you can set the relative point of origin to 10, 10 to place the points of origin for both elements 10 pixels to the right and 10 pixels below the corresponding center points of each element.

## Flipping Elements

You can flip elements on their horizontal or vertical axes. The axis for each element is determined by its point of origin. For more information on how to change the point of origin, see [Moving the Origin of an Element](#) on page 129.

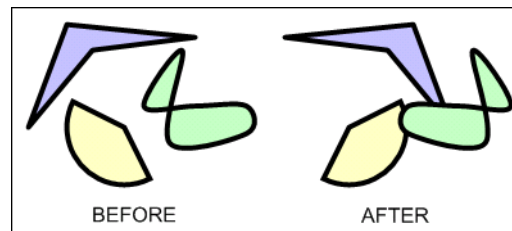
### To flip elements vertically

- 1 Select one or more elements.
- 2 On the **Arrange** menu, point to **Transform**, and then click **Flip Vertical**. The selected elements are flipped vertically on their horizontal axis.



### To flip elements horizontally

- 1 Select one or more elements.
- 2 On the **Arrange** menu, point to **Transform**, and then click **Flip Horizontal**. The selected elements are flipped horizontally on their vertical axis.



## Locking and Unlocking Elements

When you lock elements, they cannot be:

- Moved.
- Resized.
- Rotated.
- Aligned.
- Flipped.

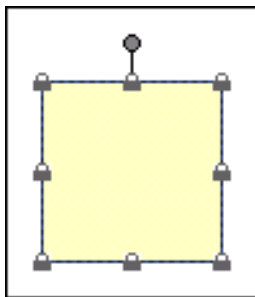
You also cannot change the point of origin in locked elements. To enable these functions again, you must unlock the elements.

### To lock elements

- 1 Select all elements that you want to lock.
- 2 Do one of the following:
  - On the **Arrange** menu, click **Lock**.
  - In the Properties Editor, set the Locked property to **True**.



The selected elements appear with lock icons at their handles.



### To unlock elements

- 1 Select all elements that you want to unlock.
- 2 Do one of the following:
  - On the **Arrange** menu, click **Unlock**.
  - In the Properties Editor, set the Locked property to **False**.



The lock icons disappear from the handles of the selected elements.

## Making Changes Using Undo and Redo

If you want to reverse a change you made in the Symbol Editor, use the undo function. After you undo a change, you can also redo that change by using the redo function.

You can undo one single change, or any number of changes that you have previously done. You can also redo any number of changes. These can be selected from a list.

### To undo a single change

- ◆ Do one of the following:
  - Press CTRL + Z.
  - On the **Edit** menu, click **Undo**.

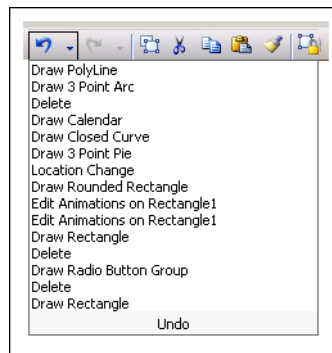
### To redo a single change

- ◆ Do one of the following:
  - Press CTRL + Y.
  - On the **Edit** menu, click **Redo**.

### To undo a specified number of previous changes



- 1 On the toolbar, click the Undo icon. The Undo list appears with a description of what the changes were.

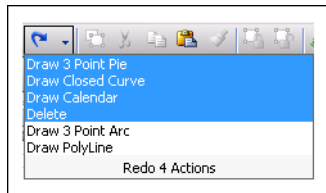


- 2 Select a change from the list. The changes up to and including the selected item are undone.



### To redo a specified number of previously undone changes

- 1 On the toolbar, click the Redo icon. The Redo list appears with a description of what the undone changes were.
- 2 Select a the change from the list. The changes down to and including the selected item are redone.



## Working with Groups of Elements

You can group together multiple elements. This is useful to bind certain element together so that they are not inadvertently moved. The group is treated as a new element.

You can:

- Create a group from one or more elements.
- Ungroup the elements without losing their original configuration information.
- Add more elements to an existing group.
- Remove elements from a group.
- Edit the elements of a group without having to ungroup them.

### Creating a Group of Elements

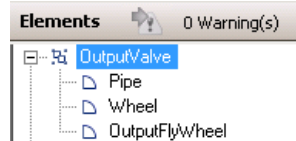
After you create elements, you can group them. Grouping elements lets you manage the elements as one unit.

Groups are assigned default names when you create them, such as Group1, Group2, and so on. After you create a group, you can rename it.

Groups can have properties that are different than the properties of the elements. For more information, see Properties of Groups on page 34.

**To create a group**

- 1 Select the elements you want as part of the new group.
- 2 On the **Arrange** menu, point to **Grouping**, and then click **Group**. The elements are combined into a group. The group is listed in the Elements List.
- 3 Rename the group as required. To do this:
  - a In the Elements List, click the group name and click again. The group name is in edit mode.



- a Type a new name and click ENTER. The group is renamed.
- b You can also rename a group or elements by changing the Name property in the Properties Editor.

## Ungrouping

After you create a group, you can ungroup it if you no longer want it.

If the group included elements and other groups, when you ungroup, the original elements and groups again exist as independent items. To ungroup any subgroups, you must select each one and ungroup it separately.

If you ungroup a set of elements and elements already exist with the names of the grouped elements, then the newly ungrouped elements are renamed.

**To ungroup**

- 1 Select the groups you want to ungroup.
- 2 On the **Arrange** menu, point to **Grouping**, and then click **Ungroup**. The groups is converted to the original elements. The group name is removed from the Elements List and the element names appear.

## Adding Elements to Existing Groups

After you create a group, you can add elements or other groups to an existing group.

For example, you can combine a group that represents a valve with another group that represents a tank to create a new group that can be called a tank unit.

You can add:

- Elements to groups.
- Groups to the primary selected group.

### To add elements to an existing group

- 1 On the canvas, select the group and also elements and groups that you want to add.
- 2 Right-click a selected element or on the group, point to **Grouping**, and then click **Add to Group**. The selected elements are added to the group.

---

**Note** You can also add elements to existing groups by using the Elements List in similar way.

---

## Removing Elements from Groups

After you create a group, you can remove elements from the group. This lets you remove one or more elements you no longer want in that group.

Removing elements from the group removes them from the canvas. It also removes any scripts or animations you added to the element.

### To remove an element from a group

- 1 On the canvas, select the group with the elements that you want to remove.
- 2 Click the group again to enter inline editing mode.
- 3 Select the elements that you want to remove from the group.
- 4 Right-click a selected elements, point to **Grouping**, and then click **Remove from Group**. The selected elements are removed from the group.

---

**Note** You can also remove elements from existing groups by using the Elements List in similar way.

---



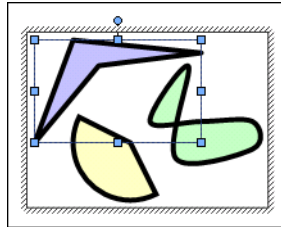
## Editing Components within a Group

You can edit components within a group without having to dissolve the group. Do this by:

- Selecting the element in Elements List.
- Using the **Edit Group** command on the shortcut menu.
- Slowly double-clicking to enter inline editing mode.

### To edit components within a group by using the Elements List

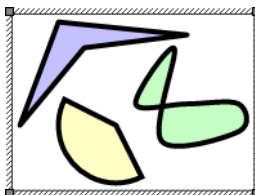
- 1 In the Elements List, expand the group that contains the element that you want to edit.
- 2 Select the element that you want to edit. The element appears selected in the group and the group is outlined with a diagonal pattern.



- 3 Edit the element with the Properties Editor, by mouse or by menu according to your requirements.
- 4 Click outside the group.

### To edit components within a group by using the Edit Group command

- 1 On the canvas, select the group that you want to edit.
- 2 On the menu **Edit**, click **Edit Group 'GroupName'**. The group is outlined with a diagonal pattern.



- 3 Select the element that you want to edit.
- 4 Edit the element with the Properties Editor, by mouse, by menu or pop-up menu according to your requirements.
- 5 Click outside the group.

---

**Note** If you move the position of an element in a group outside the group, the group size is automatically changed to incorporate the new position of the element.

---

## Using Path Graphics

You can join a set of open elements, such as lines, to create a new closed element. The new closed element is called a path graphic.

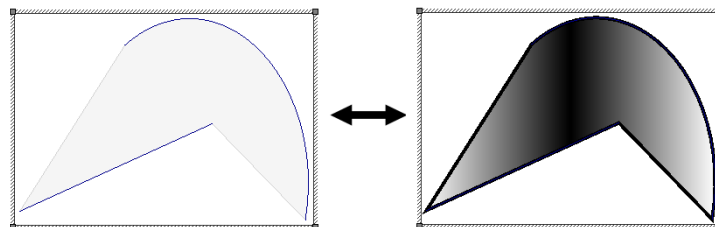
You can:

- Create a path graphic by joining open elements.
- Break the path graphic into its elements.
- Edit the path graphic in its entirety or by editing its elements.
- Add new elements to the path graphic.
- Remove elements from the path graphic.

You can view a path graphic in two modes:

- Element mode shows you the individual elements contained in the path graphic and determine its shape. Elements that make up the path graphic are shown as blue lines. The points where the elements are connected are shown as grey lines.
- Path mode shows you the path graphic in its final rendering, including fill styles and lines styles.

When you are in inline editing mode, you can switch between both modes by pressing the space bar. This lets you preview the path graphic without leaving the inline editing mode.

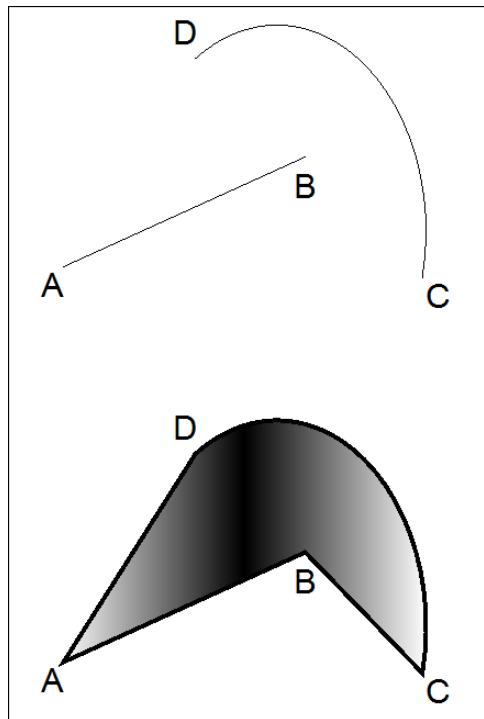


## Creating a Path Graphic

You can create a path graphic from one or more open elements such as lines, polylines, curves, and arcs.

The path graphic is created according to the start and end points and the z-order of the open elements that create it.

For example, if you draw a line from point A to point B, then an arc from point C to point D, and then join these elements in a path graphic, the path graphic is described by a straight edge from points A to B, a straight edge from points B to C, a curved edge from points C to D, and closed by a straight edge from points D to A.



**Note** If the Path Graphic doesn't appear as you expected after you create it, then you can swap the end points or change the z-order of one or more elements. For more information, see [Swapping the End Points of an Element in a Path Graphic](#) on page 145 and [Changing the Z-order of an Element in a Path Graphic](#) on page 146.

### To create a path graphic

- 1 Select one or more open elements.
- 2 On the **Arrange** menu, point to **Path**, and then click **Combine**. A new path graphic is created from the selected open elements.



## Breaking the Path of a Path Graphic

You can break the path of a path graphic so that it is broken into its individual open elements. When you do so, the path graphic loses its unique properties such as fill style and line style.

### To break the path of a path graphic

- 1 Select one or more path graphics.
- 2 On the **Arrange** menu, point to **Path**, and then click **Break**.



## Changing a Path Graphic

You can edit an existing path graphic on the canvas by accessing the individual elements of which it consists. For each individual element, you can:

- Move.
- Rotate.
- Change size.
- Change start and sweep angles if the elements are arcs.
- Change control points if the elements are curves or polylines.
- Swap the end points of an element in a path graphic.
- Change the z-order of the elements in a path graphic.

The path graphic is updated while you edit the individual elements.

## Moving Elements in a Path Graphic

You can move elements in a path graphic. If you move an element outside of the path graphic boundary, the boundary is redrawn to include the moved element.

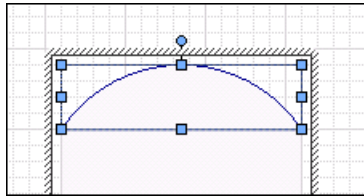
### To move an element within a path graphic

- 1 Select the path graphic you want to edit.
- 2 Do one of the following:
  - a On the **Edit** menu, click **Edit Path**.
  - b Slowly double-click the path graphic.

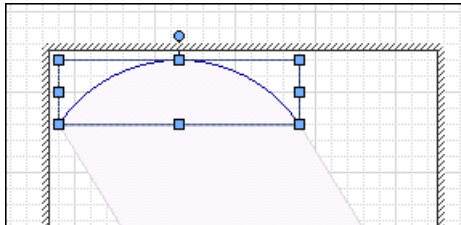


The path graphic appears in element mode.

- 3 Select the individual element within the path graphic you want to move. You can also do this by selecting the element in the Elements List.



- 4 Click a solid part of the element and drag it to the new position. The element is moved.



- 5 Click outside the path graphic on the canvas. The path graphic is shown in path mode.

## Resizing Elements in a Path Graphic

You can resize elements in a path graphics. If you resize an element outside of the path graphic boundary, the boundary is redrawn to include the resized element.

### To resize an element within a path graphic

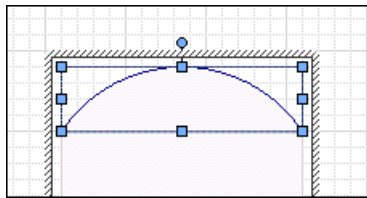
- 1 Select the path graphic you want to edit.
- 2 Do one of the following:



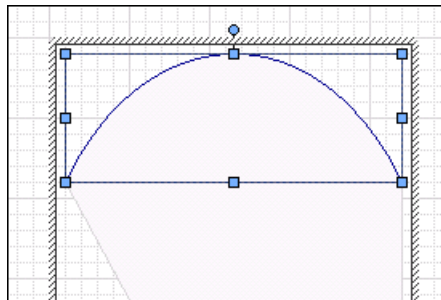
- a On the **Edit** menu, click **Edit Path**.
- b Slowly double-click the path graphic.

The path graphic appears in element mode.

- 3 Select the individual element within the path graphic you want to resize. You can also do this by selecting the element in the Elements List.



- 4 Click and drag any of the resize handles of the selected element. The element is resized.



- 5 Click outside the path graphic on the canvas. The path graphic is shown in path mode.

## Editing Start and Sweep Angles of Elements in a Path Graphic

If your path graphic contains arcs, you can edit the start and sweep angles of these elements. If changing the angle of an element causes it to overlap the path graphic boundary, the boundary is redrawn to include the changed element.

### To edit start or sweep angle of an element within a path graphic

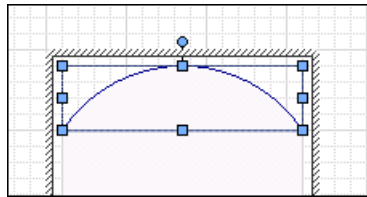
- 1 Select the path graphic you want to edit.
- 2 Do one of the following:



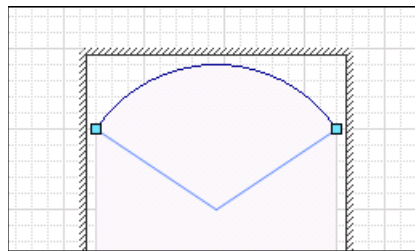
- a On the **Edit** menu, click **Edit Path**.
- b Slowly double-click the path graphic.

The path graphic appears in element mode.

- 3 Select the individual element within the path graphic for which you want to change the start or sweep angle. You can also do this by selecting the element in the Elements List.



- 4 Click the element again. The element appears in edit mode with its start angle and sweep angle.



- 5 Click outside the path graphic on the canvas. The path graphic is shown in path mode.

### Editing Element Control Points in a Path Graphic

If your path graphic contains curves or polylines, you can edit the control points of these elements. If changing the control points of the element causes it to overlap the path graphic boundary, the boundary is redrawn to include the changed element.

#### To edit control points of an element within a path graphic

1 Select the path graphic you want to edit.

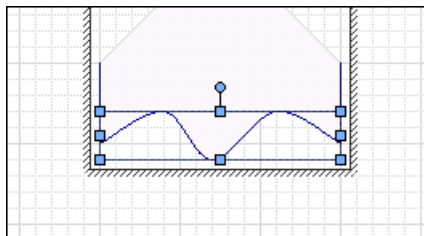
2 Do one of the following:

a On the **Edit** menu, click **Edit Path**.

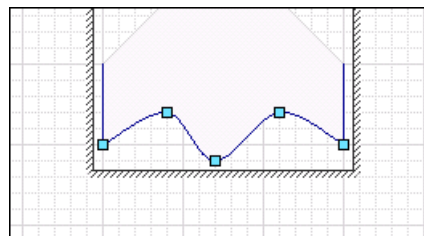
b Slowly double-click the path graphic.

The path graphic appears in element mode.

3 Select the curve or polyline element within the path graphic for which you want to change the control points. You can also do this by selecting the element in the Elements List.



4 Click the element again. The element appears in edit mode with its control points.



5 Drag any of the control points to shape the curve or polyline.

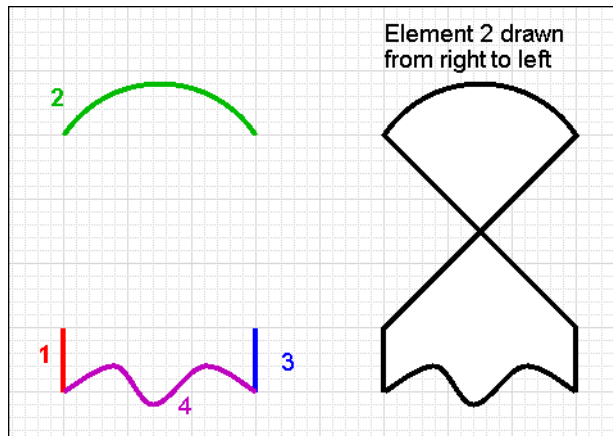
6 Click outside the path graphic on the canvas. The path graphic is shown in path mode.



## Swapping the End Points of an Element in a Path Graphic

The path graphic is created by following the direction in which you draw its elements.

If a path graphic does not appear as expected, this can be caused by drawing an element in a different direction as intended. You can see this if one of the path graphic edges appears crossed over when connecting to the previous and next element.



You can fix this by swapping the end points of the element where this appears.

### To swap the end points of an element within a path graphic

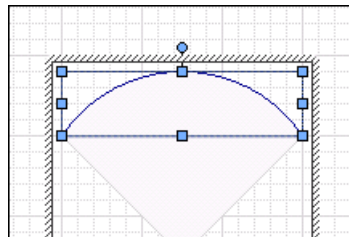
- 1 Select the path graphic you want to edit.
- 2 Do one of the following:



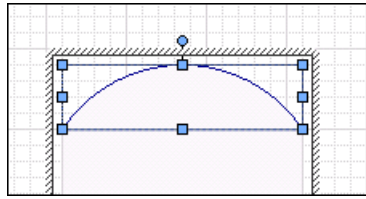
- a On the **Edit** menu, click **Edit Path**.
- b Slowly double-click the path graphic.

The path graphic appears in element mode.

- 3 Select the individual element within the path graphic for which you want to swap the end points. You can also do this by selecting the element in the Elements List.



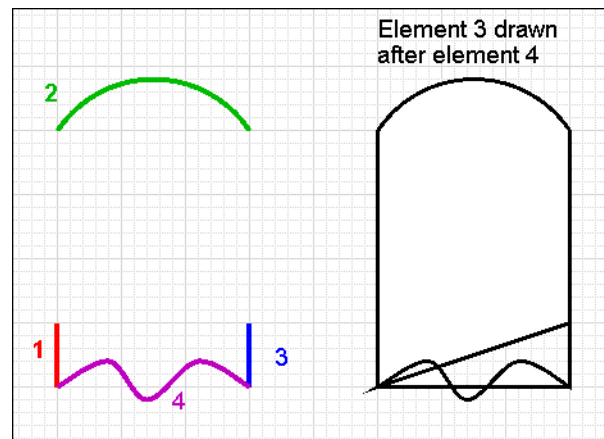
- 4 Right-click that element and select **Path, Swap End Points** on the context menu. The end points of the selected element are swapped and the path graphic is updated accordingly.



- 5 Click outside the path graphic on the canvas. The path graphic is shown in path mode.

### Changing the Z-order of an Element in a Path Graphic

If a path graphic does not appear as expected, this can be caused by drawing an element in a different z-order as intended. You can see this if one of the path graphic edges jumps across the path graphic area.



You can fix this by changing the z-order of the element where this appears.

---

**Note** The z-order of elements in a path graphic is only applicable within the path graphic.

---

**To change the z-order of an element within a path graphic**

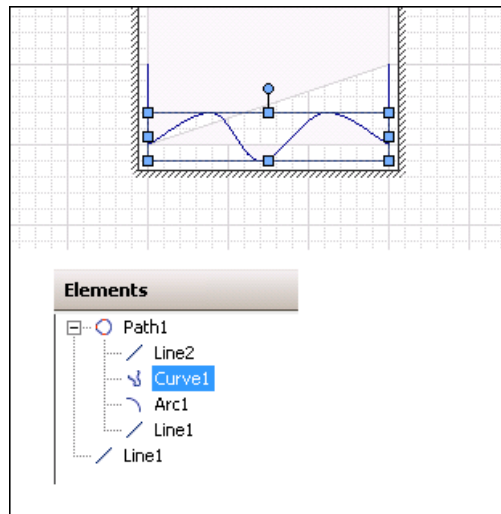
- 1 Select the path graphic you want to edit.
- 2 Do one of the following:



- a On the **Edit** menu, click **Edit Path**.
- b Slowly double-click the path graphic.

The path graphic appears in element mode.

- 3 Select the individual element within the path graphic for which you want to change the z-order. You can also do this by selecting the element in the Elements List.



**Note** You can see the elements in their z-order in the Elements List. Alternatively, you can select one from the Elements List and change its z-order.

- 4 On the **Arrange** menu, point to **Order**, and then click:
  - **Send To Back** to send the element to the back of the set of elements of the path graphic.
  - **Send Backward** to send the element one order backward.
  - **Sent To Front** to send the element to the front of the set of elements of the path graphic.
  - **Send Forward** to send the element one order forward.
- 5 Click outside the path graphic on the canvas. The path graphic is shown in path mode.

## Adding Elements to an Existing Path Graphic

You can easily add elements to an existing path graphic. You can add:

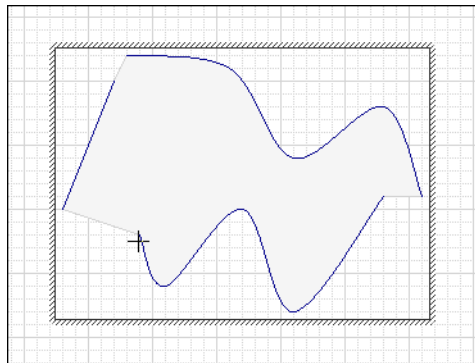
- New elements, which you draw while the path graphic is in edit mode.
- Existing elements, which are already on the canvas.

You can only add open elements such as lines, polylines, curves, and arcs to an existing path graphic.

You can only set the origin of a new element within the frame of the existing path graphic. If you click anywhere outside the path graphic, the edit mode is exited and the element you are drawing is a new element.

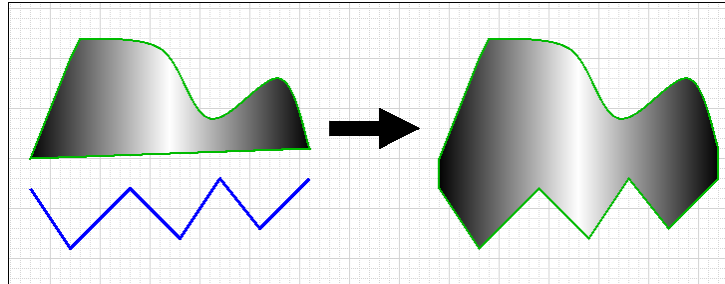
### To add new elements to an existing path graphic

- 1 Select the path graphic to which you want to add a new element.
- 2 On the **Edit** menu, click **Edit Path**. The path graphic appears in element mode.
- 3 Select the new element you want to add from the Tools panel.
- 4 Draw the element as you would normally. While you are drawing the element, the path graphic is updated.



**To add existing elements to an existing path graphic**

- 1 Select the path graphic and all elements that you want to add to the path graphic.
- 2 Right-click a solid part of a selected element, point to **Path**, and then click **Add To Path**. The selected elements are added to the selected path graphic.



## Removing Elements from a Path Graphic

You can remove individual elements from a path graphic. The elements are not deleted, but appear outside the path graphic.

You cannot remove the last element of a path graphic.

**To remove elements from a path graphic**

- 1 Select the path graphic from which you want to delete individual elements.
- 2 On the **Edit** menu, click **Edit Path**. The path graphic appears in element mode.
- 3 **SHIFT** + click one or more elements to remove.

---

**Note** You can also select the elements to remove from the Elements List by holding **CTRL** key during the selection.

---

- 4 Right-click any selected element, point to **Path**, and then click **Remove From Path**. The selected element is removed from the path graphic and the path graphic is updated accordingly.



# Chapter 6

## Editing Common Properties of Elements and Symbols

Some properties are common to most types of elements, such as fill, line styles, and visibility. You can:

- Edit the name of an element.
- Edit the fill properties of an element.
- Edit the line properties of an element.
- Edit the text properties of an element.
- Set the style.
- Set the transparency level of an element.
- Tweaking colors and style for an element's gradient style.
- Enable and disable elements for run-time interaction.
- Change the visibility of an element.
- Change the tab order of an element.
- Use the Format Painter to format elements.
- Edit the general properties of a symbol.

For more information about properties that are specific to certain types of elements, see [Setting Symbol and Element-Specific Properties](#) on page 193.

## Editing the Name of an Element

The name of an element uniquely identifies the element on the drawing surface.

When you draw a new element on the drawing surface, it is assigned a default name. You can then change its name in the Properties Editor or the Elements List.

Element names are case-insensitive and unique within the same element hierarchy. It is possible to have two elements with the same name if one is, for example, in a group and the other outside that group.

### To change an element's name in the Properties Editor

- 1 Select the element on the drawing surface.
- 2 In the Properties Editor, click the value for the **Name** box.
- 3 Type a new name and click **ENTER**.

### To change an element's name in the Elements List

- 1 Select the element in the Elements List.
- 2 Click the element in the Elements List again.
- 3 Type a new name and click **ENTER**.

## Editing the Fill Properties of an Element

You can configure the following fill properties for an element:

- Fill style as solid color, gradient, pattern or texture
- Unfilled style
- Fill orientation, relative to the element or to the screen
- Fill behavior, which determines if the object is to be filled horizontally, vertically, or both
- Horizontal fill direction
- Vertical fill direction
- Percent of horizontal fill
- Percent of vertical fill



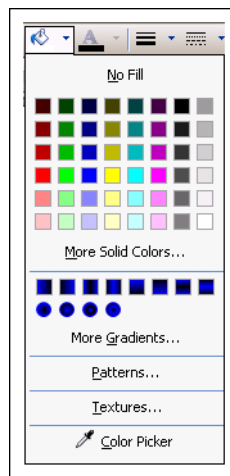
## Setting Fill Style

You can configure the fill style of one or more elements. You can do this to:

- Selected elements on the toolbar.
- Style properties in the Properties Editor.
- Nested style properties, such as just one color of a multi-colored gradient.

### To configure the fill style of an element with the toolbar

- 1 Select one or more elements you want to configure.
- 2 On the toolbar, click the **Fill Color** icon. The fill style list appears.



- 3 Configure the fill color. Do any of the following:
  - Click **No Fill** to configure an empty element.
  - Click a predefined solid color in the display.
  - Click **More Solid Colors** to open the **Style Selection** dialog box and select a solid color.
  - Click **Color Picker** to select a color from the screen.
- 4 Configure the fill gradient, pattern, or texture. Do any of the following:
  - Click a predefined gradient.
  - Click **More Gradients** to open the **Style Selection** dialog box and configure a gradient.
  - Click **Patterns** to open the **Style Selection** dialog box and select a pattern.
  - Click **Textures** to open the **Style Selection** dialog box and select a texture.

For more information about the **Style Selection** dialog box, see [Setting Style](#) on page 166.

**To configure the fill style by setting style properties**

- 1 Select one or more elements.
- 2 In the Properties Editor, locate the `FillStyle` property.
- 3 Click the browse button to open the **Style Selection** dialog box. For more information about the **Style Selection** dialog box, see *Setting Style* on page 166.

**To configure the fill style by setting gradient color style properties**

- 1 Select one or more elements with gradient fill style.
- 2 In the Properties Editor, locate the `Color1`, `Color2`, and `Color3` properties.
- 3 Click the browse button for any of these to set the selected gradient color from the **Style Selection** dialog box. For more information, see *Setting Style* on page 166.

## Setting Unfilled Style

You can configure an element's unfilled style. The unfilled style of an element determines the element's unfilled portion at design time and run time.

**To configure the unfilled style of an element**

- 1 Select one or more elements.
- 2 In the Properties Editor, click **UnfilledStyle**.
- 3 Click the browse button in the **UnfilledStyle** line. The **Style Selection** dialog box appears.
- 4 Select a solid color, gradient, pattern, or texture. For more information about the **Style Selection** dialog box, see *Setting Style* on page 166.
- 5 Click **OK**.

## Setting Fill Orientation

You can configure an element's fill orientation in the Properties Editor. The fill orientation property determines if the fill style is relative to the screen or element.

- If relative to the screen, the gradient, pattern, or texture does not rotate with the element.
- If relative to the element, the gradient, pattern, or texture rotates with the element.

### To configure an element's fill orientation

- 1 Select one or more elements you want to configure.
- 2 In the Properties Editor, click **FillOrientation**.
- 3 From the list in the same line, click **RelativeToScreen** or **RelativeToGraphic**.

## Setting Fill Behavior

You can set the fill behavior of an element. The fill can be:

- Horizontal.
- Vertical.
- Both horizontal and vertical.

### To set an element's fill behavior

- 1 Select one or more elements you want to configure.
- 2 In the Properties Editor, set the property **FillBehavior** to one of the following:
  - **Horizontal**
  - **Vertical**
  - **Both**

## Setting Horizontal Fill Direction and Percentage

An element can fill:

- From left to right.
- From right to left.

You can also set the amount you want the element to be horizontally filled by as a percentage.

### To set an element's horizontal fill direction and percentage

- 1 Select one or more elements you want to configure.
- 2 In the Properties Editor, set the `HorizontalDirection` property to:
  - **Right** to fill from left to right.
  - **Left** to fill from right to left.
- 3 For the `HorizontalPercentFill` property, type a percentage (0 - 100) in the value box.

## Setting Vertical Fill Direction and Percentage

An element can fill:

- From bottom to top.
- From top to bottom.

You can also set the amount you want the element to be vertically filled by as a percentage.

### To set an element's vertical fill direction and percentage

- 1 Select one or more elements you want to configure.
- 2 In the Properties Editor, set the `VerticalDirection` property to:
  - **Top** to fill from bottom to top.
  - **Bottom** to fill from top to bottom.
- 3 For the `VerticalPercentFill` property, type a percentage (0 - 100) in the value box.

## Editing the Line Properties of an Element

You can set the line properties for any element that contains lines, such as:

- Lines and polylines.
- Rectangles, rounded rectangles, and ellipses.
- Curves, closed curves, and polygons.
- Arcs, pies, and chords.
- Text boxes.

You can set the:

- Start and end points for lines, arcs, and H/V lines.
- Line weight, which is the thickness of a line.
- Line pattern, which is the continuity of a line. For example, a continuous line, a dotted line, a dashed line, or a combination.
- Line style, which is the fill style of a line.
- Shape and size of the end points of a line. For more information, see [Setting Line End Shape and Size](#) on page 195.

---

**Note** You can also set the element's line properties in the **Line Format** properties group in the Properties Editor.

---

### Setting Start or End Points of a Line

After you draw a line or H/V line, you can change its start or end points in the Properties Editor.

#### To set the line or H/V line start or end point

- 1 Select a line or H/V line.
- 2 In the Properties Editor, type coordinate values X, Y for the **Start** or **End** properties.

## Setting the Line Weight

You can set a line weight from 0 pixels to 255 pixels for any element that contains lines. You can set the line weight using the **Format** menu, the toolbar, or the `LineWeight` property in the Properties Editor.

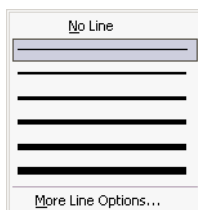
---

**Note** Extreme weight settings can cause unexpected behavior, especially with curves and line end styles.

---

### To set the line weight using the Format menu

- 1 Select one or more elements.
- 2 On the **Format** menu, click **Line Weight**.



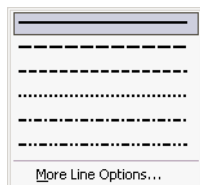
- 3 To use a predefined line weight, select it from the list.
- 4 To use another line weight, click **More Line Options**. The **Select Line Options** dialog box appears. In the **Weight** box, type a new line weight from 0 to 255 and then click **OK**.

## Setting the Line Pattern

You can set the line pattern for any element that contains lines. The line pattern specifies the continuity of a line (continuous, dotted, dashed) and not its fill properties.

### To set the line pattern

- 1 Select one or more elements.
- 2 On the **Format** menu, click **Line Pattern**.



- 3 To use a predefined line pattern, select it from the list.
- 4 To use another line pattern, click **More Line Options**. The **Select Line Options** dialog box appears. In the **Pattern** list, select a pattern, and then click **OK**.

---

**Note** You can also set the line pattern by changing the `LinePattern` property in the Properties Editor.

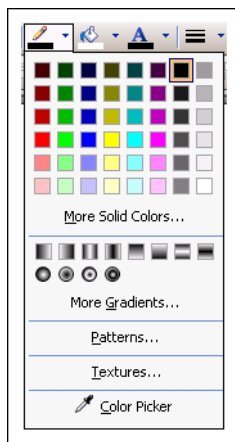
---

## Setting the Line Style

You can set the line style for any element that contains lines. Setting the line style is similar to setting the fill style. You can also set the solid color, gradient, pattern, and texture for a line.

### To set the line style

- 1 Select one or more elements.
- 2 On the toolbar, click the **Line Color** icon. The line style list appears.



- 3 Configure the line color. Do any of the following:
  - Click a predefined solid color in the display.
  - Click **More Solid Colors** to open the **Style Selection** dialog box and select a solid color.
  - Click **Color Picker** to select a color from the screen.
- 4 Configure the line gradient, pattern, or texture. Do any of the following:
  - Click a predefined gradient.
  - Click **More Gradients** to open the **Style Selection** dialog box and configure a gradient.
  - Click **Patterns** to open the **Style Selection** dialog box and select a pattern.
  - Click **Textures** to open the **Style Selection** dialog box and select a texture.

For more information about the **Style Selection** dialog box, see [Setting Style](#) on page 166.

---

**Note** You can also set the element's line style in the Properties Editor. If you do this, you can configure the solid color, gradient, pattern, or texture in the **Style Selection** dialog box. For more information, see [Setting Style](#) on page 166.

---

## Setting the Text Properties of an Element

You can set the following for text, text box, and button elements:

- The text that appears
- The format in which the text appears
- The font of the text
- The alignment of the text
- The text style

You can also substitute strings in text, text box, and button elements.

## Setting the Displayed Text

You can set the text of a text element, text box, or button in the canvas or by changing the Text property in the Properties Editor.

### To set the text to display

- 1 Select the text element, text box or button on the canvas.
- 2 On the **Edit** menu, click **Edit Text**. The selected element appears in edit mode.
- 3 Type a text string and click **ENTER**.



## Setting the Text Display Format

You can configure how values are shown for the text in a text box or button. For example, as a rounded float with the format `#.###`.

You can format the text display for the:

- Text element and the button element in the same way as in the InTouch HMI or with the `TextFormat` property in the Properties Editor.
- Text box element only with the `TextFormat` property.

### To set the text display format

- 1 Select a text element, text box, or button.
- 2 In the Properties Editor, type a format for the `TextFormat` property.



## Setting the Text Font

You can change the font style and font size of a text using:

- The **Format** menu.
- The Font property in the Properties Editor.
- Lists on the toolbar.

### To set the text font, font style, and size

- 1 Select a text element, a text box, or a button element on the canvas.
- 2 On the **Format** menu, click **Fonts**. The **Font** dialog box appears.
- 3 Set the font, font style, size, and effects.
- 4 Click **OK**.



## Setting the Text Color

You can set the text color as a solid color, a gradient, a pattern, or a texture.

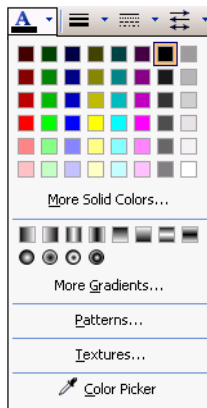
---

**Note** You can also change the text color in the Properties Editor with the TextColor property.

---

### To set the text color

- 1 Select a text element, a text box, or a button element on the canvas.
- 2 Click the **Text Color** icon.



- 3 Configure the text color. Do any of the following:
  - Click a predefined solid color in the display.
  - Click **More Solid Colors** to open the **Style Selection** dialog box and select a solid color.
  - Click **Color Picker** to select a color from the screen.

- 4 Configure the text gradient, pattern, or texture. Do any of the following:
  - Click a predefined gradient.
  - Click **More Gradients** to open the **Style Selection** dialog box and configure a gradient.
  - Click **Patterns** to open the **Style Selection** dialog box and select a pattern.
  - Click **Textures** to open the **Style Selection** dialog box and select a texture.

For more information about the **Style Selection** dialog box, see [Setting Style](#) on page 166.

## Setting the Text Alignment

You can change the horizontal and vertical positioning of text within a text box element or button element.

You can also change the positioning for a text element. If the text is modified at design time or run time, the alignment sets how the element boundary changes to fit around the modified text.

---

**Note** You can also set the text alignment in the Properties Editor by setting the **Alignment** property.

---

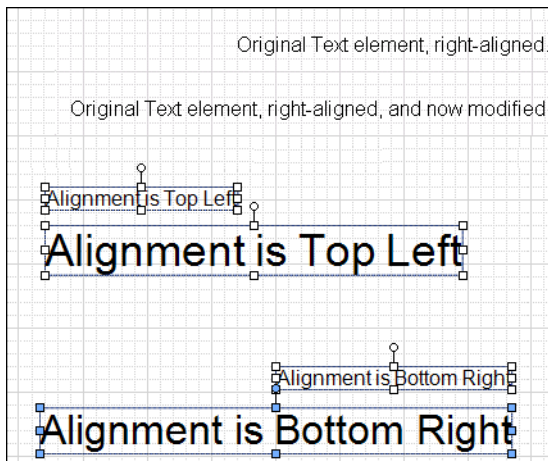
If the element is a text box or a button, then the text is aligned accordingly.

Top Left	Top Center	Top Center
Middle Left	Middle Center	Middle Right
Bottom Left	Bottom Center	Bottom Right

Top Left	Top Center	Top Right
Middle Left	Middle Center	Middle Right
Bottom Left	Bottom Center	Bottom Right

If the element is a text element and you then modify the text at design time or run time, the text is anchored to the point of alignment.

- Text right alignments move additional text further over to the left.
- Text left alignments move additional text to the right.
- Changes in font size leave the point of alignment unchanged and modify the frame accordingly.



#### To set the text alignment

- 1 Select a text element, text box element or button element on the canvas.
- 2 On the **Format** menu, point to **Text Alignment**, and then click the appropriate command:

Click this command	To
<b>Top Left</b>	Align the text at the top left frame handle.
<b>Top Center</b>	Align the text at the top middle frame handle.
<b>Top Right</b>	Align the text at the top right frame handle.
<b>Middle Left</b>	Align the text at the middle left frame handle.
<b>Middle Center</b>	Align the text in the middle of the element.
<b>Middle Right</b>	Align the text at the middle right frame handle.
<b>Bottom Left</b>	Align the text at the bottom left frame handle.
<b>Bottom Center</b>	Align the text at the bottom center frame handle.
<b>Bottom Right</b>	Align the text at the bottom right frame handle.

## Substituting Strings

You can search and replace strings of any element that have the Text property on your canvas. You can use the basic mode to replace strings in a list.

You can also use advanced functions, such as find and replace, ignore, case-sensitivity, and wildcards.

You cannot substitute static strings that are used in an Radio Button Group, List Box or Combo Box.

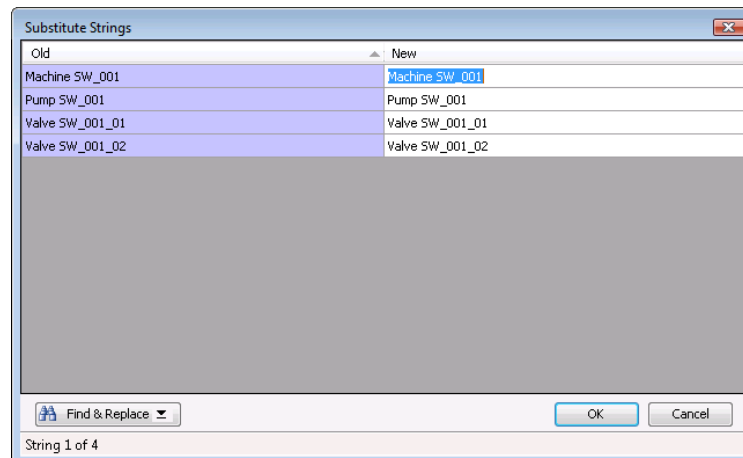
If you substitute strings for a text element in an embedded symbol, that text element is not updated if you change the source symbol text. For example, an embedded symbol contains a text graphic with the string "SomeTextHere". You substitute "SomeTextHere" with "MyText". You then changes the source symbol text from "SomeTextHere" to "Wonderware". The text in the embedded symbol will still show "MyText".

### To substitute strings in a symbol by using the list

- 1 Select one or more elements.
- 2 Do one of the following:
  - Press CTRL + L.
  - On the **Special** menu, click **Substitute Strings**.



The **Substitute Strings** dialog box appears.



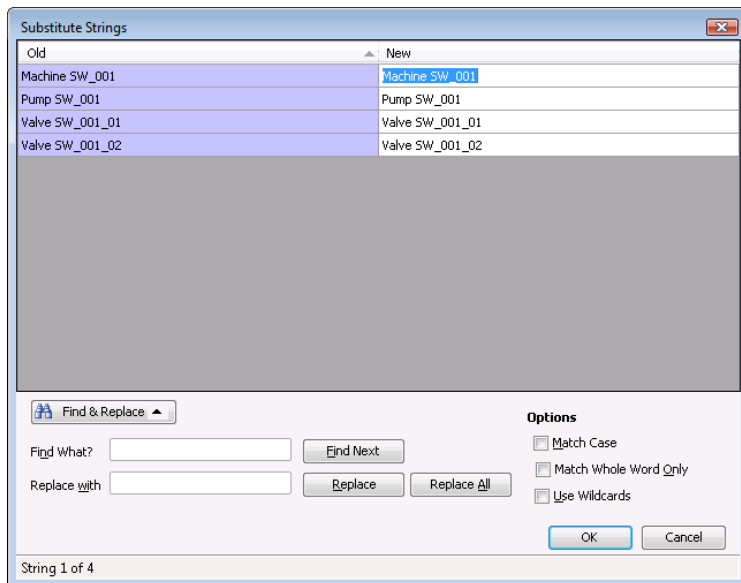
- 3 In the **New** column, type the text to be replaced.
- 4 Click **OK**.

### To substitute strings in a symbol by using advanced functions

- 1 Select one or more elements.
- 2 Do one of the following:
  - Press CTRL + E.
  - On the **Special** menu, click **Substitute Strings**.

The **Substitute Strings** dialog box appears.

- 3 Click **Find & Replace**. The dialog box expands and shows advanced options.



- 4 Configure the search strings. Do any of the following:
  - To find specific strings in the list, type a string in the **Find What** box and click **Find Next** to find the next string.
  - To replace a selected found string with another string, type a string in the **Replace with** box and click **Replace**.
  - To replace multiple strings, type values in the **Find What** and **Replace with** boxes and click **Replace all**.

- 5 Configure the search options. Do any of the following:
  - If you want the search to be case-sensitive, click **Match Case**.
  - To find only entire words that match your search string, click **Match Whole Word Only**.
  - To use wildcards, click **Use Wildcards**. Use an asterisk (\*) to search for any sequence of characters. Use a question mark (?) to search for strings with one variable character.
- 6 Click **OK**.

## Setting Style

You can set the fill, line, and text style from various places in the ArchestrA Symbol Editor using the **Style Selection** dialog box. The **Style Selection** dialog box is common to any element for which you can set a solid color, gradient, pattern, or texture. You can also set the transparency of the style.

Because you can open the **Style Selection** dialog box from different places in the ArchestrA Symbol Editor, the dialog box header can be different.

Also, not all tabs may be available. For example, for setting one color of a gradient in the Properties Editor, you can only select a solid color from the **Style Selection** dialog box.

## Setting a Solid Color

You can set a solid color using the **Solid Color** tab in the **Style Selection** dialog box. You can set a solid color from the:

- Standard palette.
- Color disc and bar.
- Value input boxes.
- Color picker.
- Custom palette.

You can also:

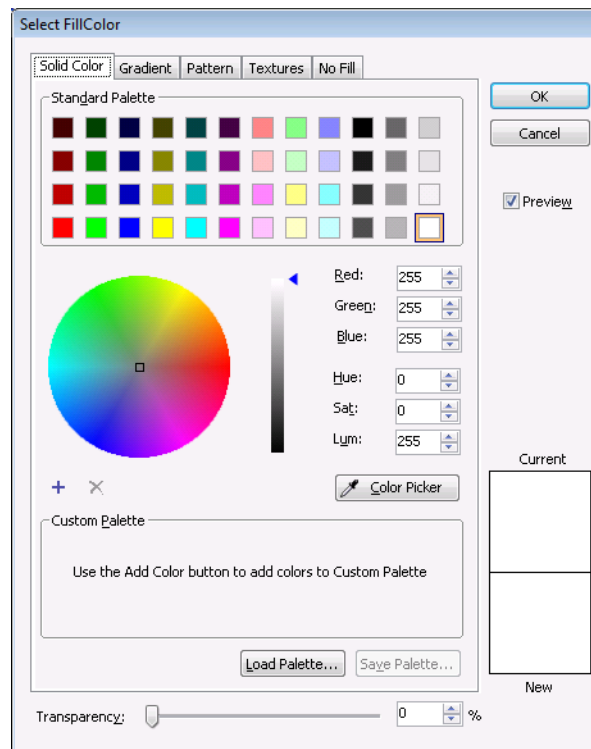
- Add the new color to the custom palette.
- Remove a color from the custom palette.
- Save the custom palette.
- Load a custom palette.

## Setting a Solid Color from the Standard Palette

You can set a solid color from the standard palette using the **Solid Color** tab in the **Style Selection** dialog box. The standard palette is a set of 48 predefined colors you can use to quickly select a solid color.

### To set a solid color from the Standard Palette

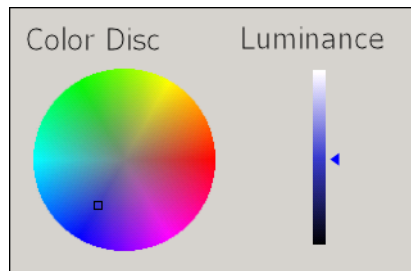
- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.



- 2 In the **Standard Palette** area, click a color. The new color appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

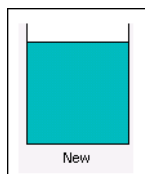
### Setting a Solid Color from the Color Disc and Bar

You can set a solid color using the color disc and bar on the **Solid Color** tab in the **Style Selection** dialog box. The color disc and bar let you graphically select the color and the luminance (brightness).



#### To set a solid color from the color disc and bar

- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.
- 2 Click on the color disk to select a color. The bar is updated and shows the selected color in varying degrees of luminance (brightness).
- 3 Click on the bar to select a luminance (brightness). The new color appears in the **New** color box on the right of the dialog box.



- 4 Click **OK**.

### Setting a Solid Color with the Value Input Boxes

You can set a solid color by typing values that define the color, such as:

- Red component (0-255).
- Green component (0-255).
- Blue component (0-255).
- Hue (0-255).
- Saturation (0-255).
- Luminance (0-255).



**To set a solid color with the value input boxes**

- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.
- 2 In the **Red**, **Green**, **Blue**, **Hue**, **Sat.**, and **Lum.** boxes, type respective values. The resulting color appears in the **New** color box on the right of the dialog box and also on the color wheel and bar.
- 3 Click **OK**.

**Setting a Solid Color with the Color Picker**

You can set a solid color by using the color picker on the **Solid Color** tab in the **Style Selection** dialog box. The color picker lets you select a color from anywhere on the screen, even outside the IDE application.

**To set a solid color with the color picker**

- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.
- 2 Click the **Color Picker** button. The color picker pointer appears.
- 3 Select a color from anywhere on the screen by moving the mouse. As you move the mouse, the new color appears in the **New** color box on the right of the dialog box.
- 4 Click the mouse to complete the color selection.
- 5 Click **OK**.

**Setting a Solid Color from the Custom Palette**

You can set a solid color from the custom palette on the **Solid Color** tab in the **Style Selection** dialog box.

The custom palette is a set of colors that you want to frequently use. You can save the custom palette to a .pal file or load a custom palette from a .pal file.

To use colors from the custom palette, you must first add them. For more information, see *Adding and Removing Colors in the Custom Palette* on page 170.

**To set a solid color from the custom palette**

- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.
- 2 In the **Custom Palette** area, select a color. The new color appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

### Adding and Removing Colors in the Custom Palette

You can add up to 36 solid colors to the custom palette. You can also remove any colors from the custom palette.

You cannot add a color that is already in the custom palette.

#### To add a solid color to the custom palette

- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.
- 2 Add the color. Do any of the following:
  - Select a solid color from the custom palette.
  - Select a solid color from the color disc and bar.
  - Type values for red, green, blue, hue, saturation, and luminance.
  - Select a solid color with the color picker.

The new solid color appears in the **New** color box on the right of the dialog box.



- 3 Click the add button above **Custom Palette**. The solid color is added to the **Custom Palette** area.

#### To remove a solid color from the custom palette

- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.
- 2 In the **Custom Palette** area, select the solid color you want to remove.
- 3 Click the delete button above **Custom Palette**. The solid color is removed from the custom palette.



### Saving and Loading the Custom Palette

You can save the current custom palette or load a previously saved custom palette. The custom palette is loaded from or saved to a Windows Palette file (.pal).

After you save or load a custom palette, the .pal file is not connected to the symbol in any way.

#### To save a custom palette

- 1 In the **Style Selection** dialog box, click the **Solid Color**.
- 2 Click the **Save Palette** button. The **Save Palette** dialog box appears.
- 3 Browse to the location where you want to save the custom palette, type a name, and then click **Save**. The custom palette is saved as a palette file.

**To load a custom palette**

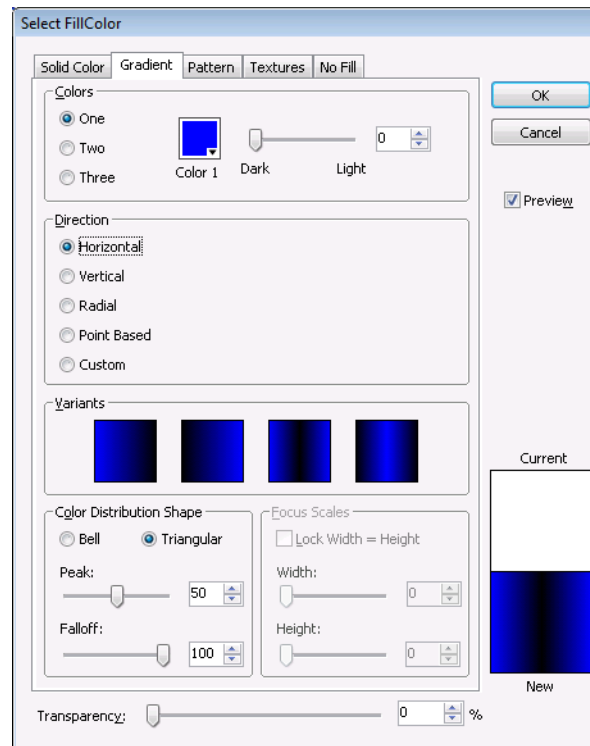
- 1 In the **Style Selection** dialog box, click the **Solid Color** tab.
- 2 Click the **Load Palette** button.
- 3 If you currently have colors in the custom palette, a message appears. Click **Yes** to continue and overwrite the current colors in the custom palette.
- 4 In the **Load Palette** dialog box, browse to the location of the palette file, select it, and then click **Open**. The custom palette is loaded from the selected file.

## Setting a Gradient

You can configure gradients by the:

- Number of colors - 1, 2 or 3.
- Direction - horizontal, vertical, radial, point based, or customized.
- Variant - depending on your selection for the number of colors and direction.
- Color distribution shape - bell or triangular with options to configure the center and falloff.
- Focus scales - width and height.

You set a gradient on the **Gradient** tab in the **Style Selection** dialog box.



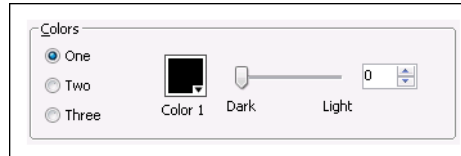
### Setting the Number of Colors for a Gradient

You can set the number of colors you want to use in a gradient.

- If you use one color, the gradient is between this solid color and a specified shade of black to white.
- If you use two colors, the gradient is between these two colors.
- If you use three colors, the gradient is between these three colors in sequence.

**To set a gradient using one color**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Colors** area, click **One**. A color selection box and a slider for the dark to light selection appears.



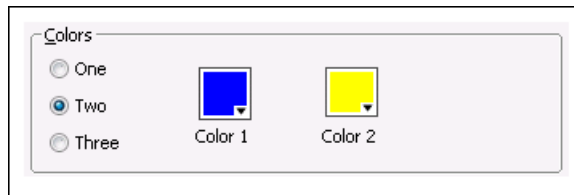
- 3 Click the color selection box to open the **Select Solid Color 1** dialog box. Select a solid color and click **OK**. For more information about this dialog box, see *Setting a Solid Color* on page 166.



- 4 Move the slider between **Dark** and **Light**. The new gradient appears in the **New** color box on the right of the dialog box.
- 5 Click **OK**.

**To set a gradient using two colors**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Colors** area, click **Two**. Two color selection boxes appear.



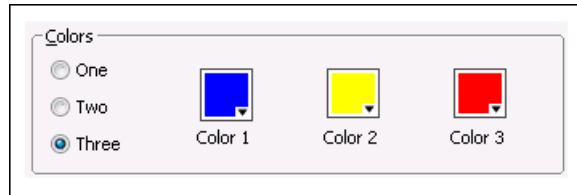
- 3 Click the **Color 1** or **Color 2** color field to select a color from the **Select Solid Color** dialog box. For more information about this dialog box, see *Setting a Solid Color* on page 166.



- The new gradient appears in the **New** color box on the right of the dialog box.
- 4 Click **OK**.

**To set a gradient for three colors**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Colors** area, select **Three**. Three color selection boxes appear.



- 3 Click the **Color 1**, **Color 2** or **Color 3** color field to select a color from the **Select Solid Color** dialog box. For more information about this dialog box, see *Setting a Solid Color* on page 166.



The new gradient appears in the **New** color box on the right of the dialog box.

- 4 Click **OK**.

**Setting the Direction of the Gradient**

You can configure the direction of the gradient to be one of the following:

- Horizontal - from side to side
- Vertical - up and down
- Radial - circular from the center outwards
- Point based - from the inside outwards in linear fashion depending on the bounding points
- Custom angle - across the element at a specified angle

**To set a horizontal gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Direction** area, click **Horizontal**. The new gradient appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

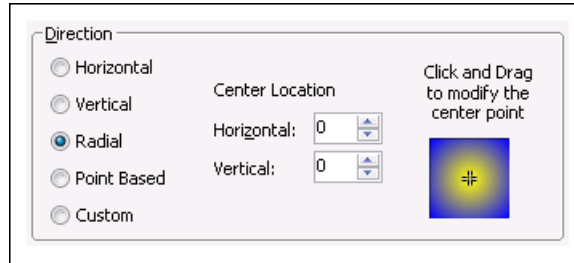
**To set a vertical gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Direction** area, click **Vertical**. The new gradient appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.



**To set a radial gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Direction** area, click **Radial**.



- 3 Set the center location. Do any of the following:
  - In the **Horizontal** and **Vertical** boxes, type values for the center location.
  - Click and drag the center point in the adjacent box.

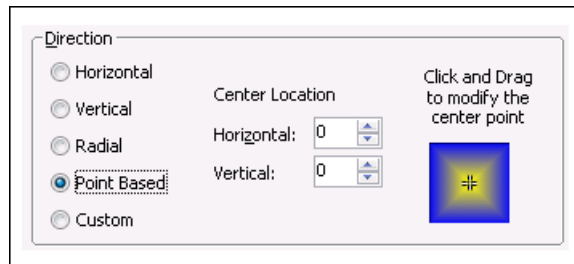


The new gradient appears in the **New** color box on the right of the dialog box.

- 4 Click **OK**.

**To set a point based gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Direction** area, click **Point Based**.



- 3 Set the center location. Do any of the following:
  - In the **Horizontal** and **Vertical** text boxes, type values for the center location.
  - Click and drag the center point in the adjacent box.

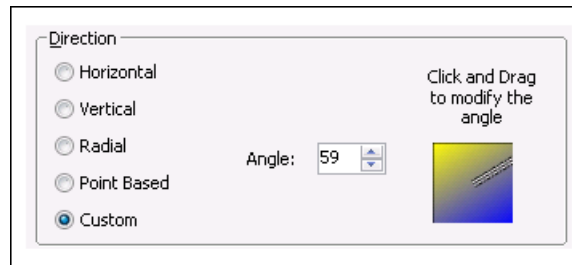


The new gradient appears in the **New** color box on the right of the dialog box.

- 4 Click **OK**.

**To set the custom angle of a gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Direction** area, click **Custom**.



- 3 Set the angle. Do any of the following:
  - In the **Angle** text box, type a value for the angle.
  - Click and drag the angle bar in the adjacent box.



The new gradient appears in the **New** color box on the right of the dialog box.

- 4 Click **OK**.

**Changing the Variant of a Gradient**

You can change the variant of a gradient. The variants are alternate gradients with the same colors you can quickly select.

**To change the variant of a gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Variants** area, click on a variant gradient.



The new gradient appears in the **New** color box on the right of the dialog box.

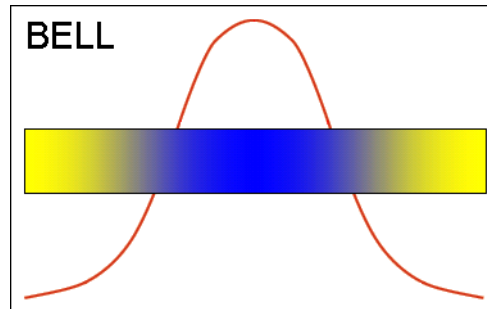
- 3 Click **OK**.



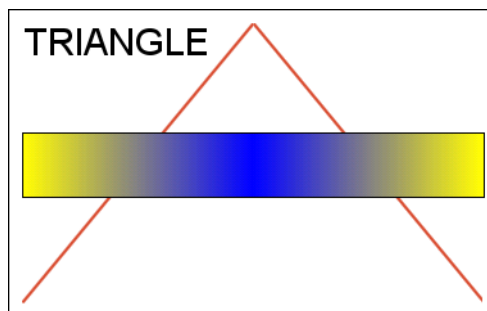
## Setting the Color Distribution Shape

You can configure the distribution shape of a gradient with one or two colors. The shape can be a bell or a triangle.

- In a bell-shaped distribution, the gradient from one color to the next falls at a slow rate, then increasingly at a larger rate, then drops down to small rate again.

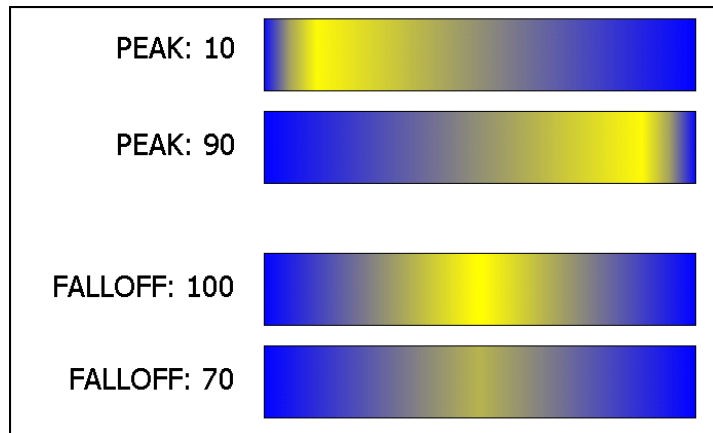


- In a triangular distribution, the gradient from one color to the next rises and falls at the same rate.



You can also configure the peak and the falloff.

- The peak specifies the offset of the gradient if it has one or two colors.
- The falloff specifies the amplitude of the gradient if it has one or two colors.



Additionally, you can configure the center point of a radial or point based gradient if it is defined by three colors.

#### To use a bell-shaped gradient

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Color Distribution Shape** area, click **Bell**. The new gradient appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

#### To use a triangular gradient

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Color Distribution Shape** area, click **Triangular**. The new gradient appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

#### To set the peak of a gradient with one or two colors

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Color Distribution Shape** area, do one of the following:
  - Use the **Peak** slider to specify the peak.
  - In the **Peak** box, type a value from 0 to 100.

The new gradient appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

**To set the falloff of a gradient with one or two colors**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Color Distribution Shape** area, do one of the following:
  - Use the **Falloff** slider to specify the peak.
  - In the **Falloff** box, type a value from 0 to 100.

The new gradient appears in the **New** color box on the right of the dialog box.

- 3 Click **OK**.

**To set the center point of a radial or point based gradient with three colors**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Color Distribution Shape** area, do one of the following:
  - Use the **Center** slider to specify the peak.
  - In the **Center** box, type a value from 0 to 100.

The new gradient appears in the **New** color box on the right of the dialog box.

- 3 Click **OK**.

**Setting the Focus Scales of a Gradient**

You can set the focus scales of a radial or point based gradient. The focus scales acts as a magnification of the gradient. You can set the width or the height of the focus scales.

Width	Height	Appearance
0	0	
50	0	
0	50	
50	50	

You can also lock the width and the height.

**To set the width of the focus scales for a gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Focus Scales** area, do one of the following:
  - Use the **Width** slider to specify the width.
  - In the **Width** text box, type a width value.The new gradient appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

**To set the height of the focus scales for a gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 In the **Focus Scales** area, do one of the following:
  - Use the **Height** slider to specify the height.
  - In the **Height** text box, type a height value.The new gradient appears in the **New** color box on the right of the dialog box.
- 3 Click **OK**.

**To lock the width and the height of the focus scales for a gradient**

- 1 In the **Style Selection** dialog box, click the **Gradient** tab.
- 2 Click **Lock Width = Height**. The width and height are set equal. When you make changes to the width or the height, they are set equal to each other.
- 3 Click **OK**.

## Setting a Pattern

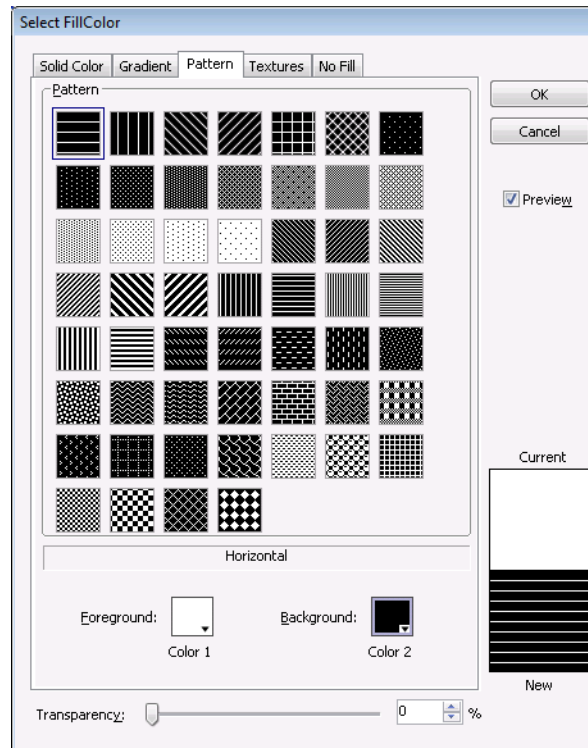
You can set a pattern for an element. The following table describes the pattern options:

Pattern	Options
Horizontal	Simple, Light, Narrow, Dark, Dashed
Vertical	Simple, Light, Narrow, Dark, Dashed
Percent	05, 10, 20, 25, 30, 40, 50, 60, 70, 75, 80, 90
Grid	Small, Large, Dotted
Checker Board	Small, Large
Diagonals	Forward, Backward, Dashed Upward/Downward, Light/Dark/Wide Upward/Downward
Diamond	Dotted, Outlined, Solid
Cross	Diagonal
Brick	Horizontal, Diagonal
Confetti	Small, Large
Others	Zig Zag, Wave, Weave, Plaid, Divot, Shingle, Trellis, and Sphere

Patterns consist of the foreground color and the background color that you can change.

**To set a pattern**

- 1 In the **Style Selection** dialog box, click the **Pattern** tab.



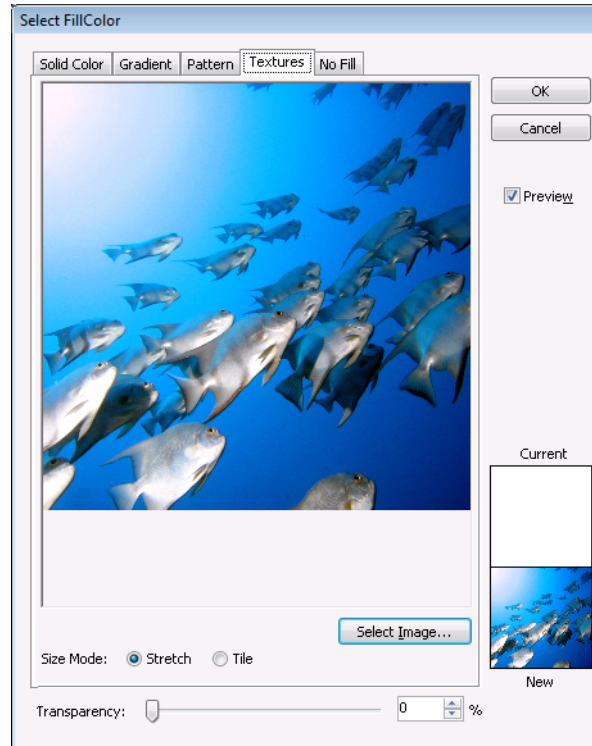
- 2 Select a pattern. The new pattern appears in the **New** color box on the right of the dialog box.
- 3 If you want to change the foreground color of the pattern, click the **Foreground** color selection box. The **Style Selection** dialog box appears. Select a solid color and click **OK**.
- 4 If you want to change the background color of the pattern, click the **Background** color selection box. The **Style Selection** dialog box appears. Select a solid color and click **OK**.  
For more information about setting a solid color, see *Setting a Solid Color* on page 166.
- 5 Click **OK**.

## Setting a Texture

Textures are images you can use as styles for lines, fills and text. You can stretch the image or tile the image across the entire element to be filled.

### To set a texture

- 1 In the **Style Selection** dialog box, click the **Textures** tab.



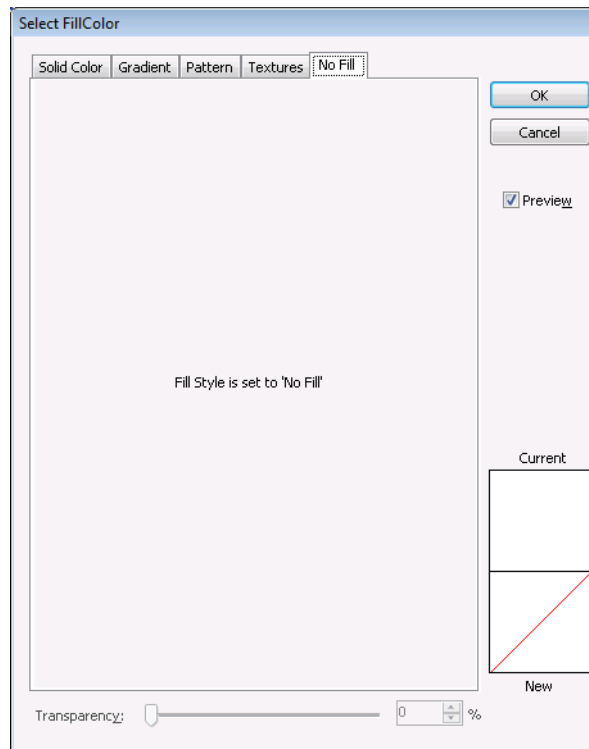
- 2 Click **Select Image**. The **Open** dialog box appears. You can import the following image formats: .BMP, .GIF, .JPG, .JPEG, .TIF, .TIFF, .PNG, .ICO, .EMF. Animated GIF images are not supported.
- 3 Browse to and select an image file and click **Open**. The new pattern appears in the **New** color box on the right of the dialog box.
- 4 Configure the size mode. Do one of the following:
  - Click **Tile** to create a pattern that repeats itself.
  - Click **Stretch** to enlarge (or shrink) the pattern across the selected element.
- 5 Click **OK**.

## Setting the Style to No Fill

You can set the style to “No Fill”. For example if you set the fill style of a rectangle element to No Fill, the background of the rectangle appears transparent.

### To set the No Fill style

- 1 In the **Style Selection** dialog box, click the **No Fill** tab.



The No Fill style appears as a red cross-through line in the **New** color box on the right of the dialog box.

- 2 Click **OK**.

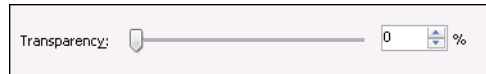


## Setting the Transparency of a Style

You can set the transparency of a solid color, gradient, pattern, or texture.

### To set the transparency of a style

- 1 Open the **Style Selection** dialog box.



- 2 At the bottom of the dialog box, do one of the following:
  - Drag the **Transparency** slider handle.
  - In the **Transparency** text box, type a percentage value.
 The new style appears in the **New** color box.
- 3 Click **OK**.

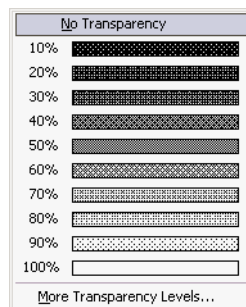
## Setting the Transparency Level of an Element

You can set the transparency level of an element. Levels range from 0 percent (opaque) to 100 percent (transparent).

Transparency of a group of elements behaves in a special way. For more information, see *Properties of Groups* on page 34.

### To set the transparency level of an element

- 1 Select one or more elements.
- 2 On the **Format** menu, click **Transparency**.



- 3 To use a predefined level, select it from the list.
- 4 To use a different level, click **More Transparency Levels**. The **Select Transparency Level** dialog box appears. Type a transparency level in the **Transparency** text box or use the slider to select a transparency level. Click **OK**.

---

**Note** You can also set the transparency level by changing the **Transparency** property in the Properties Editor.

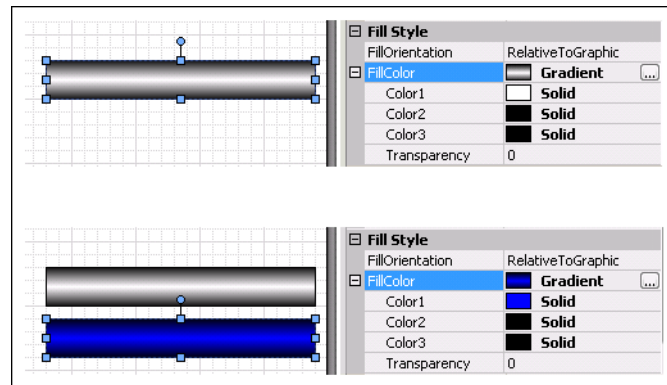
---

## Tweaking the Colors and Transparency of a Gradient

You can easily change the colors and transparency of an element with a gradient style.

For example, you can create pipes with a gradient style of different colors. You can change the pipe color, but still keep the 3-D appearance.

You do this in the Properties Editor using the Color1, Color2, Color3, and Transparency sub-properties.



### To tweak the colors and transparency of a gradient

- 1 Select the element for which you want to change colors or transparency.
- 2 In the Properties Editor, locate the appropriate style setting. This can be:
  - FillColor
  - LineColor
  - TextColor
  - UnFillColor
- 3 Click the + icon to expand the property. The Color1, Color2, Color3, and Transparency sub-properties are shown.
- 4 Do one of the following:
  - Click the color box of one of the color sub-properties.
  - Type a new value for the transparency and click **ENTER**.
- 5 Click the browse button. The **Style Selection** dialog box appears.
- 6 Select a color from the **Style Selection** dialog box and click **OK**. The solid color is applied to the selected element.

## Enabling and Disabling Elements for Run-Time Interaction

You can enable or disable elements so that the run time user cannot use any interaction animations, such as:

- User input.
- Horizontal and vertical sliders.
- Pushbuttons.
- Action scripts.
- Showing and hiding symbols.

Other animations such as horizontal fills and tooltips continue to work as expected.

### To enable an element for run-time interaction

- 1 Select one or more elements you want to enable.
- 2 In the Properties Editor **Runtime Behavior** group, set the Enabled property to True.

### To disable an element for run-time interaction

- 1 Select one or more elements you want to disable.
- 2 In the Properties Editor **Runtime Behavior** group, set the Enabled property to False.

## Changing the Visibility of Elements

You can configure elements to be hidden or shown at run time.

The visibility of an element does not affect its animations. Even when an element is invisible, its animations continue to be evaluated.

### To configure an element to be shown at run time

- 1 Select one or more elements you want to have shown at run time.
- 2 In the Properties Editor **Runtime Behavior** group, set the Visible property to True.

### To configure an element to be hidden at run time

- 1 Select one or more elements you want to have hidden at run time.
- 2 In the Properties Editor **Runtime Behavior** group, set the Visible property to False.

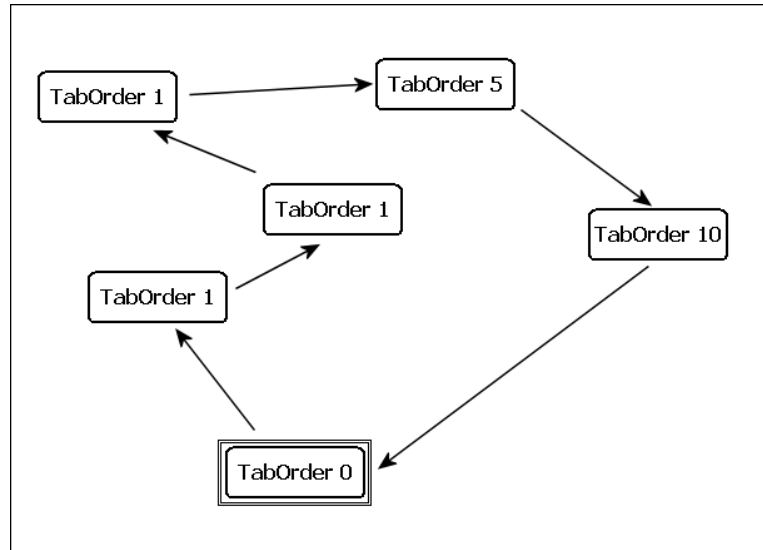
## Editing the Tab Order of an Element

You can configure the elements on the canvas so that at run time you can use the TAB key to put each element in focus in a specified sequence. This sequence is called the tab order.

By default, when you place elements on the canvas, they have a tab order number of 0. Elements with the same tab order number are placed into focus by tabbing at run time according to their z-order. This means they are tabbed through at run time according to their position in the Elements List.

You can override the tab order by assigning a unique index number to the TabOrder property of each element.

Lower tab order numbers take precedence over higher tab order numbers. You must change this value to determine the tab order sequence.



You must also make sure that the TabStop property of each element is set to true. When the TabStop property is set to true, you can use the TAB key at run time to switch to the selected element.

### To edit the element's tab order

- 1 Select the element for which you want to set the tab order.
- 2 In the Properties Editor, ensure that the TabStop property is set to True.
- 3 Type a unique value for the TabOrder property.

## Using the Format Painter to Format Elements

You can apply formatting of one element to other elements quickly by using the format painter. You can apply the format of one element:

- One time to other elements.
- In repetitive mode to other elements.

When you use the format painter, it copies the following formats of the element if applicable to the target elements:

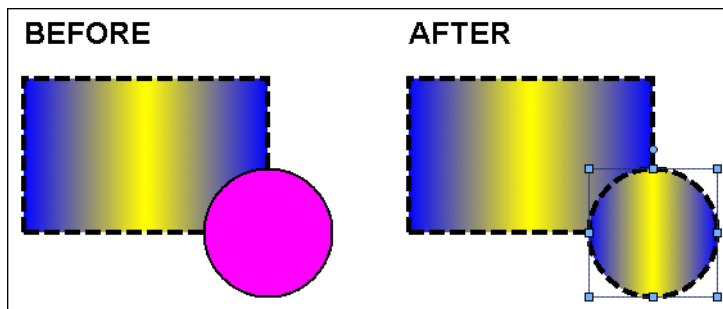
- Font family, size, and style
- Text style, alignment, and word wrap settings
- Line style, weight, pattern, and ends
- Transparency
- Fill style, orientation, behavior, horizontal percent fill, and vertical percent fill
- Unfilled style
- Horizontal and vertical direction properties

You cannot use the format painter for:

- The status element.
- An element that is part of a path.
- Groups of elements.
- Elements in different hierarchy groups.

### To copy the format of an element one time

- 1 Select the element with the format you want to copy.
- 2 On the **Edit** menu, click **Format Painter**. The pointer appears as the format painter cursor.
- 3 Select the element you want to apply the format to. The format is applied to the clicked element.





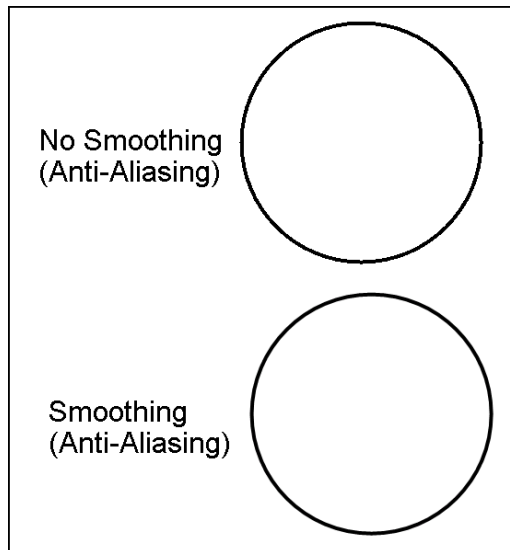
### To copy the format of an element in repetitive mode

- 1 Select the element with the format you want to copy.
- 2 On the toolbar, double-click the Format Painter icon. The pointer appears as the format painter cursor.
- 3 Click each element you want to apply the format to. The format is applied to the clicked element.
- 4 Repeat Step 3 for any other elements you want to apply the format to.
- 5 When you are done, press the ESC key.

## Editing the General Properties of a Symbol

You can configure the general properties of a symbol. The general properties determine the overall appearance and behavior of the symbol. You can:

- Add a meaningful description to your symbol.
- Enable anti-aliasing, or smoothing, for your symbol to improve its appearance. The anti-aliasing filter essentially blurs the elements slightly at the edges.



- Allow or prevent the opening of more than one symbol or display from a symbol. One example is a symbol with multiple Show Symbol animations. If this option is enabled, you can open more than one pop-up and each pop-up is modeless.

**To edit the description of a symbol**

- 1 Click on the canvas so that no elements are selected.
- 2 In the Properties Editor, type a meaningful description for the Description property.

**To use smoothing (anti-aliasing) for a symbol**

- 1 Click on the canvas so that no elements are selected.
- 2 In the Properties Editor, select True for the Smoothing property.

**To allow multiple pop-ups for a symbol**

- 1 Click on the canvas so that no elements are selected.
- 2 In the Properties Editor, select True for the MultiplePopupsAllowed property.





# Chapter 7

## Setting Symbol and Element-Specific Properties

You can configure symbol-specific and element-specific properties. For properties that are common to all or most elements, see [Editing Common Properties of Elements and Symbols](#) on page 151.

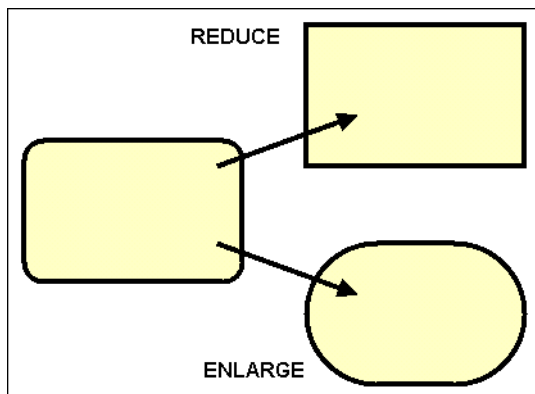
You can configure:

- General properties of a symbol.
- Radius of rounded rectangles.
- Shape and end appearance of lines and H/V lines.
- Auto-sizing and word-wrapping in text boxes.
- Image-specific properties.
- Button-specific properties.
- Control points and tension in curves.
- Angles in pies, chords, and arcs.
- Status elements.
- Windows common controls.

## Setting the Radius of Rounded Rectangles

You can specify the radius, in pixels, of the corners of rounded rectangles. The radius determines their “roundness”. You can:

- Enlarge or reduce the radius of the rounded rectangle on the fly. The easiest way to do this is with the keyboard.
- Set the radius of the rounded rectangle to a specific value using the Properties Editor.



Rounded rectangles maintain their radius when their size is changed. If the symbol containing rounded rectangles is embedded into an InTouch window and resized, the radius is not affected. This can have adverse affects on the graphic representation of your symbol.

### To enlarge the radius of a rounded rectangle

- 1 Select one or more rounded rectangles on the canvas.
- 2 Press and hold **SHIFT** and the **+** key on the number pad. The radius is enlarged, and the rounded rectangle becomes more round.

### To reduce the radius of a rounded rectangle

- 1 Select one or more rounded rectangles on the canvas.
- 2 Press and hold **SHIFT** and the minus (**-**) key on the number pad. The radius is reduced, and the rounded rectangle becomes more rectangular.

### To set the radius of a rounded rectangle exactly

- 1 Select one or more rounded rectangles on the canvas.
- 2 In the Properties Editor, change the value for Radius property and click **ENTER**. The selected rounded rectangles are updated accordingly.

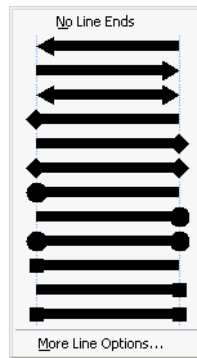
## Setting Line End Shape and Size

You can set the line end shape and size for any element that contains open lines such as lines, H/V lines, polylines, curves, and arcs.

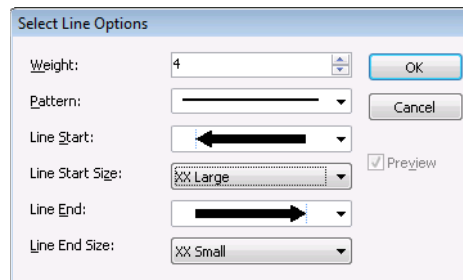
For a line end, you can set the shape to be an arrowhead, diamond, circle, or square. You can set the size if the line end shape is an arrowhead.

### To set the line end shape

- 1 Select one or more elements.
- 2 On the **Format** menu, click **Line Ends**.



- 3 To use a predefined line end shape, select it from the list.
- 4 To use another line shape, click **More Line Options**. The **Select Line Options** dialog box appears. Do the following:



- a In the **Line Start** list, click a shape for the start of the line.
- b In the **Line End** list, click a shape for the end of the line.
- c Click **OK**.

**To set the size of the line arrowheads**

- 1 Select one or more open line elements.
- 2 On the **Format** menu, click **More Line Options**. The **Select Line Options** dialog box appears.
- 3 Select a size on the **Line Start Size** list if the line starts with an arrowhead. Valid sizes are: XX Small, X Small, Small, Medium Small, Medium, Medium Large, Large, X Large, XX Large.
- 4 Select a size on the **Line End Size** list if the line ends with a shape.
- 5 Click **OK**.

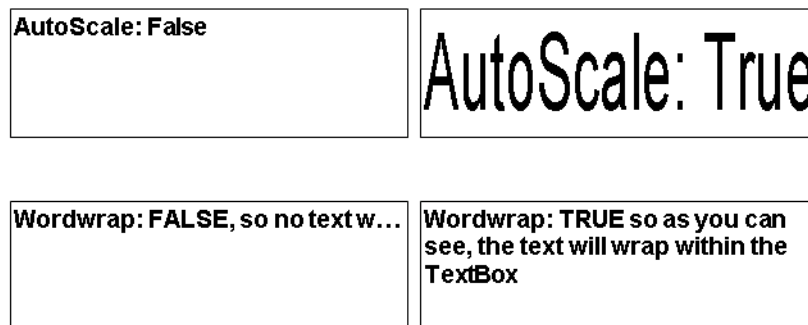
---

**Note** You can also set the line end shapes by changing the **StartCap** and **EndCap** properties in the Properties Editor.

---

## Setting Auto Scaling and Word Wrapping for a Text Box

You can configure a text box to auto scale the text or to word wrap the text within the text box.



- For auto scaling, the text is resized to fit the text box.
- For word wrapping, the text in a text box continues on the next line.

**To auto scale the text in a text box**

- 1 Select one or more text boxes.
- 2 In the Properties Editor, set the **AutoScale** property to true.

**To word wrap the text in a text box**

- 1 Select one or more text boxes.
- 2 In the Properties Editor, set the **WordWrap** property to true.

## Using Images

You can place images on the canvas. This is a two step process:

- 1 Draw a frame which specifies the target size of the image.
- 2 Import the image from an image file.

After you place an image on the canvas, you can:

- Set the display mode (ImageStyle).
- Set the image alignment (ImageAlignment).
- Set the transparency color (HasTransparentColor, TransparentColor properties).
- Open the image in an image editing application.
- Select a different image for the image element.

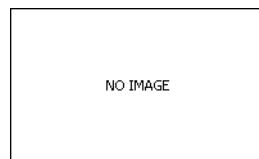
## Placing an Image on the Canvas

You can place an image on the canvas. The image data must come from an image file. You can import the following image formats: .BMP, .GIF, .JPG, .JPEG, .TIF, .TIFF, .PNG, .ICO, .EMF.

You cannot use animated GIF images.

### To place an image on the canvas

- 1 In the Tools panel, select the image icon.
- 2 Click the canvas where you want to place the image and drag the mouse to draw a rectangle that will contain your image.



- 3 Release the mouse button. The **Open** dialog box appears.
- 4 Browse to and select an image file, and then click **Open**. The image is loaded into the image frame.

If the image frame is smaller than the image, the image is cropped to fit into the frame. If the image frame is larger than the image, the image appears in its original size.

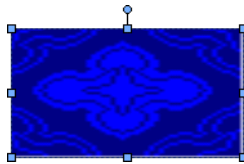
## Setting the Image Display Mode

You can set the way the image appears on the canvas.

- In normal mode, the image is not stretched or tiled. You can resize the image frame with the resizing handles.
- In stretch mode, the image is stretched so that it fills its frame.
- In tile mode, the image is repeated so that a tiled pattern that fills its frame is created.
- In auto mode, the image frame is enlarged or reduced to the image size. The resizing handles are locked. When the image style of an image element is Auto, you cannot change its size.

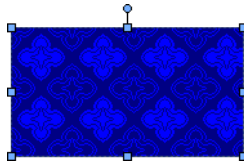
### To stretch an image to the image frame

- 1 Select the image element you want to stretch.
- 2 In the Properties Editor, select **ImageStyle**.
- 3 In the list, click **Stretch**. The image is stretched to the image frame.



### To tile an image in an image frame

- 1 Select the image element you want to tile.
- 2 In the Properties Editor, select **ImageStyle**.
- 3 In the list, click **Tile**. The image is tiled to fill the image frame.



### To set an image frame size to its image size

- 1 Select the image element you want to adjust.
- 2 In the Properties Editor, select **ImageStyle**.
- 3 In the list, click **Auto**. The image frame is enlarged or reduced to the image size.



## Setting the Image Alignment

The image alignment specifies where the image appears in an image frame. By default, images appear in the center of the image frame. You can change this setting to one of the following:

- Top left, top center, or top right
- Middle left, center, or middle right
- Bottom left, bottom center, or bottom right

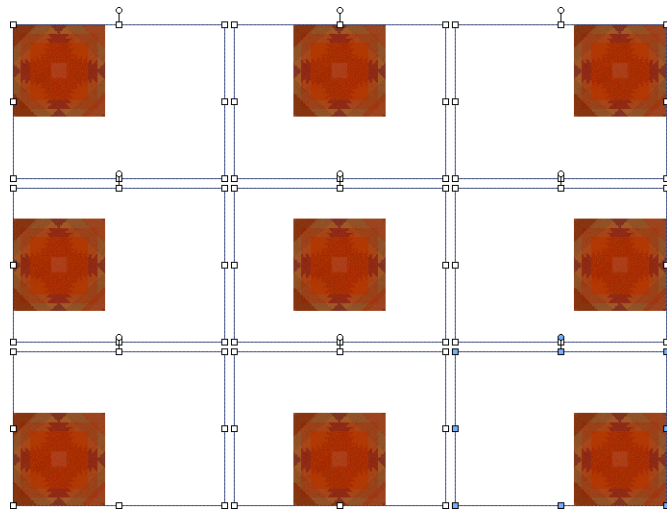
---

**Note** You can also set the image alignment in the **ImageAlignment** property in the Properties Editor.

---

### To set the image alignment

- 1 Select the image element with the image you want to align.
- 2 In the Properties Editor, select **ImageAlignment**.
- 3 In the list, click one of the following options: **TopLeft**, **TopCenter**, **TopRight**, **MiddleLeft**, **Centers**, **MiddleRight**, **BottomLeft**, **BottomCenter** or **BottomRight**. The image is aligned accordingly in the image frame.



## Setting the Image Color Transparency

You can use image color transparency to specify that a color within an image is partially or entirely transparent. When you configure image transparency, you must:

- Enable color transparency for images.
- Specify the color that is to be transparent.

Setting the image color transparency is different than setting the transparency of the image element, as it only applies to one color. Image transparency applies to the entire image.

### To enable image color transparency

- 1 Select the image element.
- 2 In the Properties Editor, select **HasTransparentColor**.
- 3 In the list, click **True**.

### To set the transparency color for an image

- 1 Select the image element.
- 2 On the **Edit** menu, click **Select Image Transparent Color**.  
The pointer becomes a color picker.
- 3 Click the color you want to use as the transparency color.  
The image is updated with the new transparency color.

---

**Note** You can also select a transparency color with the **TransparentColor** property in the Properties Editor. For more information about setting the color, see [Setting a Solid Color](#) on page 166.


---

## Editing the Image

You can edit the image in an image element by opening it in an image editing application.

You can specify the image editor by changing the designer preferences. For more information, see [Setting the Image Editing Application](#) on page 201.

### To edit an image

- 1 Select the image element with the image you want to edit.
- 2  On the **Edit** menu, click **Edit Image**. The image is opened with the associated image editing application.
- 3 Make changes to the image as needed, save the image and close the image editing application. The image is updated on the canvas.

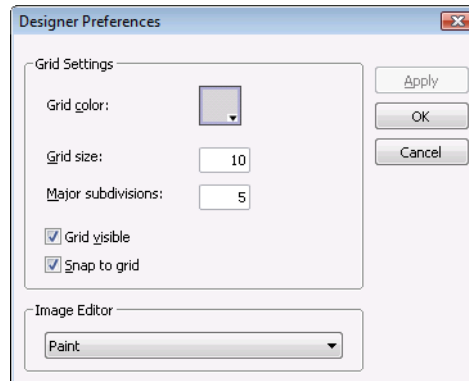


## Setting the Image Editing Application

You can specify the image editor that opens when you select an image for editing. You can select a currently registered image editing application or add one.

### To set the image editing application

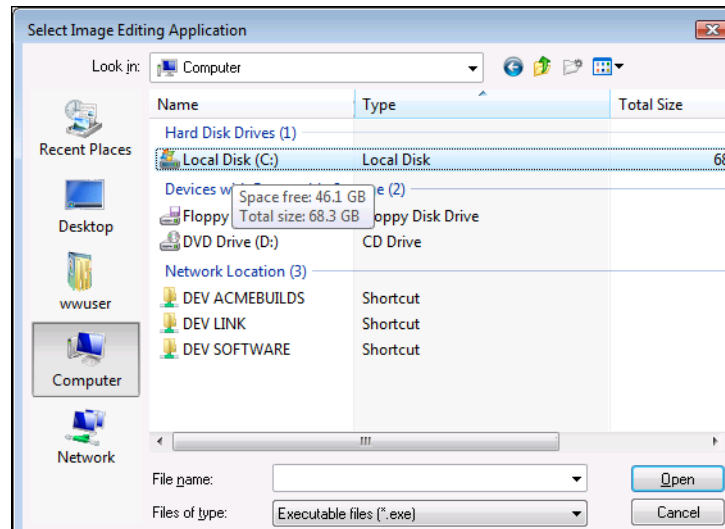
- 1 On the **Special** menu, click **Preferences**. The **Designer Preferences** dialog box appears.



- 2 Select an image editor from the Image Editor list.

### To add an image editing application

- 1 On the **Special** menu, click **Preferences**. The **Designer Preferences** dialog box appears.
- 2 In the **Image Editor** list, click **Choose Custom Editor**. The **Select Image Editing Application** dialog box appears.



- 3 Browse to and select the executable file of the image editing application and click **Open**. The image editor is added to the list.

## Selecting a Different Image

You can change the current image of an image element by selecting a new image.

### To select a different image

- 1 Select the image element with the image you want to change.
- 2 On the **Edit** menu, click **Select Image**. The **Open** dialog box appears.
- 3 Browse to and select an image file, and then click **Open**. The image is loaded into the image frame.

---

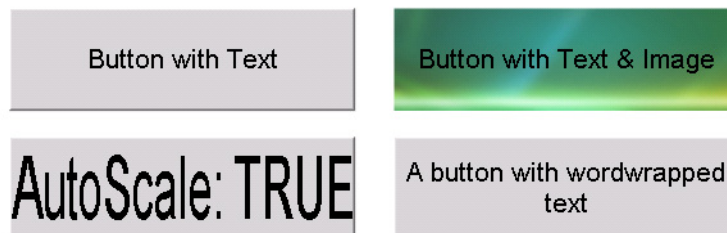
**Note** You can also select a different image by clicking the browse button in the **Image** property in the Properties Editor.

---

## Using Buttons

You can add a text caption or an image to buttons that belong to ArchestrA symbols. If a button includes a text caption, you can:

- Automatically scale the font size
- Configure the text to wrap within the button



## Automatically Scaling Text in Buttons

You can automatically scale text so that the font size is adapted to the button size.

### To automatically scale text in buttons

- 1 Select the button element on the canvas.
- 2 In the Properties Editor, set the `AutoScale` property to `True`.

## Wrapping Text in Buttons

You can wrap text in buttons.

### To wrap text in buttons

- 1 Select the button element on the canvas.
- 2 In the Properties Editor, set the `WordWrap` property to `True`.

## Configuring Buttons with Images

You can use buttons with an image in ArcestrA Symbols.

- The “up” image appears after a button is released and returns to the up position during run time
- The “down” image appears after a button is pressed and locks in the down position during run time

You can edit an up image or a down image after you assign it to a button.

### To use a down image or up image on a button

- 1 Select the button element on the canvas.
- 2 In the Properties Editor, select **Image** in the property **ButtonStyle** list.
- 3 Click the browse button of the `UpImage` property and select an image in the **Open** dialog box. This is the image that appears on the button by default and also when the button is released.
- 4 Click the browse button of the `DownImage` property and select an image in the **Open** dialog box. This image appears after the button is clicked.

### To edit an up image or a down image of a button

- 1 Right-click the button element on the canvas. The context menu appears.
- 2 Click **Edit Button Image**, then click one of the following:
  - **Edit Up Image**
  - **Edit Down Image**

The up image or down image is opened in the default image editor.

- 3 Edit the image.
- 4 Save the image and close the image editor. The up image or down image is updated.

## Editing Control Points

Control points determine the shapes of polylines, polygons, curves, and closed curves. To change the shape of these elements after they have been placed on the canvas, you can:

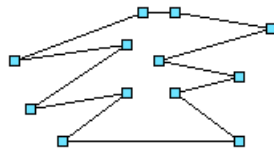
- Move individual control points.
- Add or remove control points.

## Moving Control Points

After you place a polyline, polygon, curve, or closed curve on the canvas, you can change its shape by editing its control points.

**To move the control points of a polyline, polygon, curve, or closed curve**

- 1 Select the polyline, polygon, curve, or closed curve.
- 2 On the **Edit** menu, click **Edit Control Points**. The control points of the element are shown.



- 3 Click a control point you want to change and drag it to the new location. The element is updated accordingly.
- 4 Repeat the previous step for all control points you want to change.

## Adding and Removing Control Points

You can add or remove control points from polylines, polygons, curves, and closed curves.

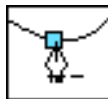
### To add control points to a curve or closed curve

- 1 Select the curve or closed curve.
- 2 On the **Edit** menu, click **Edit Control Points**. The control points of the element are shown.
- 3 Press and hold the SHIFT key.
- 4 Move the mouse over the curve or closed curve at the point you want to add a control point. The pointer appears as a pen with a plus symbol.
- 5 Click the curve or closed curve. The control point is added to the curve or closed curve.
- 6 Repeat the last step for any other control points you want to add.
- 7 When you are done, release the SHIFT key.



### To delete control points from a curve or closed curve

- 1 Select the curve or closed curve.
- 2 On the **Edit** menu, click **Edit Control Points**. The control points of the element are shown.
- 3 Press and hold the SHIFT key.
- 4 Move the mouse over the control point you want to remove. The pointer appears as a pen with a minus symbol.
- 5 Click the control point. The control point is removed from the curve or closed curve.
- 6 Repeat the last step for any other control points you want to remove. You must have at least two control points.
- 7 When you are done, release the CTRL key.



## Changing the Tension of Curves and Closed Curves

After you place a curve or a closed curve, you can change its tension. The tension specifies how tightly the curve bends through the control points. Valid range are float values from 0 (tightly) to 2 (loosely).

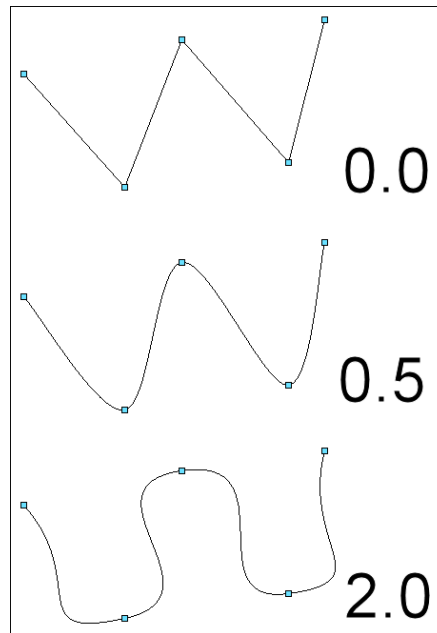
---

**Note** You can also change the tension of a curve or closed curve by changing the value for the Tension property in the Properties Editor.

---

### To edit the tension of a curve or closed curve

- 1 Select the curve or closed curve.
- 2 In the Properties Editor, type a float value from 0 to 2 for the Tension property.



## Changing Angles of Arcs, Pies and Chords

After you place an arc, pie, or chord, you can change the start and sweep angles of elements. You can change the angles to any integer degree from 0 to 359. When you change the angles, you can press the SHIFT and CTRL keys to make the angle snap to multiples of 15 or 45 degrees.

You can also move the start angle and sweep angle at the same time. The object appears to be rotated around its arc/pie/chord center point while keeping the same center point angle.

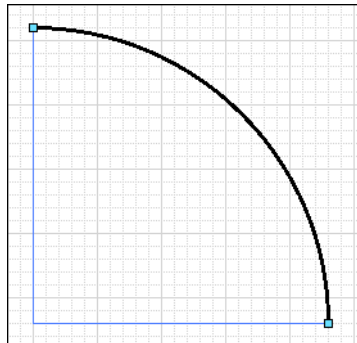
---

**Note** You can also change the start or sweep angle of an arc, pie or chord in the StartAngle or SweepAngle properties in the Properties Editor.

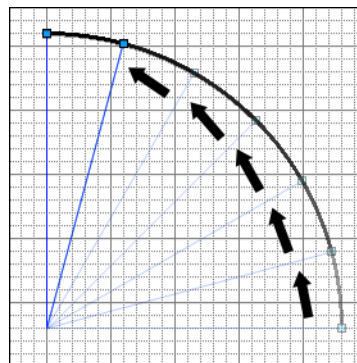
---

### To change the start or sweep angle of an arc, pie, or chord

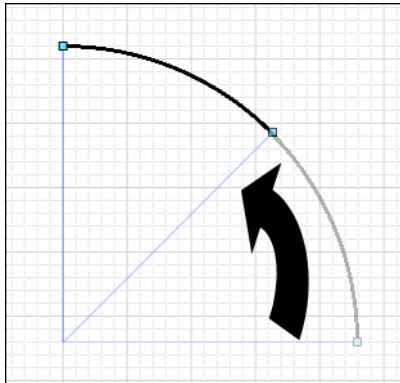
- 1 Select the arc, pie, or chord.
- 2 On the **Edit** menu, click **Edit Start and Sweep Angles**. The start and sweep angle handles appear on the selected element.



- 3 If you want the angle to be multiples of 15 degrees, press and hold the SHIFT key.



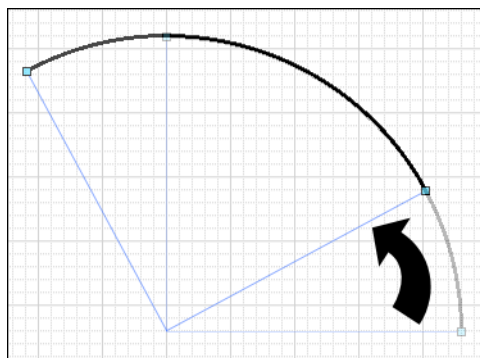
- 4 If you want to the angle to be multiples of 45 degrees, press and hold the CTRL key.



- 5 Grab the start angle or the sweep angle handle and drag it to the new location. The element is updated accordingly.

**To change the start and sweep angles of an arc, pie, or chord together**

- 1 Select the arc, pie, or chord.
- 2 On the **Edit** menu, click **Edit Start and Sweep Angles**. The start and sweep angle handles appear on the selected element.
- 3 Select the start angle or the sweep angle handle and keep the mouse button down.
- 4 Press and hold the ALT key.
- 5 If you want additionally either angles to be multiples of 15 degrees, press and hold the SHIFT key.
- 6 If you want additionally either angles to be multiples of 45 degrees, press and hold the CTRL key.
- 7 Drag the mouse. The start angle and sweep angle are changed accordingly.



- 8 When you are done, release the mouse button and then any keys.



## Monitoring and Showing Quality and Status

You can configure your symbol to show non-good status and quality of attributes in different ways:

- A status element shows a specific icon depending on the quality and status of configured attributes or elements. See [Using Status Elements](#) on page 209.
- The text, fill, or line appearance of elements is overridden depending on the quality and status of the attributes they reference. See [Overriding Element Appearance Depending on Quality and Status of its Attributes](#) on page 212.
- Elements are drawn with an outline depending on the quality and status of the attributes they reference. See [Overriding Element Appearance Depending on Quality and Status of its Attributes](#) on page 212.

### Using Status Elements

Status elements show a specified symbol depending on the quality and status of:

- Attributes configured for specific animated elements.
- One or more specified attributes.

You can assign status elements to an animation in three steps:

- 1 Draw the status element on the canvas.
- 2 Associate the status element with animated elements on the canvas and/or attributes that provide the quality and status data to be monitored.
- 3 If needed, configure the appearance of the status element.

#### Drawing the Status Element on the Canvas

You can easily place a status element on the canvas to show an icon that indicates quality and status of attributes contained in selected animated elements and/or specified attributes.

You do this as you would with any other element. For more information, see [Drawing and Dragging Elements](#) on page 94.

## Configuring the Status Element

You can associate the status element with:

- Animated elements that use attributes that provide the quality and status that is to be monitored.
- Attributes that provide the quality and status that is to be monitored.

In both cases, the appearance is set by the settings in the **Configure Quality and Status Display** dialog box.

For more information on how to configure this animation, see [Configuring Animation for a Status Element](#) on page 328.

## Setting the Appearance of a Status Element

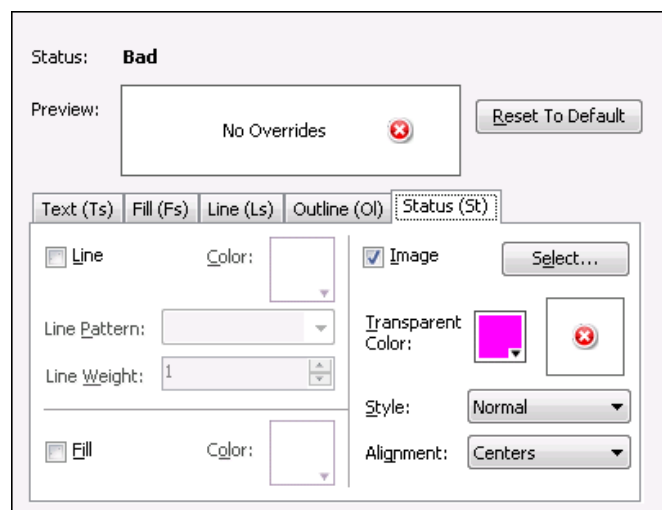
You can set the appearance of a status element depending on the quality and status of its referenced attributes and/or attributes used in its referenced elements.

You can also preview the appearance of a status element. For more information, see [Previewing all Status Appearances](#) on page 218.

You can reset the appearance of a status element to its default. For more information, see [Resetting an Override Appearance to its Default](#) on page 219.

### To set the default appearance of a status element

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Quality and Status Display**. The **Configure Quality and Status Display** dialog box appears.
- 2 Enable **Enable Quality and Status Display**.
- 3 Select a status or quality from the **Status Style Overrides** list.
- 4 Click the **Status** tab.



- 5 To set the line style:
  - a Select **Line**.
  - b Click the color box.
  - c Select a style from the **Style Selection** dialog box. For more information, see *Setting Style* on page 166.
- 6 To set the line pattern, select **Line Pattern** and select a line pattern from the adjacent list.
- 7 To set the line weight, select **Line Weight** and type a line weight in the adjacent box.
- 8 To set the fill style:
  - a Select **Fill**.
  - b Click the color box.
  - c Select a style from the **Style Selection** dialog box. For more information, see *Setting Style* on page 166.
- 9 To show an image for the status element:
  - a Select **Image**.
  - b Click **Select**.
  - c Select an image file from the **Open** dialog box and click **Open**.
- 10 To set the transparent color for the image:
  - a Click the color box next to **Transparent Color**.
  - b Select a solid color from the **Style Selection** dialog box. For more information, see *Setting Style* on page 166.
- 11 To set the image style, select from the **Style** list:
  - Click **Normal** if you want the image to not stretch or tile.
  - Click **Stretch** to stretch the image so it fills its frame.
  - Click **Tile** to repeat the image so that a tiled pattern that fills its frame is created.
  - Click **Auto** to enlarge or reduce the image frame to the image size.

- 12 To set the image alignment, select from the **Alignment** list:
- Click **MiddleLeft** to align in the middle on the left side.
  - Click **TopLeft** to align at the top left corner.
  - Click **BottomLeft** to align at the bottom left corner.
  - Click **Centers** to place the center point of the image directly over the center point of the frame.
  - Click **TopCenter** to align at the center on the top side.
  - Click **BottomCenter** to align at the center on the bottom side.
  - Click **MiddleRight** to align in the middle on the right side.
  - Click **TopRight** to align at the top right corner.
  - Click **BottomRight** to align at the bottom right corner.
- 13 Click **OK**.

## Overriding Element Appearance Depending on Quality and Status of its Attributes

You can configure any animated element to appear differently depending on the quality and status of its associated attributes.

For animated elements, you can:

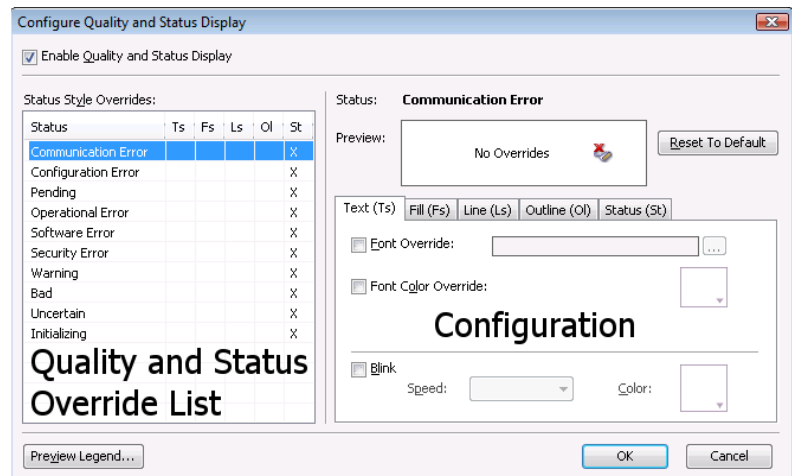
- Override the appearance of the text font, style, and blinking.
- Override the appearance of the fill style and blinking.
- Override the appearance of the line style, weight, pattern, and blinking.
- Preview all status appearances in one dialog box.
- Reset the status appearances to their defaults.
- Use an outline to indicate a specified status or quality.

---

**Note** Instead of overriding the appearance of elements on the canvas, you can use a status element. The status element shows an icon representing quality and status of monitored attributes.

---

You can configure the appearance overrides and Status element overrides in the **Configure Quality and Status Display** dialog box, which you access from the IDE **Configure** menu.



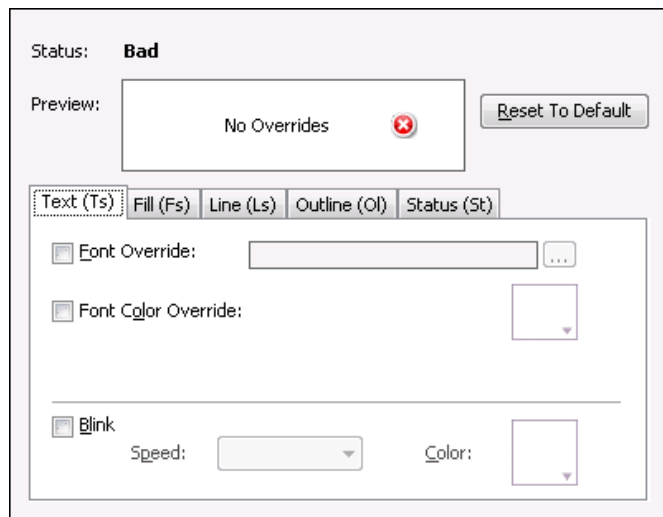
### Overriding the Text Appearance of Elements to Indicate Non-Good Status or Quality

You can configure the Galaxy so that the text appearance of animated elements with attributes that have non-good status or quality are overridden with a specific text appearance.

**To override the text appearance of an element specified by a status element**

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Quality and Status Display**. The **Configure Quality and Status Display** dialog box appears.
- 2 Enable **Enable Quality and Status Display**.
- 3 Select a status or quality from the **Status Style Overrides** list.

- 4 Click the **Text** tab.



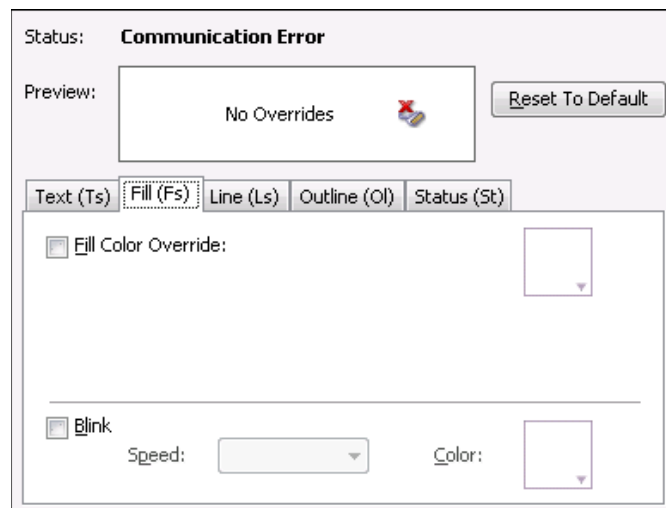
- 5 To override the font, select **Font Override**, click the browse button and select a font from the **Font** dialog box.
- 6 To override the font style:
  - a Select **Font Color Override**.
  - b Click the color box.
  - c Select a style from the **Style Selection** dialog box. For more information, see [Setting Style](#) on page 166.
- 7 To override the text blink behavior:
  - a Select **Blink**.
  - b Select a blinking speed from the **Speed** list.
  - c Click the color box.
  - d Select a style from the **Style Selection** dialog box. For more information, see [Setting Style](#) on page 166.
- 8 Click **OK**.

## Overriding the Fill Appearance of Elements to Indicate Non-Good Status or Quality

You can configure the Galaxy so that the fill appearance of animated elements with attributes that have non-good status or quality are overridden with a specific fill appearance.

### To override the fill appearance of an element specified by a Status element

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Quality and Status Display**. The **Configure Quality and Status Display** dialog box appears.
- 2 Enable **Enable Quality and Status Display**.
- 3 Select a status or quality from the **Status Style Overrides** list.
- 4 Click the **Fill** tab.



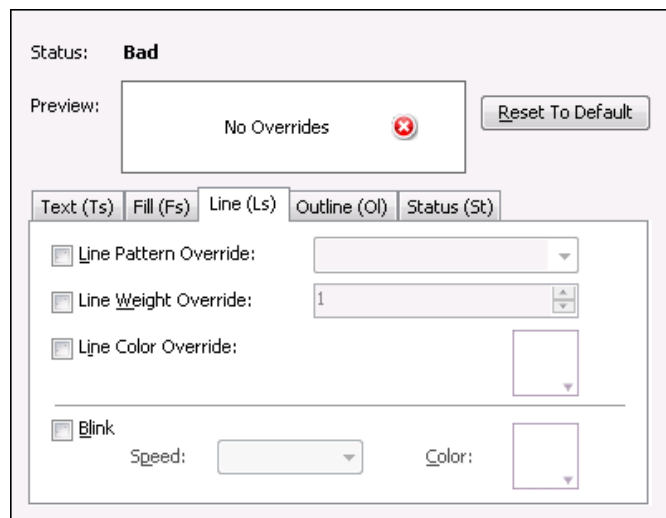
- 5 To override the fill style:
  - a Select **Fill Style Override**.
  - b Click the color box.
  - c Select a style from the **Style Selection** dialog box. For more information, see [Setting Style](#) on page 166.
- 6 To override the fill blink behavior:
  - a Select **Blink**.
  - b Select a blinking speed from the **Speed** list.
  - c Click the color box.
  - d Select a style from the **Style Selection** dialog box. For more information, see [Setting Style](#) on page 166.

## Overriding the Line Appearance of Elements to Indicate Non-Good Status or Quality

You can configure a Galaxy so that the line appearance of animated elements with attributes that have non-good status or quality are overridden with a specific line appearance.

### To override the line appearance of elements specified by a Status element

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Quality and Status Display**. The **Configure Quality and Status Display** dialog box appears.
- 2 Enable **Enable Quality and Status Display**.
- 3 Select a status or quality from the **Status Style Overrides** list.
- 4 Click the **Line** tab.



- 5 To override the line pattern, select **Line Pattern Override** and select a line pattern from the adjacent list.
- 6 To override the line weight, select **Line Weight Override** and type a new line weight in the adjacent box.
- 7 To override the line style:
  - a Select **Line Color Override**.
  - b Click the color box.
  - c Select a style from the **Style Selection** dialog box. For more information, see [Setting Style](#) on page 166.



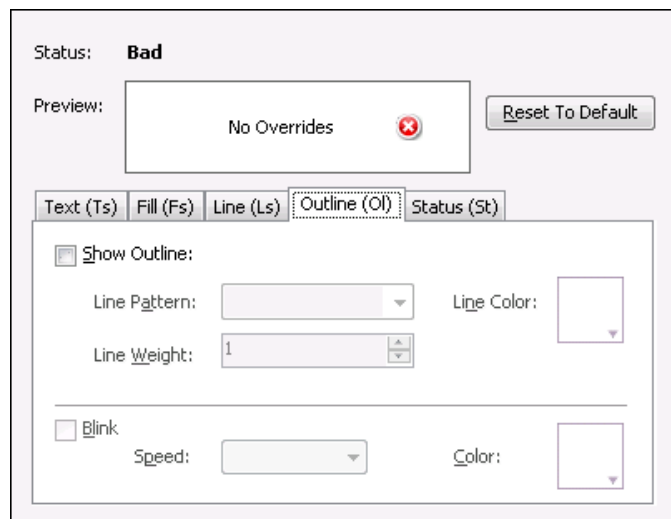
- 8 To override the line blink behavior:
  - a Select **Blink**.
  - b Select a blinking speed from the **Speed** list.
  - c Click the color box.
  - d Select a style from the **Style Selection** dialog box. For more information, see Setting Style on page 166.
- 9 Click **OK**.

## Adding Outlines to Elements to Indicate Non-Good Status or Quality

You can configure the Galaxy so that animated elements with attributes that have non-good status or quality are shown with an outline.

### To add outlines to elements to indicate non-good status or quality

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Quality and Status Display**. The **Configure Quality and Status Display** dialog box appears.
- 2 Enable **Enable Quality and Status Display**.
- 3 Select a status or quality from the **Status Style Overrides** list.
- 4 Click the **Outline** tab.



- 5 Select **Show Outline**.
- 6 To set the line pattern, select **Line Pattern** and select a line pattern from the adjacent list.
- 7 To set the line weight, select **Line Weight** and type a line weight in the adjacent box.

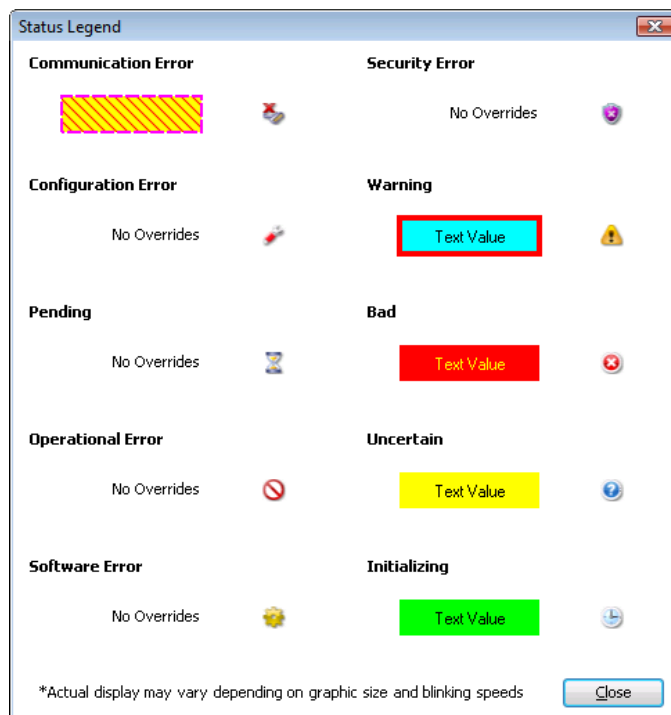
- 8 To set the line style:
  - a Click the color box next to **Line Color**.
  - b Select a style from the **Style Selection** dialog box. For more information, see *Setting Style* on page 166.
- 9 To set the line blink behavior:
  - a Select **Blink**.
  - b Select a blinking speed from the **Speed** list.
  - c Click the color box.
  - d Select a style from the **Style Selection** dialog box. For more information, see *Setting Style* on page 166.

### Previewing all Status Appearances

You can preview all override appearances at the same time.

#### To preview all override appearances

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Quality and Status Display**. The **Configure Quality and Status Display** dialog box appears.
- 2 Click **Preview Legend**. The **Status Legend** dialog box appears.



- 3 Click **Close**.

## Resetting an Override Appearance to its Default

For any status, you can reset:

- Text, fill, and line overrides.
- Outline settings.
- Status element settings.

### To reset a status appearance to its default

- 1 On the **Galaxy** menu, point to **Configure**, and then click **Quality and Status Display**. The **Configure Quality and Status Display** dialog box appears.
- 2 Select a status or quality from the **Status Style Overrides** list.
- 3 Click **Reset to Default**. All text, fill, and line overrides, status element icons, and outline settings are reset to their defaults.

## Using Windows Common Controls

You can add the following Windows common controls to your symbol:

- Radio button group
- Check box
- Edit box
- Combo box
- Calendar control
- DateTime picker
- List box

You can place these Windows common controls as you would any other element, by selecting them from the Tools panel, clicking on the canvas to position it and, with exception of the calendar control, dragging a rectangle to set the size.

After placing the control on the canvas, you can then configure:

- Background color and text color (with exception of the DateTime Picker control).
- Other control-specific properties in the Properties Editor.
- Control-specific animations.
- The common Value property in scripting to read from and write to the Windows common control at run time.

## Changing Background Color and Text Color of Windows Common Controls

You can change the background color and text color of all Windows common controls with exception of the DateTime Picker control.

The background color and text color of the Windows common controls can be only solid colors, not gradients, patterns, nor textures.

### To set the background color of a Windows common control

- 1 Select the Windows common control.
- 2 In the Properties Editor, click the browse button of the FillColor property. The **Style Selection** dialog box appears.
- 3 Select a solid color and click **OK**. For more information, see Setting a Solid Color on page 166. The Windows common control background color changes accordingly.

### To set the text color of a Windows common control

- 1 Select the Windows common control.
- 2 In the Properties Editor, click the browse button of the TextColor property. The **Style Selection** dialog box appears.
- 3 Select a solid color and click **OK**. For more information, see Setting a Solid Color on page 166. The Windows common control text color changes accordingly.

## Reading and Writing the Selected Value at Run Time

You can use the Value property that is common to all Windows common controls. It is not visible in the Properties Editor. You can use the value property in a script or other animation links.

The following table shows you the data type, a description on how the value property is used, and an example for each Windows common control.

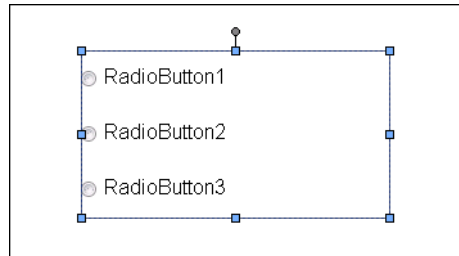
Control	Data Type	Description	Example
Radio Button Group	Boolean, Integer, Real or String	Reads the value of the selected item, or selects the item with that value if it exists.	<code>RadioButtonGroup1.Value = "Mixing";</code>
Check Box	Boolean	Sets or reads the checked status.	<code>CheckBox1.Value = 1;</code>
Edit Box	String	Sets or reads the text contents.	<code>EditBox1.Value = "Hello World";</code>
Combo Box	Integer	Reads the value of the selected item, or selects the item with that value if it exists.	<code>ComboBox1.Value = 5;</code>
Calendar	Time	Sets or reads the selected date.	<code>Calendar1.Value = "11/15/2006 11:12:34 AM";</code>
DateTime Picker	Time	Sets or reads the selected date and time.	<code>DateTimePicker1.Value = "11/15/2006 2:55:12 PM";</code>
List Box	Integer	Reads the value of the selected item, or selects the item with that value if it exists.	<code>ListBox1.Value = "John Smith";</code>

For more information about scripting, see [Adding and Maintaining Symbol Scripts](#) on page 353.

For more information about the value property, see [Alphabetical List of Properties](#) on page 444.

## Configuring Radio Button Group Controls

You can use a Radio Button Group control element to exclusively select an option from a group of options at run time.



You can set the:

- 3D appearance of buttons.
- Layout of the radio button group options.

You can also use properties that are specific to the Radio Button Group control in scripting. At run time you can access the script to view and modify the Radio Button Group control.

### Setting the 3D appearance of a Radio Button Group Control

You can set the 3D appearance of a radio button group control. This affects how the option circles appear.

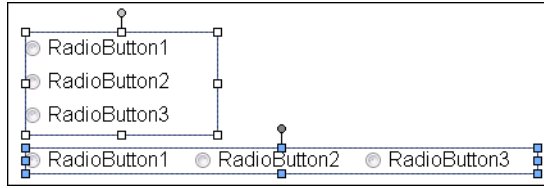
- Three-dimensional appearance
- Flat appearance in same color as option text

#### To set the 3D appearance of a radio button group control

- 1 Select the radio button group control.
- 2 In the Properties Editor, select from the list for the `ControlStyle` property:
  - Click **ThreeD** for a three-dimensional appearance.
  - Click **Flat** for a flat two-dimensional appearance in the same color as the option text.

## Setting the Layout of the Radio Button Group Options

You can set the layout of the radio button group options in a vertical or horizontal direction.



### To set the layout of the radio button group options

- 1 Select the radio button group control.
- 2 In the Properties Editor, select from the list for the Layout property:
  - Click **Vertical** to arrange the options under each other.
  - Click **Horizontal** to arrange the options next to each other.

---

**Note** You can set this option also in the radio button group animation dialog box.

---

## Using Radio Button Group-Specific Properties at Run Time

You can use properties that are specific to the Radio Button Group control at run-time. These properties are:

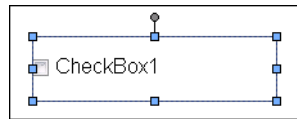
- **Count** - returns the number of radio buttons in the Radio Button Group control.
- **SelectedValue** - reads the value of the selected item, or selects the item with that value if it exists.

These properties are available when you browse for a Radio Button Group control in the Galaxy Browser. For more information about scripting, see *Adding and Maintaining Symbol Scripts* on page 353.

For more information about the properties, see *Alphabetical List of Properties* on page 444.

## Configuring Check Box Controls

You can use a Check Box control for users to reset a Boolean attribute during run time.



You can set the following properties of the Check Box control:

- Default state, checked or unchecked.
- Caption text of the Check Box control button.
- 3D appearance of the Check Box control button..

### Setting the Default State of a Check Box Control

You can set the default state of a check box control to be checked or unchecked.

#### To set the default state of a check box control

- 1 Select the Check Box control.
- 2 In the Properties Editor, select from the list for the **Checked** property:
  - Click **False** to use an unchecked check box by default.
  - Click **True** to use a checked check box by default.

### Setting the Caption Text of a Check Box Control

You can set the caption text of a Check Box control.

#### To set the caption text of a Check Box control

- 1 Select the Check Box control.
- 2 In the Properties Editor, type a text string in the Caption property value box.



## Setting the 3D appearance of a Check Box Control

You can set the appearance of the check box within the control to be either flat or three-dimensional.



Three-dimensional appearance



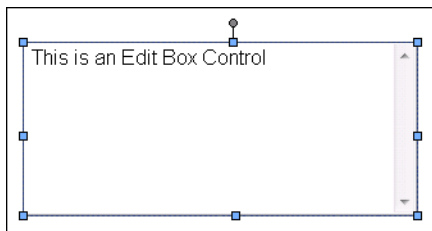
Flat appearance in same color as caption text

### To set the 3D appearance of a Check Box control

- 1 Select the check box control.
- 2 In the Properties Editor, select from the list for the `ControlStyle` property:
  - Click **ThreeD** for a three-dimensional check box.
  - Click **Flat** for a flat two-dimensional check box in the same color as the caption text.

## Configuring Edit Box Controls

You can use an Edit Box control to create a box during run time in which users can enter text or view text.



You can configure the following properties of an Edit Box control:

- Set the default text.
- Wrap text to the next line in the edit box at design time and run time.
- Configure it so that the run-time text is read-only.

### Setting the Default Text in an Edit Box Control

You can set the default text that appears in an edit box control during run time.

#### To set the default text in an Edit Box control

- 1 Select the edit box control.
- 2 In the Properties Editor, type a text in the Text property. The text appears in the edit box control at design time. At run time, it can be overwritten with the value of a configured attribute.

### Configuring the Text to Wrap in an Edit Box Control

You can configure the edit box control to wrap text at design time and run time. This lets you view and type strings in a more compact way.

#### To configure text-wrapping in an edit box control

- 1 Select the edit box control.
- 2 In the Properties Editor, select from the list for the Multiline property:
  - Click **True** to enable text-wrapping at run time.
  - Click **False** to disable text-wrapping at run time.

### Configuring the Text to be Read-Only in an Edit Box Control

You can configure the Edit Box control to only show text at run time and prevent the run-time user from writing back to the associated attribute.

#### To configure the text to be read-only in an Edit Box control

- 1 Select the edit box control.
- 2 In the Properties Editor, set the ReadOnly property to **True**.

---

**Note** To enable writing back to the associated attribute at run time, you can set the **ReadOnly** property to **False**.

---

## Configuring Combo Box Controls

You can use Combo Box controls to select an option from a foldable list.



You can configure :

- Drop-down type of combo box control.
- Width of the drop-down list.
- Integral height flag of the drop-down list to avoid clipping of the items in simple combo box controls.
- Maximum number of items to appear in the drop-down list.

You can also use properties that are specific to the Combo Box control in scripting. At run time, you can access the script to view and modify the items in the Combo Box control.

### Setting the Type of Combo Box Control

You can use one of the following combo box control types:

- Simple - no drop-down list, allows entry of values
- DropDown - has a drop-down list, allows entry of values
- DropDownList - has a drop-down list, does not allow entry of any value

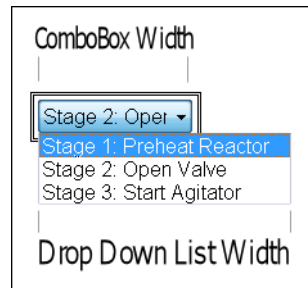
#### To set the type of Combo Box control

- 1 Select the combo box control.
- 2 In the Properties Editor, select from the list for the DropDownType property:
  - Simple
  - DropDown
  - DropDownList

### Setting the Width of the Drop-Down List

You can set the width of the expanded drop-down list when the user clicks on it. This setting can be used to save space of the folded combo box control at run time.

Typically you set the drop-down list width greater than the width of the combo box on the canvas.



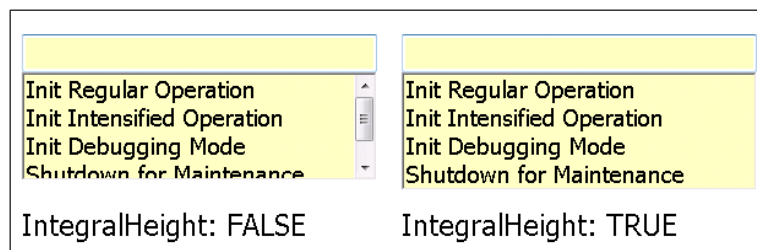
If you set the drop-down list width smaller than the combo box control width on the canvas, the drop-down list is the same width as the combo box control.

#### To set the width of the combo box drop-down list

- 1 Select the combo box control.
- 2 In the Properties Editor, type a width value in the DropDownWidth property value box.

### Avoiding Clipping of Items in the Simple Combo Box Control

You can avoid clipping of items in the simple combo box control list by setting the IntegralHeight property to true. The combo box list height is then changed to ensure that no items appear clipped.



#### To avoid clipping of items in the drop-down list

- 1 Select the combo box control.
- 2 In the Properties Editor, select **True** as the value for the IntegralHeight property.

### Setting the Maximum Number of Items to Appear in the Combo Box Drop-Down List

You can limit the number of items that appear at any given time in the combo box drop-down list.

#### To set the maximum number of items to appear in the drop-down list

- 1 Select the combo box control.
- 2 In the Properties Editor, type the maximum number as a value for the `MaxDropDownItems` property.

### Using Combo Box-Specific Properties at Run Time

You can use properties that are specific to the Combo Box control at run time.

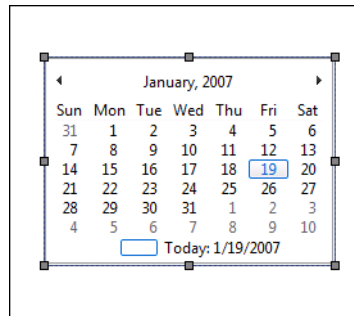
- The `Count` property returns the number of items in a Combo Box control.
- The `NewIndex` property returns the index of the last item added to the Combo Box list.

These properties are available when you browse for a Combo Box control in the Galaxy Browser. For more information about scripting, see [Adding and Maintaining Symbol Scripts](#) on page 353.

For more information about the properties, see [Alphabetical List of Properties](#) on page 444.

## Configuring Calendar Controls

You can use the Calendar control to select a date from one or more monthly calendar sheets.



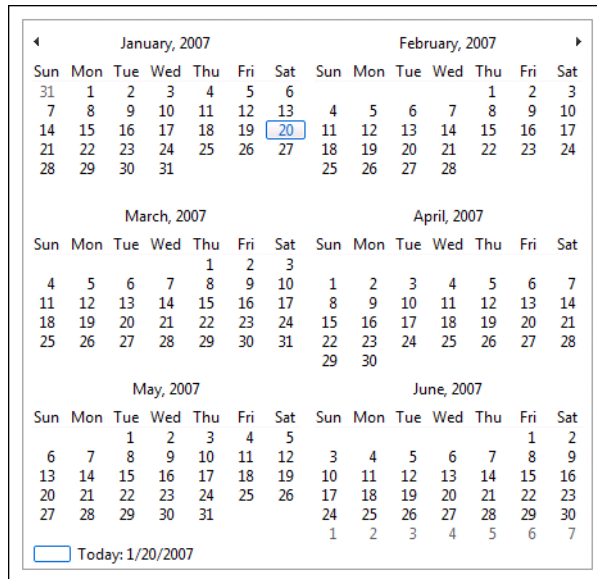
You can:

- Set the number of calendar month sheets to be shown.
- Set the first day of the week.
- Show or hide today's date on the bottom of the control.
- Set the fill and text colors of the calendar title.
- Set the text color for trailing dates.
- Set the date value of the Calendar Control that is used as default at run time.

### Setting the Number of Calendar Month Sheets

You can set the number of calendar month sheets to be shown by specifying the number of month columns and month rows. The number of columns and rows in the calendar control depends on the font size and the width of the calendar control.

For example, you can show six months in a calendar control by specifying two columns and three rows.



#### To set the number of calendar month sheets

- 1 Select the Calendar control.
- 2 In the Properties Editor, configure the calendar properties:
  - For the CalendarColumns property, specify the number of columns in the calendar control.
  - For the CalendarRows property, specify the number of rows in the calendar control.

#### Setting the First Day of the Week

You can set the first day of the week for the calendar control. This is the day that appears on the most left column of each calendar month sheet.

You can set it to:

- The default as defined by the operating system.
- Any day of the week.

#### To set the first day of the week

- 1 Select the Calendar control.
- 2 In the Properties Editor, select from the list for the FirstDayOfWeek property:
  - Click **Default** to use the operating system setting.
  - Click the day of the week.

## Showing or Hiding Today's Date on a Calendar Control

You can show or hide today's date on the bottom of a calendar control

### To show or hide today's date on the bottom of a calendar control

- 1 Select the Calendar control.
- 2 In the Properties Editor, set the ShowToday property to one of the following:
  - **True** to show today's date
  - **False** to hide today's date

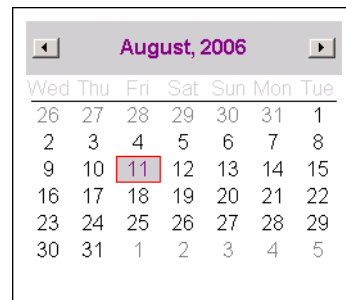
## Setting Title Fill Color and Text Color on a Calendar Control

You can set the title fill color and title text color on a calendar control.

When you change the title fill color, this also affects the:

- Color of the week days.
- Fill color of the indication box of today's date.

When you change the title text color, this also affects the text color of the indication box of today's date.



### To change the title fill color of a Calendar control

- 1 Select the Calendar control.
- 2 In the Properties Editor, click the browse button for the TitleFillColor property. The **Style Selection** dialog box appears. For more information, see Setting Style on page 166.
- 3 Select a color and click **OK**. The title fill color is changed accordingly.



**To change the title fill color of a Calendar control**

- 1 Select the Calendar control.
- 2 In the Properties Editor, click the browse button for the `TitleTextColor` property. The **Style Selection** dialog box appears. For more information, see [Setting Style](#) on page 166.
- 3 Select a color and click **OK**. The title text color is changed accordingly.

**Setting the Text Color for Trailing Dates in a Calendar Control**

You can set the text color for dates outside the month for any month sheet in a calendar control.

**To set the text color for trailing dates**

- 1 Select the Calendar control.
- 2 In the Properties Editor, click the browse button for the `TrailingTextColor` property. The **Style Selection** dialog box appears. For more information, see [Setting Style](#) on page 166.
- 3 Select a color and click **OK**. The text color of the trailing dates is changed accordingly.

**Setting the Default Value of the Calendar Control**

You can set the default value of the Calendar Control. The default value is a date that the control uses when it is shown the first time.

**To set the default value of the calendar control**

- 1 Select the Calendar control.
- 2 In the Properties Editor, set the `DefaultValue` property to the date value you want to use as default at run time.

## Configuring DateTime Picker Controls

Use the DateTime Picker control to select a date or time.

Saturday , January 20, 2007 ▾

You can configure the DateTime Picker control to show:

- A long format, such as Friday, August 11, 2008.
- A short format, such as 8/11/2008.
- Just the time, such as 9:16:36 PM.
- A custom time format, such as 8/11/2008 9:16:36 PM.

You can also set the default value of the DateTime Picker control.

### To set the long date format

- 1 Select the DateTime Picker control.
- 2 In the Properties Editor, set the Format property to **Long**.

### To set the short date format

- 1 Select the DateTime Picker control.
- 2 In the Properties Editor, set the Format property to **Short**.

### To set only time display

- 1 Select the DateTime Picker control.
- 2 In the Properties Editor, set the Format property to **Time**.

### To set a custom date/time format

- 1 Select the DateTime Picker control.
- 2 In the Properties Editor, set the Format property to **Custom**.

- 3 Type the time format in the value box for the **CustomFormat** property. Use the following letters as placeholders:

h	The one or two-digit hour in 12-hour format.
hh	The two-digit hour in 12-hour format. Single digit values are preceded by a zero.
H	The one or two-digit hour in 24-hour format.
HH	The two-digit hour in 24-hour format. Single digit values are preceded by a zero.
t	The one-letter AM/PM abbreviation ("AM" is shown as "A").

tt	The two-letter AM/PM abbreviation ("AM" is shown as "AM").
m	The one or two-digit minute.
mm	The two-digit minute. Single digit values are preceded by a zero.
s	The one or two-digit seconds.
ss	The two-digit seconds. Single digit values are preceded by a zero.
d	The one or two-digit day.
dd	The two-digit day. Single digit day values are preceded by a zero.
ddd	The three-character day-of-week abbreviation.
dddd	The full day-of-week name.
M	The one or two-digit month number.
MM	The two-digit month number. Single digit values are preceded by a zero.
MMM	The three-character month abbreviation.
MMMM	The full month name.
y	The one-digit year (2001 is shown as "1").
yy	The last two digits of the year (2001 is shown as "01").
yyyy	The full year (2001 is shown as "2001").

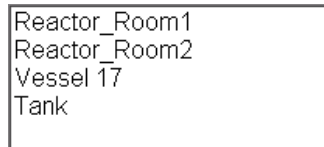
You can use any other characters, except “g” in the property. These characters then appear at design time and run time in the control.

#### To set the default value in a DateTime Picker control

- 1 Select the DateTime Picker control.
- 2 In the Properties Editor, set the DefaultValue property to the date and time value you want to use as default at run time.

## Configuring List Box Controls

You can create a list box for users to select an option from a scrollable list during run time.



You can:

- Configure the list box to avoid clipping of its contained items. When you set the Integral Height flag, the list box control is resized so that no items are clipped.
- Specify if you want the control to be scrollable in horizontal direction at run time. This enables the user to see the full text if the item captions are wider than the control itself.
- Use properties that are specific to the List Box control in scripting. At run time you can access the script to view and modify the items in the List Box control.

### Avoiding Clipping of Items in the List Box Control List

In the list of a List Box control, some items may appear vertically clipped. You can configure the List Box control to avoid this clipping by setting the IntegralHeight property.

#### To avoid clipping of items in the List Box control

- 1 Select the list box control.
- 2 In the Properties Editor, select **True** as value for the IntegralHeight property.

### Using a Horizontal Scroll Bar in a List Box Control

You can configure a horizontal scroll bar in a List Box Control so that at run time the user can scroll the list horizontally to see items that are wider than the control.

#### To configure a horizontal scroll bar

- 1 Select the List Box control.
- 2 In the Properties Editor, select **True** as value for the HorizontalScrollbar property.

### Using List Box-Specific Properties at Run Time

You can use properties that are specific to the List Box control at run time.

- The `Count` property returns the number of items in a List Box control.
- The `NewIndex` property returns the index of the last item added to the List Box list.
- The `SelectedValue` property reads the value of the selected item, or selects the item with that value if it exists.
- The `TopIndex` property returns the index of the top most item in the list.

These properties are available when you browse for a List Box control in the Galaxy Browser. For more information about scripting, see [Adding and Maintaining Symbol Scripts](#) on page 353.

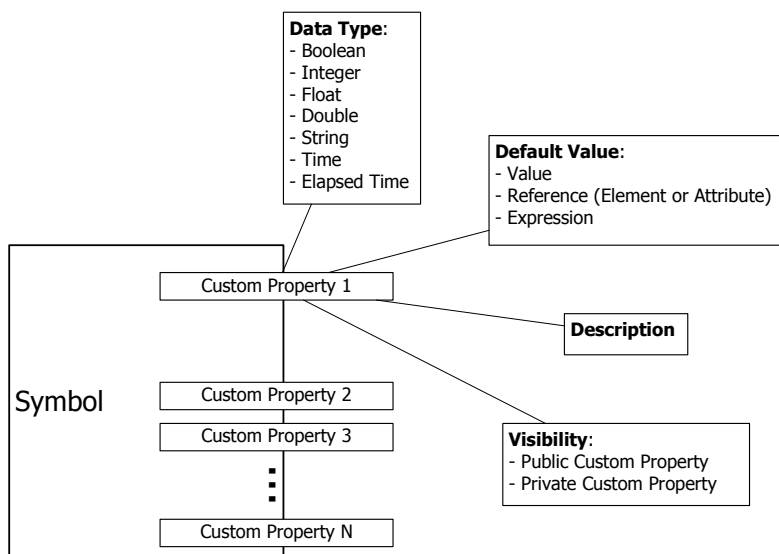
For more information about the properties, see [Alphabetical List of Properties](#) on page 444.



# Chapter 8

## Using Custom Properties

You can configure and use custom properties to extend the functionality of symbols and use them in combination with InTouch tags. You can use binding with custom properties to dynamically change the reference of a custom property.

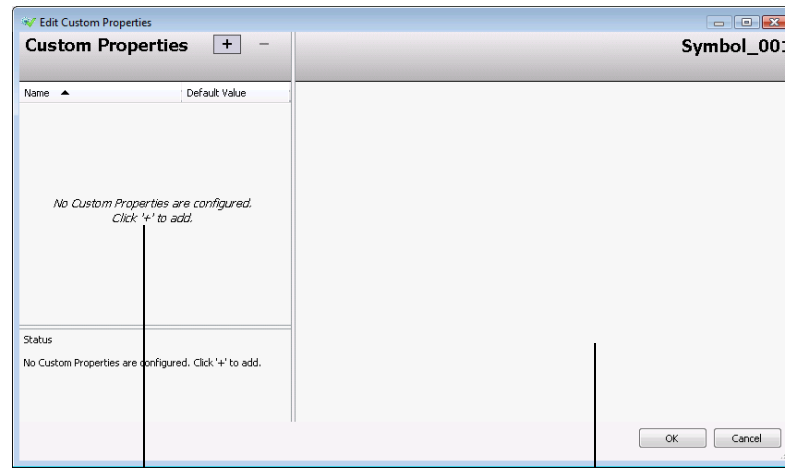


## About Custom Properties

You use custom properties to extend the standard properties of a symbol or an embedded symbol. You can associate custom properties with functionality you want exposed and that you want to be reusable. You can also use custom properties to connect an embedded ArchestrA Symbol to InTouch tags.

## Managing Custom Properties

You manage all custom properties of a symbol using the **Edit Custom Properties** dialog box.



List of Custom Properties

Custom Properties  
Configuration Panel

From the **Custom Properties** dialog box, you can:

- Add and delete custom properties.
- Set the types and data types of custom properties.
- Set the default values of custom properties.
- Determine the visibility of each custom property.
- Add a description for each custom property.
- Validate and clear custom properties.

You can also:

- Rename custom properties.
- Link custom properties to external sources.
- Override custom properties with new values.
- Revert custom property values to their default values.

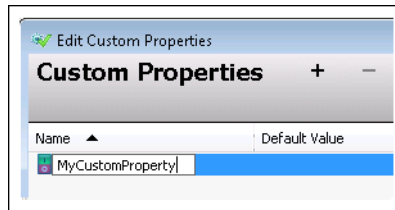


## Adding and Deleting Custom Properties

You can add and delete custom properties from a symbol.

### To add a custom property

- 1 Click the canvas to cancel any selected elements.
- 2 On the **Special** menu, click **Custom Properties**. The **Edit Custom Properties** dialog box appears.
- 3 Click the **Add** icon. A new line is added in the custom properties list.
- 4 Type a name for the new custom property and click **ENTER**.



You can see the name of the symbol and the custom property in the header of the dialog box.



- 5 Configure the custom property on the right side of the **Edit Custom Properties** dialog box. For more information see [Configuring Custom Properties](#) on page 242.
- 6 Click **OK**.

### To delete a custom property

- 1 Click the canvas to to cancel any selected elements.
- 2 On the **Special** menu, click **Custom Properties**. The **Edit Custom Properties** dialog box appears.
- 3 Select the custom property you want to delete and click the **Remove** icon. When a message appears requesting confirmation to delete the custom property, click **Yes**. The custom property is removed from the custom properties list.
- 4 Click **OK**.



## Configuring Custom Properties

You can configure custom properties when you create them or at a later point of time.

### To configure a custom property




- 1 Click the canvas of the symbol.
- 2 On the **Special** menu, click **Custom Properties**. The **Edit Custom Properties** dialog box appears.
- 3 Select the custom property you want to edit. The configuration for the selected custom property appears at the right of the dialog box.





The screenshot shows a dialog box titled 'MyCustomProperty' for 'Symbol\_001'. The dialog has a header with a small icon of a symbol and a red checkmark. Below the header, there are several sections:

- Data Type:** A dropdown menu set to 'Boolean'.
- Default Value:** A text input field containing 'True'.
- Visibility:** Two radio buttons: 'Public (Property can be seen when symbol is embedded)' (selected) and 'Private (Property is hidden when symbol is embedded)'.
- Description:** A large empty text area.

**Note** The header of the configuration area shows you the symbol name, for example Symbol\_001, on the right and the custom property name on the left, for example MyCustomProperty. It can be accessed from scripting as Symbol\_001.MyCustomProperty.

- 4 In the **Data Type** list, click the data type of the custom property. You can select one of the following:

Data Type	Symbol
Boolean	
Double	
Elapsed Time	

Data Type	Symbol
Float	
Integer	
String	
Time	



- 5 If you want to:
- Make the property read-only at design time and prevent further changes to it when the symbol is embedded into another symbol, click the Lock icon.
  - Make the property read-only at run time and prevent its value being changed, click the Lock icon.

- 6 In the **Default Value** box, type a literal value, reference, or expression or browse for a reference using the **Browse** icon.



- 7 If the selected data type is String, Time or Elapsed Time, you can click the **T** icon or tag icon.



- Select the **T** icon to indicate that the default value is a static value.
- Select the tag icon to indicate that the default value is a reference to a value.

- 8 In the **Visibility** box, configure how the symbol is visible. Do one of the following:

- Click **Public** if you want the custom property to be visible and can be used in a source symbol if the symbol is embedded.
- Click **Private** if you want the custom property to be hidden and no reference be made to it outside of the defining symbol.

- 9 In the **Description** box, type a meaningful description for the custom property.

## Validating Custom Properties

You can validate custom properties to track down and avoid configuration errors.

### To validate a custom property

- 1 Click on the canvas to cancel any selected elements.
- 2 On the **Special** menu, click **Custom Properties**. The **Edit Custom Properties** dialog box appears.
- 3 Select the custom property you want to validate and click the **Validate** icon. Required boxes are highlighted by a red box, possible errors appear in the status area under the custom properties list.



Status

The property has the following error: Invalid Value. This field cannot be empty. [Custom Property MyCustomProperty] To configure an empty reference, use '---' as the default value.

## Clearing the Configuration of Custom Properties

You can clear the configuration of custom properties. This resets the properties to their default values.

### To clear the configuration of a custom property

- 1 In the **Edit Custom Properties** dialog box, select the custom property.
- 2 Click the **Clear** icon. The configured values are reset to their default values.



## Renaming Custom Properties

You can rename custom properties.

### To rename a custom property

- 1 In the **Edit Custom Properties** dialog box, select the custom property.
- 2 Click the custom property again. The custom property is in edit mode.
- 3 Type the new custom property name and click **ENTER**. The custom property is renamed.

## Linking Custom Properties to External Sources

You can link custom properties of a symbol directly to external sources by:

- Configuring AutomationObjects that point to external sources and then point the custom property at the corresponding attribute reference.
- Configuring a special InTouch reference syntax in the **Default Value** box. When you embed the symbol into an InTouch window, the referenced InTouch tags connect to the tags of the InTouch HMI.

For more information, see *Connecting Animations with Custom Properties* on page 258, *Connecting Animations with InTouch Tags* on page 259 and the *InTouch HMI and ArcestrA Integration Guide*.

## Overriding Custom Properties

You can override the custom property default values of embedded symbols within symbols in the ArcestrA Symbol Editor or InTouch WindowMaker.

---

**Note** When you override the custom property, it appears bold in the custom property list.

---

You can override the following custom property values:

- Default value
- Visibility, but only from public to private, not private to public
- Description
- Locked state
- String mode setting

You cannot override the data type of a custom property.

## Reverting to Original Custom Property Values

After you override a custom property value, you can revert to the original custom property value. This can be done for overridden custom properties of embedded symbols in other symbols and in InTouch WindowMaker.

### To revert to the original custom property value



- ◆ In the **Edit Custom Properties** dialog box, click the Revert icon. The custom property value reverts to its original value.

## Examples of Using Custom Properties

Possible uses for using custom properties are:

- A “TankLevel” custom property of type Writable Attribute can be given a value of “me.pv”.
- A “MaxFillLevel” Custom Property of type Expression can be given a value of “Me.MaxCapacity - 200”.

A more extensive example on how to use custom properties in embedded symbols in an InTouch window can be found in the *InTouch HMI and ArchestrA Integration Guide*.

## Using Binding in Custom Properties

ArchestrA object scripting supports a type called “Indirect”. It allows you to bind a variable to a reference and read and write to it. This is done using the BindTo() method.

---

**Note** The BindTo() method binds a variable to a reference as long as the symbol is shown.

---

For example, the local script variable ptr is defined and bound to the reference ud1\_001.Int1.

```
dim ptr as indirect;
ptr.BindTo("ud1_001.Int1");
```

Within the same script you can use the indirect variable ptr to read from and write to the attribute ud1\_001.Int1.

ArchestrA Symbols also use scripting in the same way as the scripting of Application Server.

However, as an ArchestrA Symbol can be embedded into an InTouch window and run anonymously, the time it takes to connect to the reference can be longer than one scan cycle.

For that reason, you cannot use the indirect variable immediately after it is bound to a reference to read from and write to it.

```
dim ptr as indirect;
ptr.BindTo("ud1_001.Int1");
ptr = 42;
```

In the example, the value 42 cannot be written to the reference `ud1_001.Int1` as the binding takes longer.

To avoid this problem, you can modify your ArchestrA Symbol script to write the value after it is ensured that the binding is complete. The completion of the binding is indicated by the quality of the indirect variable.

You can configure a loop in the script to query for the quality and use the indirect variable to read from and write to the reference when its quality is good.

---

**Note** Make sure to include an exit condition in your script, so that the script doesn't "hang" in case the binding cannot be made.

---

The following example script shows you how to do this:

```
dim ptr as indirect;
dim timeout;
ptr.BindTo("ud1_001.Int1");
while (IsGood(ptr)==0); {if quality not good}
    timeout=timeout+1; {increase the timer}
    if timeout>10000 then {if timer reaches threshold}
        exit while; {continue script execution}
    endif;
endwhile; {otherwise just loop for a while}
ptr=42; {try to write to value to the reference}
```

A while loop is included in the script before the first write attempt. The while loop provides additional time for the symbol to connect to the reference. If the quality is good, then the script exits from the while loop.

---

**Note** Similar behavior can occur when you try to bind to a reference of an object that is hosted on a different AppEngine.

---

**Caution** If possible, use the `IOSetReferences QuickScript` function instead of the `BindTo()` method. The `BindTo()` method can be CPU-intensive.

---

## Changing the Expression or Reference of a Custom Property at Run Time

You can change the expression or reference of a custom property at run time by calling the following method on the symbol using a client script:

```
SetCustomPropertyValue(System.String name,  
    System.String value, System.Boolean isConstant);
```

You can select this method using the Element Browser from within the Symbol Editor. This method is supported only for ArcestrA client scripts.

This method has three parameters:

- *name* - Name of the custom property to be modified on the symbol. This parameter is of type string, and it can be a reference or a constant.
- *value* - The new value to be set. This parameter is of type string, and it can be an expression, reference, or constant. If the value is given in quotes(""), then the value is considered a reference. If the value is given without quotes, then the value of the expression is assigned as a constant.
- *type* - The type of the value. This parameter is of type Boolean, which means that the second parameter is reference or constant only if the custom property (specified in the *name* parameter) is of string and time type. The *type* parameter has no meaning if the custom property is an integer, float, Boolean, or double type.

The whole expression or reference of the custom property is replaced with the new value, regardless if it is overridden or not. No partial replacement is supported.

Only public custom properties on the symbol can be changed.

When the method executes, it overrides any modifications done by previous `IOSetRemoteReference()` calls from a native InTouch script.



---

# Chapter 9

## Animating Graphic Elements

You can use animations to change the appearance of graphic elements at run time. Animations are driven by data that comes from ArchestrA attribute values and expressions as well as element properties that can be changed in WindowMaker.

You can use:

- **Visualization animations** such as visibility, fill style, line style, text style, blinking, percent fill horizontal, percent fill vertical, horizontal location, vertical location, width, height, orientation, value display or tooltip.
- **Interaction animations** such as disable, user input, horizontal slider, vertical slider, pushbutton, action script, show symbol or hide symbol.
- **Element-specific animations** for the Status element and Windows common control elements.

Each element in your ArchestrA Symbol can have one or more animations. You can disable and enable individual animations. You can also cut, copy and paste animations between elements. Only animations supported by the target element are pasted.

You can also substitute references and strings in animations.

---

**Note** Not all animations are available for all element types. Some animations do not make logical sense, such as line style with a text element. You cannot select or copy these invalid combinations.

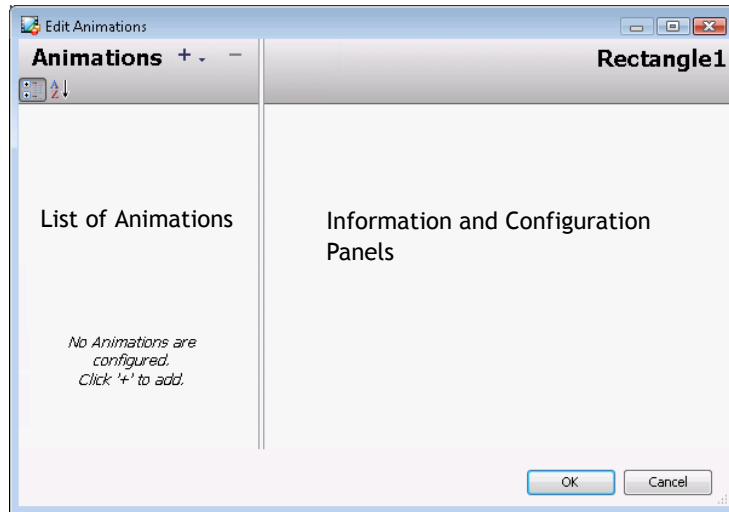
---

## Adding an Animation to an Element

You can add one or more animations to a single element in your ArchestrA Symbol.

### To add an animation to an element

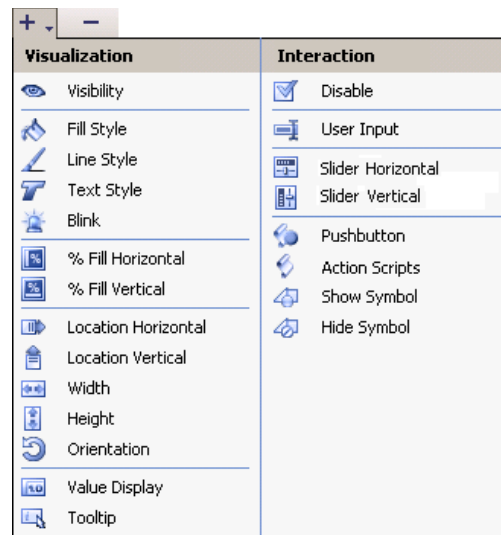
- 1 Select the element to which you want to add an animation.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.



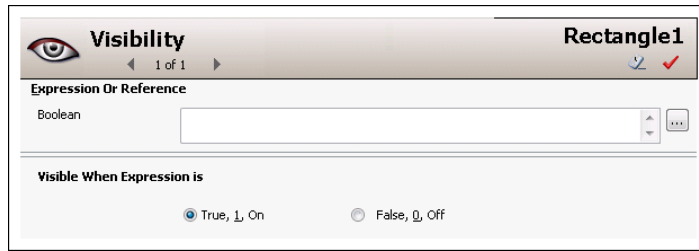
You can also add an animation from the Animation Summary in the lower right corner of the ArchestrA Symbol Editor.



- 3 Click the **Add** icon. The list of animations appears.



- 4 Select an animation from the list. The animation is added to the Animation list. You can configure the selected animation from the **Edit Animations** dialog box.



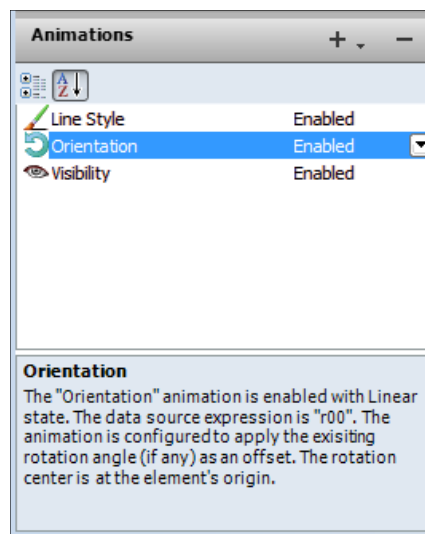
**Note** Depending on the animation type, you may get an animation state selection panel instead. For more information, see [Reviewing which Animations are Assigned to an Element](#) on page 251.

## Reviewing which Animations are Assigned to an Element

You can review which animations are assigned to an element and change the number of animations or their configuration at the same time.

### To review which animations are assigned to an element

- 1 Select the element. The assigned animations appear in the Animation Summary in the lower right of the ArchestrA Symbol Editor.



You can also review which animations are assigned to an element by double-clicking it.

- 2 Select an animation to view further information on how the element is configured with that animation.

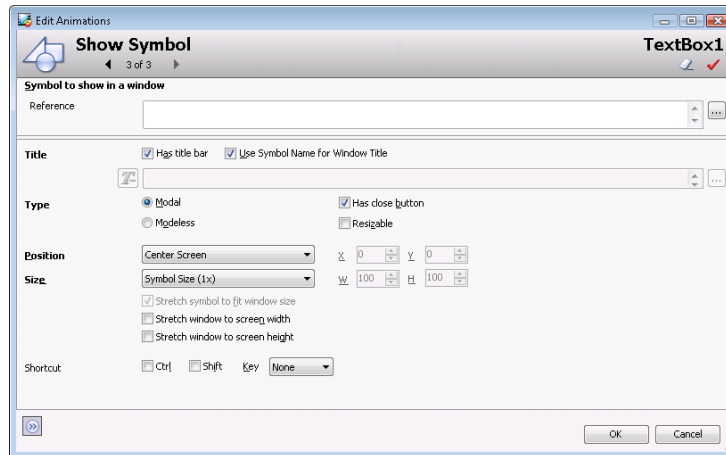
## Showing and Hiding the Animation List

You can show or hide the Animation list. If you hide the Animation list, the configuration space expands, giving you more space to configure the animations.

### To hide the Animation list



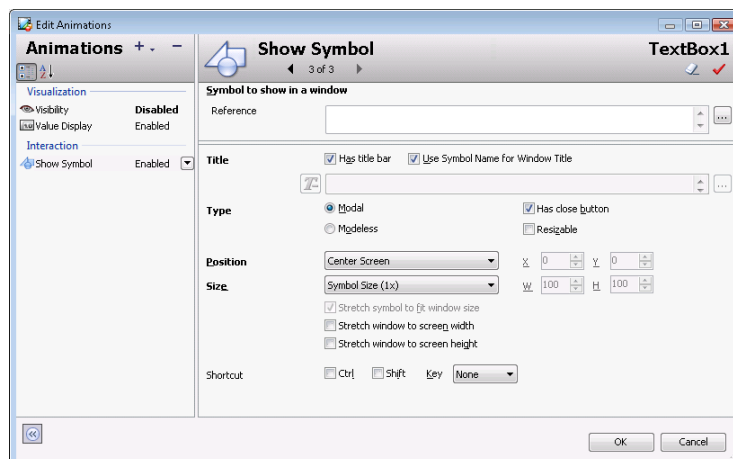
- ◆ In the **Edit Animations** dialog box, click the **Hide** icon. The Animation list hides and the configuration space expands.



### To show the Animation list



- ◆ In the **Edit Animations** dialog box, click the **Show** icon. The Animation list appears and the configuration space reduces to its default width.



## Removing Animations from an Element

You can remove an animation from an element by using the **Edit Animations** dialog box. You can remove animations from an element for:

- Individual animations
- All animations at the same time

### To remove an animation from an element

- 1 Select the element in which you want to remove an animation.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.

You can also remove an animation from the Animation Summary in the lower right of the ArcestraA Symbol Editor.

- 3 Select the animation you want to remove from the Animation list.



- 4 Click the **Remove** icon. A message appears.
- 5 Click **Yes**. The animation is removed from the list and no longer associated with the element.

### To remove all animations from an element

- 1 Select one or more elements from which you want to remove all animations.
- 2 Do one of the following:
  - Right-click, point to **Animations** and then click **Clear**.
  - On the **Edit** menu, point to **Animations**, and then click **Clear**.

All animations are removed from the selected elements.

## Enabling and Disabling Animations

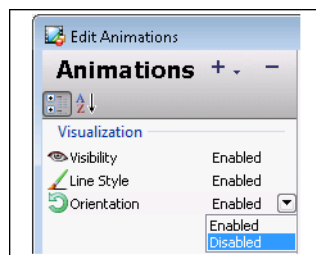
You can enable or disable animations for an element. When you disable an animation, its configuration is not lost. This lets you see, for example, each animation independently from each other.

### To disable an animation

- 1 Select the element with the animation you want to disable.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.

You can also disable animations from the Animation Summary in the lower right corner of the ArcestrA Symbol Editor.

- 3 Locate the animation you want to disable from the Animation list on the left of the dialog box.
- 4 Select **Disabled** from the list of that row.



- 5 Repeat for any other animations you want to disable and click **OK** when you are done.

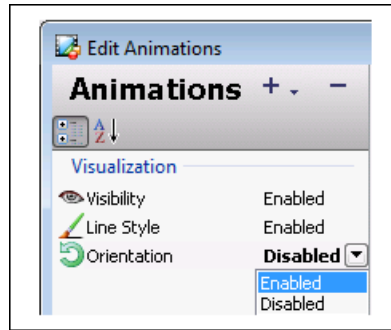
### To enable an animation

- 1 Select the element with the animation you want to enable.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.

You can also enable animations from the Animation Summary of the ArcestrA Symbol Editor.

- 3 Locate the animation you want to enable from the Animation list.

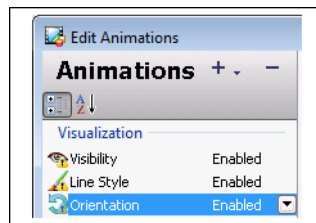
- 4 Select **Enabled** from the list of that row.



- 5 Repeat for any other animations you want to enable and click **OK** when you are done.

## Validating the Configuration of an Animation

You can validate the configuration of an animation. If the configuration contains an error, an exclamation mark appears next to the **Animation** icon.



Examples of animation configuration errors include:

- Animation is disabled
- Syntax errors such as data mismatches
- Required values not specified
- Specified values out of valid range

### To validate the configuration of an animation

- 1 Select the element that contains the animations you want to validate.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Select the animation you want to validate.
- 4 Click the **Validate** icon. The currently selected animation is validated. Possible errors are highlighted.



## Clearing the Configuration from an Animation

You can clear all data from the configuration boxes of an animation and reset the settings to their defaults.

### To clear all data from the configuration boxes of an animation



- 1 In the **Edit Animations** dialog box, select the animation.
- 2 In the configuration panel, click the **Clear** icon. All data from the configuration boxes is cleared and the settings are reset to their defaults.

## Connecting Animations with Data Sources

You can connect animations to

- ArcestrA attributes
- Element properties
- Custom properties
- InTouch tags

For some input boxes, you can specify if the configuration is a static value or a reference by setting the input mode.

## Connecting Animations with ArcestrA Attributes

You connect the element animation and appearance with an ArcestrA attribute. The ArcestrA attribute provides values at run time that control the behavior and appearance of the element.

For example, a rectangle element fill animation can be connected to the run-time value of the ArcestrA attribute Tank\_001.PV.

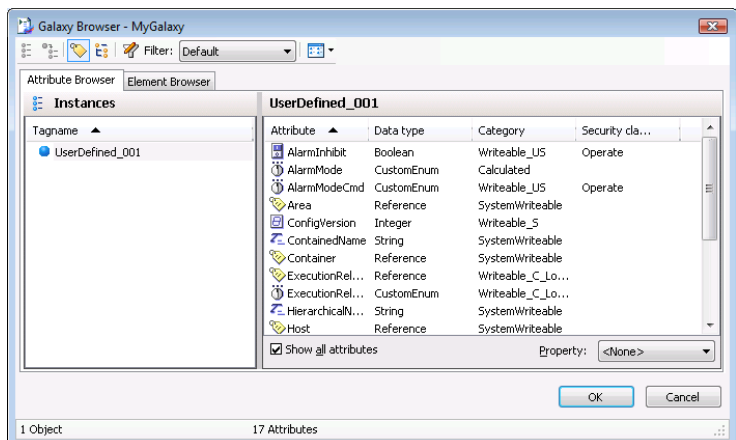
You can browse all ArcestrA attributes and Element Properties with the Galaxy Browser.

### To connect animations to attribute references using the Galaxy Browser

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Select the animation from the Animation list.
- 4 Select the parameter.



- Click the browse button. The **Galaxy Browser** appears.



- Click the **Attribute Browser** tab.
- Select an AutomationObject from the list shown in the left pane. The attributes associated with the selected AutomationObject are shown in the right pane.
- Select an attribute from the right pane and click **OK**. The selected attribute reference appears in the configuration box.
- Repeat for any other animation parameters. Click **OK** when you are done.

## Connecting Animations with Element Properties

You can connect the element animation and appearance with a property of any element on the canvas.

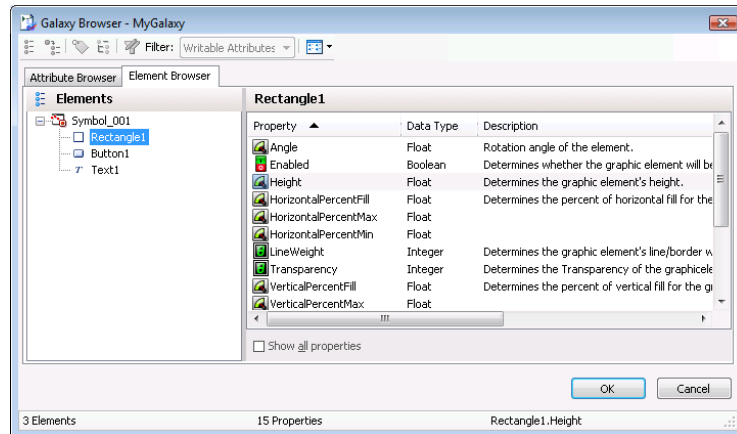
You can browse the properties of all elements on the canvas with the Galaxy Browser.

You cannot connect animations to properties of elements that are part of an embedded symbol on the canvas. You can connect animations to the public custom properties of embedded symbols.

### To connect animations to element property references using the Galaxy Browser

- Select the element.
- On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- Select the animation from the Animation list.
- Select the parameter.
- Click the browse button. The **Galaxy Browser** appears.

- Click the **Element Browser** tab to show the **Element Browser** page.



- From the Elements List, select an element. The right pane shows the properties of the selected element.
- Select a property and click **OK**. The selected element and property appears in the configuration box.

## Connecting Animations with Custom Properties

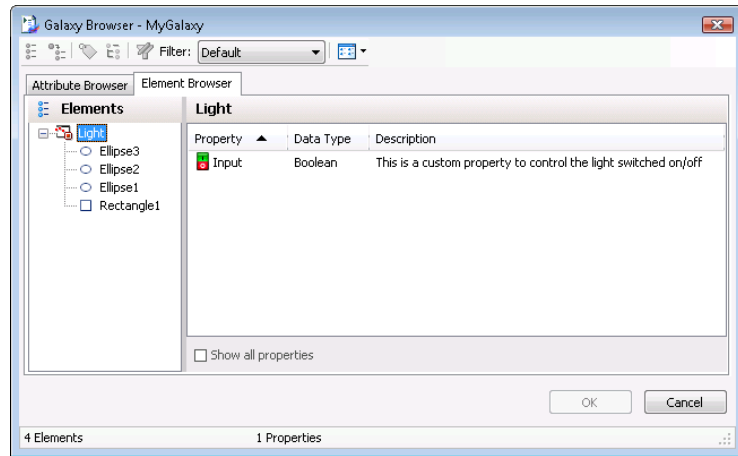
You can connect the element animation and appearance with a custom property of:

- The current symbol
- An embedded symbol on the canvas

### To connect animations to element property references using the Galaxy Browser

- Select the element.
- On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- Select the animation from the Animation list.
- Select the parameter.
- Click the browse button. The **Galaxy Browser** appears.

- 6 Click the **Element Browser** tab. The **Element Browser** tab appears.



- 7 From the Elements List on the left, select the symbol. The right pane shows the custom properties and other properties of the selected symbol.
- 8 Select a custom property and click **OK**. The selected custom property appears in the configuration box.

## Connecting Animations with InTouch Tags

You can connect an element animation and appearance to an InTouch tag. The InTouch tag provides values at run time that control the animation and appearance of the element.

You can connect an element animation to an InTouch tag by:

- Configuring a reference with the **intouch:tag** syntax.
- Using a custom property and configuring the custom property in the embedded ArchestrA Symbol in InTouch to reference an InTouch tag. For more information, see the *InTouch HMI and ArchestrA Integration Guide*.
- Configuring an ArchestrA attribute reference to the managed InTouchViewApp object that contains the InTouch tags as attributes. The InTouchViewApp object uses the functionality of an InTouchProxy object.
- Configuring an ArchestrA attribute reference to an InTouchProxy object that contains the InTouch tags as items. This is a special case of configuring an ArchestrA attribute reference.

### Using the InTouch:Tagname Syntax

When you use the **intouch:tagname** syntax, the animation connects to the InTouch tag of the node the symbol is used. There are some restrictions on how you can use this syntax:

- Unlike in Application Server, you cannot use true and false as Boolean values. Use 1 and 0 instead.
- If you want to make a reference to an InTouch SuperTag, use the following syntax instead:

```
attribute ("intouch:SuperTag\Member")
```

### Connecting Animations with InTouchViewApp Attributes

To be able to browse for InTouch tags, you must first:

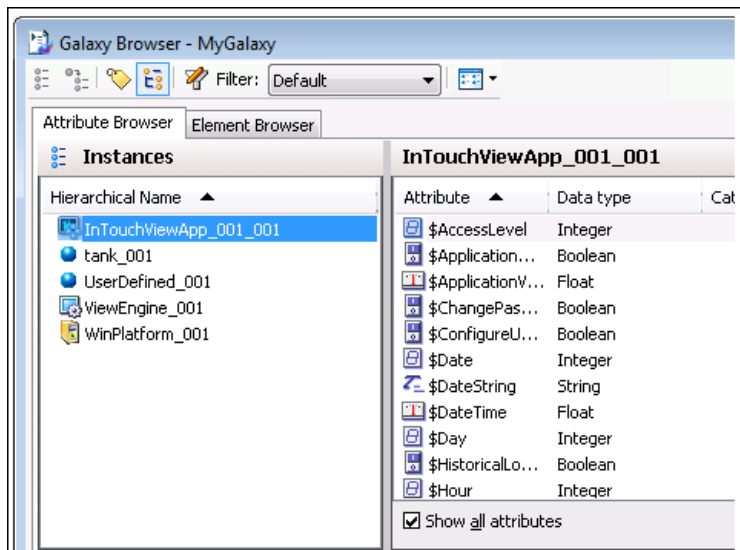
- Create a managed InTouch application by deriving an InTouchViewApp template and configuring it in WindowMaker.
- Derive an instance of the InTouchViewApp derived template.

The InTouch tags are represented by attributes of the InTouchViewApp object instance.

#### To connect animations to InTouch tags

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Select the animation from the Animation list.
- 4 Select the parameter.
- 5 Click the browse button. The **Galaxy Browser** appears.

- 6 Select the InTouchViewApp object that corresponds to the managed InTouch application. The right panel shows the InTouch tags.



- 7 Select a tag and click **OK**. The selected ArchestrA reference to an InTouch tag appears in the configuration box.

## Setting the Input Mode

In some boxes you can enter a value or expression that uses static and/or references to attributes and element properties. Boxes that support both input methods have an **Input Mode** selection icon.

Select:



- **Static Mode input** icon to specify literal static value or expression such as 3.141 or “Test”.



- **Reference Mode input** icon to specify a reference to an attribute or element property such as: Tank\_001.PV.

---

**Note** To use static string values with or without references in Reference mode, you can enclose them with double-quotes such as: “Description: “+Tank\_001.Desc

---

## Managing Animations

You can easily manage animations in the **Edit Animations** dialog box. You can:

- Change the way the list of animations appears.
- Switch easily between multiple animations of an element.

You can also do this for the Animation Summary in the lower right corner of the ArchestrA Symbol Editor.

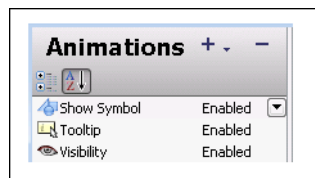
## Organizing the Animation List

You can organize the list of animations alphabetically or by category.

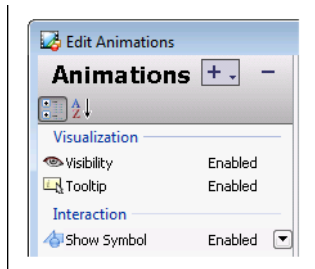
**To organize the Animation list**

- ◆ In the **Edit Animations** dialog box, click the:

- **Alphabetic sort** icon to sort alphabetically.



- **Category** icon to sort by category.

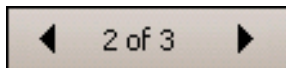


## Switching between Animations

If you configure more than one animation for an element, you can easily switch between their configuration panels without having to use the Animation list. This is particularly useful when the Animation list is hidden.

### To switch between animations

- ◆ In the **Edit Animations** dialog box, on the configuration panel click the left or right arrow icon.



The configuration panel changes to the configuration panel of the previous or next animation.

## Configuring Common Types of Animations

Every animation type has its own set of configuration parameters. This section shows you how to configure each type of animation and what references it can use.

You can configure:

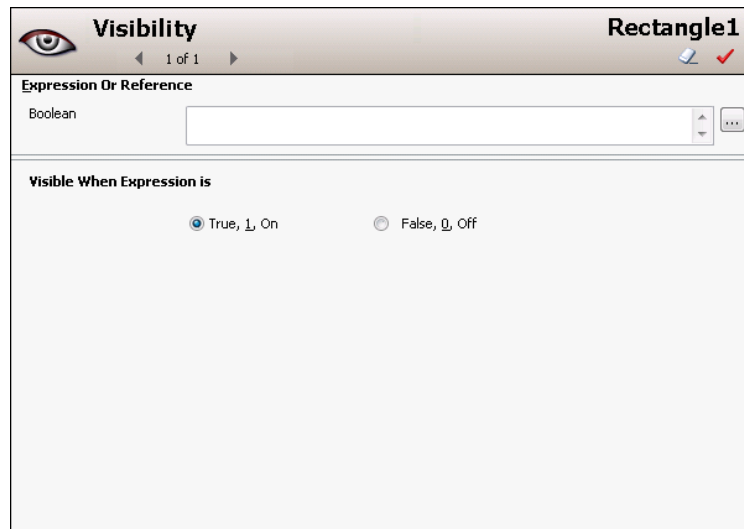
- Visualization animations such as:
  - Visibility animations
  - Fill style, line style or text style animations
  - Blink animations
  - Horizontal or vertical percent fill animations
  - Horizontal or vertical location animations
  - Width or height animations
  - Orientation animations
  - Value display animations
  - Tooltip animations
- Interaction animations such as:
  - Disable animation
  - User input animation
  - Horizontal and vertical slider animations
  - Pushbutton animations
  - Action script animations
  - Show or hide animations

## Configuring a Visibility Animation

You can configure an element with a visibility animation.

### To configure an element with a visibility animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Visibility**. The visibility animation is added to the Animation list and the **Visibility** configuration panel appears.



- 4 In the **Boolean** box, type a Boolean numeric value, attribute reference or expression.
- 5 Select **True, 1, On** if you want the element to show, when the expression is true, otherwise select **False, 0, Off**.

## Configuring a Fill Style Animation

You can configure an element with a:

- Boolean fill style animation.
- Truth table fill style animation.

The truth table fill style animation lets you:

- Associate expressions of any data type supported by ArchestrA with a fill style.
- Define as many fill styles as you require and associate each one with a condition.

You can define the conditions by specifying a comparison operator (=, >, >=, <, <=) and a breakpoint, which itself can be a value, an attribute reference, or an expression.



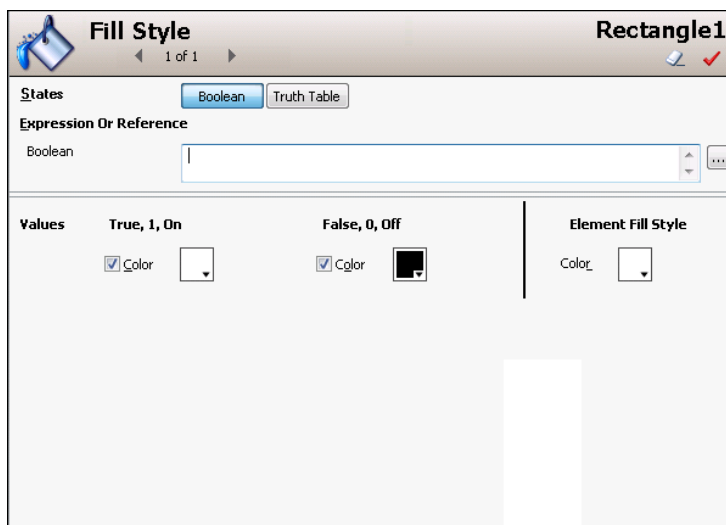
You can add conditions, delete conditions, and also change the order in which the conditions are processed.

### Configuring a Boolean Fill Style Animation

You can configure an element with a discrete fill style animation.

To configure an element with a Boolean fill style animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Fill Style**. The fill style animation is added to the Animation list and the **Fill Style** state selection panel appears.
- 4 Click the **Boolean** button. The **Boolean Fill Style** configuration panel appears.



- 5 In the **Boolean** box, type a Boolean numeric value, attribute reference or expression.
- 6 Clear **Color** in the **True, 1, On** area or **False, 0, Off** area if you do not want a different fill style for the true or false condition than the default fill style.
- 7 In the **True, 1, On** area, click the color box to configure the fill color when the expression is true. The Style Selection dialog box appears. For more information, see Setting Style on page 166.
- 8 In the **False, 0, Off** area, click the color box to configure the fill color when the expression is false. The Style Selection dialog box appears. For more information, see Setting Style on page 166.
- 9 Click **OK**.

**To set default fill style in a Boolean fill style animation**

- 1 Open the **Edit Animations** dialog box, **Boolean Fill Style** panel.
- 2 In the **Element Fill Style** area, click the color box to select a style from the Style Selection dialog box.

**To use default fill style in a Boolean fill style animation**

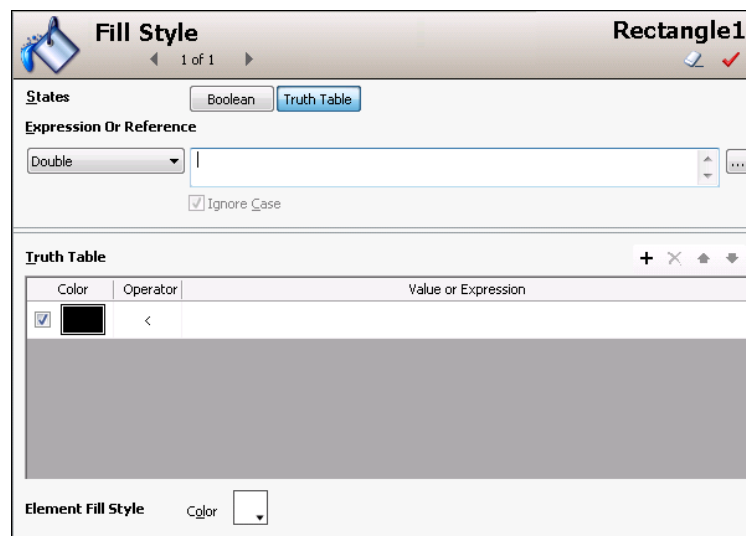
- 1 Open the **Edit Animations** dialog box, **Boolean Fill Style** panel.
- 2 Clear **Color** to use the corresponding default fill style.

**Configuring a Truth Table Fill Style Animation**

You can configure an element with a fill style animation based on a truth table.

**To configure an element with a truth table fill style animation**

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Fill Style**. The fill style animation is added to the Animation list and the **Fill Style** state selection panel appears.
- 4 Click the **Truth Table** button. The **Truth Table Fill Style** configuration panel appears.



- 5 In the **Expression Or Reference** area:
  - Select the data type of the expression from the list.
  - Type a value, attribute reference or expression in the text box.

- 6 If the data type of the expression is string or internationalized string, you can specify to ignore the case by selecting **Ignore Case**.
- 7 In the **Truth Table**, click the color box in the **Color** column. The Style Selection dialog box appears. For more information, see [Setting Style](#) on page 166.
- 8 In the **Operator** column, select the comparison operator.
- 9 In the **Value or Expression** column, type a value, attribute reference, or expression.
- 10 To add further conditions, see [To add a condition to a truth table fill style animation](#) on page 267.
- 11 Click **OK**.

#### To set the default fill style for a truth table fill style animation

- 1 Open the **Edit Animations** dialog box, **Truth Table Fill Style** panel.
- 2 In the **Element Fill Style** area, click the color box. The Style Selection dialog box appears. For more information, see [Setting Style](#) on page 166.



#### To use the default fill style in a truth table fill style animation

- 1 Open the **Edit Animations** dialog box, **Truth Table Fill Style** panel.
- 2 Locate the condition for which you want to set the style to default style.
- 3 Clear the mark for that condition in the **Color** column of the truth table. The associated style is the same as the style for the **Element Fill Style**.

#### To add a condition to a truth table fill style animation

- 1 Open the **Edit Animations** dialog box, **Truth Table Fill Style** panel.
- 2 Click the **Add** icon. An additional condition is added to the truth table.



Color	Operator	Value or Expression
<input checked="" type="checkbox"/> 	>=	Standards_TankLevel.LevelMax
<input checked="" type="checkbox"/> 	<	<input type="text"/>

- 3 Configure color, operator and breakpoint value according to your requirements.

**To delete a condition from an analog fill style animation**

- 1 Open the **Edit Animations** dialog box, **Truth Table Fill Style** panel.
- 2 Select the condition you want to delete.
- 3 Click the **Remove** icon. The condition is removed.

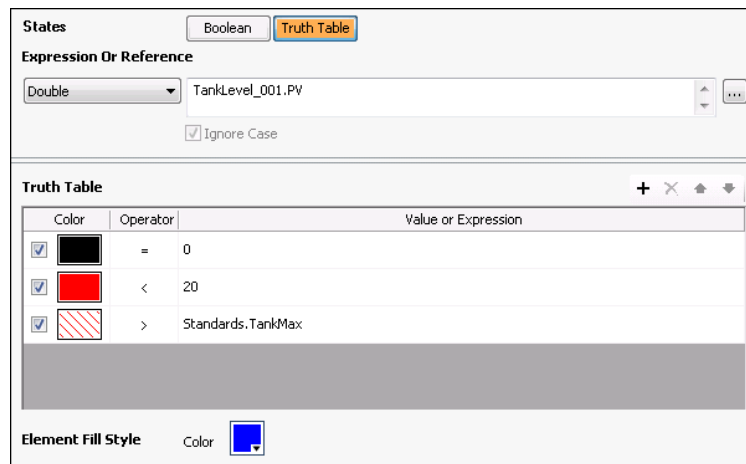
**To change the processing order of fill style conditions**

- 1 Open the **Edit Animations** dialog box, **Truth Table Fill Style** panel.
- 2 Select the condition you want to move up or down the condition list in order for it to be processed sooner or later.
- 3 Click the:
  - **Arrow up** icon to move the condition up in the truth table.
  - **Arrow down** icon to move the condition down in the truth table.



For example, you want to model an analog fill color animation that describes the following conditions:

- When the attribute TankLevel\_001.PV is 0 then the fill style is solid black.
- When the attribute TankLevel\_001.PV is smaller than 20, then the fill style is solid red.
- When the attribute TankLevel\_001.PV is greater than the attribute Standards.TankMax then the fill style is red with a diagonal pattern.
- In all other cases, the fill style is solid blue.



## Configuring a Line Style Animation

You can configure an element with a:

- Boolean line style animation.
- Truth table line style animation.

The truth table line style animation lets you:

- Associate expressions of any data type supported by ArcestrA with a line style.
- Define as many line styles as you want and associate each one with a condition.

You can define the conditions by specifying an comparison operator ( $=$ ,  $>$ ,  $>=$ ,  $<$ ,  $<=$ ) and a breakpoint, which itself can be a value, an attribute reference or an expression.

You can add conditions, delete conditions and also change the order in which the conditions are processed.

### Configuring a Boolean Line Style Animation

You can configure an element with a Boolean line style animation. You can use a new style or use all or parts of the default appearance of a line for:

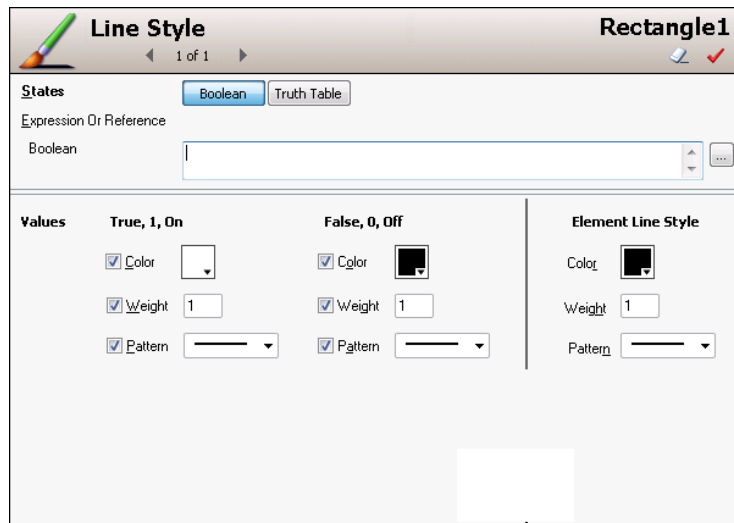
- Line style.
- Line thickness.
- Line pattern.

**To configure an element with a Boolean line style animation**

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Line Style**. The line style animation is added to the Animation list and the **Line Style** state selection panel appears.



- Click the **Boolean** button. The **Boolean Line Style** configuration panel appears.



- In the **Boolean** box, type a Boolean numeric value, attribute reference or expression.
- In the **True, 1, On** area, click the **Color** box to configure the line style when the expression is true. The Style Selection dialog box appears. For more information, see Setting Style on page 166.
- In the **Weight** box, type a value for the line thickness when the expression is true.
- From the **Pattern** list, select a line pattern for the line when the expression is true.
- Repeat the above steps for the false condition in the **False, 0, Off** area.
- Click **OK**.

#### To set default line style, thickness and/or pattern in a Boolean line style animation

- Open the **Edit Animations** dialog box, **Boolean Line Style** panel.
- In the **Element Line Style** area, select a style, type a value for the width and select a pattern for the default Boolean line style.

### To use default line style, thickness and/or pattern in a Boolean line style animation

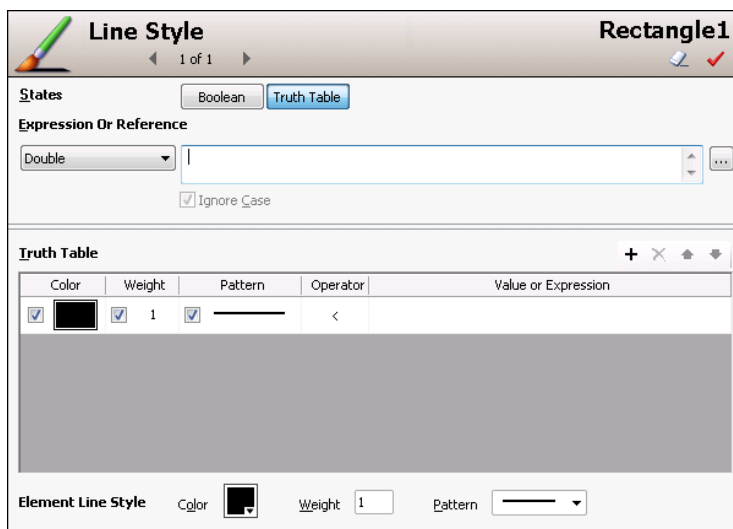
- 1 Open the **Edit Animations** dialog box, **Boolean Line Style** panel.
- 2 In the **True, 1, On** or **False, 0, Off** areas, clear **Color**, **Weight** and/or **Pattern** to use the corresponding default style, weight and/or pattern.

### Configuring a Truth Table Line Style Animation

You can configure an element with a truth table line style animation.

#### To configure an element with a truth table line style animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Line Style**. The line style animation is added to the Animation list and the **Line Style** state selection panel appears.
- 4 Click the **Truth Table** button. The **Truth Table Line Style** configuration panel appears.



- 5 In the **Expression or Reference** box:
  - Select the data type of the expression from the list.
  - Type a value, attribute reference or expression in the text box.
- 6 If the data type of the expression is string or internationalized string, you can specify to ignore the case by selecting **Ignore Case**.

- 7 In the **Truth Table**, click the color box in the **Color** column. The Style Selection dialog box appears. For more information, see Setting Style on page 166.
- 8 Select the truth options. Do one of more of the following:
  - In the **Weight** column, type a value for the line weight.
  - In the **Pattern** column, select a line pattern.
  - In the **Operator** column, select the comparison operator.
  - In the **Value or Expression** column, type a value, attribute reference or expression.
  - To add further conditions, see To add a condition to a truth table line style animation on page 272.
- 9 Click **OK**.

#### To set the default line style, width or pattern for a truth table line style animation

- 1 Open the **Edit Animations** dialog box, **Truth Table Line Style** panel.
- 2 In the **Element Line Style** area, select a style, type a value for the width and select a pattern for the default truth table line style.

#### To use the default line style, width or pattern in a truth table line style animation

- 1 Open the **Edit Animations** dialog box, **Truth Table Line Style** panel.
- 2 Locate the condition for which you want to change the line style, width or pattern.
- 3 To use the default line style for the condition, clear the mark in the **Color** column of the truth table.
- 4 To use the default line width for the condition, clear the mark in the **Width** column of the truth table.
- 5 To use the default line pattern for the condition, clear the mark in the **Pattern** column of the truth table.

#### To add a condition to a truth table line style animation



- 1 In the **Edit Animations** dialog box, **Truth Table Line Style** panel, click the **Add** icon. An additional condition is added to the truth table.
- 2 Configure color, weight, pattern, operator and breakpoint value according to your requirements.



**To delete a condition from an analog line color animation**

- 1 In the **Edit Animations** dialog box, **Truth Table Line Style** panel, select the condition you want to delete.
- 2 Click the **Remove** button. The condition is removed.

**To change the processing order of line style conditions**

- 1 Open the **Edit Animations** dialog box, **Truth Table Line Style** panel
- 2 Select the condition you want to move up or down the condition list in order for it to be processed sooner or later.
- 3 Click the:
  - **Arrow up** icon to move the condition up in the truth table.
  - **Arrow down** icon to move the condition down in the truth table.



## Configuring a Text Style Animation

You can configure an element with a:

- Boolean text style animation.
- Truth table text style animation.

The truth table text style animation lets you:

- Associate expressions of any data type supported by ArcestrA with a text style.
- Define as many text styles as you want and associate each one with a condition.

You can define the conditions by specifying an comparison operator ( $=$ ,  $>$ ,  $>=$ ,  $<$ ,  $<=$ ) and a breakpoint, which itself can be a value, an attribute reference or an expression.

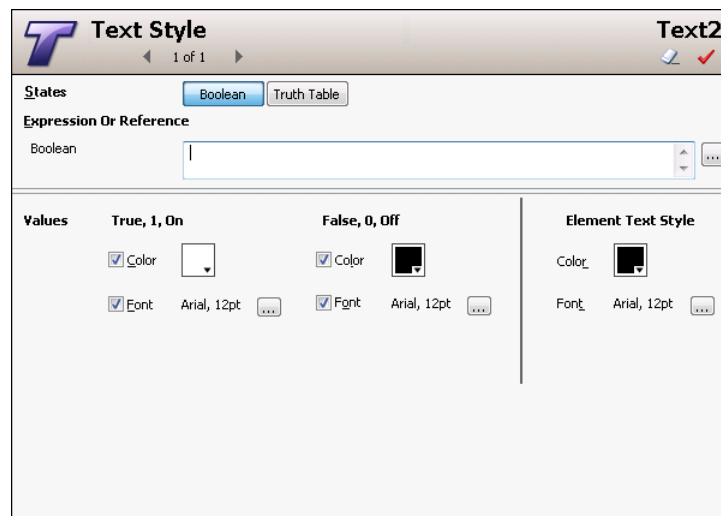
You can add conditions, delete conditions and also change the order in which the conditions are processed.

## Configuring a Boolean Text Style Animation

You can configure an element with a Boolean text style animation.

To configure an element with a Boolean text style animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Text Style**. The text style animation is added to the Animation list and the **Text Style** state selection panel appears.
- 4 Click the **Boolean** button. The **Boolean Text Style** configuration panel appears.



- 5 In the **Boolean** box, type a Boolean numeric value, attribute reference or expression.
- 6 In the **True, 1, On** area, click the **Color** box to configure the text style when the expression is true. The Style Selection dialog box appears. For more information, see Setting Style on page 166.
- 7 Click the browse button for the **Font** box, to select a font, font style and size for the text when the expression is true.
- 8 Repeat the above steps for the false condition in the **False, 0, Off** area.
- 9 Click **OK**.

### To set default text style and/or font in a Boolean text style animation

- 1 Open the **Edit Animations** dialog box, **Boolean Text Style** panel.
- 2 In the **Element Text Style** area, select a style and/or a font for the default Boolean text style.

### To use default text style and/or font in a Boolean text style animation

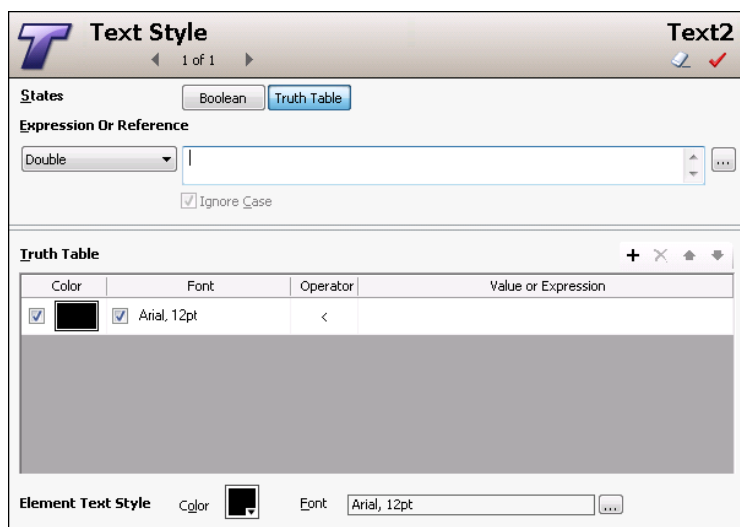
- 1 Open the **Edit Animations** dialog box, **Boolean Text Style** panel.
- 2 In the **True, 1, On** or **False, 0, Off** areas, clear **Color** and/or **Font** to use the corresponding default style and/or font.

## Configuring a Truth Table Text Style Animation

You can configure an element with a truth table text style animation.

### To configure an element with a truth table text style animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Text Style**. The text style animation is added to the Animation list. The **Text Style** information page appears.
- 4 Click the **Truth Table** button. The **Truth Table Text Style** configuration panel appears.



- Select the data type of the expression from the list.
- Type a value, attribute reference or expression in the text box.

- 5 If the data type of the expression is string or internationalized string, you can specify to ignore the case by selecting **Ignore Case**.
- 6 In the **Truth Table**, click the color box in the **Color** column. The **Style Selection** dialog box appears. For more information, see *Setting Style* on page 166.
- 7 Select the truth options. Do one of more of the following:
  - Click on the cell in the **Font** column to select a font.
  - In the **Operator** column, select the comparison operator.
  - In the **Value or Expression** column, type a value, attribute reference or expression.
  - To add further conditions, see *To add a condition to a truth table text style animation* on page 276.
- 8 Click **OK**.

#### To set the default text style or font for a truth table text style animation

- 1 Open the **Edit Animations** dialog box, **Truth Table Text Style** panel.
- 2 In the **Element Text Style** area, select a style and a font for the default truth table text style.

#### To use the default text style or font in a truth table text style animation

- 1 Open the **Edit Animations** dialog box, **Truth Table Text Style** panel.
- 2 Locate the condition for which you want to change the text style or font.
- 3 To use the default text style for the condition, clear the mark in the **Color** column of the truth table.
- 4 To use the default font for the condition, clear the mark in the **Font** column of the truth table.

#### To add a condition to a truth table text style animation



- 1 In the **Edit Animations** dialog box, **Truth Table Text Style** panel, click the **Add** icon. An additional condition is added to the truth table.
- 2 Configure style, font, operator and breakpoint value according to your requirements.

**To delete a condition from a truth table text style animation**

- 1 In the **Edit Animations** dialog box, **Truth Table Text Style** panel, select the condition you want to delete.
- 2 Click the **Remove** button.

**To change the processing order of text style conditions**

- 1 Open the **Edit Animations** dialog box, **Truth Table Text Style** panel
- 2 Select the condition you want to move up or down the condition list in order for it to be processed sooner or later.
- 3 Click the:
  - **Arrow up** icon to move the condition up in the truth table.
  - **Arrow down** icon to move the condition down in the truth table.



## Configuring a Blink Animation

You can configure an element with a blink animation. You can specify:

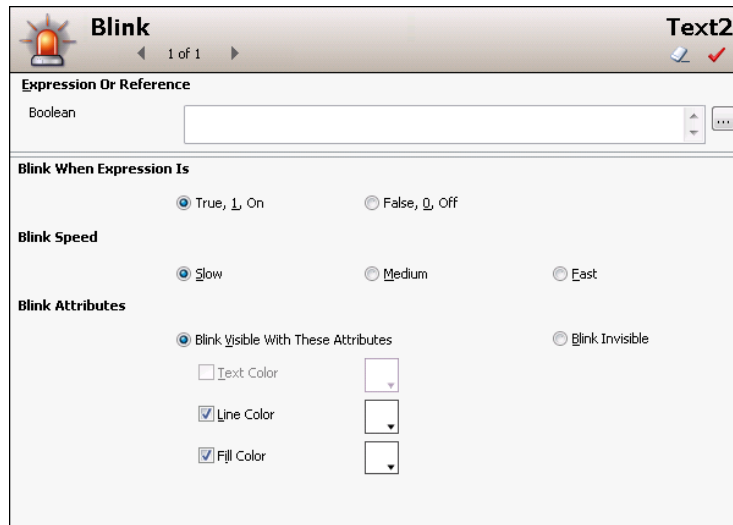
- The blinking speed: slow, medium or fast.
- If the element should blink invisibly or if it should blink with specified colors.

**To configure an element with a blink animation**

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.



- 3 Click the **Add** icon and select **Blink**. The blink animation is added to the Animation list and the **Blink** configuration panel appears.



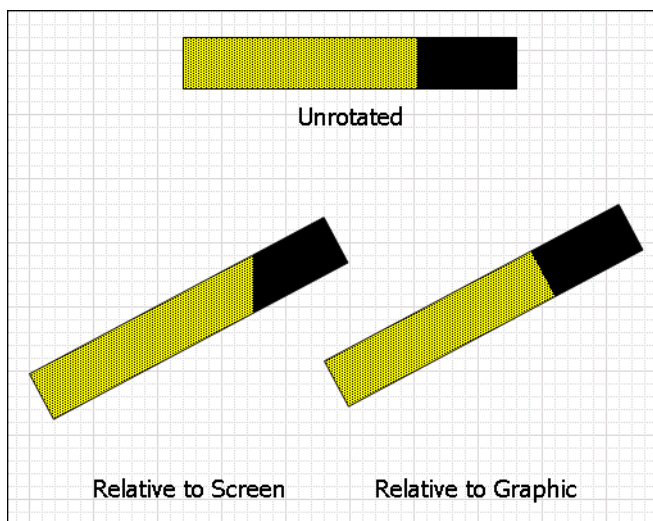
- 4 In the **Boolean** box, type a Boolean numeric value, attribute reference or expression.
- 5 In the **Blink When Expression Is** area, select:
  - **True, 1, On** to enable blinking when the expression is true.
  - **False, 0, Off** to enable blinking when the expression is false.
- 6 In the **Blink Speed** area, select **Slow**, **Medium** or **Fast** for the blinking speed.
- 7 In the **Blink Attributes** area, select **Blink Visible With These Attributes** or **Blink Invisible**.
- 8 If you select **Blink Visible With These Attributes**, you can configure the styles used at run time for the text, line and fill component of the element. Click on the corresponding color box, and the Style Selection dialog box appears. For more information, see Setting Style on page 166.
- 9 Click **OK**.

## Configuring a Percent Fill Horizontal Animation

You can configure an element with a percent fill horizontal animation.

Besides specifying the expressions that determine how much of the element is filled at run time, you can also specify:

- **Fill direction:** from left to right or right to left.
- **Unfill color:** the style of the background when the element has 0 percent filling.
- **Fill orientation:** if the filling is in relation to the element or to the screen. This affects how the fill appears if the orientation of the element changes. If the fill is in relation to the screen and the element or symbol are rotated, the fill remains in relation to the screen.




---

**Note** The fill orientation is a common setting to the percent fill horizontal and percent fill vertical animations.

---

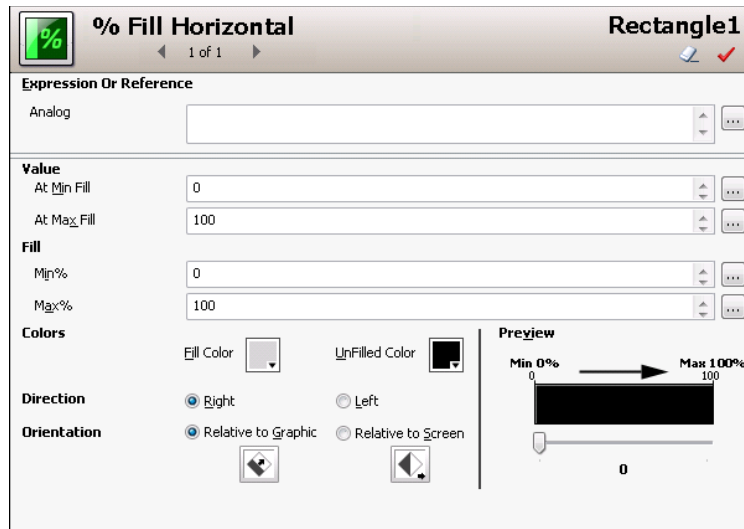
You can also preview how the percent fill horizontal animation appears at run time.

### To configure an element with a percent fill horizontal animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.



- Click the **Add** icon and select **% Fill Horizontal**. The percent fill horizontal animation is added to the Animation list and the **% Fill Horizontal** configuration panel appears.

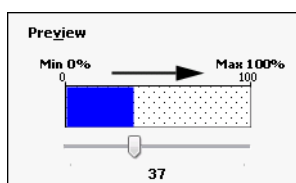


- Specify the settings. Do one or more of the following:
  - In the **Analog** box, type an analog value, attribute reference or expression.
  - In the **Value - At Min Fill** box, type an analog value, attribute reference or expression that causes the minimum percent of filling at run time.
  - In the **Value - At Max Fill** box, type an analog value, attribute reference or expression that causes the maximum percent of filling at run time.
  - In the **Fill - Min%** box, type an analog value, attribute reference or expression to specify the minimum percent of filling.
  - In the **Fill - Max%** box, type an analog value, attribute reference or expression to specify the maximum percent of filling.
  - In the **Colors** area, click the:
    - Fill Color** box to select a style from the Style Selection dialog box. This is the fill style of the element.
    - Unfilled Color** box to select a style from the Style Selection dialog box. This is the unfilled fill style of the element.

For more information, see [Setting Style](#) on page 166.



- 5 In the **Direction** area, select:
  - **Right** - to fill from left to right.
  - **Left** - to fill from right to left.
- 6 In the **Orientation** area, select:
  - **Relative to Graphic** - so that the filling is in relation to the element and the filling rotates with the element.
  - **Relative to Screen** - so that the filling is in relation to the screen and the filling does not rotate with the element.
- 7 You can preview your configuration by using the slider in the **Preview** area. Drag the slider to see how different values affect the appearance at run time.



- 8 Click **OK**.

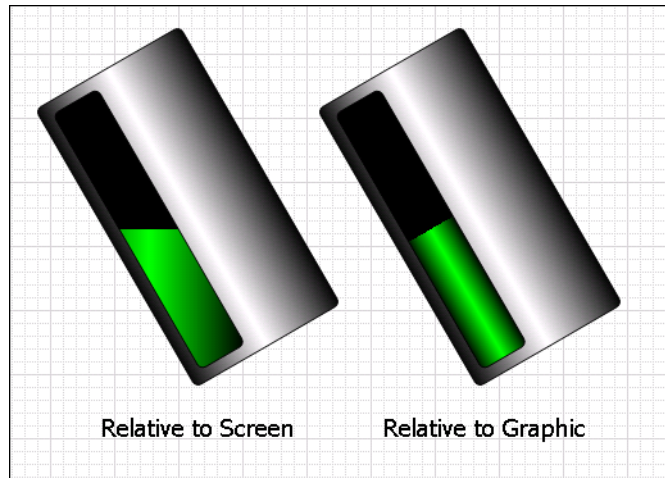
## Configuring a Percent Fill Vertical Animation

You can configure an element with a percent fill vertical animation.

Besides specifying the expressions that determine how much of the element is filled at run time, you can also specify:

- **Fill direction:** from lower to top or top to lower.
- **Unfill color:** the style of the background when the element has 0 percent filling.

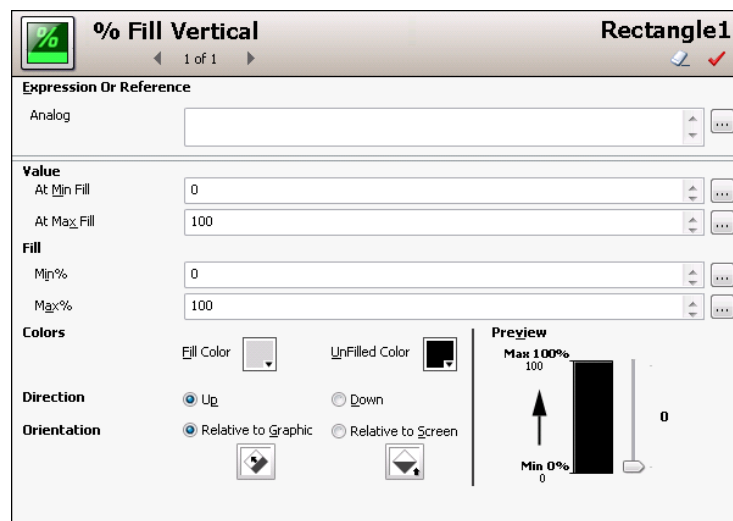
- **Fill orientation:** if the filling is in relation to the element or to the screen. This affects how the fill appears if the orientation of the element changes. If the fill is in relation to the screen and the element or symbol are rotated, the fill remains in relation to the screen.



**Note** The fill orientation is a common setting to the percent fill horizontal and percent fill vertical animations.

#### To configure an element with a percent fill vertical animation

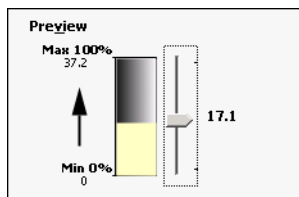
- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **% Fill Vertical**. The percent fill vertical animation is added to the Animation list and the **% Fill Vertical** configuration panel appears.



- 4 In the **Analog** box, type an analog value, attribute reference or expression.
- 5 In the **Value-At Min Fill** box, type an analog value, attribute reference or expression that causes the minimum percent of filling at run time.
- 6 In the **Value-At Max Fill** box, type an analog value, attribute reference or expression that causes the maximum percent of filling at run time.
- 7 In the **Fill-Min%** box, type an analog value, attribute reference or expression to specify the minimum percent of filling.
- 8 In the **Fill-Max%** box, type an analog value, attribute reference or expression to specify the maximum percent of filling.
- 9 In the **Colors** area, click the:
  - **Fill Color** box to select a style from the Style Selection dialog box. This is the fill style of the element.
  - **Unfilled Color** box to select a style from the Style Selection dialog box. This is the unfilled fill style of the element.

For more information, see [Setting Style](#) on page 166.

- 10 In the **Direction** area, select:
  - **Up** - to fill from lower to top.
  - **Down** - to fill from top to lower.
- 11 In the **Orientation** area, select:
  - **Relative to Graphic** - so that the filling is in relation to the element and the filling rotates with the element.
  - **Relative to Screen** - so that the filling is in relation to the screen and the filling does not rotate with the element.
- 12 You can preview your configuration by using the slider in the **Preview** area. Drag the slider to see how different values affect the appearance at run time.



- 13 Click **OK**.

## Configuring a Horizontal Location Animation

You can configure an element with a horizontal location animation.

### To configure an element with a horizontal location animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Location Horizontal**. The horizontal location animation is added to the Animation list and the **Location Horizontal** configuration panel appears.



Location Horizontal		Rectangle1
Expression Or Reference		
Analog	<input type="text"/>	...
Value		
At Left End	<input type="text" value="0"/>	...
At Right End	<input type="text" value="100"/>	...
Movement		
To Left	<input type="text" value="0"/>	...
To Right	<input type="text" value="100"/>	...

- 4 In the **Analog** box, type an analog value, attribute reference or expression.
- 5 In the **Value-At Left End** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement-To Left** value.
- 6 In the **Value-At Right End** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement-To Right** value.
- 7 In the **Movement-To Left** box, type an analog value, attribute reference or expression for the maximum offset to the left.
- 8 In the **Movement-To Right** box, type an analog value, attribute reference or expression for the maximum offset to the right.
- 9 Click **OK**.

## Configuring a Vertical Location Animation

You can configure an element with a vertical location animation.

To configure an element with a vertical location animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Location Vertical**. The vertical location animation is added to the Animation list and the **Location Vertical** configuration panel appears.



- 4 In the **Analog** box, type an analog value, attribute reference or expression.
- 5 In the **Value - At Top** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement - Up** value.
- 6 In the **Value - At lower** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement - Down** value.
- 7 In the **Movement - Up** box, type an analog value, attribute reference or expression for the maximum offset upwards.
- 8 In the **Movement - Down** box, type an analog value, attribute reference or expression for the maximum offset downwards.
- 9 Click **OK**.

## Configuring a Width Animation

You can configure an element with a width animation. You can also specify if the element is to be anchored to its left, center, right side or origin.

### To configure an element with a width animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Width**. The width animation is added to the Animation list and the **Width** configuration panel appears.



- 4 In the **Analog** box, type an analog value, attribute reference or expression.
- 5 In the **Value-At Min Size** box, type an analog value, attribute reference or expression that corresponds to the minimum width specified by the **Width-Min%** value.
- 6 In the **Value-At Max Size** box, type an analog value, attribute reference or expression that corresponds to the maximum width specified by the **Width-Max%** value.
- 7 In the **Width-Min%** box, type an analog value, attribute reference or expression for the minimum width in percent of the original element.
- 8 In the **Width-Max%** box, type an analog value, attribute reference or expression for the maximum width in percent of the original element.

- 9 In the **Anchor** area, select:
  - **Left** - to specify that the left of the element is anchored.
  - **Center** - to specify that the horizontal center of the element is anchored.
  - **Right** - to specify that the right side of the element is anchored.
  - **Origin** - to specify that the origin of the element is anchored.
- 10 Click **OK**.

## Configuring a Height Animation

You can configure an element with a height animation. You can also specify if the element is to be anchored to its top side, middle, lower side or origin.

### To configure an element with a height animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Height**. The height animation is added to the Animation list and the **Height** configuration panel appears.



The screenshot shows the 'Height' configuration panel for 'Rectangle1'. The panel has a title bar with a green up/down arrow icon, the text 'Height', and 'Rectangle1' on the right. Below the title bar is a navigation bar with '1 of 1' and a red checkmark. The main area is divided into sections: 'Expression Or Reference' with an 'Analog' input field; 'Value' with 'At Min Size' (0) and 'At Max Size' (100) input fields; 'Height' with 'Min%' (0) and 'Max%' (100) input fields; and 'Anchor' with radio buttons for 'Top', 'Middle' (selected), 'Bottom', and 'Origin'.

- 4 In the **Analog** box, type an analog value, attribute reference or expression.
- 5 In the **Value-At Min Size** box, type an analog value, attribute reference or expression that corresponds to the minimum height specified by the **Height-Min%** value.

- 6 In the **Value-At Max Size** box, type an analog value, attribute reference or expression that corresponds to the maximum height specified by the **Height-Max%** value.
- 7 In the **Height-Min%** box, type an analog value, attribute reference or expression for the minimum height in percent of the original element.
- 8 In the **Height-Max%** box, type an analog value, attribute reference or expression for the maximum height in percent of the original element.
- 9 In the **Anchor** area, select:
  - **Top** - to specify that the top side of the element is anchored.
  - **Middle** - to specify that the vertical center of the element is anchored.
  - **lower** - to specify that the lower side of the element is anchored.
  - **Origin** - to specify that the origin of the element is anchored.
- 10 Click **OK**.

## Configuring an Orientation Animation

You can configure an element with an orientation animation. You can also:

- Specify a different orientation origin.
- Ignore or accept the design-time orientation of the element on the canvas.
- Preview the orientation at run time with a slider.

### To configure an element with an orientation animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.





- 3 Click the **Add** icon and select **Orientation**. The orientation animation is added to the Animation list and the **Orientation** configuration panel appears.

**Orientation** Rectangle1

← 1 of 1 →

**Expression Or Reference**

Analog

---

**Value**

At CCW End

At CW End

**Orientation**

CCW

CW

**Orientation Offset**

Replace Current

Use Current as Offset Orientation

**Current Relative Origin**

dX  dY

**Preview**

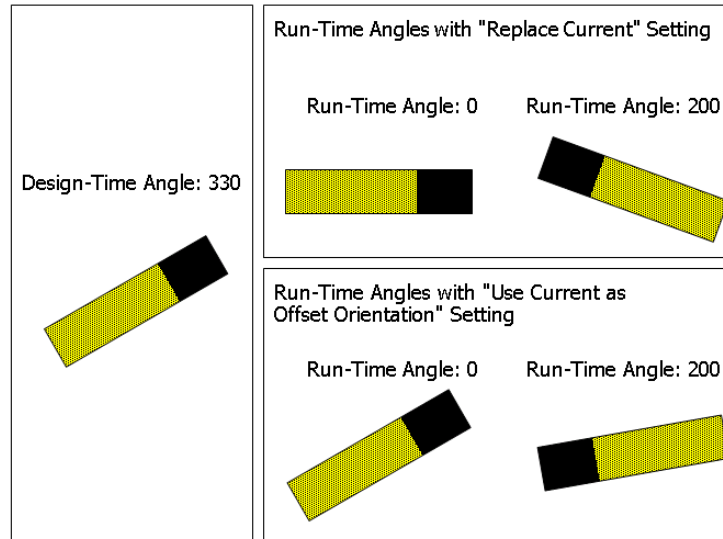
Max/Min Current

100 0

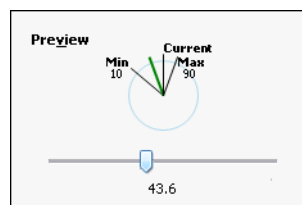
0

- 4 In the **Analog** box, type an analog value, attribute reference or expression.
- 5 In the **Value-At CCW End** box, type an analog value, attribute reference or expression that corresponds to the maximum angle in degrees for the counter-clockwise orientation as specified by the **Orientation-CCW** value.
- 6 In the **Value-At CW End** box, type an analog value, attribute reference or expression that corresponds to the maximum angle in degrees for the counter-clockwise orientation as specified by the **Orientation-CW** value.
- 7 In the **Orientation-CCW** box, type an analog value, attribute reference or expression for the maximum orientation in counter-clockwise direction in degrees.
- 8 In the **Orientation-CW** box, type an analog value, attribute reference or expression for the maximum orientation in clockwise direction in degrees.

- 9 In the **Orientation Offset** area, select:
- **Replace Current** to ignore the design-time orientation of the element as it appears on the canvas and to use absolute orientation.
  - **Use Current as Offset Orientation** to orientate the element at run time in relation to its design-time orientation on the canvas.



- 10 If you use current as offset orientation, you can type an offset value in the text box next to **Use Current as Offset Orientation**. This affects the orientation of the element on the canvas.
- 11 In the **Current Relative Origin** area, type values in the **dX** and **dY** boxes to specify the rotation origin as offset from the element center point. This affects the point of origin of the element on the canvas.
- 12 You can preview the orientation and how run-time values affect the appearance of the element, by dragging the slider in the **Preview** area.



- 13 Click **OK**.

## Configuring a Value Display Animation

You can configure an element with a value display animation. You can show:

- A Boolean value as a Message.
- An Analog value.
- A string value.
- A time or date value.
- The tag name, hierarchical name or contained name of the hosting object.

### Configuring a Boolean Value Display Animation

You can configure an element to show a Boolean value as a message.

To configure an element with a Boolean value display animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Value Display**. The value display animation is added to the Animation list and the **Value Display** state selection panel appears.
- 4 Click the **Boolean** button. The **Boolean Value Display** configuration panel appears.



- 5 In the **Boolean** box, type a Boolean value, attribute reference or expression.
- 6 In the **True Message** box, type a value, attribute reference or expression for the text display when the expression is true.

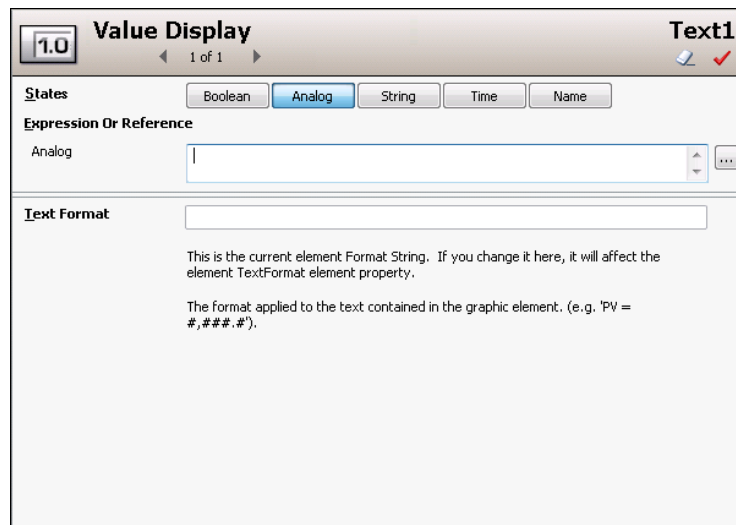
- 7 In the **False Message** box, type a value, attribute reference or expression for the text display when the expression is false.
- 8 Click **OK**.

### Configuring an Analog Value Display Animation

You can configure an element to show an analog value.

To configure an element with an analog value display animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Value Display**. The value display animation is added to the Animation list and the **Value Display** state selection panel appears.
- 4 Click the **Analog** button. The **Analog Value Display** configuration panel appears.



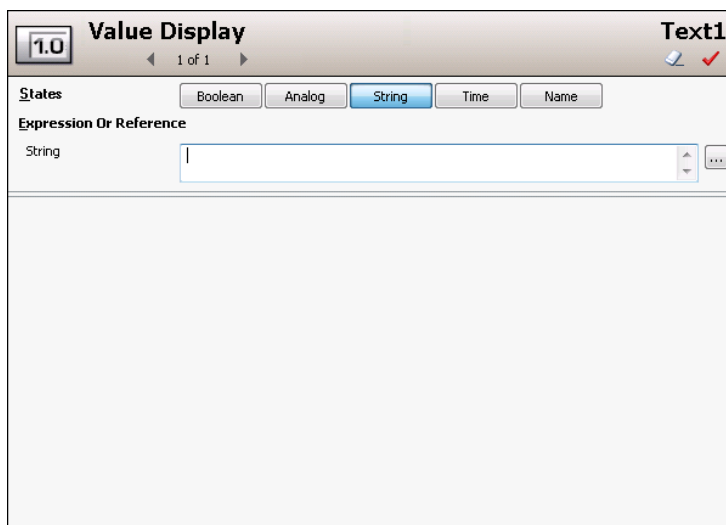
- 5 In the **Analog** box, type an analog value, attribute reference or expression.
- 6 In the **Text Format** box, type a format for the value output. If you change this value, the **TextFormat** property of the element also changes. For example: `PV = #.##`
- 7 Click **OK**.

## Configuring a String Value Display Animation

You can configure an element to show a string value.

### To configure an element with a string value display animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Value Display**. The value display animation is added to the Animation list and the **Value Display** state selection panel appears.
- 4 Click the **String** button. The **String Value Display** configuration panel appears.



- 5 In the **String** box, type a string value, attribute reference or expression.
- 6 Click **OK**.

## Configuring a Time Value Display Animation

You can configure an element to show a time value.

Use the following letters to set the time format:

h	The one or two-digit hour in 12-hour format.
hh	The two-digit hour in 12-hour format. Single digit values are preceded by a zero.
H	The one or two-digit hour in 24-hour format.
HH	The two-digit hour in 24-hour format. Single digit values are preceded by a zero.
t	The one-letter AM/PM abbreviation ("AM" appears as "A").
tt	The two-letter AM/PM abbreviation ("AM" appears as "AM").
m	The one or two-digit minute.
mm	The two-digit minute. Single digit values are preceded by a zero.
s	The one or two-digit seconds.
ss	The two-digit seconds. Single digit values are preceded by a zero.
d	The one or two-digit day.
dd	The two-digit day. Single digit day values are preceded by a zero.
ddd	The three-character day-of-week abbreviation.
dddd	The full day-of-week name.
M	The one or two-digit month number.
MM	The two-digit month number. Single digit values are preceded by a zero.
MMM	The three-character month abbreviation.
MMMM	The full month name.
y	The one-digit year (2001 appears as "1").
yy	The last two digits of the year (2001 appears as "01").
yyyy	The full year (2001 appears as "2001").

The format for elapsed time is:

"[-][DDDDDD] [HH:MM:]SS[.ffffff]"

Use the following letters to set the elapsed time format:

DDDDDD	The number of days. Valid values are 0 to 999999.
HH	The two-digit hour in 24-hour format. Single digit values are preceded by a zero. Valid values are 00 to 23.
MM	The two-digit month number. Single digit values are preceded by a zero. Valid values are 00 to 59.
SS	The two-digit seconds. Single digit values are preceded by a zero. Valid values are 00 to 59.
ffffff	Optional fractional seconds to right of decimal, and can be one through seven digits.

---

**Note** You can use any other characters, except “g” in the property. These characters then appear at design time and run time in the control.

---

#### To configure an element with a time value display animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Value Display**. The value display animation is added to the Animation list and the **Value Display** state selection panel appears.
- 4 Click the **Time** button. The **Time Value Display** configuration panel appears.



**Value Display** Text1

1.0 1 of 1

States Boolean Analog String Time Name

**Expression Or Reference**

Time or Elapsed Time

**Text Format**

This is the current element Format String. If you change it here, it will affect the element TextFormat element property.

The format applied to the text contained in the graphic element. (e.g. 'Start Date =' MM/dd/yyyy hh:mm).

- 5 In the **Time or Elapsed Time** box, type a time or elapsed time value, attribute reference or expression.
- 6 In the **Text Format** box, type a format for the value output. If you change this value, the TextFormat property of the element also changes.
- 7 Click **OK**.

### Configuring a Name Display Animation

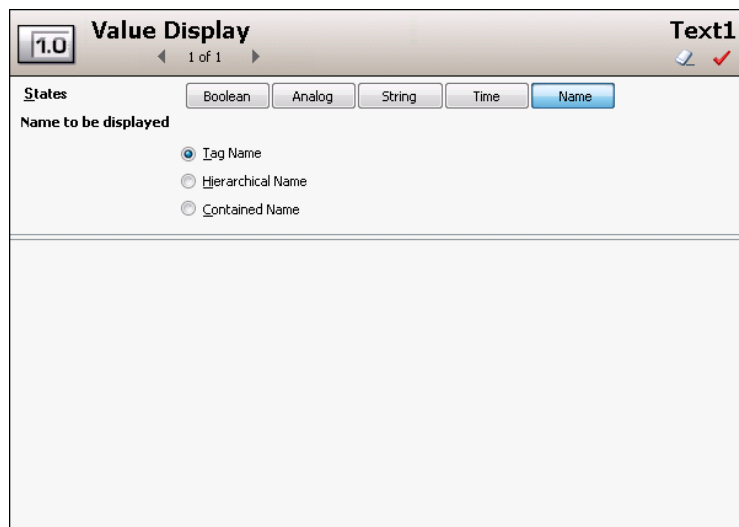
You can configure an element to show the tag name, hierarchical name or contained name of the AutomationObject that is hosting it.

For example if the AutomationObject hosting the symbol is named Valve\_001 and Valve\_001 is contained in Pump\_001 and has a contained name of InletValve, then configuring an element with the value display animation with:

- **Tag Name** shows Valve\_001 at run time
- **Hierarchical Name** shows Pump\_001.InletValve at run time
- **Contained Name** shows InletValve at run time

#### To configure an element with a name display animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Value Display**. The value display animation is added to the Animation list and the **Value Display** state selection panel appears.
- 4 Click the **Name** button. The **Name Display** configuration panel appears.





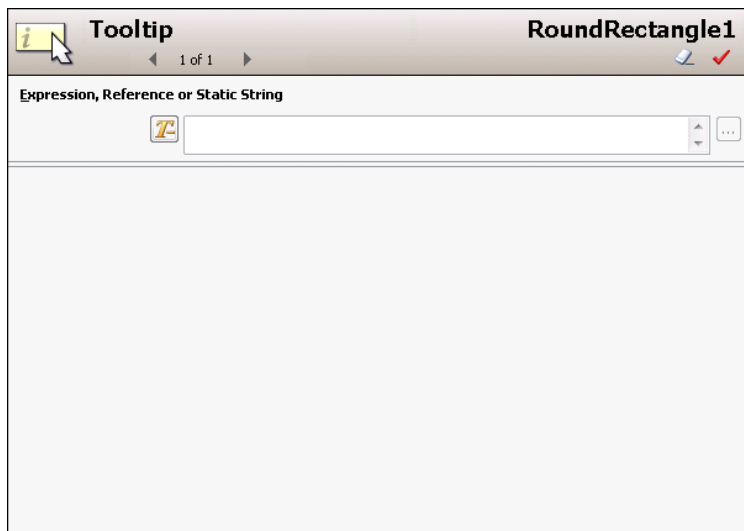
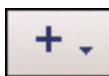
- 5 Select:
  - **Tag Name** to show the tag name of the hosting AutomationObject.
  - **Hierarchical Name** to show the hierarchical name of the hosting AutomationObject.
  - **Contained Name** to show the contained name of the hosting AutomationObject.
- 6 Click **OK**.

## Configuring a Tooltip Animation

You can configure an element with a tooltip animation.

### To configure an element with a tooltip animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Tooltip**. The tooltip animation is added to the Animation list and the **Tooltip** configuration panel appears.



- 4 In the **Expression** box, type:
  - A static value and make sure the **Input Mode** icon is set to static.
  - An attribute reference or expression and make sure the **Input Mode** icon is set to attribute or reference.
- 5 Click **OK**.

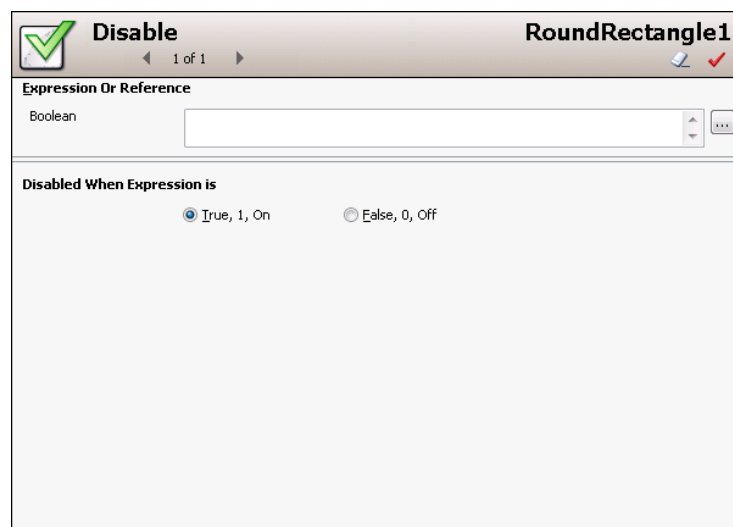


## Configuring a Disable Animation

You can configure an element with a disable animation. This lets you disable user interaction with an element depending on a run-time value or expression.

### To configure an element with a disable animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Disable**. The disable animation is added to the Animation list and the **Disable** configuration panel appears.



- 4 In the **Boolean** box, type a Boolean numeric value, attribute reference or expression.
- 5 In the **Disabled When Expression is** area, select:
  - **True, 1, On** in which case the element is disabled at run time whenever the expression is true, and enabled whenever the expression is false.
  - **False, 0, Off** in which case the element is disabled at run time whenever the expression is false, and enabled whenever the expression is true.
- 6 Click **OK**.

## Configuring a User Input Animation

You can configure an element with a user input animation for the following data types:

- Boolean
- Analog (integer, float, double)
- String
- Time
- Elapsed time

### Configuring a User Input Animation for a Discrete Value

You can configure an element with a user input animation for a Boolean value.

To configure an element with a user input animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **User Input**. The user input animation is added to the Animation list and the **User Input** state selection panel appears.
- 4 Click the **Boolean** button. The **Boolean Value User Input** configuration panel appears.



The screenshot shows the 'User Input' configuration panel for 'RoundRectangle1'. The panel is titled 'User Input' and 'RoundRectangle1'. It features a 'States' section with buttons for 'Boolean', 'Analog', 'String', 'Time', and 'Elapsed Time'. The 'Boolean' button is selected. Below this is a 'Reference' section with a text field containing a vertical bar and a dropdown arrow. The 'Message to User' section has a text field with a 'T' icon. The 'Prompt' section includes 'True Message' (ON) and 'False Message' (OFF) fields, each with a 'T' icon. The 'Shortcut' section has checkboxes for 'Ctrl' and 'Shift', and a 'Key' dropdown set to 'None'. The 'Interaction' section has an 'Input Only' checkbox. The 'Display Value' section includes 'True Message' (ON) and 'False Message' (OFF) fields, each with a 'T' icon.

- 5 Specify the options. Do one or more of the following:
  - In the **Boolean** box, type an attribute reference or browse for one by using the browse button.
  - In the **Message to User** box, type a value, attribute reference or expression. This is the text that appears as prompt on the Boolean value input dialog box at run time.
  - In the **Prompt - True Message** box, type a value, attribute reference or expression. This is the text that appears on the button that causes the attribute to be set to true.
  - In the **Prompt - False Message** box, type a value, attribute reference or expression. This is the text that appears on the button that causes the attribute to be set to false.
  - Specify that the input dialog box appears by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
  - If you don't want the discrete value display element to show the True Message and False Message, select **Input Only**.
  - In the **Display Value - True Message** box, type a value, attribute reference or expression. This is the text that appears on the canvas when the associated attribute is true.
  - In the **Display Value - False Message** box, type a value, attribute reference or expression. This is the text that appears on the canvas when the associated attribute is false.
  - Make sure that the input modes of all boxes are set correctly. Click the **Input Mode** icon to set a static value or an attribute reference or expression.



- 6 Click **OK**.

## Configuring a User Input Animation for an Analog Value

You can configure an element with a user input animation for an analog value.

### To configure an element with a user input animation for an analog value

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **User Input**. The user input animation is added to the Animation list and the **User Input** state selection panel appears.
- 4 Click the **Analog** button. The **Analog Value User Input** configuration panel appears.



- 5 In the **Analog** box, type an attribute reference or browse for one by using the browse button.
- 6 In the **Message to User** box, type a value, attribute reference or expression. This is the text that appears as prompt on the analog value input dialog box at run time.
- 7 Make sure that the input mode of the **Message to User** box is set correctly. Click the **Input Mode** icon to set a static value or an attribute reference or expression.



- 8 If you want to restrict the range of input values, you can do so in the **Value Limits** area by:
  - First selecting **Restrict Values**.
  - The Typing values, attribute references or expressions for the **Minimum** and **Maximum** boxes.
- 9 Specify that the **Input** dialog box appears by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 10 If you don't want the analog value display element to show the analog input result on the canvas, select **Input Only**.
- 11 To use the keypad to type the analog value, select **Use Keypad**.
- 12 If you want to format your text during input, type a valid text format in the **Text Format** box. For example: PV = #.##
- 13 Click **OK**.

### Configuring a User Input Animation for a String Value

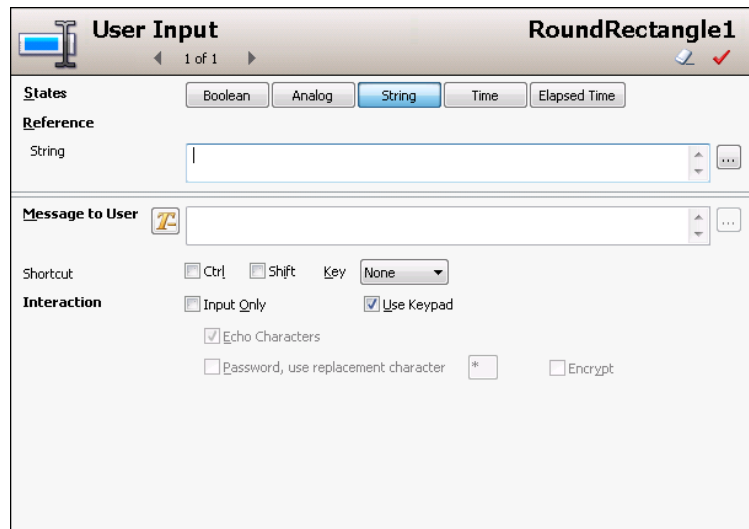
You can configure an element with a user input animation for a string value.

#### To configure an element with a user input animation for a string value

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **User Input**. The user input animation is added to the Animation list and the **User Input** state selection panel appears.



- 4 Click the **String** button. The **String Value User Input** configuration panel appears.



- 5 In the **String** box, type an attribute reference or browse for one by using the browse button.
- 6 In the **Message to User** box, type a value, attribute reference or expression. This is the text that appears as prompt on the string value input dialog box at run time.
- 7 Make sure that the input mode of the **Message to User** box is set correctly. Click the **Input Mode** icon to set a static value or an attribute reference or expression.
- 8 You can specify that the **Input** dialog box appears by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 9 If you don't want the string value display element to show the string input result on the canvas, select **Input Only**.
- 10 If you want to use the keypad to type the string value, select **Use Keypad**.
- 11 If you select **Input Only** and want to see placeholders during the input at run time, select **Echo Characters**.
- 12 If you are configuring a password input:
- Select **Password**.
  - Type in the replacement character in the adjacent box.
  - Select **Encrypt** if you want to encrypt the string that holds the password.
- 13 Click **OK**.

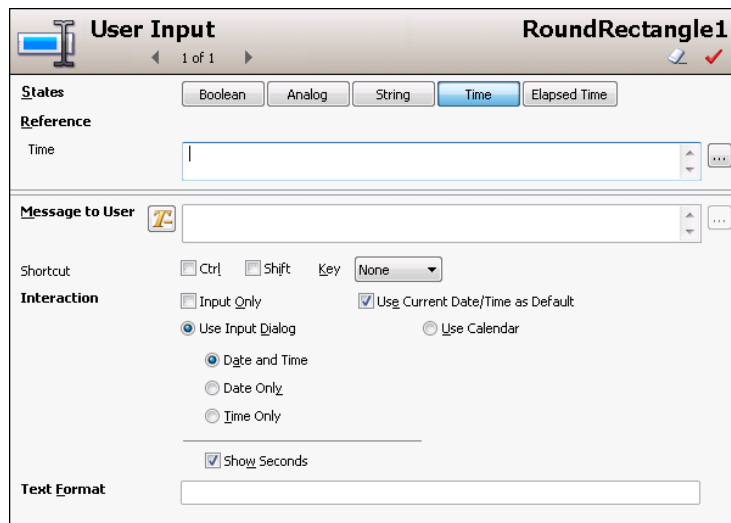


## Configuring a User Input Animation for a Time Value

You can configure an element with a user input animation for a time value.

**To configure an element with a user input animation for a time value**

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **User Input**. The user input animation is added to the Animation list and the **User Input** state selection panel appears.
- 4 Click the **Time** button. The **Time Value User Input** configuration panel appears.



- 5 In the **Time** box, type an attribute reference or browse for one by using the browse button.
- 6 In the **Message to User** box, type a value, attribute reference or expression. This is the text that appears as prompt on the time value input dialog box at run time.
- 7 Make sure that the input mode of the **Message to User** box is set correctly. Click the **Input Mode** icon to set a static value or an attribute reference or expression.
- 8 Specify that the **Input** dialog box appears by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 9 If you don't want the time value display element to show the time input result on the canvas, select **Input Only**.





- 10 To use the current date and time as default, select **Use Current Date/Time as Default**.
- 11 Select:
- **Use Input Dialog** to use the **Time User Input** dialog box at run time to type date and time values in individual boxes.
  - **Use Calendar** to use the **Time User Input** dialog box at run time to type a date with the calendar control.
- 12 If you select **Use Input Dialog** to type the time value, you can select:
- **Date and Time** to type date and time.
  - **Date** to only type a date.
  - **Time** to only type a time.

Select **Show Seconds** if you also want to input seconds.

- 13 If you want to format your text after input, type a valid text format in the **Text Format** box. Use the following letters to set the time format:

h	The one or two-digit hour in 12-hour format.
hh	The two-digit hour in 12-hour format. Single digit values are preceded by a zero.
H	The one or two-digit hour in 24-hour format.
HH	The two-digit hour in 24-hour format. Single digit values are preceded by a zero.
t	The one-letter AM/PM abbreviation ("AM" appears as "A").
tt	The two-letter AM/PM abbreviation ("AM" appears as "AM").
m	The one or two-digit minute.
mm	The two-digit minute. Single digit values are preceded by a zero.
s	The one or two-digit seconds.
ss	The two-digit seconds. Single digit values are preceded by a zero.
d	The one or two-digit day.
dd	The two-digit day. Single digit day values are preceded by a zero.
ddd	The three-character day-of-week abbreviation.
dddd	The full day-of-week name.
M	The one or two-digit month number.
MM	The two-digit month number. Single digit values are preceded by a zero.
MMM	The three-character month abbreviation.

MMMM	The full month name.
y	The one-digit year (2001 appears as "1").
yy	The last two digits of the year (2001 appears as "01").
yyyy	The full year (2001 appears as "2001").

**Note** You can use any other characters, except “g” in the property. These characters then appear at design time and run time in the control.

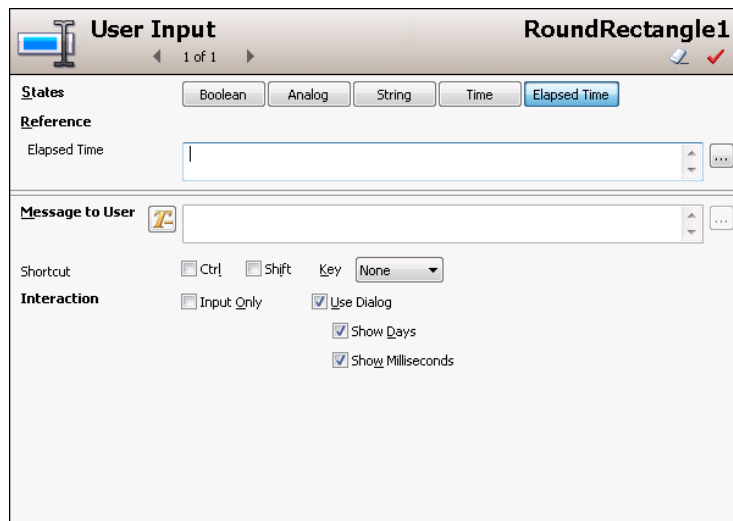
14 Click **OK**.

### Configuring a User Input Animation for an Elapsed Time Value

You can configure an element with a user input animation for an elapsed time value.

**To configure an element with a user input animation for an elapsed time value**

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **User Input**. The user input animation is added to the Animation list and the **User Input** state selection panel appears.
- 4 Click the **Elapsed Time** button. The **Elapsed Time Value User Input** configuration panel appears.



- 5 In the **Elapsed Time** box, type an attribute reference or browse for one by using the browse button.



- 6 In the **Message to User** box, type a value, attribute reference or expression. This is the text that appears as prompt on the elapsed time value input dialog box at run time.
- 7 Make sure that the input mode of the **Message to User** box is set correctly. Click the **Input Mode** icon to set a static value or an attribute reference or expression.
- 8 Specify that the **Input** dialog box appears by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 9 If you don't want the elapsed time value display element to show the time elapsed input result on the canvas, select **Input Only**.
- 10 Select **Use Dialog** to use the **Elapsed Time User Input** dialog box to type the elapsed time value at run time.
- 11 If you select **Use Dialog** to type the elapsed time value, you can optionally select:
  - **Show Days** if you also want to input days.
  - **Show Milliseconds** if you also want to input milliseconds.
- 12 Click **OK**.

## Configuring a Horizontal Slider Animation

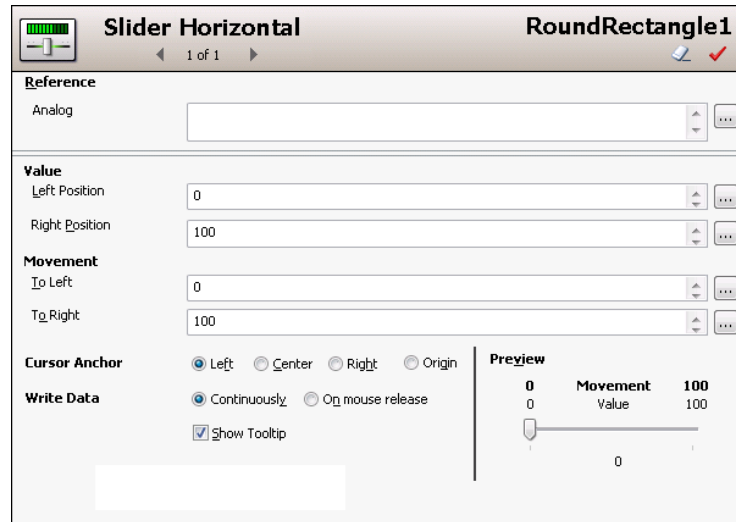
You can configure an element with a horizontal slider animation. This lets you drag an element at run time in horizontal direction and write a corresponding value back to an attribute.

### To configure an element with a horizontal slider animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.

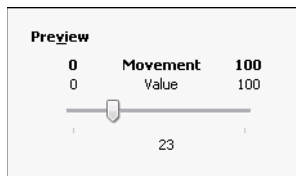


- Click the **Add** icon and select **Slider Horizontal**. The horizontal slider animation is added to the Animation list and the **Slider Horizontal** configuration panel appears.



- In the **Analog** box, type an attribute reference or browse for one by using the browse button.
- In the **Value - Left Position** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement - To Left** value.
- In the **Value - Right Position** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement - To Right** value.
- In the **Movement - To Left** box, type an analog value, attribute reference or expression for the maximum offset to the left in pixels.
- In the **Movement - To Right** box, type an analog value, attribute reference or expression for the maximum offset to the right in pixels.
- You can select where the cursor is anchored to the element when it is dragged at run time. In the **Cursor Anchor** area, select:
  - Left** to anchor the element at its left.
  - Center** to anchor the element at its center point.
  - Right** to anchor the element at its right side.
  - Origin** to anchor the element at its point of origin.
- You can select if position data from the slider is written continuously to the attribute, or only one time when the mouse button is released. In the **Write Data** area, select **Continuously** or **On mouse release**.
- If you want a tooltip to appear on the element showing the current value during dragging, select **Show Tooltip**.

- Preview the movement as it appears in run time by dragging the slider in the **Preview** area.



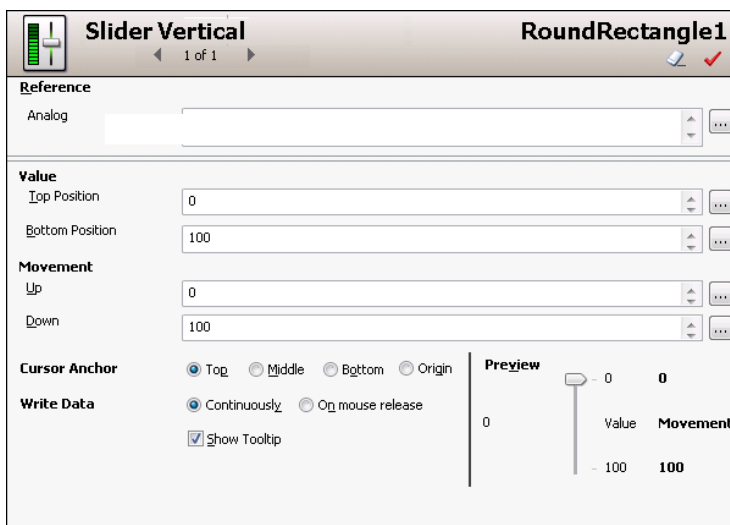
- Click **OK**.

## Configuring a Vertical Slider Animation

You can configure an element with a vertical slider animation.

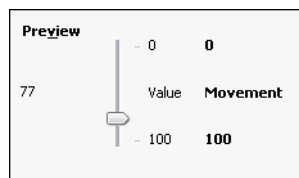
To configure an element with a vertical slider animation

- Select the element.
- On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- Click the **Add** icon and select **Slider Vertical**. The vertical slider animation is added to the Animation list and the **Slider Vertical** configuration panel appears.



- In the **Analog** box, type an attribute reference or browse for one by using the browse button.
- In the **Value - Top Position** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement - Up** value.
- In the **Value - lower Position** box, type an analog value, attribute reference or expression that corresponds to the offset specified by the **Movement - Down** value.

- 7 In the **Movement - Up** box, type an analog value, attribute reference or expression for the maximum offset upwards in pixels.
- 8 In the **Movement - Down** box, type an analog value, attribute reference or expression for the maximum offset downwards in pixels.
- 9 You can select where the cursor is anchored to the element when it is dragged at run time. In the **Cursor Anchor** area, select:
  - **Top** to anchor the element at its top side.
  - **Middle** to anchor the element at its middle point.
  - **lower** to anchor the element at its lower side.
  - **Origin** to anchor the element at its point of origin.
- 10 You can select if position data from the slider is written continuously to the attribute, or only one time when the mouse button is released. In the **Write Data** area, select **Continuously** or **On mouse release**.
- 11 If you want a tooltip to appear on the element showing the current value during dragging, select **Show Tooltip**.
- 12 Preview the movement as it appears in run time by dragging the slider in the **Preview** area.



- 13 Click **OK**.

## Configuring a Pushbutton Animation

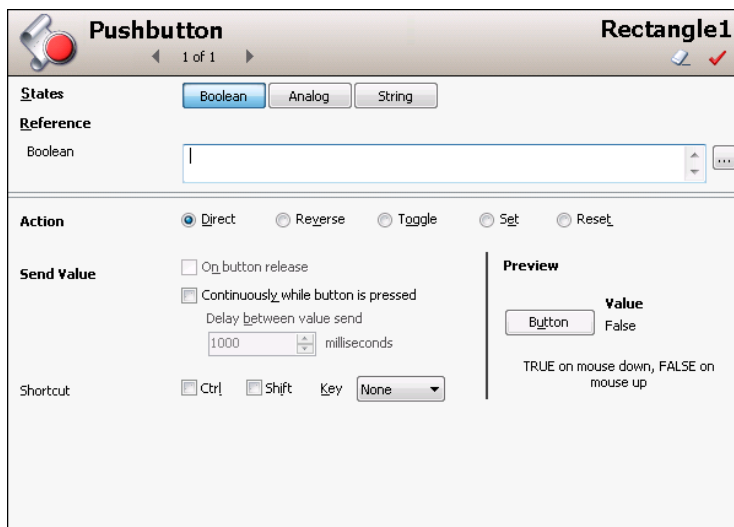
You can configure an element with a pushbutton animation to change Boolean, analog or string references.

### Configuring a Pushbutton Animation for a Boolean Value

You can configure an element with a pushbutton to change a Boolean value.

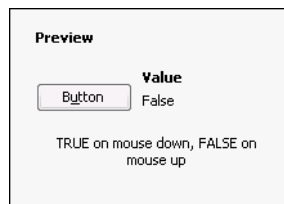
To configure an element with a pushbutton animation to change a Boolean value

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Pushbutton**. The pushbutton animation is added to the Animation list and the **Pushbutton** state selection panel appears.
- 4 Click the **Boolean** button. The **Boolean Pushbutton** configuration panel appears.



- 5 In the **Boolean** box, type a Boolean attribute reference or browse for one by using the browse button.

- 6 In the **Action** list, select:
  - **Direct** so the value becomes true when the element is clicked and the mouse button held. The value returns to false when the mouse button is released.
  - **Reverse** so the value becomes false when the element is clicked and the mouse button held. The value returns to true when the mouse button is released.
  - **Toggle** so the value becomes true if it is false and false if it is true when the element is clicked.
  - **Set** so the value is set to true when the element is clicked.
  - **Reset** so the value is set to false when the element is clicked.
- 7 If you select **Toggle**, **Set** or **Reset** as action, you can configure the action to be performed when the mouse button is released instead of pressed down. To do this, select **On button release**.
- 8 If you select **Direct**, **Reverse**, **Reset** or **Set** as action, you can configure the value to be written:
  - Continuously by selecting **Continuously while button is pressed**. Also specify the frequency the value is to be sent, by typing a value in the **Delay between value send** box.
  - One time by clearing **Continuously while button is pressed**.
- 9 Specify that the pushbutton action is executed by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 10 Preview the pushbutton run-time behavior by clicking **Button** in the **Preview** area.



- 11 Click **OK**.

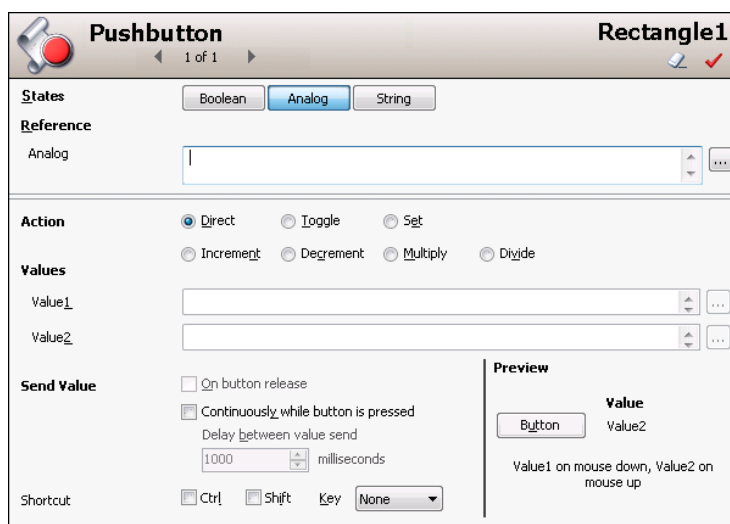


## Configuring a Pushbutton Animation for an Analog Value

You can configure an element with a pushbutton to set an analog value.

### To configure an element with a pushbutton animation to set an analog value

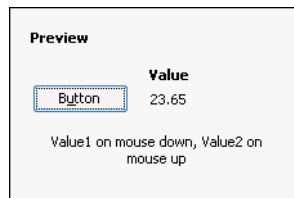
- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Pushbutton**. The pushbutton animation is added to the Animation list and the **Pushbutton** state selection panel appears.
- 4 Click the **Analog** button. The **Analog Pushbutton** configuration panel appears.



- 5 In the **Analog** box, type an attribute reference or browse for one by using the browse button.
- 6 From the **Action** list, select:
  - **Direct** so the value becomes Value1 when the element is clicked and the mouse button held. The value returns to Value2 when the mouse button is released.
  - **Toggle** so the value becomes Value1 if it is Value2 and Value2 if it is Value1 when the element is clicked.
  - **Set** so the value is set to Value1 when the element is clicked
  - **Increment** so the value is increased by Value1.
  - **Decrement** so the value is decreased by Value1.
  - **Multiply** so the value is multiplied with Value1.
  - **Divide** so the value is divided by Value1.

- 7 In the boxes **Value1** and, if applicable, **Value2**, type analog values, attribute references or references.
- 8 You can configure the value to be written when the mouse button is released instead. Select **On button release**. This does not apply if you select Direct as action.
- 9 You can configure the value to be written:
  - Continuously by selecting the **Continuously while button is pressed**. Also specify the frequency the value is to be sent, by typing a value in the **Delay between value send** box.
  - One time by clearing the **Continuously while button is pressed**.

This does not apply if you select Toggle as action.
- 10 Specify that the pushbutton action is executed by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 11 Preview the pushbutton run-time behavior by clicking **Button** in the **Preview** area. Click the button multiple times to preview the value changes over a period of time.



- 12 Click **OK**.

### Configuring a PushButton Animation for a String Value

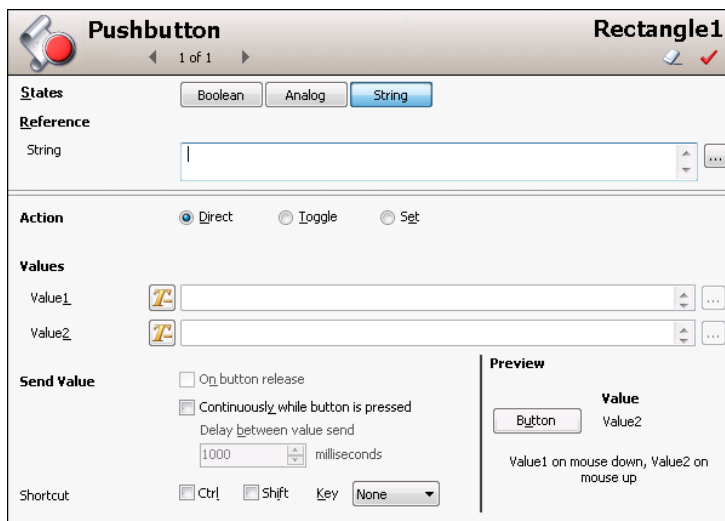
You can configure an element with a pushbutton to set a string value.

#### To configure an element with a pushbutton animation to set a string value

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Pushbutton**. The pushbutton animation is added to the Animation list and the **Pushbutton** state selection panel appears.



- 4 Click the **Analog** button. The **String Pushbutton** configuration panel appears.

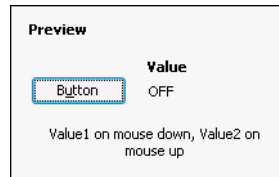


- 5 In the **String** box, type an attribute reference or browse for one by using the browse button.
- 6 From the **Action** list, select:
- **Direct** so the value becomes Value1 when the element is clicked and the mouse button held. The value returns to Value2 when the mouse button is released.
  - **Toggle** so the value becomes Value1 if it is Value2 and Value2 if it is Value1 when the element is clicked
  - **Set** so the value is set to Value1 when the element is clicked.
- 7 In the boxes **Value1** and, if applicable, **Value2**, type string values, attribute references or references.
- 8 Make sure that the input modes of the **Value1** and **Value2** boxes are set correctly. Click the **Input mode** icons to set a static values or an attribute references or expressions.
- 9 You can configure the value to be written when the mouse button is released instead. Select **On button release**. This does not apply if you select Direct as action.
- 10 You can configure the value to be written:
- Continuously by selecting the **Continuously while button is pressed**. Also specify the frequency the value is to be sent, by typing a value in the **Delay between value send** box.
  - One time by clearing the **Continuously while button is pressed**.

This does not apply if you select Toggle as action.



- 11 Specify that the pushbutton action is executed by pressing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 12 Preview the pushbutton run-time behavior by clicking **Button** in the **Preview** area.



- 13 Click **OK**.

## Configuring an Action Script Animation

You can configure an element with an action script animation.

You can assign multiple action scripts to one element that are activated in different ways such as:

Use this...	To activate the action script when the...
<b>On Primary Click/Key Down</b>	primary mouse button or a specific key is pressed.
<b>While Primary Click/Key Down</b>	primary mouse button or a specific key is pressed and held.
<b>On Primary Click/Key Up</b>	primary mouse button or a specific key is released
<b>On Primary Double Click</b>	primary mouse button is double-clicked.
<b>On Secondary Down</b>	secondary mouse button is pressed.
<b>While Secondary Down</b>	secondary mouse button is pressed and held.
<b>On Secondary Up</b>	secondary mouse button is released.
<b>On Secondary Double Click</b>	secondary mouse button is double-clicked.
<b>On Center Down</b>	center mouse button is pressed.
<b>While Center Down</b>	center mouse button is pressed and held.
<b>On Center Up</b>	center mouse button is released.

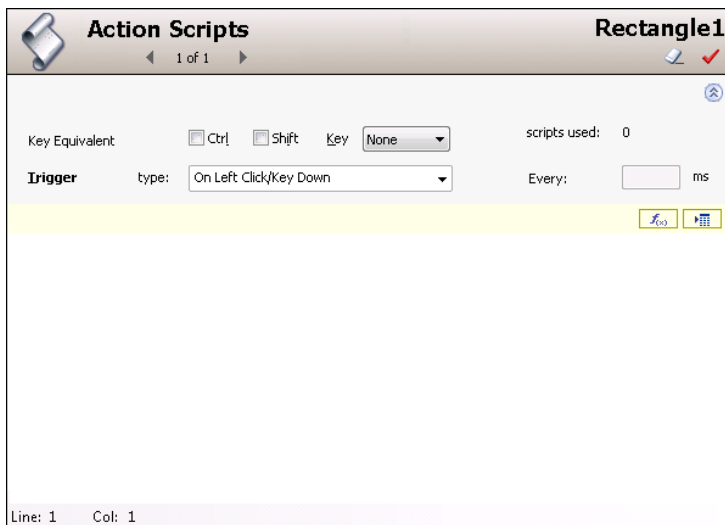
Use this...	To activate the action script when the...
<b>On Center Double Click</b>	center mouse button is double-clicked.
<b>On Mouse Over</b>	pointer is moved over the element.
<b>On Mouse Leave</b>	pointer is moved out of the element.
<b>On Startup</b>	element is first shown in WindowViewer.
<b>While Mouse Over</b>	pointer is over the element.



**Note** To expand the available space for your script you can use the **Expansion** buttons to hide the script header and/or the Animation list.

**To configure an element with an action script animation**

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Action Scripts**. The action scripts animation is added to the Animation list and the **Action Scripts** configuration panel appears.



- 4 From the **Trigger type** list, select the trigger that activates the action script at run time.
- 5 If you select a trigger type that starts with “While”, type how frequently the action script is executed at run time in the **Every** box.
- 6 If you select the trigger types **On Mouse Over** or **On Mouse Leave**, the **Every** box label shows **After** instead. Type a value in the **After** box. This value specifies after what delay the action script is executed at run time.
- 7 Specify a trigger type that involves pressing a key is run by typing a key or key combination. In the **Shortcut** area. Select a shortcut key in the **Key** list. Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 8 Create your script in the action script window.
- 9 Click **OK**.

---

**Note** For more information about scripts, see Adding and Maintaining Symbol Scripts on page 353.

---

## Configuring a Show Symbol Animation

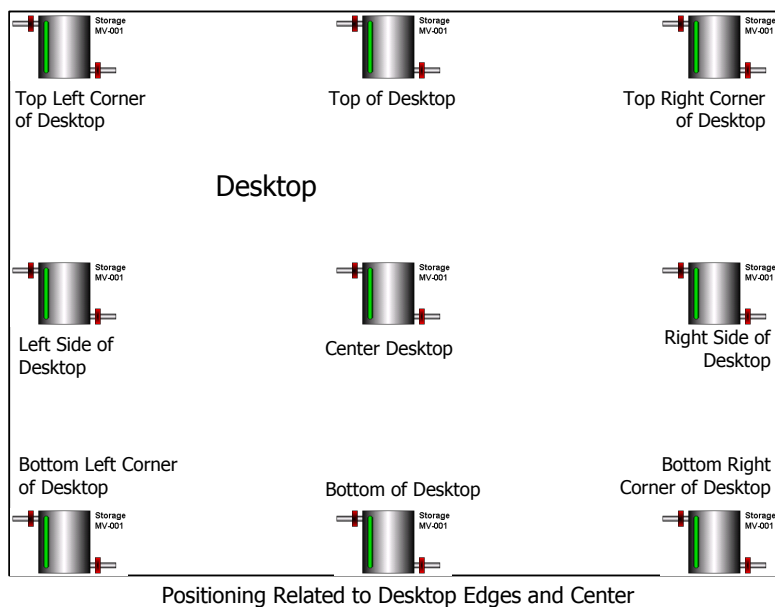
You can configure an element with a show symbol animation. A Show symbol animation shows a specified symbol in a new dialog box, when the element is clicked on.

You can configure:

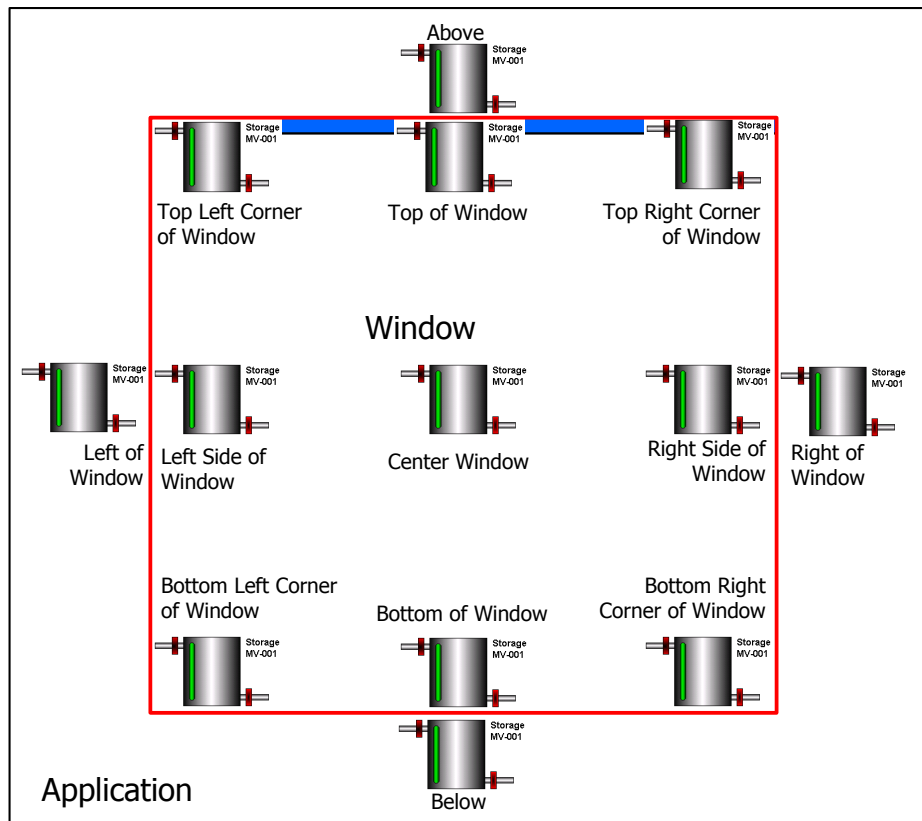
- Which symbol appears in the new window.
- If the window has a title bar, and if so if it has a caption.
- If the window is modal or modeless.
- If the window has a close button.
- If the window can be resized.
- The initial window position.
- The size of the window.

You can configure the position to be in relation of the:

- Desktop, such as at edges, corners, or at center.



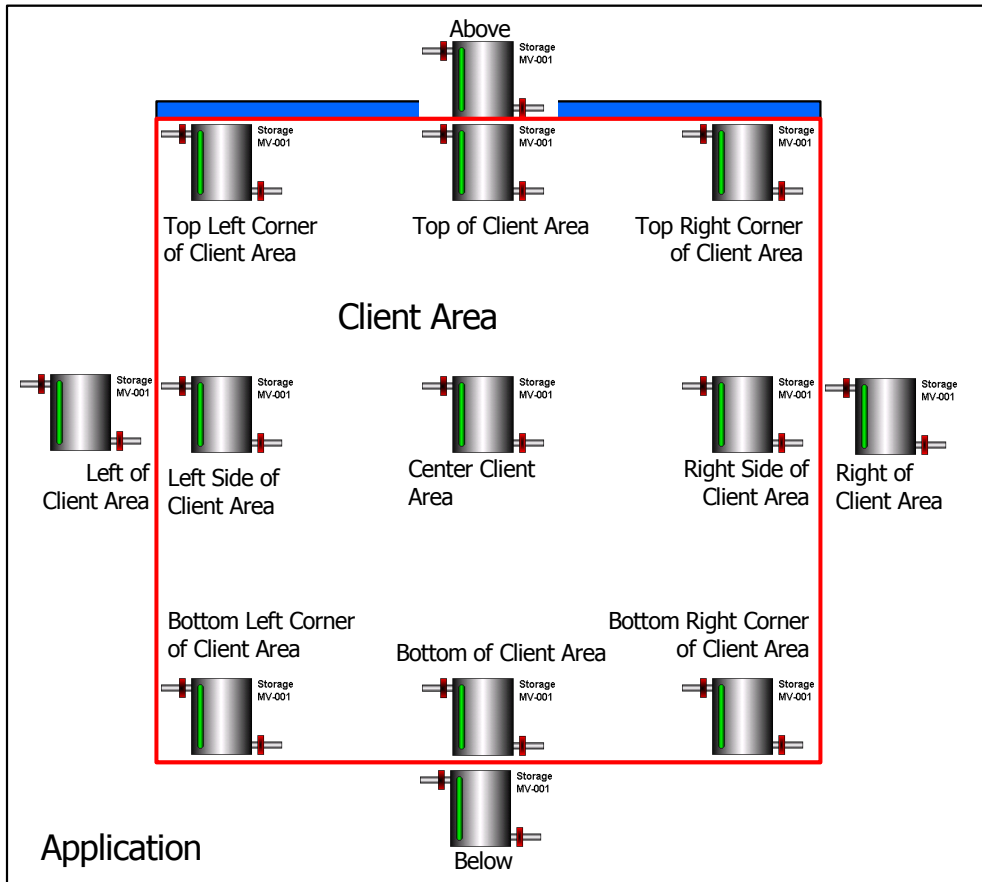
- Window, such as at one of its edges, its corners, its center or above, below, to the left or right. The window area includes the title bar if it appears.



Positioning Related to Window

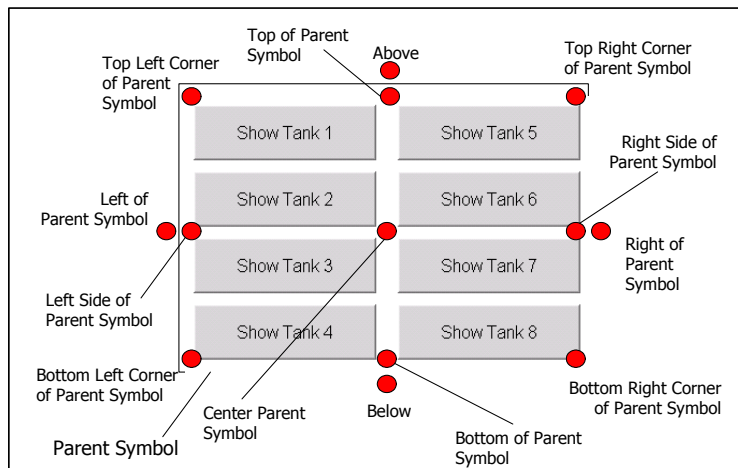


- Client Area. In InTouch client area is the active drawing area of an InTouch window excluding the title bar.



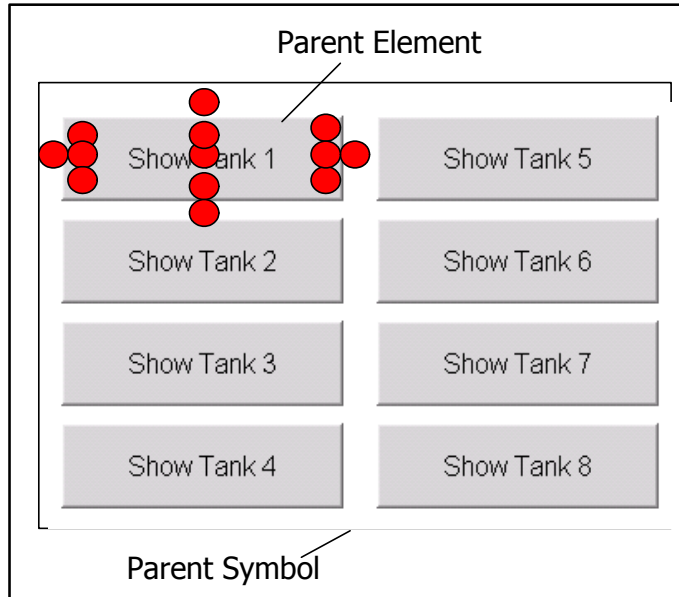
Positioning Related to Client Area

- Source Symbol, in which case the show symbol window is positioned in relation to the entire source symbol that contains the element that called the window.



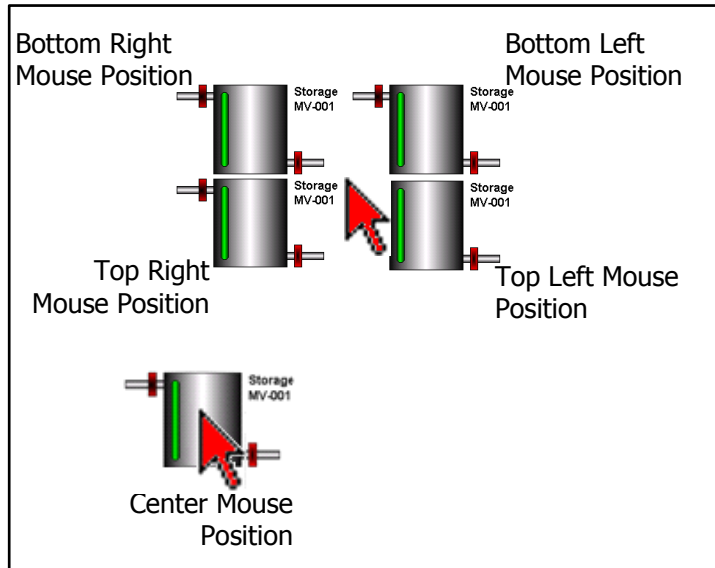
Positioning Related to Parent Symbol

- Parent Element, in which case the show symbol window is positioned in relation just to the element that called the show symbol window.



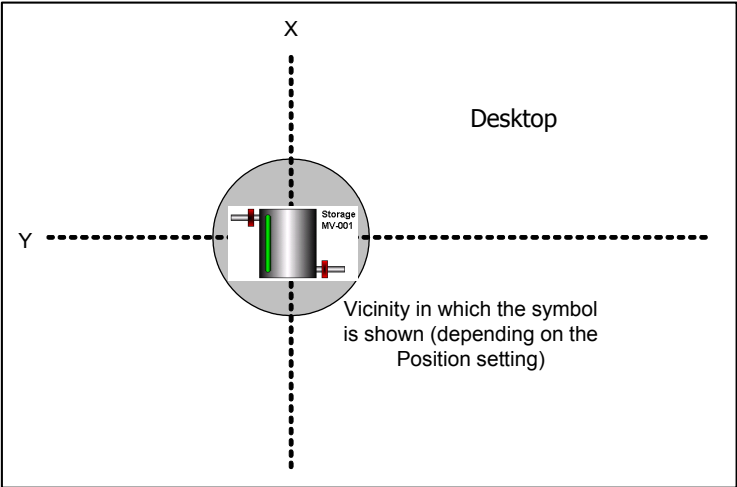
Positioning Related to Parent Element

- Mouse, in which case the show symbol window is positioned in relation to the pointer coordinates.



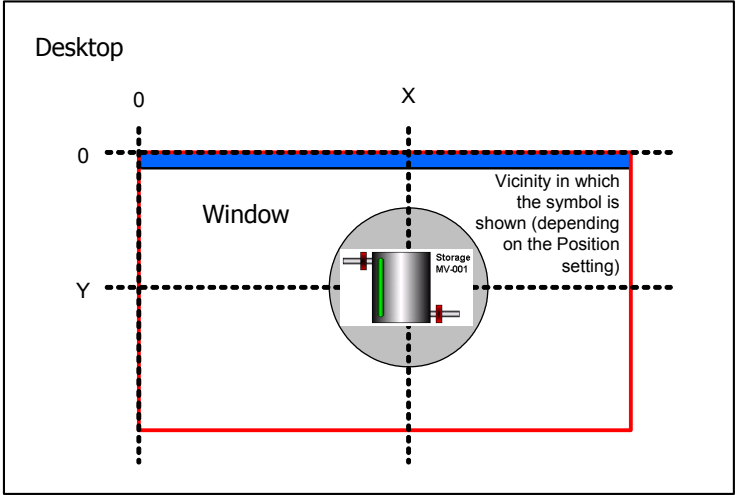
Selected Positionings Related to Mouse Pointer

- Desktop coordinates. The symbol is placed in the vicinity of coordinates that relate to the desktop.



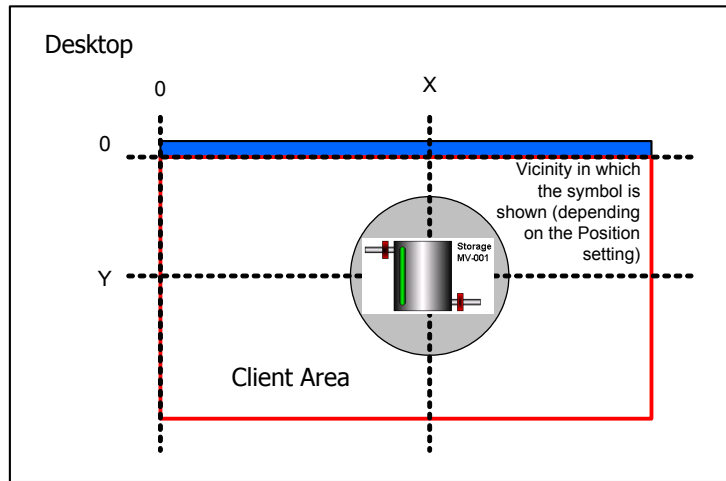
X, Y Positioning Related to Desktop

- Window coordinates. The symbol is placed in the vicinity of coordinates that relate to the window, including the title bar if shown.



X, Y Positioning Related to Window

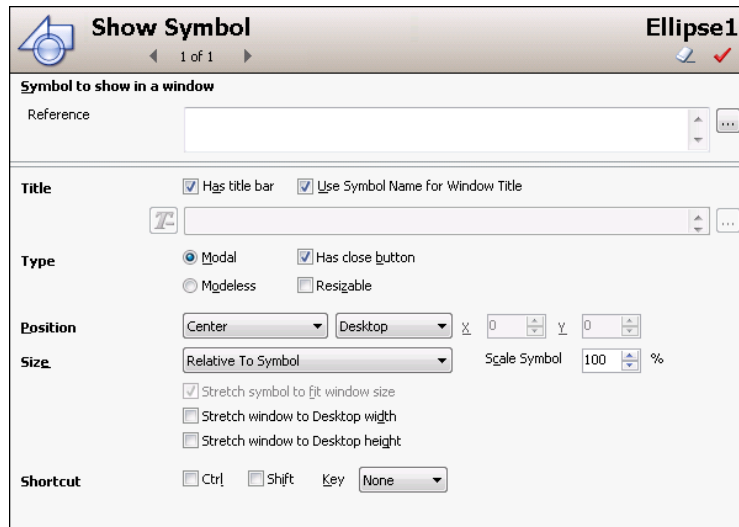
- Client Area coordinates. The symbol is placed in the vicinity of coordinates that relate to the client area.



X, Y Positioning Related to Client Area

### To configure an element with a show symbol animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Show Symbol**. The Show Symbol animation is added to the Animation list.



- 4 Configure the symbol. Do one or more of the following:
  - a In the **Reference** box, type a symbol name or browse for one by using the browse button.
  - b To add a title bar to the symbol, select **Has Title Bar**.
  - c To use the symbol name as window title, select **Use Symbol Name for Window Title**.

- d Select the window type, **Modal** or **Modeless**.
  - e To add a Close button, select **Has Close Button**.
  - f To add resize controls, select **Resizable**.
- 5 Select where you want the window to appear by selecting a position in the **Position** lists. The first list contains positions that are in relation to the item of the second list. Select one of the following:
- **Center**
  - **Top Left Corner**
  - **Top Right Corner**
  - **Left**
  - **Right of**
  - **Lower**
  - **Below**
  - **Above**
  - **Top**
  - **Left of**
  - **Right Side**
  - **Lower Left Corner**
  - **Lower Right Corner**

From the second list, select the item the position is referring to:

- **Desktop** relative to the entire desktop.
  - **Window** relative to the window.
  - **Client Area** relative to the client area.
  - **Parent Symbol** relative to the entire symbol that calls it.
  - **Parent Element** relative to the element or element group that calls it.
  - **Mouse** relative to the pointer.
  - **Desktop X,Y** relative to a specified coordinate on the desktop.
  - **Window X,Y** relative to a specified coordinate of the window.
  - **Client Area X,Y** relative to a specified coordinate of the client area.
- 6 If you select **Desktop X,Y** or **Window X,Y** or **Client Area X,Y** as position, type the new coordinates in the **X** and **Y** value boxes.
- 7 Select how large you want the window to be in the **Size** list. You can select:
- **Relative to Symbol** to make the window size the same as the size of the symbol.
  - **Custom Width and Height** to specify a width and height.

Depending on your selection of the item the symbol is referring to, you can select:

- **Relative to Desktop** to adjust the window size relative to the size of the desktop.
  - **Relative to Window** to adjust the window size relative to window that contains the symbol that calls it.
  - **Relative to Client Area** to adjust the window size relative to the client area.
  - **Relative to Parent Symbol** to adjust the window size relative to the size of the symbol that calls it.
  - **Relative to Parent Element** to adjust the window size relative to the size of the element that calls it.
- 8 Continue specifying position information.
    - a If you select **Relative...** as size, enter a scaling percentage in the **Scale Symbol** box.
    - b If you select **Custom Width and Height** as size, type the new width and height in the **W** and **H** boxes.
    - c If you select **Desktop, Window, Client Area, Parent Symbol** or **Parent Element** as referred item, you can configure the object to be stretched horizontally or vertically. Do one or both of the following:
      - Select **Stretch symbol to fit ... width** and enter a height in the **H** box.
      - Select **Stretch symbol to fit ... height** and enter a width in the **W** box.
  - 9 You can specify that the symbol window appears by pressing a key or key combination. In the **Shortcut** area:
    - a Select a shortcut key in the **Key** list.
    - b Select **CTRL** and/or **SHIFT** to combine the shortcut key with the **CTRL** key and/or **SHIFT** key.
  - 10 Click **OK**.

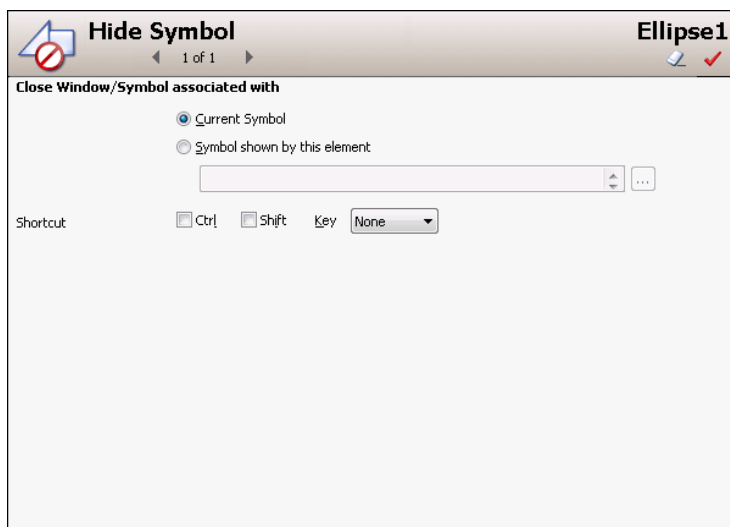
## Configuring a Hide Symbol Animation

You can configure an element with a hide symbol animation. The hide symbol animation lets you close:

- The current symbol
- A symbol that is shown by a specified element.

### To configure an element with a hide symbol animation

- 1 Select the element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Add** icon and select **Hide Symbol**. The hide symbol animation is added to the Animation list and the **Symbol Hide** configuration panel appears.



- 4 Select:
  - **Current Symbol** if you want to close the currently shown symbol.
  - **Symbol shown by an element** if you want to close a symbol that appears by that element. Type the element name in the adjacent box.
- 5 You can specify that the symbol window closes by pressing a key or key combination. In the **Shortcut** area:
  - a Select a shortcut key in the **Key** list.
  - b Select **CTRL** and/or **SHIFT** to combine the shortcut key with the CTRL key and/or SHIFT key.
- 6 Click **OK**.

## Configuring Element-Specific Animations

Some elements have their own unique animation type that can only be used for that element type. You cannot remove their unique animation, but depending on the element you can add and remove other common animations.

The elements with specific animations are:

- Status element
- Windows common controls

### Configuring Animation for a Status Element

You can configure the Status element with a DataStatus animation to indicate quality and status from:

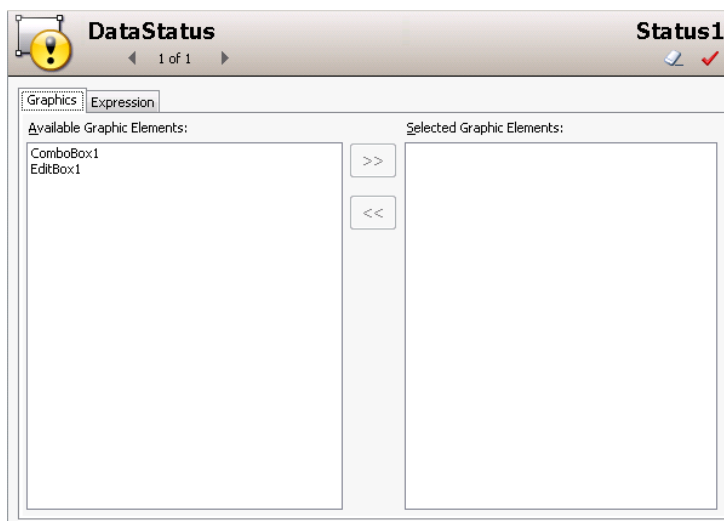
- ArcestrA attributes used in elements with animation.
- ArcestrA attributes directly.

The appearance of the Status element depends on the settings in the **Configure Quality and Status Display** dialog box. For more information, see *Configuring Animation for a Status Element* on page 328.

The DataStatus animation is only used by the Status element and cannot be removed from the Status element.

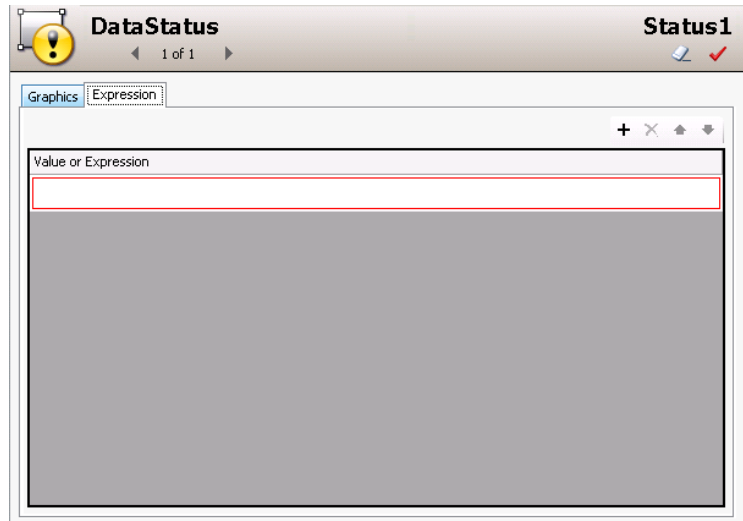
#### To configure a DataStatus animation

- 1 Select the Status element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears, showing the DataStatus configuration panel.





- 3 In the **Available Graphic Elements** list, select all elements for which you want to monitor their attribute quality and status.
- 4 Click the >> button to add them to the **Selected Graphic Elements** list.
- 5 Click the **Expression** tab. The **Expression** panel appears.



- 6 In the **Value or Expression** list, type a value or expression that can be a literal, or a reference or element property.

---

**Tip** You can also browse for the reference by clicking the browse button.

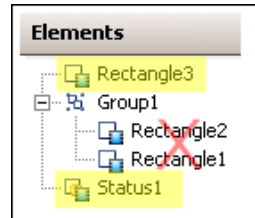
---

- 7 To add more values or expressions, click the **Add** button. An additional row is added for data input.
- 8 Click **OK**.

### Restrictions of the Status Element

The Status element must be in the same hierarchical level as the animated elements with the attributes you want to monitor.

If you move elements out of their hierarchical level after you associate them with a Status element, for example, by grouping them, their attributes are no longer monitored.



To avoid this problem, move a new Status element in the hierarchical level you want to monitor, or associate it directly with the attributes you want to monitor.

## Configuring a Radio Button Group Animation

The Radio Button Group animation is only used by the Radio Button Group element.

You can create a:

- **Static** radio button group - uses static captions and values that you define in the configuration panel.
- **Array** radio button group - uses captions and values contained in an AutomationObject array.
- **Enum** radio button group - uses captions and values contained in an enum data type of an AutomationObject.

## Configuring a Static Radio Button Group Animation

You can configure a radio button group with static values and captions.

To configure a static radio button group animation

- 1 Select the radio button group element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears and the **Static Radio Button Group** configuration panel appears on the right side.

- 3 In the **Reference** box, type an attribute reference that is to be tied to the selected value at run time.  
You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.

- 4 In the **Static Values and Captions** list, configure the captions of the radio button group and also the values that correspond to them to:
  - **Add an option** - click the **Add** icon.
  - **Delete an option** - select it in the list and click the **Remove** icon.
  - **Move an option up** the list - select it in the list and click the **Arrow up** icon.
  - **Move an option down** the list - select it in the list and click the **Arrow down** icon.



- 5 To use the values themselves as captions, select **Use Values as Captions**.
- 6 Orientate the radio button group in vertical or horizontal direction. Select **Vertical** or **Horizontal**.
- 7 Click **OK**.

## Configuring an Array Radio Button Group Animation

You can configure a radio button group with values from an array and captions.

To configure an array radio button group animation

- 1 Select the radio button group element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Array** button. The **Array Radio Button Group** configuration panel appears on the right side.

- 4 In the **Reference** box, type an attribute reference that is to be tied to the selected value at run time.  
You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.
- 5 In the **Array Reference** box, type or browse for an array attribute. The **Array Values and Captions** list shows the values from the array reference.
- 6 To define your own captions, clear **Use Values as Captions** and type them in the list.
- 7 To format the value before it appears as a caption, type a text format string in the **Format** box, for example `#.###`. Preceding zeroes are ignored if the array data type is numeric.

- 8 Set **Items Sorting** to:
  - **None** to show the items in the order they are in the array attribute.
  - **Ascending** to show the items sorted in ascending order.
  - **Descending** to show the items sorted in descending order.
- 9 Orientate the radio button group in vertical or horizontal direction. Select **Vertical** or **Horizontal**.
- 10 Click **OK**.

For example, you want to create a Radio Button Group in your symbol with the following options. The values to be written to the target attribute are contained in the user-defined attribute array called **Options** of an AutomationObject called **UD**.

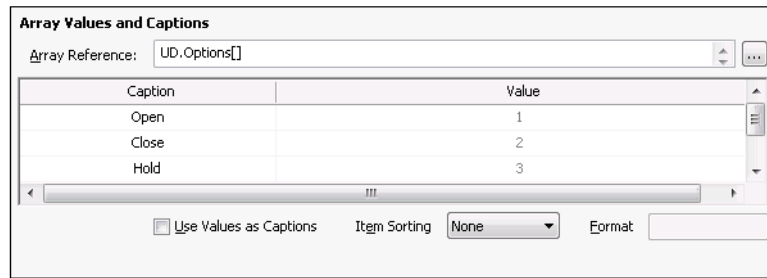
Option	Value to be written
Open	1
Close	2
Hold	3
Report Error	4
Unknown	99

The user-defined attribute array **Options** of the **UD** object can appear as follows:

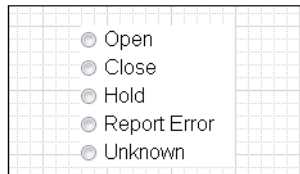
The screenshot shows the configuration window for a User-Defined Attribute (UDA) named 'Options'. The 'UDA name' is 'Options'. The 'Data type' is set to 'Integer' and the 'Category' is 'User writeable'. Under the 'Value' section, the checkbox 'This is an array' is checked, and the 'Number of elements' is set to 5. A table below shows the array elements:

Index	Value
1	1
2	2
3	3
4	4
5	99

You can configure the array reference of the Radio Button Group as follows:



The Radio Button Group element appears as follows:

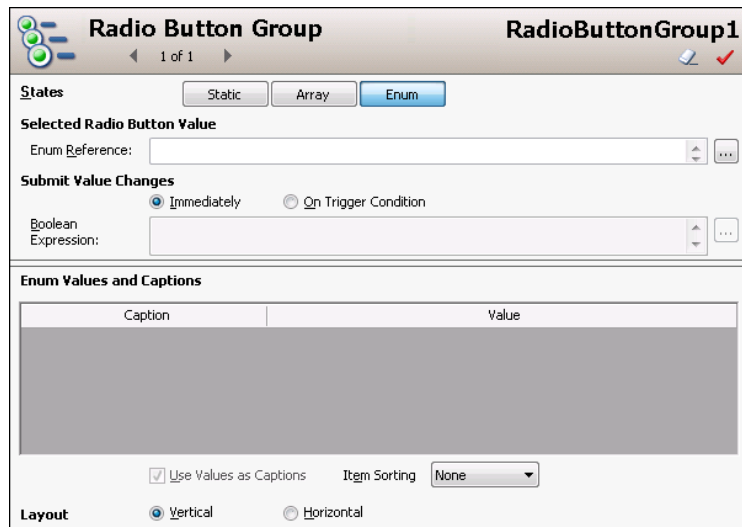


### Configuring an Enum Radio Button Group Animation

You can configure a radio button group with values from an enum attribute and captions.

To configure an enum radio button group animation

- 1 Select the radio button group element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Enum** button. The **Enum Radio Button Group** configuration panel appears on the right side.



- 4 In the **Enum Reference** box, type an enum attribute reference. The **Enum Values and Captions** list shows the values from the enum reference.  
You can select when to submit the value changes. For more information, see Submitting the Value Changes on page 349.
- 5 To define your own captions, clear **Use Values as Captions** and type them in the list.
- 6 Set **Items Sorting** to:
  - **None** to show the items in the order they are in the enum attribute.
  - **Ascending** to show the items sorted in ascending order.
  - **Descending** to show the items sorted in descending order.
- 7 Orientate the radio button group in vertical or horizontal direction. Select **Vertical** or **Horizontal**.
- 8 Click **OK**.

## Configuring a Check Box Animation

The Check Box animation is only used by the Check Box element.

### To configure a Check Box animation

- 1 Select the Check Box element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears and the **Check Box** configuration panel appears on the right side.

The screenshot shows the configuration panel for a Check Box element. The panel is titled "Check Box" and "CheckBox1". It contains three main sections:

- Checked value:** A "Boolean" field with a dropdown arrow and a "..." button.
- Submit Value Changes:** Two radio buttons: "Immediately" (selected) and "On Trigger Condition". Below them is a "Boolean Expression" field with a dropdown arrow and a "..." button.
- Caption:** A "String" field containing "CheckBox1". Below it is a checkbox labeled "Override caption at Runtime with the following expression". If checked, there is a "String Expression" field with a dropdown arrow and a "..." button.

- 3 In the **Checked value - Boolean** box, type an attribute reference. The attribute reference is tied to the selected state of the check box control at run time.  
You can select when to submit the value changes. For more information, see *Submitting the Value Changes* on page 349.
- 4 To set the caption of the check box at run-time, select **Override caption at Runtime with the following expression** and type a string value or attribute reference or expression in the **String Expression** box.
- 5 Click **OK**.

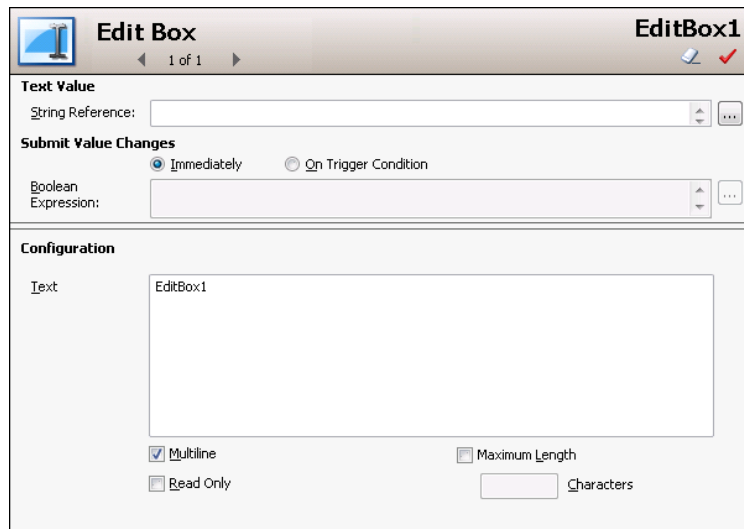
## Configuring an Edit Box Animation

The Edit Box animation is only used by the Edit Box element. You cannot remove this animation from the Edit Box element, but you can add certain common animations.

You can also use Edit Box-specific methods in scripting to get and set the text at run time. You can browse these methods in the Galaxy Browser with the Edit Box selected. For more information on these methods, see *Configuring Edit Box Methods* on page 364.

### To configure an Edit Box animation

- 1 Select the Edit Box element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears and the **Edit Box** configuration panel appears on the right side.





- 3 In the **String Reference** box, type an string attribute reference. The string attribute reference is tied to the text in the edit box at run time.

You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.

---

**Tip** Use the **On Trigger Condition** option to set when the Edit Box element writes the run-time value to the reference. This avoids a conflict between the run-time value of the Edit Box and run-time value of the reference.

---

- 4 In the **Configuration** area, select:
  - **Multiline** to wrap the text into multiple lines in the edit box.
  - **Read-Only** to use the edit box to only show text and not allow text input.
  - **Maximum Length** to limit the maximum numbers of characters you can type in the edit box control. You can specify the maximum number in the **Characters** box.

Enter a default text in the **Text** box.

## Configuring a Combo Box Animation

The Combo Box animation is only used by the Combo Box element.

You can create a:

- **Static** combo box - uses static captions and values that you define in the configuration panel.
- **Array** combo box - uses captions and values contained in an AutomationObject array.
- **Enum** combo box - uses captions and values contained in an enum data type of an AutomationObject.

You can also use Combo Box-specific methods in scripting to perform various functions at run time. You can browse these methods in the Galaxy Browser with the Combo Box selected.

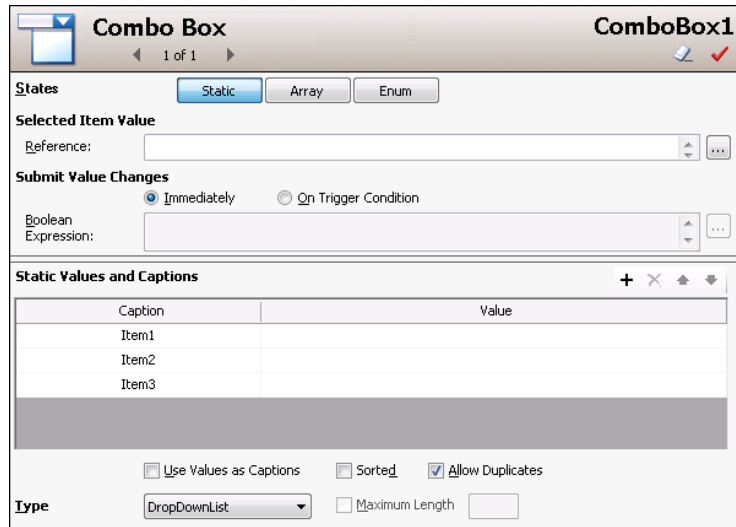
For more information on these methods, see [Configuring Combo Box and List Box Methods](#) on page 365.

## Configuring a Static Combo Box Animation

You can configure a combo box with static values and captions.

To configure a static combo box animation

- 1 Select the combo box element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears and the **Static Combo Box** configuration panel appears on the right side.



- 3 In the **Reference** box, type an attribute reference that is to be tied to the selected value at run time. You can select when to submit the value changes. For more information, see Submitting the Value Changes on page 349.
- 4 In the **Static Values and Captions** list, configure the captions of the combo box and also the values that correspond to them:



- **Add an option** - click the **Add** icon.
- **Delete an option** - select it in the list and click the **Remove** icon.
- **Move an option up** the list - select it in the list and click the **Arrow up** icon.
- **Move an option down** the list - select it in the list and click the **Arrow down** icon.

- 5 Specify how you want to use captions. Do one of more of the following:
  - To use the values themselves as captions, select **Use Values as Captions**.
  - To alphabetically sort the captions, select **Sorted**.
  - To allow duplicate captions, select **Allow Duplicates**.

---

**Note** If you clear **Allow Duplicates** and click **OK**, all duplicate captions are removed from combo box on the canvas. The captions are case-insensitive, so that for example “item1” is considered a duplicate of “Item1”. The removal of the duplicate items is reflected when you re-open the **Edit Animations** dialog box.

---

- 6 Select the type of combo box from the **Type** list. Select:
  - **Simple** - at run time you can type a value, or select one by using arrow up and arrow down buttons. However, you cannot see the list of values.
  - **DropDown** - at run time you can type a value, or select one from the list.
  - **DropDownList** - at run time you can only select a value from the list, but not type one.
- 7 Click **OK**.

### Configuring an Array Combo Box Animation

You can configure a combo box with values from an array and captions.

#### To configure an array combo box animation

- 1 Select the combo box element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.

- Click the **Array** button. The **Array Combo Box** configuration panel appears on the right side.

- In the **Reference** box, type an attribute reference that is to be tied to the selected value at run time. The **Array Values and Captions** list shows the values from the array reference.  
You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.
- To define your own captions, clear **Use Values as Captions** and type them in the list.
- If you want to format the value before it appears as a caption, type a text format string in the **Format** box, for example `#.###`. Preceding zeroes are ignored if the array data type is numeric.
- Set **Items Sorting** to:
  - None** to show the items in the order they are in the array attribute.
  - Ascending** to show the items sorted in ascending order.
  - Descending** to show the items sorted in descending order.
- Click **OK**.

## Configuring an Enum Combo Box Animation

You can configure a combo box with values from an enum attribute and captions.

To configure an enum combo box animation

- 1 Select the combo box element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Enum** button. The **Enum Combo Box** configuration panel appears on the right side.

The screenshot shows the 'Enum Combo Box' configuration panel for 'ComboBox1'. It has a title bar with a close button, a tab icon, and a refresh button. Below the title bar are three buttons: 'Static', 'Array', and 'Enum' (which is selected). The panel is divided into several sections:

- States:** Contains buttons for 'Static', 'Array', and 'Enum'.
- Selected Item Value:** Includes an 'Enum Reference' text box with up/down arrows and a menu icon.
- Submit Value Changes:** Features two radio buttons: 'Immediately' (selected) and 'On Trigger Condition'. Below is a 'Boolean Expression' text box with up/down arrows and a menu icon.
- Enum Values and Captions:** A table with two columns: 'Caption' and 'Value'. The table area is currently empty.
- Options:** A checked checkbox for 'Use Values as Captions', an 'Item Sorting' dropdown menu set to 'None', and an unchecked checkbox for 'Maximum Length'.
- Type:** A dropdown menu set to 'DropDownList' and an unchecked checkbox for 'Maximum Length'.

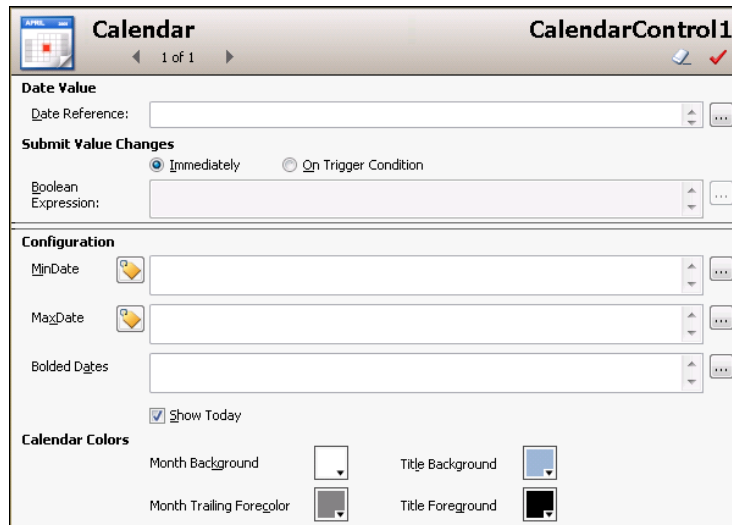
- 4 In the **Enum Reference** box, type an enum attribute reference. The **Enum Values and Captions** list shows the values from the enum reference. You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.
- 5 To define your own captions, clear **Use Values as Captions** and type them in the list.
- 6 Set **Items Sorting** to:
  - **None** to show the items in the order they are in the enum attribute.
  - **Ascending** to show the items sorted in ascending order.
  - **Descending** to show the items sorted in descending order.
- 7 Click **OK**.

## Configuring a Calendar Control Animation

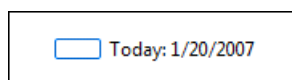
The Calendar Control animation is only used by the Calendar Control element. The Calendar Control date format depends on the regional settings of the operating system.

### To configure a Calendar control animation

- 1 Select the Calendar control element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears and the **Calendar** configuration panel appears on the right side.



- 3 In the **Date Reference** box, type a Time attribute reference that is to be tied to the selected value at run time.  
You can select when to submit the value changes. For more information, see Submitting the Value Changes on page 349.
- 4 To restrict the date the user can select at run time, specify limits as follows:
  - In the **MinDate** box, type a lower limit for the date.
  - In the **MaxDate** box, type an upper limit for date.
- 5 To show some dates as bold, in the **Bolded Dates** box, type a reference that points to an attribute array with time data type.
- 6 To show today's date on the calendar control, select **Show Today**.



- 7 To change the colors of the calendar control, click in the **Calendar Colors** area the following color boxes:

- Month Background
- Month Trailing Forecolor
- Title Background
- Title Foreground

The Style Selection dialog box appears and you can select a solid color.

- 8 Click **OK**.

## Configuring a DateTime Picker Animation

The DateTime Picker animation is only used by the DateTime Picker element.

### To configure a DateTime Picker animation

- 1 Select the DateTime Picker control element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears and the **DateTime Picker** configuration panel appears.

The screenshot shows the configuration panel for a DateTime Picker. It includes the following sections:

- DateTime Value:** A text box for 'Time Reference'.
- Submit Value Changes:** Radio buttons for 'Immediately' (selected) and 'On Trigger Condition'. A text box for 'Boolean Expression'.
- Configuration:** A dropdown for 'Format' (set to 'Long') showing the format 'dddd, MMMM dd, yyyy'. Text boxes for 'MinTime' and 'MaxTime'.
- Calendar DropDown Colors:** Four color selection boxes for 'Month Background', 'Month Trailing Forecolor', 'Title Background', and 'Title Foreground'.

- 3 In the **Time Reference** box, type a Time attribute reference that is to be tied to the selected value at run time.

You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.

- 4 To set the datetime format, select one of the following from the **Format** list:
- **Long** to show the date and time in the long format of the operating system, for example: Thursday, August 03 2006.
  - **Short** to show the date and time in the short format of the operating system, for example: 8/3/2006.
  - **Time** to show just the time in the time format of the operating system, for example: 3:46:09 PM.
  - **Custom** to specify your own time format. Use the following letters to set the time format:
 

h	One or two-digit hour in 12-hour format.
hh	Two-digit hour in 12-hour format. Single digit values are preceded by a zero.
H	One or two-digit hour in 24-hour format.
HH	Two-digit hour in 24-hour format. Single digit values are preceded by a zero.
t	One-letter AM/PM abbreviation ("AM" appears as "A").
tt	Two-letter AM/PM abbreviation ("AM" appears as "AM").
m	One or two-digit minute.
mm	Two-digit minute. Single digit values are preceded by a zero.
s	One or two-digit seconds.
ss	Two-digit seconds. Single digit values are preceded by a zero.
d	One or two-digit day.
dd	Two-digit day. Single digit day values are preceded by a zero.
ddd	Three-character day-of-week abbreviation.
dddd	Full day-of-week name.
M	One or two-digit month number.
MM	Two-digit month number. Single digit values are preceded by a zero.
MMM	Three-character month abbreviation.
MMMM	Full month name.
y	One-digit year (2001 appears as "1").
yy	Last two digits of the year (2001 appears as "01").
yyyy	Full year (2001 appears as "2001").



---

**Note** You can use any other characters, except “g” in the property. These characters then appear at design time and run time in the control.

---

- 5 To restrict the date the user can select at run time, you can specify limits in the:
  - **MinDate** box - type a lower limit for the date.
  - **MaxDate** box - type an upper limit for date.
- 6 To change the colors of the calendar control that drops down, click in the **Calendar Colors** area the following color boxes:
  - **Month Background**
  - **Month Trailing Forecolor**
  - **Title Background**
  - **Title Foreground**

The Style Selection dialog box appears and you can select a solid color.

## Configuring a List Box Animation

The List Box animation is only used by the List Box element.

You can create a:

- **Static** list box - uses static captions and values that you define in the configuration panel.
- **Array** list box - uses captions and values contained in an AutomationObject array.
- **Enum** list box - uses captions and values contained in an enum data type of an AutomationObject.

You can also use List Box-specific methods in scripting to perform various functions at run time. You can browse these methods in the Galaxy Browser with the List Box selected.

For more information on these methods, see [Configuring Combo Box and List Box Methods](#) on page 365.

## Configuring a Static List Box Animation

You can configure a list box with static values and captions.

### To configure a static list box animation

- 1 Select the list box element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears and the **Static List Box** configuration panel appears on the right side.

Static Values and Captions	Value
Item1	
Item2	
Item3	

- 3 In the **Reference** box, type an attribute reference that is to be tied to the selected value at run time.  
You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.
- 4 In the **Static Values and Captions** list, configure the captions of the list box and also the values that correspond to them. To:



- **Add an option** - click the **Add** icon.
- **Delete an option** - select it in the list and click the **Remove** icon.
- **Move an option up** the list - select it in the list and click the **Arrow up** icon.
- **Move an option down** the list - select it in the list and click the **Arrow down** icon.

- 5 Specify how you want to use captions. Do one of more of the following:
  - If you want to use the values themselves as captions, select **Use Values as Captions**.
  - If you want to alphabetically sort the captions, select **Sorted**.
  - If you want to allow duplicate captions, select **Allow Duplicates**.
- 6 Click **OK**.

### Configuring an Array List Box Animation

You can configure a list box with values from an array and captions.

To configure an array list box animation

- 1 Select the list box element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Array** button. The **Array List Box** configuration panel appears on the right side.

- 4 In the **Reference** box, type an attribute reference that is to be tied to the selected value at run time.  
You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.
- 5 In the **Array Reference** box, type or browse for an array attribute. The **Array Values and Captions** list shows the values from the array reference.
- 6 To define your own captions, clear **Use Values as Captions** and type them in the list.

- 7 To format the value before it appears as a caption, type a text format string in the **Format** box, for example **#.###**. Preceding zeroes are ignored if the array data type is numeric.
- 8 Set **Items Sorting** to:
  - **None** to show the items in the order they are in the array attribute.
  - **Ascending** to show the items sorted in ascending order.
  - **Descending** to show the items sorted in descending order.
- 9 Click **OK**.

### Configuring an Enum List Box Animation

You can configure a list box with values from an enum attribute and captions.

#### To configure an enum list box animation

- 1 Select the radio button group element.
- 2 On the **Special** menu, click **Edit Animations**. The **Edit Animations** dialog box appears.
- 3 Click the **Enum** button. The **Enum List Box** configuration panel appears on the right side.

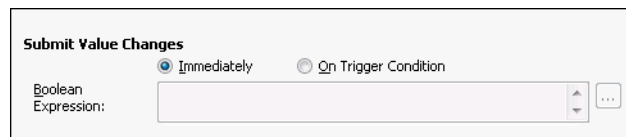
- 4 In the **Enum Reference** box, type an enum attribute reference. The **Enum Values and Captions** list shows the values from the enum reference.
- 5 You can select when to submit the value changes. For more information, see [Submitting the Value Changes](#) on page 349.

- 6 To define your own captions, clear **Use Values as Captions** and type them in the list.
- 7 Set **Items Sorting** to:
  - **None** to show the items in the order they are in the enum attribute.
  - **Ascending** to show the items sorted in ascending order.
  - **Descending** to show the items sorted in descending order.
- 8 Click **OK**.

## Submitting the Value Changes

You can configure a Windows common control to write the data:

- Immediately when it is selected in the control at run time.
- When a specified Boolean expression becomes true.




---

**Note** The Boolean expression is a trigger that determines when the value is written from the control to the tag or attribute. If the value changes in the tag or attribute, then the value is written to the control, regardless of the trigger setting or condition.

---

### To submit value changes immediately

- 1 Open the Windows common control in the **Edit Animations** dialog box.
- 2 In the **Submit Value Changes** area, select **Immediately**.

### To submit value changes after a Boolean expression becomes true

- 1 Open the Windows common control in the **Edit Animations** dialog box.
- 2 In the **Submit Value Changes** area, select **On Trigger Condition**.
- 3 In the **Boolean Expression** box, type a Boolean expression or browse for a Boolean attribute.

## Cutting, Copying and Pasting Animations

You can cut, copy and paste animations and their configuration between different elements. This is useful when you want to duplicate the animations of one element such as a line, to a different type of element such as a polyline.

If you try to paste an animation to an element that is already configured with that animation, or does not support this animation, a message appears informing you why you cannot paste the animation.

### To copy and paste animations between elements

- 1 Select the element from which you want to copy the animations.
- 2 On the **Edit** menu, point to **Animations**, and then click **Copy**.
- 3 Select one or more elements to which you want to paste the animations.
- 4 On the **Edit** menu, point to **Animations**, and then click **Paste**. The animation links are copied from the source element to the target elements.

### To cut and paste animations between elements

- 1 Select the element from which you want to cut the animations.
- 2 On the **Edit** menu, point to **Animations**, and then click **Cut**.
- 3 Select one or more elements to which you want to paste the animations.
- 4 On the **Edit** menu, point to **Animations**, and then click **Paste**. The animation links are removed from the source element and copied to the target elements.

## Substituting References in Elements

You can search and replace references used by any element on your canvas. You can use:

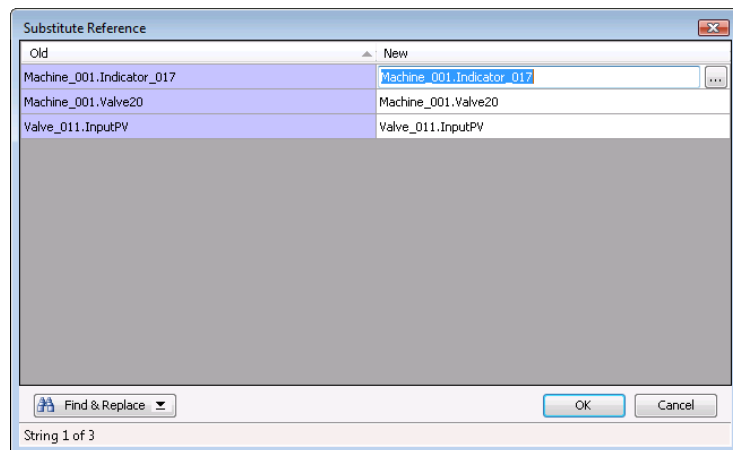
- basic mode by replacing strings in a list.
- advanced functions such as find and replace, ignore or respect case-sensitivity and wildcards.

### To substitute references in a symbol by using the list

- 1 Select one or more elements.
- 2 Do one of the following:
  - Press CTRL + E.
  - On the **Special** menu, click **Substitute References**.



The **Substitute References** dialog box appears.



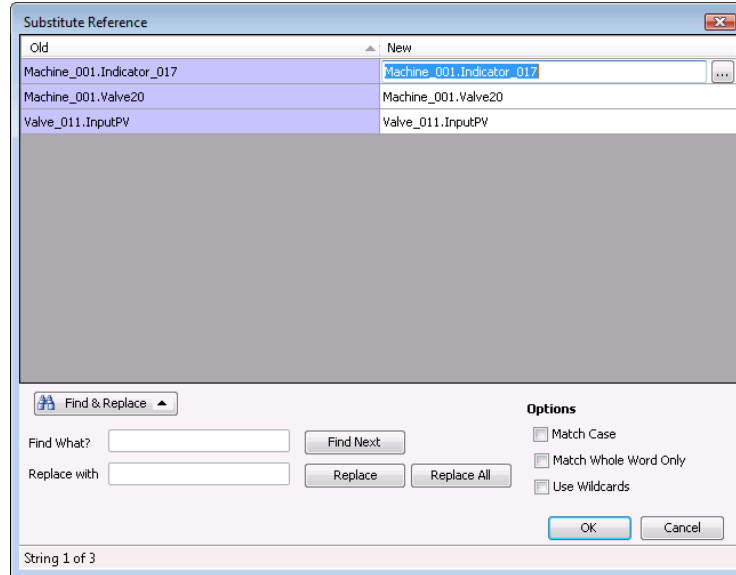
- 3 In the **New** column, type the reference to be replaced.
- 4 Click **OK**. All references are substituted accordingly in the elements.

### To substitute references in a symbol by using find and replace functions

- 1 Select one or more elements.
- 2 Do one of the following:
  - Press CTRL + E.
  - On the **Special** menu, click **Substitute References**.

The **Substitute References** dialog box appears.

- 3 Click **Find & Replace**. The dialog box expands and shows find and replace parameters.



- 4 Specify your find and replace options. Do one of more of the following:
  - To find specific references in the list, type a string in the **Find What** box and click **Find Next** to find the next string.
  - To replace a selected found string with another string, type a string in the **Replace with** box and click **Replace**.
  - To replace multiple references, type values in the **Find What** and **Replace with** boxes and click **Replace all**.
  - To specify the search is case-sensitive, select **Match Case**.
  - To find only entire words that match your search string, select **Match Whole Word Only**.
  - To use wildcards, select **Use Wildcards**. Valid wildcards are “\*” (asterisk) and “?” (question mark). “\*” indicates any number of variable characters. For example, “s\*” to search for all strings starting with “s”. “?” indicates one single variable character. For example, “M\_7?t” to search for all strings that start with “M\_7” and end with “t” and have exactly 5 characters.
- 5 Click **OK**. All text strings are substituted accordingly in the elements.



---

# Chapter 10

## Adding and Maintaining Symbol Scripts

You can associate scripts to your symbols. Scripts can add animation to a symbol or its elements.

---

**Caution** If you configure scripts that affect more than element and symbol animation, the script processing may affect performance.

---

### About Symbol Scripts

You can configure your symbol with synchronous scripts that can be executed in run time.

You can:

- Configure the predefined scripts of a symbol.
- Add named scripts to a symbol.
- Edit existing named or predefined scripts in a symbol.
- Rename named scripts in a symbol.
- Remove named scripts from a symbol.
- Substitute references in named or predefined scripts.
- Use element methods in named or predefined scripts.

For more information about Application Server scripting, see the *Application Server User's Guide*.

## Predefined and Named Scripts

Predefined symbol scripts are similar to InTouch window scripts in that they are executed:

- One time when the symbol is shown or opened: **On Show**
- Periodically while the symbol is showing: **While Showing**
- One time when the symbol is hidden or closed: **On Hide**
- Any combination of the above

Named symbol scripts let you execute any number of scripts in run time that are triggered by values or expressions:

- Being true: **While True**
- Being false: **While False**
- Transitioning from false to true: **On True**
- Transitioning from true to false: **On False**
- Change in value and/or quality: **DataChange**

The name of named scripts can be up to 32 characters in length, contain at least one letter, and contain special characters, such as #, \$, and \_.

## Execution Order of Symbol Scripts

When the symbol is showing, the scripts execute in the following order:

- 1 On Show script.
- 2 Named scripts, not necessarily in the order that they appear in the list.

Any named script that is triggered by the DataChange trigger type is executed the first time when the reference is subscribed to. This behavior is different than the DataChange trigger behavior of Application Server scripts and can take considerable time in intermittent networks.

## Security in Symbol Scripts

If the symbol script attempts to write to attributes with Secured Write or Verified Write security classification, the script execution stops and the authentication dialog box appears.

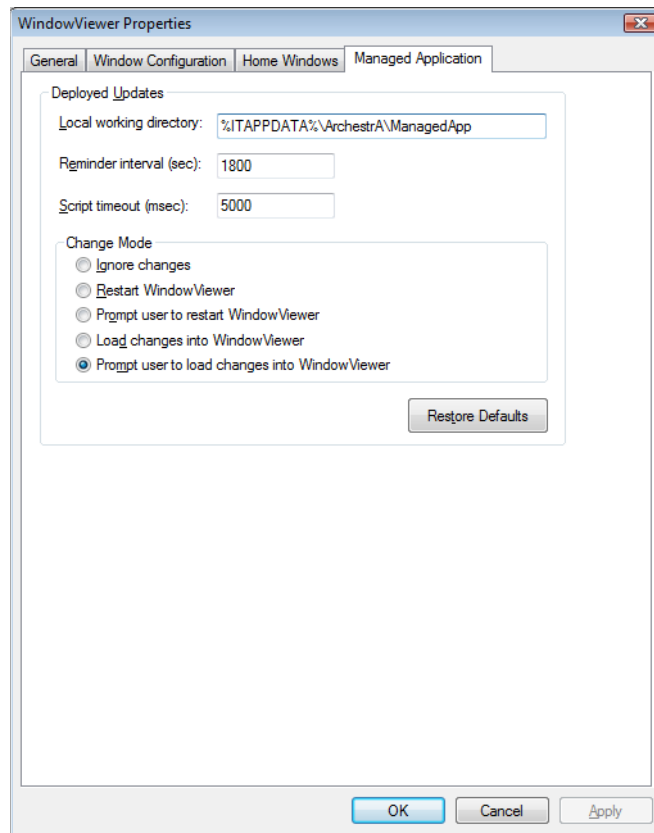
After you enter correct authentication information, the symbol script continues execution.

## Symbol Script Timeouts

You can set a timeout for a symbol script after which the script stops running and a message is written to the Logger. By default, the timeout is five seconds.

### To change the timeout for a symbol script

- 1 In WindowMaker, on the **Special** menu, point to **Configure** and click **WindowViewer**. The **WindowViewer Properties** dialog box appears.
- 2 Click the **Managed Application** tab.



- 3 In the **Script timeout (msec)** box, type a timeout value in milliseconds. Valid values are from 1 to 360,000.
- 4 Click **OK**.

## Error Handling

A symbol script does not run if it contains a syntax error. When the symbol or graphic is loaded, a message is written to the Logger.

## Configuring the Predefined Scripts of a Symbol

You can configure the predefined scripts of a symbol. The predefined scripts can consist of:

- A script that runs one time when the symbol opens (On Show).
- A script that runs periodically as long as the symbol is open (While Showing).
- A script that runs one time when the symbol closes (On Hide).

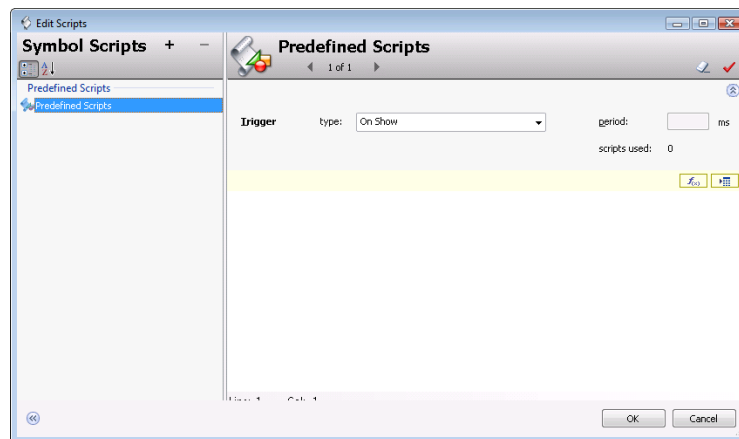
---

**Note** The **Predefined Scripts** animation cannot be deleted. It can contain scripts for each trigger type **On Show**, **While Showing** and **On Hide**.

---

**To configure the predefined scripts for a symbol**

- 1 Open the symbol in the ArcestrA Symbol Editor.
- 2 On the **Special** menu, click **Scripts**. The **Edit Scripts** dialog box appears.



- 3 In the **Trigger Type** list, click:
  - **On Show** to configure a script that runs one time when the symbol opens.

If you create an OnShow script that uses a custom property bound to an InTouch tag, there is no guarantee that the tag data is valid when the script executes. This is because of the asynchronous nature of data subscriptions within ArcestrA. Your script should first test the quality and status of the tag value before it is used in the rest of the script.
  - **While Showing** to configure a script that runs periodically while the symbol is open.
  - **On Hide** to configure a script that runs one time when the symbol closes.
- 4 If you configured a **While Showing** script, type a time period in milliseconds in the **Period** box. This specifies after how many milliseconds the action script is executed.

---

**Note** If you set the **While Showing** period too low, system performance may decrease.

---

- 5 Type your script in the main edit box. The script syntax is the same as the syntax of AutomationObject scripting.
- 6 Use the Script Function Browser and Attribute Browser to select external data.
- 7 When you are done, click **OK**. The script editor checks the syntax of the script and may inform you of invalid syntax. Click:
  - **Yes** to save changes even if the script contains errors.
  - **No** to cancel the changes and close the script dialog box.

## Adding Named Scripts to a Symbol

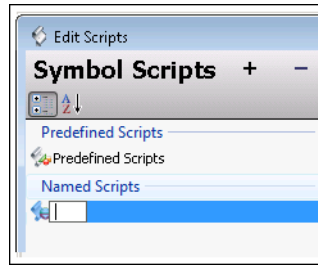
You can add named scripts to a symbol. A named script runs:

- One time when the specified values, data or expressions change.
- Periodically if the values or expressions meet a certain criterion, such as being true.

Every named script can contain only one trigger type.

### To add a named script to a symbol

- 1 Open the symbol in the ArchestrA Symbol Editor.
- 2 On the **Special** menu, click **Scripts**. The **Edit Scripts** dialog box appears.
- 3 Click the Add icon. A new entry is created in the **Named Scripts** list.



- 4 Type a name for the named script. The name appears on the right panel as header.

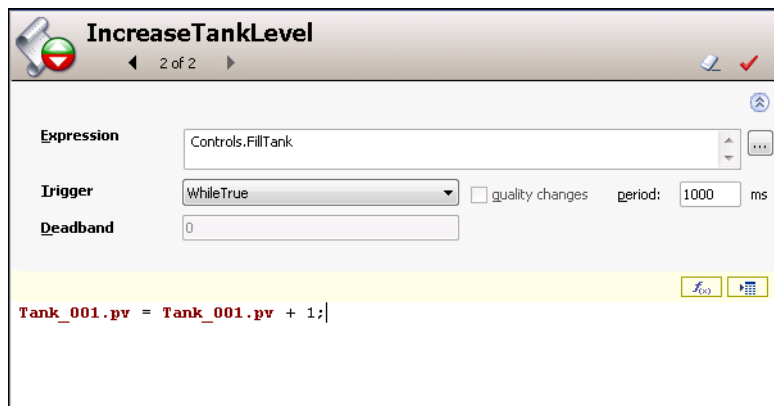


- 5 In the **Expression** box, do one of the following:
  - Type an expression, value or reference.
  - Browse for a reference.

The expression acts as data source for the script trigger.

- 6 In the **Trigger** list, click:
  - **WhileTrue** to trigger the script periodically when the expression is true.
  - **WhileFalse** to trigger the script periodically when the expression is false.
  - **OnTrue** to trigger the script one time when the expression becomes true from false.
  - **OnFalse** to trigger the script one time when the expression becomes false from true.
  - **DataChange** to trigger the script one time when the expression or its quality changes. Select the **Quality Changes** check box to trigger the script when the quality of the specified expression changes.
- 7 If you want to specify how often the script is run when the trigger condition is fulfilled, type a time delay in milliseconds in the **Trigger Period** box.
- 8 If you want to specify by how much the evaluated value must change before the script runs, type a deadband value in the **Deadband** box.
- 9 Type your script in the main edit box.
- 10 Use the Script Function Browser and Attribute Browser to select external data.
- 11 Click **OK**. The script editor validates the syntax of the script and identifies any errors. Click:
  - **Yes** to save changes even if the script contains errors.
  - **No** to cancel the changes and close the script dialog box.

In this example, while the Boolean Controls.FillTank is true, the tank level Tank\_001.pv increases by 1 unit every second.



## Editing Symbol Scripts

You can edit predefined and named symbol scripts.

### To edit symbol scripts

- 1 Open the symbol in the ArchestrA Symbol Editor.
- 2 On the **Special** menu, click **Scripts**. The **Edit Scripts** dialog box appears.
- 3 Select the script from the list. The right pane shows the script configuration.
- 4 If you are editing a predefined script, select the script trigger from the **TriggerType** list:
  - **On Show** if the action script you want to edit runs one time after the symbol opens.
  - **While Showing** if the action script you want to edit runs periodically while the symbol is open.
  - **On Hide** if the action script you want to edit is runs one time when the symbol closes.
- 5 Edit the action script in the script box.
- 6 Click **OK**.

## Renaming Scripts in a Symbol

You can rename named scripts in a symbol. When you rename the named script, the functionality of the script does not change.

### To rename a named script

- 1 Open the symbol in the ArchestrA Symbol Editor.
- 2 On the **Special** menu, click **Scripts**. The **Edit Scripts** dialog box appears.
- 3 In the **Named Scripts** list, click the script you want to rename.
- 4 Click the script again. It appears in edit mode.
- 5 Enter a new name for the script and click **ENTER**. The script is renamed.



## Removing Scripts from a Symbol

You can remove predefined or named scripts from a symbol.

### To remove predefined scripts from a symbol

- 1 Open the symbol in the ArcestrA Symbol Editor.
- 2 On the **Special** menu, click **Scripts**. The **Edit Scripts** dialog box appears.
- 3 Select **Predefined Scripts** from the list.
- 4 In the **Trigger type** list, click:
  - **On Show** if the action script you want to remove runs one time after the symbol opens.
  - **While Showing** if the action script you want to remove runs periodically while the symbol is open.
  - **On Hide** if the action script you want to remove runs one time after the symbol closes.
- 5 Delete all the script content in the script box.
- 6 Click **OK**.

### To remove named scripts from a symbol

- 1 Open the symbol in the ArcestrA Symbol Editor.
- 2 On the **Special** menu, click **Scripts**. The **Edit Scripts** dialog box appears.
- 3 Select the named script from the list.
- 4 Click the Remove icon. A message appears.
- 5 Click **Yes**. The script is removed.



## Substituting Attribute References in Scripts

You can substitute attribute references in scripts in the same way as you would with attribute references in elements. For more information, see [Substituting References in Elements](#) on page 351.

## Example of Changing Element Properties using Scripts

You can change some properties of elements using scripting. This lets you configure additional run-time behavior to your elements in addition to design-time animation of those elements.

When you write scripts for the symbol or for one of its elements, you can use the Galaxy Browser to show and select a:

- Property of an element
- Custom property of the symbol

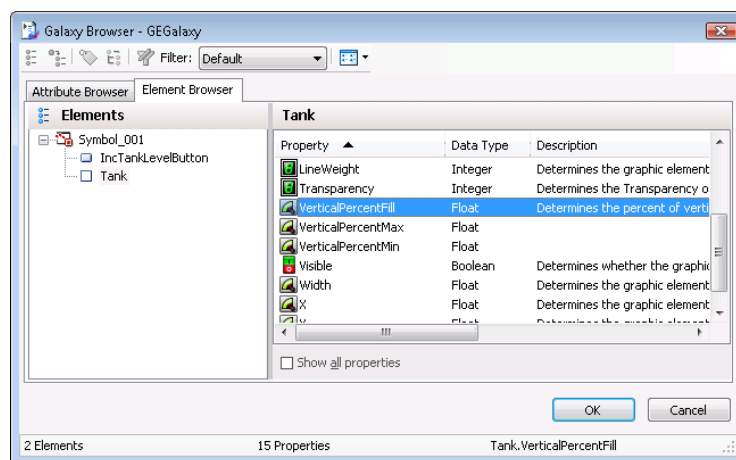
If a reference is not unique, the following order applies:

- 1 Dimensioned variable references
- 2 Graphic properties references
- 3 Custom property references
- 4 Object attribute references

### To select an element property or a symbol custom property

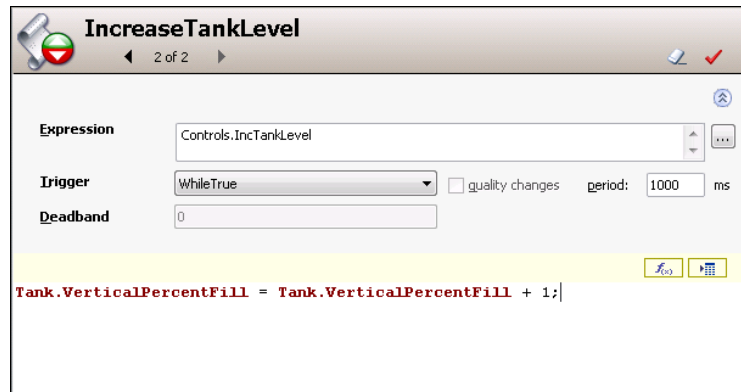


- 1 From the script window, click the Galaxy Browser icon. The **Galaxy Browser** dialog box appears.
- 2 Click the **Element Browser** tab. The Galaxy Browser shows the element names and the properties of the selected element.



- 3 Select an element or symbol from the list. The right pane shows the accessible properties of the selected element or symbol.

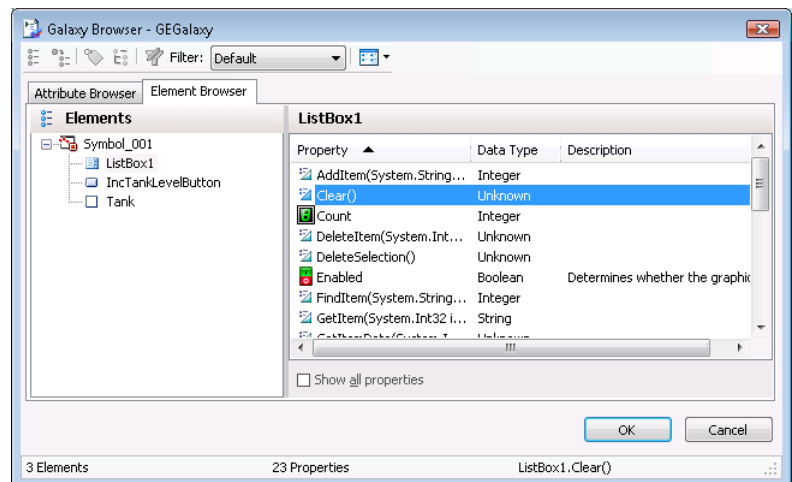
- 4 Select a property from the right pane and click **OK**. The reference appears in the script window.



## Using Methods in Scripting

Some elements, such as the Edit Box, Combo Box and List Box controls, support methods in scripting. These methods can be used to perform various functions on the elements themselves at run time.

You can see the properties and methods supported by any given element by opening the Galaxy Browser and selecting the element.



You can use the methods of the:

- Edit Box control to save and load the text at run time to and from a file.
- Combo Box and List Box controls to access and change the contents of their lists at run time.

## Configuring Edit Box Methods

You can use the methods of an Edit Box control to:

- Save the contained text at run time to a file.
- Load text into the control from a file at run time.

### To save the contained text in an Edit Box control

- ◆ In an action script, use the following method:

```
ControlName.SaveText(FileName);
```

where `ControlName` is the name of the Edit Box control and `FileName` is the name of the file in which to save the contents of the control.

The text contained in the control at run time is saved to the specified file.

If you only specify a file name, the file is saved by default in the user folder of the operating system. For example:

```
c:\documents and settings\username.
```

### To load text into an Edit Box control from a file

- ◆ In an action script, use the following method:

```
ControlName.LoadText(FileName);
```

where `ControlName` is the name of the Edit Box control and `FileName` is the name of the file you want to load the text from.

The text contained in the file is loaded into the run time contents of the Edit Box control.

If you only specify a file name, by default, the file is expected to be in the user folder of the operating system. For example:

```
c:\documents and settings\username.
```

## Configuring Combo Box and List Box Methods

The Combo Box and List Box controls have methods that you can use to access and change the items in the list at run time. Typically, you configure an action script to access these methods.

You can:

- Add and insert items into the list.
- Delete individual or all items from the list.
- Find an item in the list.
- Get the item caption based on a specified index.
- Associate the items with values.
- Load items from and save items to a file.

For more information on the methods, see [Overview of Windows Common Control List Methods](#) on page 494.

### Adding and Inserting Items into a List

You can add an individual item:

- To the end of the list.
- Above the currently selected item.

#### To add an item to a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
ControlName.AddItem("ItemCaption");
```

where `ControlName` is the name of the Combo Box or List Box control and `ItemCaption` is the new item you want to add.

The item is added to the end of the list.

#### To insert an item in a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
Controlname.InsertItem("ItemCaption");
```

where `ControlName` is the name of the Combo Box or List Box control and `ItemCaption` is the new item you want to insert.

The item is inserted above the currently selected item in the list.

## Deleting Items from a List

You can delete:

- An individual item from a list.
- The selected item from a list.
- All items from a list.

If items cannot be deleted from a list at run time, no warning message is shown. Such items include Combo Box and List Box controls configured with enums or arrays.

### To delete an individual item from a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
ControlName.DeleteItem(Index);
```

where `ControlName` is the name of the Combo Box or List Box control and `Index` is the index of the item you want to delete. The first item of the list has an index of 0.

The item at the specified index is deleted, subsequent items are moved up the list.

### To delete the selected item from a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
ControlName.DeleteSelection();
```

where `ControlName` is the name of the Combo Box or List Box control.

The selected item is deleted, subsequent items are moved up the list.

### To delete all items from a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
ControlName.Clear();
```

where `ControlName` is the name of the Combo Box or List Box control.

All items of the control are deleted.

### Finding an Item in a List

You can find an item in a Combo Box or List Box list. You specify the item caption, and the method returns the index number of the first item found. Otherwise, the method returns -1.

#### Finding an item in a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
Index = ControlName.FindItem("ItemCaption");
```

where `ControlName` is the name of the Combo Box or List Box control and `ItemCaption` is the caption of the item you are looking for.

The index is set to -1 if the item is not found, otherwise it contains the index of the first found item. The first item of the list has an index of 0.

### Reading the Caption of a Selected Item in a List

You can read the caption of a selected item in a Combo Box or List Box list.

#### Reading the caption of a selected item in a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
Caption = ControlName.GetItem(Index);
```

where `ControlName` is the name of the Combo Box or List Box control. `Index` is the index of the item for which you want to read the caption. The first item of the list has an index of 0.

`Caption` contains the item caption of the specified index.

### Associating Items with Values in a List

You can associate items with values in a Combo Box or List Box control. This is the same as using a secondary index system to identify items in the list.

You can:

- Set item data, which associates an item with a value
- Get item data, which returns the value that is associated with an item

#### To set item data in a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
ControlName.SetItemData(Index, Value);
```

where `ControlName` is the name of the Combo Box or List Box control, `Index` is the index of the item that you want to set and `Value` is the value you want to assign to the item. The first item of the list has an index of 0.

#### To get item data in a Combo Box or List Box list

- ◆ In an action script, use the following method:

```
Value = ControlName.GetItemData(Index);
```

where `ControlName` is the name of the Combo Box or List Box control and `Index` is the index of the item for which you want to get the value. The first item of the list has an index of 0.

`Value` contains the value that is assigned to the item.

### Loading and Saving Item Lists

You can load and save all items in a list from and to a file.

#### To load the item list for a Combo Box or List Box control from a file

- ◆ In an action script, use the following method:

```
ControlName.LoadList(FileName);
```

where `ControlName` is the name of the Combo Box or List Box control and `FileName` is the name of a file on the local harddrive or on the network.

If you only specify a file name, the file is expected to be in the users folder. For example: `c:\documents and settings\username`.

The list contained in the file is loaded and, if valid, the current list is overwritten.



**To save the item list for a Combo Box or List Box control to a file**

- ◆ In an action script, use the following method:

```
Controlname.SaveList (FileName) ;
```

where ControlName is the name of the Combo Box or List Box control and FileName is the name of a file on the local harddrive or on the network.

If you only specify a file name, the file is saved to the users folder. For example, c:\documents and settings\username.

The list is saved to the specified file.



# Chapter 11

## Using Client Controls

You can:

- Import and embed client controls into a symbol.
- View and edit the properties of the client control.
- Bind the properties of the client control with attributes and element references.
- Configure scripts for client control events.
- Animate client controls.
- Export a client control.
- Configure a client control with security.
- View additional client control information such as the files the client control uses and what objects and symbols are using the client control.

For information on language switching for client controls, see Chapter 14, *Switching Languages for Graphic Elements*.

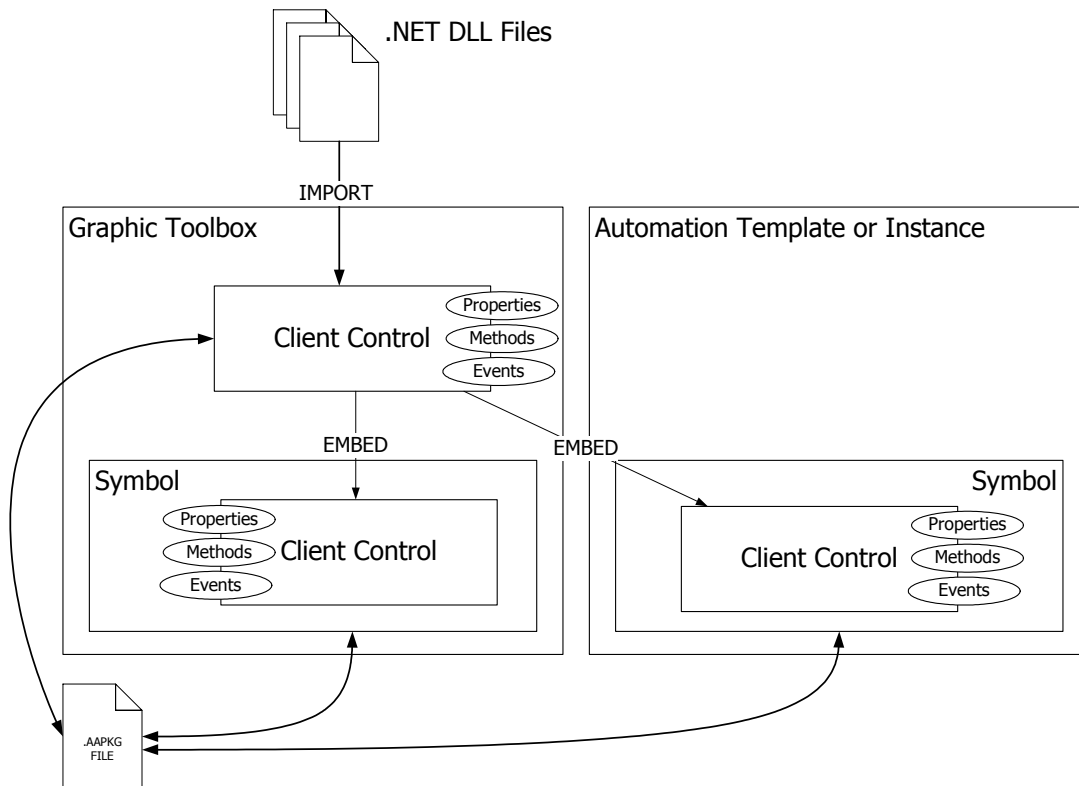
## About Client Controls

Client controls give you functionality contained in .NET controls you can use in symbols. To use this functionality, you must:

- Import the .DLL file that contains one or more client controls. The client control is imported into the Graphic Toolbox.
- Browse and embed one or more of the client controls into a new or existing symbol. The client controls appear as elements.
- View and edit the exposed client control properties.
- Bind the client control properties to ArchestrA attributes, symbol custom properties or InTouch tags. Do this using data binding animation.
- Configure scripts for client control events. Do this using the events animation.

You can then use the symbol containing the embedded client control in an InTouch application.

Wonderware ActiveFactory™ is one example of a software product that contains client controls.



## Importing Client Controls

You can import client controls into the Graphic Toolbox from .NET Dynamic Link Library (.DLL) files.

After importing client controls, you can organize them in the Graphic Toolbox as you would with ArchestrA Symbols. For more information, see [Organizing Client Controls](#) on page 376.

You can also import client controls that have previously been exported in an ArchestrA package (.aaPKG) file.

If you import a newer version of a client control that you are already using in the ArchestrA IDE or in the InTouch HMI as embedded ArchestrA Symbol, you need to restart the ArchestrA IDE and/or the InTouch HMI.

## Importing Client Controls

You can import one or more client controls from .DLL files. The client controls can be contained in one single .DLL file or span multiple files.

To import a client control, you must have security permissions to import graphic objects.

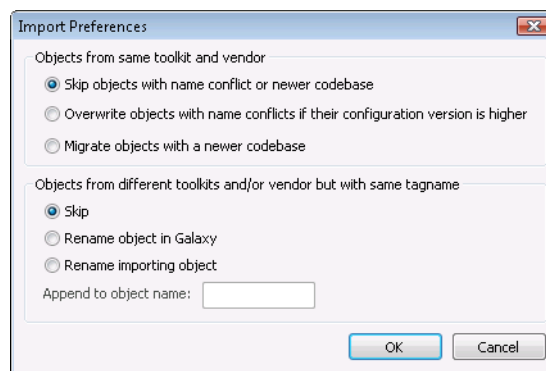
---

**Note** If you select .NET .DLL files that do not contain client controls, the import process ignores these and continues at the next .DLL file.

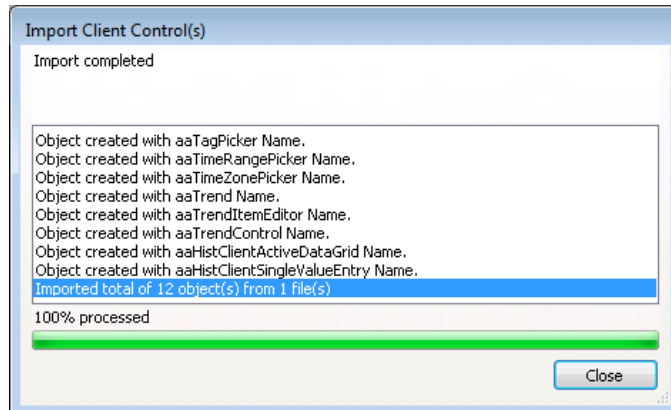
---

### To import client controls

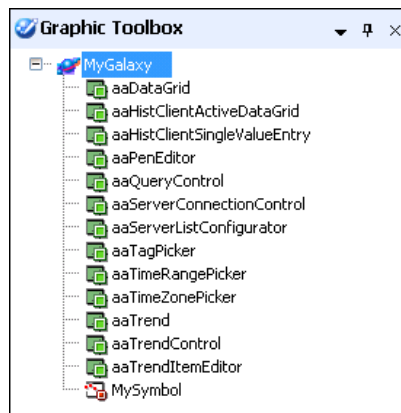
- 1 In the IDE on the **File** menu, point to **Import**, and then click **Client Control**. The **Import Client Control(s)** dialog box appears.
- 2 Select one or more .NET .DLL files that contain the client controls you want to import and click **Open**. The **Import Preferences** dialog box appears.



- 3 Select the appropriate options for the import and click **OK**. The **Import Client Control(s)** dialog box appears.



- 4 When the client controls are imported, click **Close**. The imported client controls appear in the Graphic Toolbox.



---

**Note** If the import fails, a message indicates the error in the **Import Client Control(s)** dialog box.

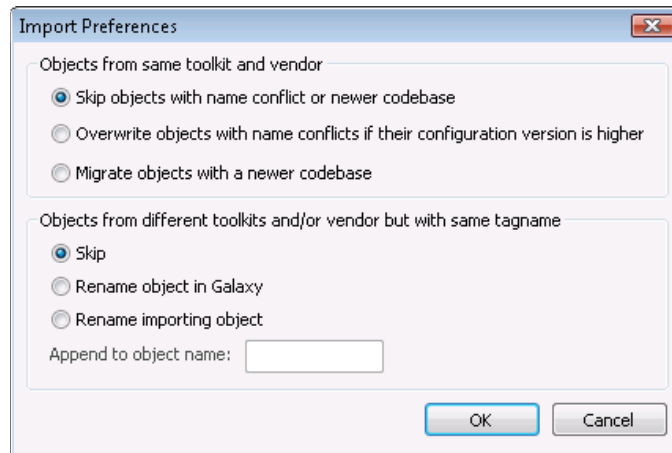
---

## Example of Installing the ActiveFactory TagPicker Control

If you install Wonderware ActiveFactory 9.2 (or higher), you can install the client controls of one of the ActiveFactory \*.DLL files. You can then use these controls to create an ArchestrA Symbol that contains the ActiveFactory TagPicker.

### To install the ActiveFactory TagPicker client control

- 1 Open the IDE.
- 2 On the **File** menu, point to **Import**, and then click **Client Control**. The **Import Client Control(s)** dialog box appears.
- 3 Browse to the C:\Program Files\Common Files\ArchestrA folder, select the aaHistClientTagPicker.dll file and click **Open**. The **Import Preferences** dialog box appears.



- 4 Select the appropriate options for the import and click **OK**.
- 5 When the import is complete, click **Close**.
- 6 Open the **Graphic Toolbox** and expand the Galaxy node. aaTagPicker is listed as a client control.

## Importing Previously Exported Client Controls

You can import one or more previously exported client controls from an ArcestrA package file (.aaPKG). Previously the client controls may have been:

- Exported without a symbol or an AutomationObject instance or template.
- Embedded in a symbol and the symbol was exported.
- Embedded in a symbol and contained in an AutomationObject instance or template and the AutomationObject was exported.

### **To import a previously exported ArcestrA package containing one or more client controls**

- ◆ Import the exported client controls the same way as you would import an AutomationObject (.aaPKG). For more information, see the *Application Server User's Guide*.

## Organizing Client Controls

You can organize the client controls within the Graphic Toolbox the same way as you would with ArcestrA Graphics. You can:

- Rename client controls.
- Move client controls in and out of Graphic Toolsets.
- Delete client controls.

For more information, see *Organizing Symbols in the Graphic Toolbox* on page 73 and *Importing and Exporting Symbols as ArcestrA Object Files* on page 77.



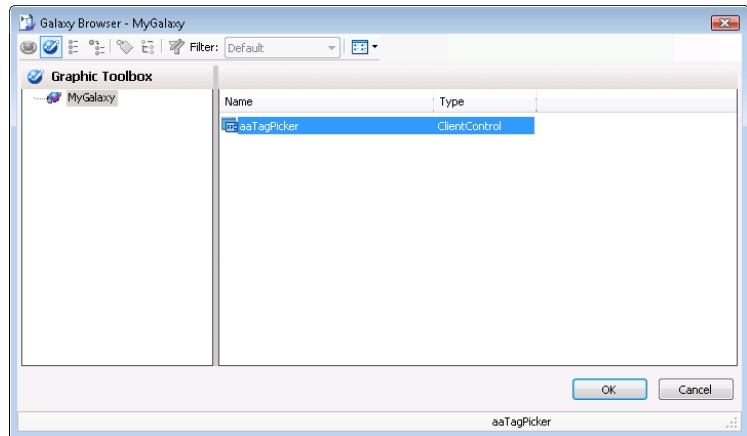
## Embedding Client Controls

You can embed an installed client control into a symbol as you would embed a symbol within another symbol.

We recommend that you not overlap client controls with other elements on the canvas. Otherwise, the client controls may not work correctly.

### To embed a client control into an ArcestraA Symbol

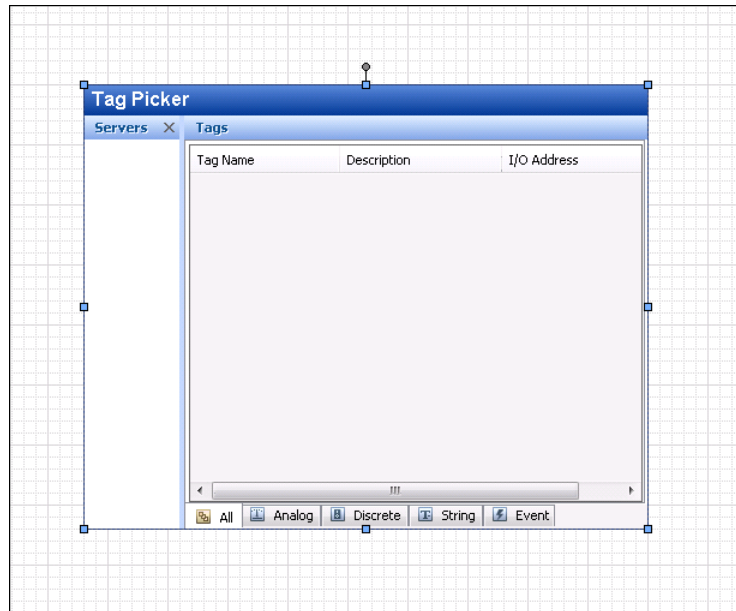
- 1 On the **Edit** menu, click **Embed Graphic Symbol**. The **Galaxy Browser** appears.



- 2 Select the **Graphic Toolbox**.
- 3 Browse to the location that contains the client control.
- 4 Select a client control from the right panel and click **OK**. The pointer changes to paste mode.



- 5 Click on the canvas where you want to embed the client control. The client control is placed onto the canvas.



## Example of Embedding the ActiveFactory TagPicker Client Control

If you install Wonderware ActiveFactory 9.2 (or higher) follow the steps of the Example of Installing the ActiveFactory TagPicker Control on page 375. Then do the following:

- 1 Create a new symbol in the Graphic Toolbox.
- 2 Open the symbol in the ArchestrA Symbol Editor.
- 3 On the **Edit** menu, click **Embed Graphic Symbol**.
- 4 Select **aaTagPicker** and click **OK**.
- 5 Click on the canvas near the top left corner. The ActiveFactory TagPicker control is placed on the canvas.

## Viewing and Changing the Properties of Client Controls

When you embed a client control into a symbol, the native properties of the client control are imported into the Properties Editor in the **Misc** group.

Also the element container of the client control has properties such as:

- Name
- X, Y, Width, Height, AbsoluteOrigin, RelativeOrigin, and Locked
- FillColor
- TextColor and Font
- Enabled, TabOrder, TabStop, and Visible

The element container properties override the native properties of the client control.

You can view and change the properties of the control in the Properties Editor.

### To view or change the properties of a client control

- 1 Select the embedded client control on the canvas.
- 2 In the Properties Editor, locate a:
  - Container property in the property categories **Graphic, Appearance, Fill Style, Text Style** or **Runtime Behavior**.
  - Native property in the **Misc** property category.
- 3 View or change the located property. For more information, see *Editing Common Properties of Elements and Symbols* on page 151.

### To reset a client control back to its original size

- ◆ On the **Edit** menu, click **Control - Original Size**. The **AutoSize** property is set to **False**.

## Example of Changing a Property of the ActiveFactory TagPicker Control

Install and embed the ActiveFactory TagPicker Control into a symbol as described in:

- Example of Installing the ActiveFactory TagPicker Control on page 375.
- Example of Embedding the ActiveFactory TagPicker Client Control on page 378.

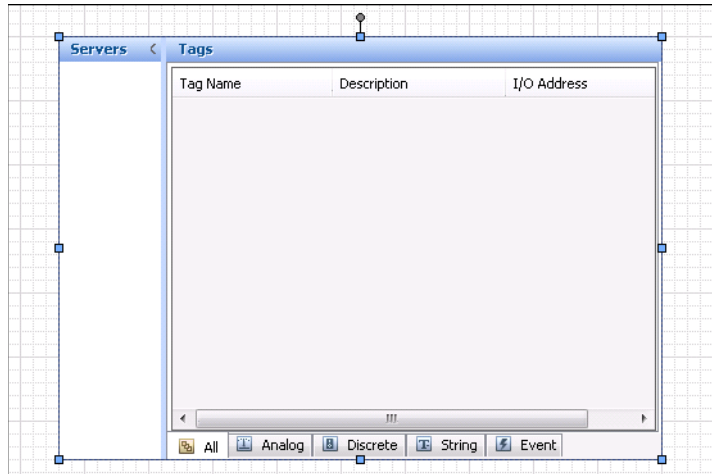
In this example the “Tag Picker” caption of the TagPicker control is hidden.

- 1 Select the embedded ActiveFactory TagPicker client control. The Properties Editor shows all properties of the client control.

Misc	
CanCopy	False
CanCut	False
CanDelete	False
CanPaste	False
CanRedo	False
CanSelectAll	False
CanUndo	False
CurrentGroup	
CurrentServer	
DescriptionFilter	
ExactMatchFilter	<b>False</b>
FilterVisible	<b>False</b>
HideCaption	<b>False</b>
HideFilterCaption	<b>False</b>
HideServersCaption	<b>False</b>
HideTagsCaption	<b>False</b>
IOAddressFilter	
SelectedTagCount	0
SelectedTags	aaTag[] Array
SingleSelectMode	<b>False</b>
SplitterOrientation	<b>Vertical</b>
TagNameFilter	
TreeVisible	<b>True</b>
TreeWidth	<b>80</b>

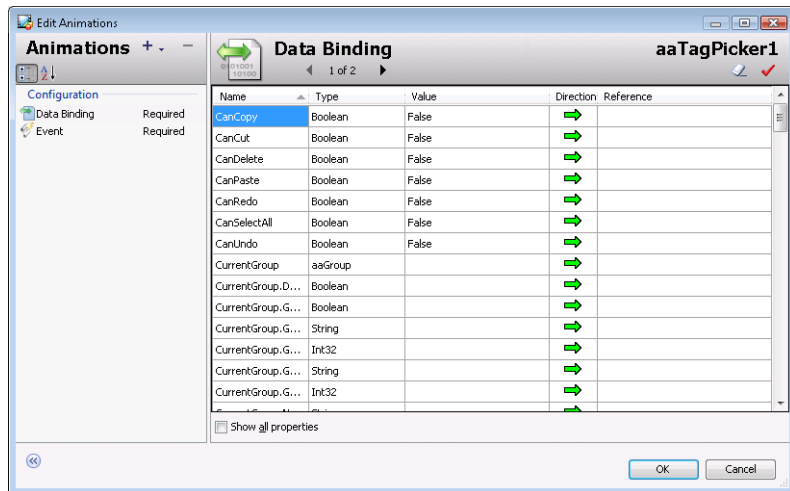
- 2 In the **Misc** property category, locate the property **HideCaption**.

- Assign the value True to it and click **ENTER**. The caption “Tag Picker” of the ActiveFactory TagPicker client control is hidden.



## Binding Client Control Properties to Attributes or Element References

You can bind the properties of an embedded client control to attributes or element references. This lets you use attributes and element references as source and consumer of data for the client control properties. You do this with the **Data Binding** animation.



The Data Binding table contains the following information:

- **Name** - name of the client control property
- **Type** - the .NET data type of the property
- **Value** - the default value of the client control property
- **Direction** - indicates if the property is read/write or just read-only



read/write property



read-only property



write-only property

- **Reference** - the attribute or element reference the property is bound to

---

**Note** You cannot remove the **Data Binding** animation.

---

#### To bind a client control property with an attribute or element reference

- 1 Double-click the embedded client control on the canvas. The **Edit Animations** dialog box appears and the **Data Binding** animation is selected by default.
- 2 Locate the client control property that you want to bind with an attribute or element reference.
- 3 Double-click the **Reference** box.
- 4 Do one of the following:
  - Type an attribute or element reference.
  - Browse for an attribute or element reference by clicking the **Browse** button.
- 5 Repeat above for any other properties you want to bind.
- 6 Click **OK**.

## Example of Data Binding in the ActiveFactory TagPicker Control

Install and embed the ActiveFactory TagPicker control into a symbol as described in:

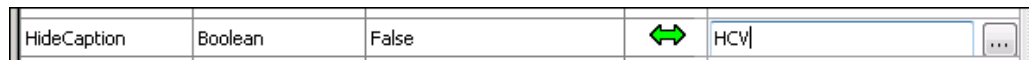
- Example of Installing the ActiveFactory TagPicker Control on page 375.
- Example of Embedding the ActiveFactory TagPicker Client Control on page 378.

In this example, the Boolean symbol custom property **HCV** controls the visibility of the ActiveFactory TagPicker caption.

Do the following:

- 1 Create a Boolean custom property and rename it **HCV**.
- 2 In the ArcestrA Symbol Editor, double-click the embedded ActiveFactory TagPicker control.
- 3 From the list of properties in the **Data Binding** configuration area, locate the **HideCaption** property.
- 4 Double-click the **Reference** box of the **HideCaption** property.
- 5 In the Galaxy Browser, select the **HCV** custom property and click **OK**.

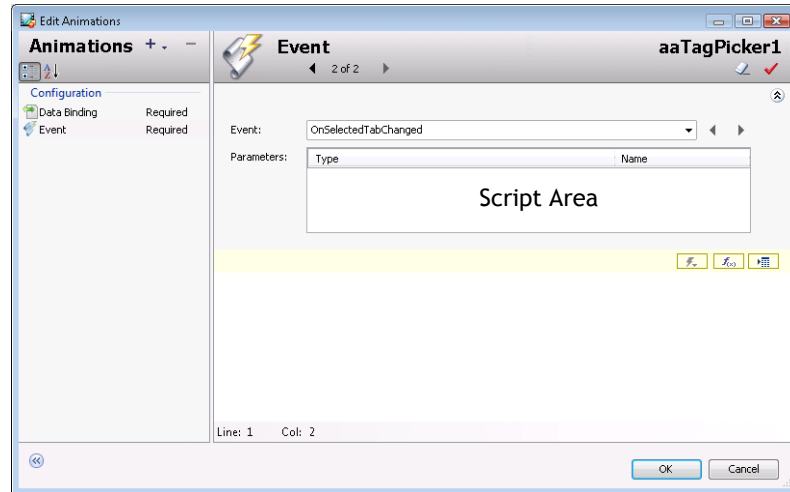
The HideCaption property is now assigned to the element reference HCV.



- 6 Click **OK**.
- 7 Place a button on the canvas and configure it with a Boolean pushbutton animation that toggles the custom property HCV.
- 8 Save and close the symbol.
- 9 Embed the symbol in a managed InTouch application and test the data binding by clicking on the button in WindowViewer. When you do so, the visibility of the caption of the TagPicker control is toggled.

## Configuring Client Control Event Scripts

You can configure an ArchestrA script that is executed when a client control event occurs. You do this using the **Event** animation.

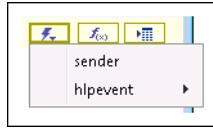


### To configure an ArchestrA script for a client control event

- 1 Double-click the embedded client control on the canvas. The **Edit Animations** dialog box appears.
- 2 In the animation list, click **Event**. The right panel shows the events configuration.
- 3 In the **Event** list, select the event for which you want to execute a script. The **Parameters** list shows for the selected event:
  - **Type**: the data type of each parameter.
  - **Name**: the name of each parameter.
- 4 In the script area, write the event script.



- 5 If you want to insert an event parameter in your script, click the **Select Event Parameter** icon. Select the parameter. The parameter name is inserted into the script at the cursor position.



- 6 If you want to configure scripts for other events, select the event from the **Event** list. The script area is cleared and you can write the script for the newly selected event.
- 7 When you are done, save and close.

## Example of Configuring an Event Script for the ActiveFactory TagPicker Control

Install and embed the ActiveFactory TagPicker control into a symbol as described in:

- Example of Installing the ActiveFactory TagPicker Control on page 375.
- Example of Embedding the ActiveFactory TagPicker Client Control on page 378.

In this example, when one of the tags is picked by double-clicking on it, a message is logged in the Log Viewer.

First however, you need to:

- Import the script function library from the file aaHistClientDatabase.dll.
- Configure a connection to a valid and running Historian Server.

### To import the script functions from aaHistClientDatabase.dll

- 1 On the **Galaxy** menu, point to **Import**, then click **Script Function Library**. The **Import Script Function Library** appears.
- 2 Browse to the aaHistClientDatabase.dll and select it. This file is by default in the C:\Program Files\Common Files\Archestra\ folder.
- 3 Click **Open**. The import starts and finishes with a message.
- 4 Click **OK**.

**To connect the TagPicker control to the Historian Server**

- 1 On the canvas, place a button next to the TagPicker control.
- 2 Double-click the button. The **Edit Animations** dialog box appears.
- 3 Add an **Action Scripts** animation to the animation list.
- 4 In the script area, type the following script:

```
Dim NewServer as
ArchestrA.HistClient.Database.aaServer;
Dim statusMessage as String;
NewServer =
aaTagPicker1.Servers.Add("MyHistorian");
NewServer.LoginID = "MyUserName";
NewServer.Password = "MyPassword";
NewServer.LogOn( statusMessage );
LogMessage ("Connection" + statusMessage);
```

- 5 In the script, replace the strings *MyHistorian*, *MyUserName* and *MyPassword* with the Historian server name, a valid user name, and a password to connect to the server.
- 6 Close the **Edit Animations** dialog box.

You can now configure the client control event to log a message every time the user picks one or more tags by double-clicking on them:

- 1 In the ArchestrA Symbol Editor, double-click on the embedded ActiveFactory TagPicker control.
- 2 In the animation list, click **Event**.
- 3 In the **Event** list, click the **OnTagsPicked** event.
- 4 In the script area, type the following:

```
LogMessage("User picked one or more tags.");
```
- 5 Save and close the **Edit Animations** dialog box.
- 6 Save and close the ArchestrA Symbol Editor.
- 7 Embed the symbol in a managed InTouch application.
- 8 Switch to run-time and connect to a valid Historian Server source.
- 9 Double-click on one of the tags in the TagPicker control.
- 10 Check the SMC Log Viewer. The message "User picked one or more tags" appears.

## Animating Client Controls

Every client control has these animation types:

- Data binding animations determine which attributes and element references can read and write to the client control.
- Event animations assign scripts to individual client control events.

You can add the following animations that correspond to the supported client control container properties:

- Visibility
- Fill Style
- Text Style
- Location Horizontal
- Location Vertical
- Width
- Height
- Tooltip
- Disable

If you configure these animations, the resulting behavior and appearance overrides the behavior and appearance given by the native properties of the client control.

### To add animation to embedded client controls

- 1 Double-click the embedded client control on the canvas. The **Edit Animations** dialog box appears.
- 2 Add animations as you would with any other element. For more information, see [Animating Graphic Elements](#) on page 249.

## Exporting Client Controls

You can export client controls as ArcestrA package files (.aaPKG). You can export them:

- Directly from the Graphic Toolbox.
- Indirectly when you export AutomationObjects or symbols that reference them.

You can import the client controls again from the exported .aaPKG files.

### To export client controls directly as ArcestrA package files

- 1 In the Graphic Toolbox, select one or more client controls that you want to export.
- 2 On the **Galaxy** menu, point to **Export**, and then click **Object(s)**.
- 3 Follow the general procedure for exporting AutomationObjects. For more information, see the *Application Server User's Guide*.

## Securing Client Controls

The client controls use the same security setting as the symbols. You can set the security for client controls and symbols in the **Security** panel of the IDE. For more information, see *Configuring Security for Symbols* on page 80.

## Viewing Additional Client Control Information

You can view:

- Which .DLL files, or assemblies, are used for the client control.
- The class name, vendor, and version.
- Which AutomationObjects and ArcestrA Symbols use the client control.

This information is contained in the Client Control Properties panels.

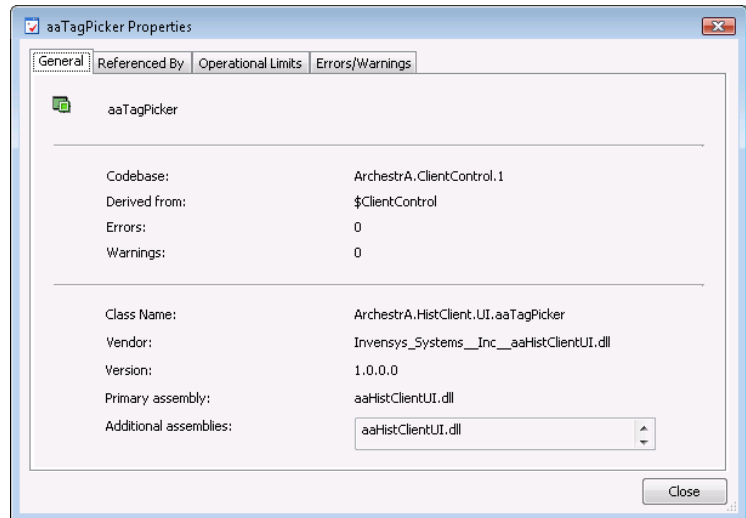
The client control properties are different than the properties of the embedded client control. The client control properties can be viewed in the IDE directly. The properties of the embedded client control can be viewed in the Properties Editor of the ArcestrA Symbol Editor.

## Viewing the Client Control Assemblies

You can view which Client Control .DLL files, or assemblies, are used for the client control.

To view the client control assemblies

- 1 In the Graphic Toolbox, select the client control.
- 2 On the **Galaxy** menu, click **Properties**. The **Properties** dialog box appears.



- 3 On the **General** tab, you can view:
  - The **Primary Assembly**, which is the main .DLL file.
  - **Additional Assemblies**, which are linked to the main .DLL file and automatically loaded.

## Viewing Class Name, Vendor, and Version of a Client Control

You can view the class name, vendor, and version of a client control in its **Properties** panel.

To view the class name, vendor, and version of a client control

- 1 In the Graphic Toolbox, select the client control.
- 2 On the **Galaxy** menu, click **Properties**. The **Properties** dialog box appears.
- 3 Click the **General** tab.

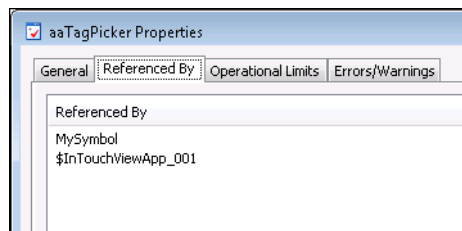
Class Name:	ArcestrA.HistClient.UI.aaTagPicker
Vendor:	Invensys_Systems__Inc__aaHistClientUI.dll
Version:	1.0.0.0

## Viewing Objects and Symbols Referencing Client Controls

You can view which AutomationObjects and ArcestrA Symbols are using a given client control. This can be viewed in the **Properties** dialog box of the client control.

### To view objects and symbols referencing a client control

- 1 In the Graphic Toolbox, select the client control.
- 2 On the **Galaxy** menu, click **Properties**. The **Properties** dialog box appears.
- 3 Click **Referenced By**. The list of objects and symbols using the client control is shown.



---

# Chapter 12

## Embedding Symbols within Symbols

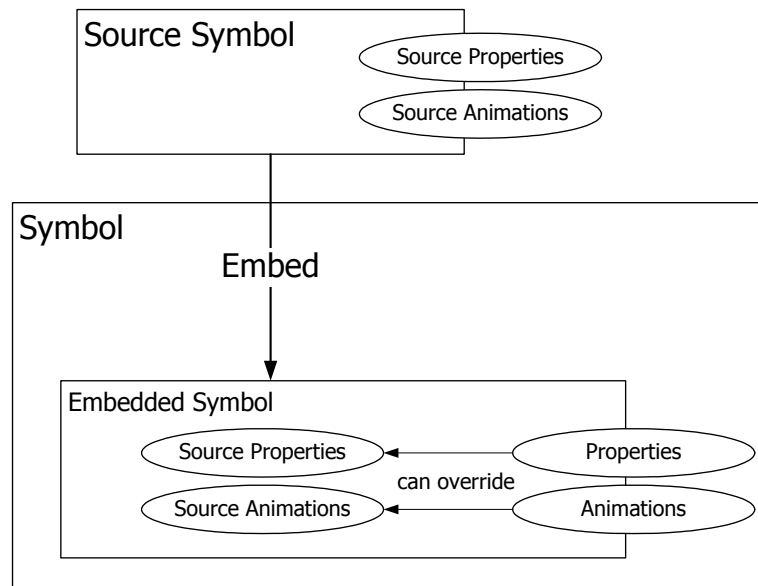
You can embed symbols into other symbols. This lets you split your visualization into modules and re-use already defined symbols. For example, you can create a valve symbol and embed it multiple times into a tank symbol.

When you embed a symbol into another symbol, you are creating a link to the original symbol. Any changes to the original symbol are propagated to all embedded instances.

You can:

- Embed a symbol within another symbol.
- Edit an embedded symbol.
- Restore an embedded symbol to the original size of its source symbol.
- Convert the embedded symbol to a group.
- Detect the source of an embedded symbol.
- Edit the source of an embedded symbol.
- Override the custom properties of the source symbol.
- Control the size propagation of an embedded symbol.

- Select an alternate symbol or same symbol of an alternate AutomationObject instance as source.
- Edit the AutomationObject that contains the source symbol.
- Create a new instance of the AutomationObject that contains the source symbol.



## Embedding Symbols

You can embed symbols from the Graphic Toolbox or an AutomationObject into other symbols.

When you embed a symbol, the animation links and scripts are inherited from the source symbol. You can only change the animations and scripts in the source symbol and all changes are propagated to the embedded symbol.

The embedded symbol appears with its original name appended by a number. The number is increased by one if you embed the same symbol again.

---

**Note** If you embed symbols that have elements outside of the coordinates (0,0) and (2000,2000), the embedded symbol clips these elements.

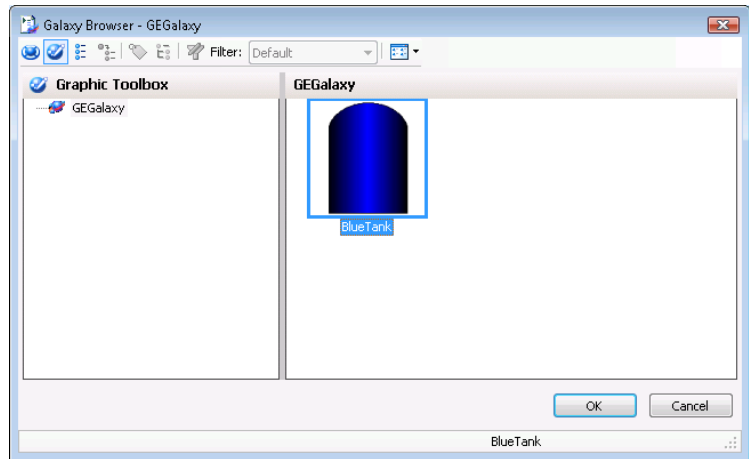
---



### To embed source symbols from the Graphic Toolbox



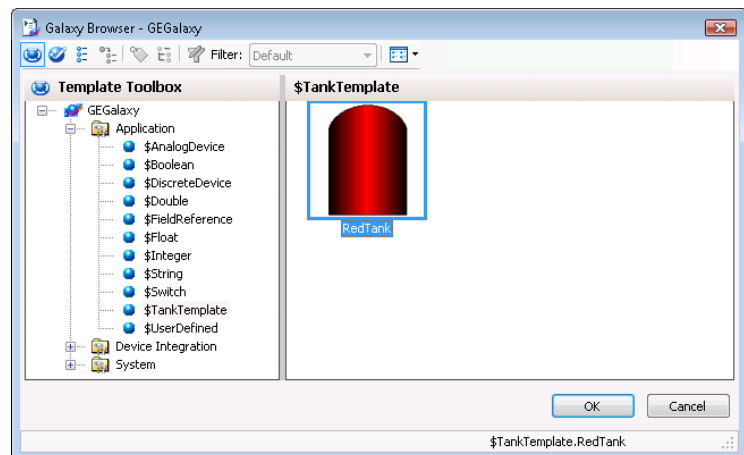
- 1 On the ArchestrA Symbol Editor **Edit** menu, click **Embed Graphic Symbol**. The Galaxy Browser appears.
- 2 Select a source symbol from the Graphic Toolbox.



- 3 Click **OK**. The pointer appears in paste mode.
- 4 Click on the canvas to place the symbol.

### To embed source symbols contained in an AutomationObject template

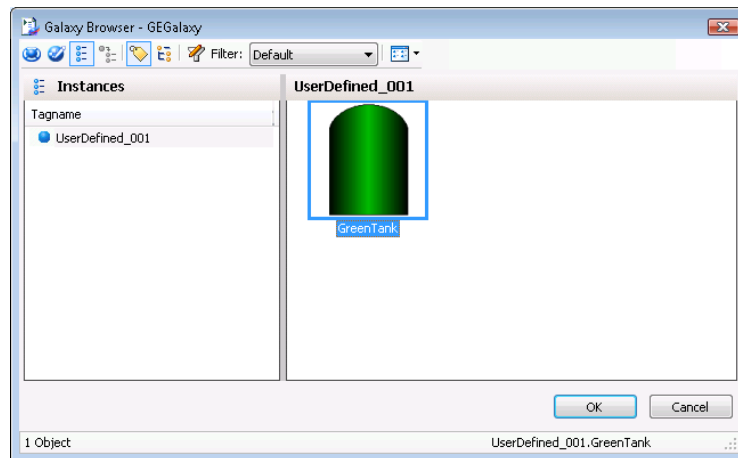
- 1 On the ArchestrA Symbol Editor **Edit** menu, click **Embed Graphic Symbol**. The Galaxy Browser appears.
- 2 Select the AutomationObject template that contains the source symbol.



- 3 Select the symbol and click **OK**. The **Create Instance** dialog box appears.
- 4 Type a name for the new instance in the **New Instance Name** box and click **OK**. The new instance of the AutomationObject is created and the pointer appears in paste mode.
- 5 Click on the canvas to place the symbol.

#### To embed source symbols contained in an AutomationObject instance

- 1 On the ArcestrA Symbol Editor **Edit** menu, click **Embed Graphic Symbol**. The Galaxy Browser appears.
- 2 Select the AutomationObject instance that contains the source symbol.



- 3 Select the source symbol and click **OK**. The pointer appears in paste mode.
- 4 Click on the canvas to place the symbol.

## Renaming Source Symbols and Hosting AutomationObjects

Generally, if you rename source symbols or their hosting AutomationObjects, embedded symbols update their references to the updated name of the renamed source symbol or hosting AutomationObject.

However, if you are using relative references and you rename the contained name of the referenced AutomationObject, the references to the embedded symbol are broken.

You can identify embedded symbols that may cause a problem by clicking on the embedded symbols and viewing the **SymbolReference** property from the Properties Editor. If the SymbolReference property contains relative references such as me or myContainer, renaming the contained name of the referenced object causes the reference to be broken.

Also, if any instance of the hosting AutomationObject is checked out, when you change the contained name of the referenced object, even after you check-in the instance:

- The change is not propagated to the instance.
- Validating the object does not indicate an error.

## Editing the Embedded Symbol

After you embed a source symbol into another symbol, its animations are inherited from the source symbol. The animation of the embedded symbol is controlled by the source symbol.

The embedded symbol itself has certain animations you can configure. The animations override the animations of the source symbol for the embedded symbol. These are:

- Visibility
- Blink
- Location Horizontal
- Location Vertical
- Width
- Height
- Orientation
- Disable

Furthermore you can override the following animations if you change the `TreatAsIcon` property of the embedded symbol to `True`:

- Tooltip
- User Input
- Slider Horizontal
- Slider Vertical
- Pushbutton
- Action Scripts
- Show Symbol
- Hide Symbol

**To override the configured animations of an embedded symbol**

- 1 Select the embedded symbol.
- 2 In the Properties Editor, change the value for the `TreatAsIcon` property to `True`.

## Overriding Custom Properties of the Source Symbol

You can override the value and description of a custom property of the embedded symbol if the custom property's visibility is set to `Public` in the source symbol.

You cannot add, delete, or rename any custom properties of an embedded symbol or change the data type. However, you can:

- Revert the value and description of the custom property to its default as defined in the source symbol.
- Set the visibility of the custom property. This has an effect if the symbol containing the embedded symbol is embedded into another symbol.

**To override the value and description of a custom property**

- 1 Select the embedded symbol on the canvas.
- 2 On the **Special** menu, click **Custom Properties**. The **Edit Custom Properties** dialog box appears.
- 3 Select the custom property you want to override with a new value or description.
- 4 In the **Default Value** box, type a new value.
- 5 In the **Description** box, type a new description.

**To revert the value and description of a custom property**

- 1 Select the embedded symbol on the canvas.
- 2 On the **Special** menu, click **Custom Properties**. The **Edit Custom Properties** dialog box appears.
- 3 Select the custom property you want to revert.
- 4 Click the Revert icon. The value and description of the selected custom property are reverted to the value and description of the same custom property in the source symbol.



## Restoring an Embedded Symbol to the Original Size of its Source Symbol

You can restore an embedded symbol to its original size as it is defined in the object or Graphic Toolbox that contains it.

**To restore an embedded symbol to its original size**

- 1 Select the embedded symbol that you want to restore to its original size.
- 2 On the **Edit** menu, point to **Embedded Symbol**, and then click **Symbol - Original Size**. The embedded symbol is restored to the original size of its source symbol.

## Converting an Embedded Symbol to a Group

You can convert an embedded symbol to a group. A converted symbol is no longer associated with its source symbol. All configuration of the embedded symbol is preserved.

If you convert an embedded symbol to a group:

- Scripts of the embedded symbol are not converted.
- You can optionally move the custom properties to the group.
- Relative references of the embedded symbol are no longer valid.

### To convert an embedded symbol to a group

- 1 Select the embedded symbol that you want to convert to a group.
- 2 On the **Edit** menu, point to **Embedded Symbol**, and then click **Convert To Group**. The embedded symbol is converted to a group.

## Detecting the Source Symbol of an Embedded Symbol

You can view the source of an embedded symbol by using the `SymbolReference` property.

### To detect the source of an embedded symbol

- 1 Select the embedded symbol on the canvas.
- 2 In the Properties Editor, view the `SymbolReference` property see what object or environment contains the source and the name of the source symbol itself. This can be:
  - `Symbol:SymbolName`.
  - `Symbol:InstanceName.SymbolName`.

## Editing the Source of an Embedded Symbol

You can edit the source of an embedded symbol by opening it in a new session of the ArcestrA Symbol Editor.

### To edit the source of an embedded symbol

- 1 Select the embedded symbol.
- 2 On the **Edit** menu, point to **Embedded Symbol**, and then click **Edit Symbol**. The source of the embedded symbol is opened in a new session of the ArcestrA Symbol Editor.
- 3 Edit the source symbol as needed and click **Save and Close**. The new session of the ArcestrA Symbol Editor is closed and the Symbol Changed icon appears in the status bar.
- 4 Double-click the Symbol Changed icon. The change is reflected in the embedded symbol.



If you do not accept the change, the embedded symbol is updated the next time you open it in the ArcestrA Symbol Editor.

## Controlling Size Propagation of Embedded Symbols

You can control the way that size changes of the source symbol are propagated to its embedded instances, which are embedded symbols. For example, a size change is:

- Resizing one of the elements in the source symbol so that the symbol boundary changes.
- Adding elements to or removing elements from the source symbol so that the symbol's boundary changes.

This feature is called dynamic size change and can be enabled or disabled.

## Setting the Anchor Point of a Source Symbol

You can set the position of the anchor point of a source symbol. The anchor point of a source symbol is by default the center point of all elements on the canvas.

You can change the position of the anchor point:

- Numerically by typing the absolute or relative anchor point position values in the Properties Editor.
- Graphically by dragging the anchor point on the canvas.

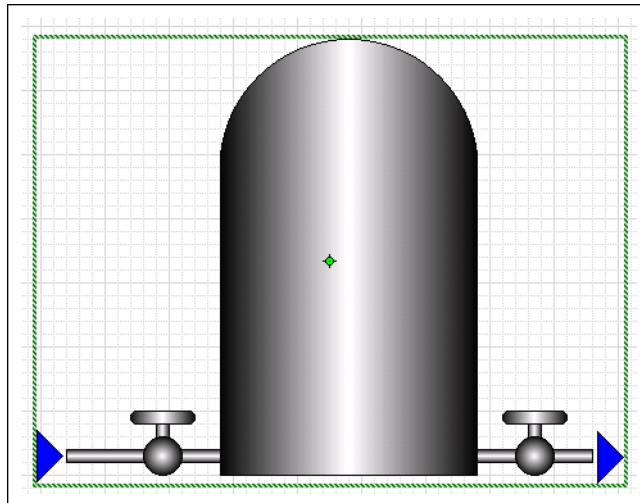
### To change the position of the anchor point numerically

- 1 Click on the canvas.
- 2 In the Properties Editor, type position values X,Y for:
  - **AbsoluteAnchor** property, where the position is relative to the top left corner of the canvas 0,0.
  - **RelativeAnchor** property, where the position is relative to the center point of all elements on the canvas.

The anchor point is changed accordingly. The **AbsoluteAnchor** and **RelativeAnchor** property values are updated accordingly.

### To change the position of the anchor point graphically

- 1 Click on the canvas.
- 2 In the Properties Editor, click the **AbsoluteAnchor** or **RelativeAnchor** property label. The anchor point of the symbol is shown.



- 3 Drag the anchor point to the new position. The **AbsoluteAnchor** and **RelativeAnchor** property values are updated accordingly.



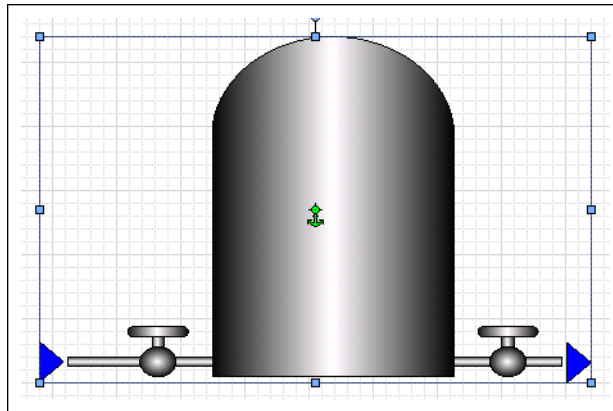
## Showing or Hiding the Anchor Points of Embedded Symbols

You can show or hide the anchor points of embedded symbols. An anchor point shows the current absolute anchor position of the embedded symbol on the canvas.

**To show or hide the anchor point of an embedded symbol**



- ◆ On the toolbar, click the **Show/Hide Embedded Symbol Anchor Points** icon. The anchor of the embedded symbol appears or disappears.

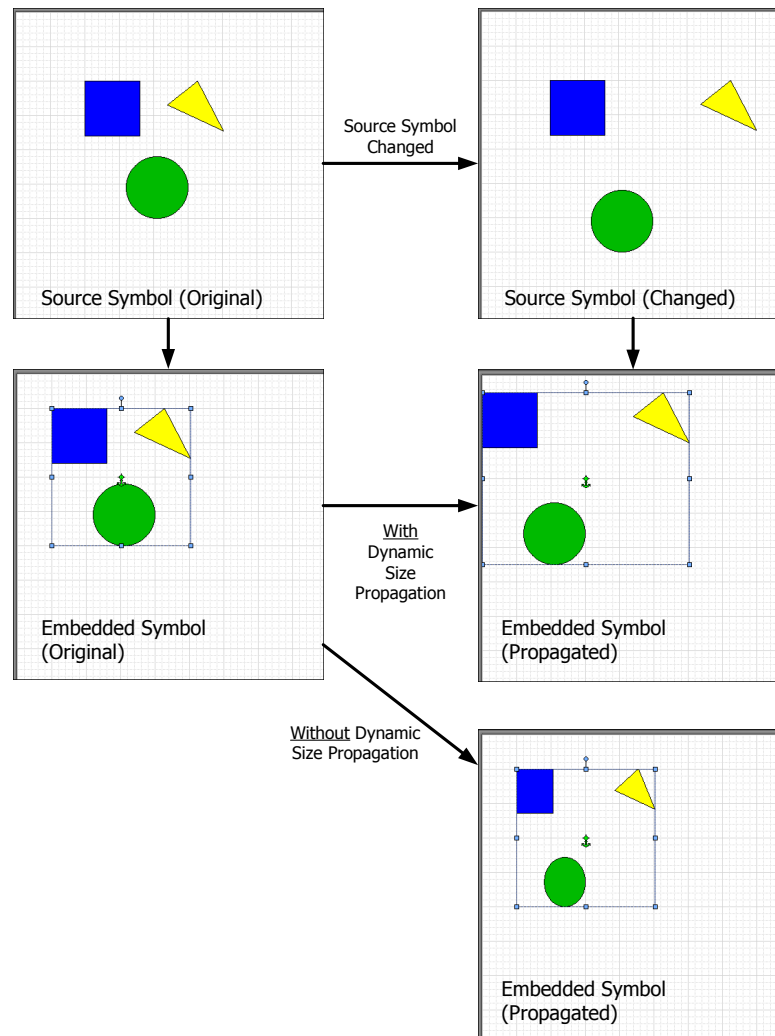


## Enabling or Disabling Dynamic Size Change of Embedded Symbols

You can enable or disable the dynamic size change of embedded symbols. The anchor points of the embedded instances are not changed by any size change to the source symbol.

If the source symbol size changes and the dynamic size change is enabled, the embedded symbol size adapts accordingly. If the dynamic size change is disabled, the embedded symbol size does not change.

In both cases the anchor points of its embedded instances do not move on the canvas.



### To enable or disable dynamic size change of an embedded symbol

- 1 Select the embedded symbol on the canvas.
- 2 On the toolbar, click the **Enable/Disable Dynamic Size Change** icon. The dynamic size change is enabled or disabled.



## Selecting Alternate Symbols and Instances

If your embedded symbol is contained in an AutomationObject instance, you can:

- Use another symbol that is contained in the same AutomationObject instance.
- Use the same symbol that is contained in a different AutomationObject instance, in which case the animation links in the symbol are redirected.

### Selecting Alternate Symbols

You can select an alternate symbol of the same AutomationObject instance to embed. The following properties are retained:

- Position and size
- Animations applied to the embedded symbol
- Override information (the TreatAsIcon property)

You can only select alternate symbols for embedded symbols contained in AutomationObject instances.

#### To select an alternate symbol

- 1 Select the embedded symbol on the canvas.
- 2 On the **Edit** menu, point to **Embedded Symbol**, and then click **Select Alternate Symbol**. The Galaxy Browser appears.
- 3 If available, select an alternate symbol that is contained in the same instance and click **OK**. The embedded symbol is updated with the new alternate symbol.

## Selecting Alternate Instances

You can select an alternate instance of the AutomationObject that contains the same embedded symbol. When you select an alternate symbol to embed, the following properties are retained:

- Position and size
- Animations applied to the embedded symbol
- Override information (the TreatAsIcon property)

### To select an alternate instance

- 1 Select the embedded symbol on the canvas.
- 2 On the **Edit** menu, point to **Embedded Symbol**, and then click **Select Alternate Instance**. The Galaxy Browser appears with a list of all instances that contain the same symbol.
- 3 Select an instance and click **OK**. All internal references of the embedded symbol update to point at the alternate instance. The name of the embedded symbol updates to reflect that it is pointing at a different instance.

## Detecting and Editing the Containing AutomationObject Instance

You can detect and edit the AutomationObject instance that contains the embedded symbol.

### To detect the AutomationObject instance that contains the source symbol

- 1 Select the embedded symbol.
- 2 In the Properties Editor, locate the OwningObject property. Its value contains the name of the object that contains the source symbol.

---

**Note** You can write to this property at run time to force the embedded symbol to point to a different AutomationObject in its references contained in animation links.

---

### To edit the AutomationObject that contains the source symbol

- 1 Select the embedded symbol.
- 2 On the **Edit** menu, point to **Embedded Symbol**, and then click **Edit Instance**. The object instance opens for editing in the IDE.
- 3 Edit the instance as needed and save your changes.

## Creating a New Instance of the Containing AutomationObject

You can create a new instance of the AutomationObject that contains an embedded symbol. The following properties of the symbol are retained:

- Position and size
- Animations
- Override information (the TreatAsIcon property)

**To create a new instance of the AutomationObject that contains an embedded symbol**

- 1** Select the embedded symbol.
- 2** On the **Edit** menu, point to **Embedded Symbol**, and then click **New Instance**. The **Create Instance** dialog box appears.
- 3** Type a name for the new instance in the **New Instance Name** box and click **OK**. The new instance of the AutomationObject is created and the references and name of the embedded symbol are updated to point at it.



# Chapter 13

## Migrating InTouch SmartSymbols

You can use InTouch SmartSymbols in ArcestrA symbols by importing (migrating) them. The SmartSymbol appearance and configuration is imported and converted to animation configuration.

The imported SmartSymbol can:

- Be added to the existing elements on the canvas.
- Replace the existing elements on the canvas.

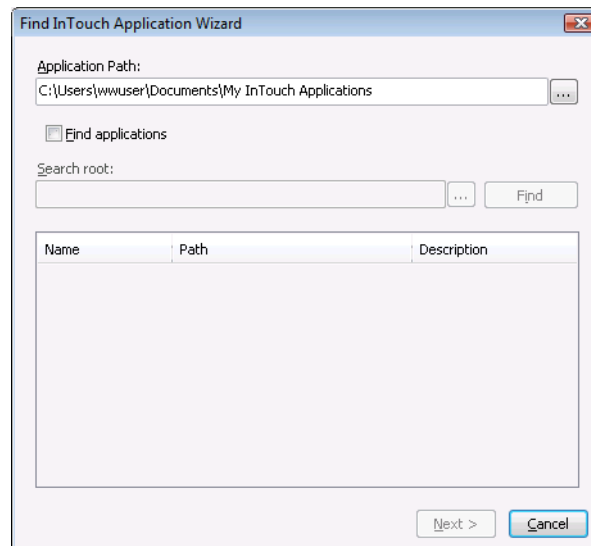
## Importing InTouch SmartSymbols into an ArcestrA Symbol

Generally, you can import any InTouch SmartSymbol into an ArcestrA symbol.

**Note** The SmartSymbol can contain objects that cannot be imported, or can be imported but have limited functionality. For a full list of these objects, see Restrictions for SmartSymbol Import on page 411.

### To import an InTouch SmartSymbol into an ArcestrA Symbol

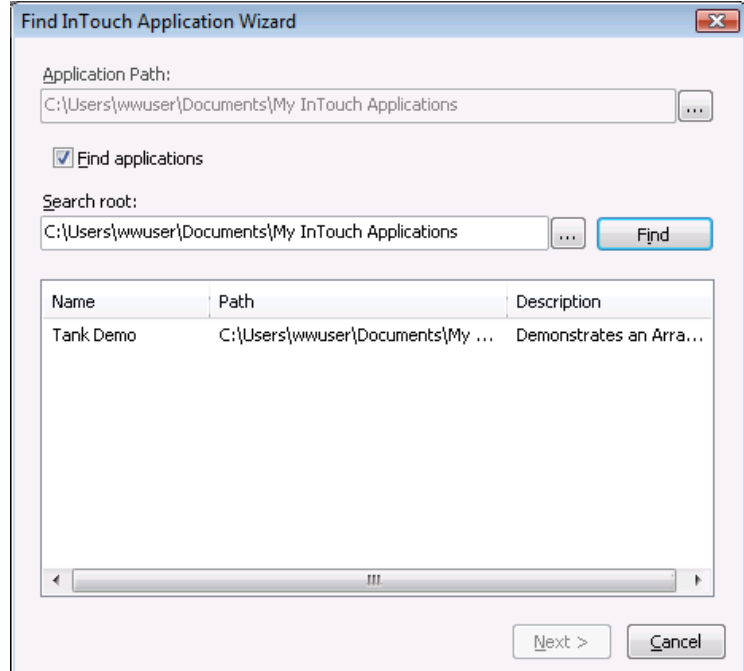
- 1 Open the ArcestrA Symbol Editor.
- 2 On the **Special** menu, click **Import InTouch SmartSymbol**. The **Find InTouch Application Wizard** appears.



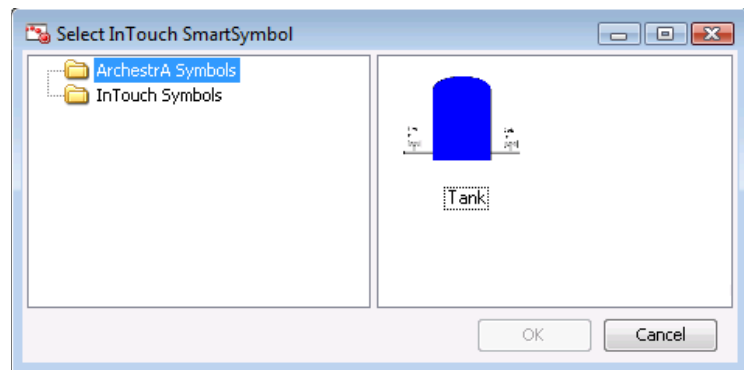
- 3 If your InTouch application is in a different folder than the default, click the browse button to browse for the path to the InTouch application.
- 4 Select **Find Applications**. The **Search Root** box shows the path under which all applications are to be searched.
- 5 If your InTouch application is not under the specified **Search Root** path, change the **Search Root** path by typing a new start folder for the search or browsing for one.



- 6 Click **Find**. All InTouch applications contained in the specified **Search Root** folder are found and listed.



- 7 Select the application from which you want to import SmartSymbols and click **Next**. The **Select InTouch SmartSymbol** dialog box appears.



- 8 Browse to the location of the SmartSymbol in the SmartSymbol hierarchy, select the SmartSymbol that you want to import, and then click **OK**.
- 9 If you already have elements on the canvas, a dialog box appears prompting if you want to replace the existing elements. Click:
- **Yes** if you want to delete the existing elements and import the SmartSymbol on an empty canvas.
  - **No** if you want to keep the existing elements and import the SmartSymbol.

- 10 If the SmartSymbol contains fonts that are currently not installed on the operating system, the **Edit Font Mapping** dialog box appears.



You can click **Continue** to accept the suggested font mapping, or change the font mapping for each individual font. To do this:

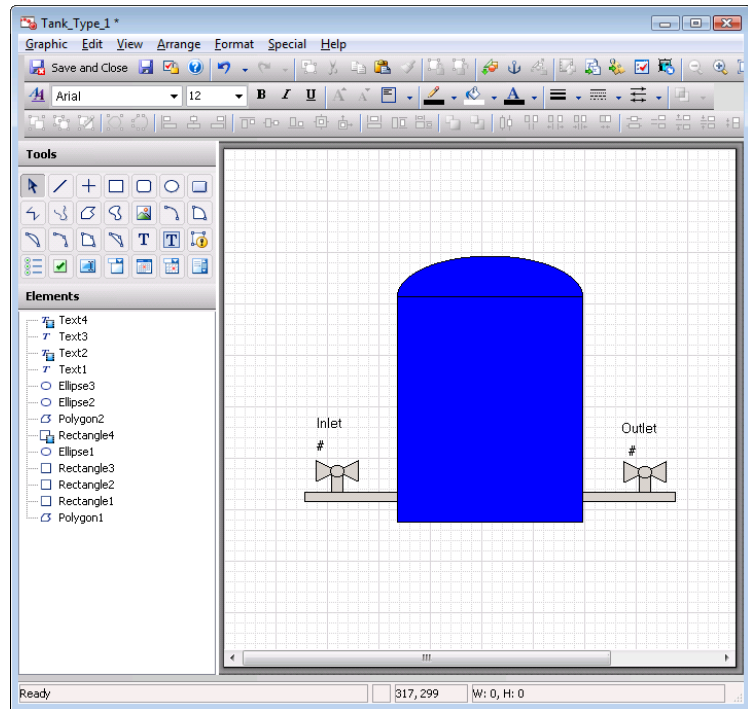
- a Click the font name in the **Mapped Font** column. A browse button appears.
- b Click the browse button. The **Supported Font Selection** dialog box appears.
- c Select a font from the list and click **OK**. The selected font appears in the **Mapped Font** column.
- d Repeat the steps for any other font you want to map to another font.

---

**Note** If you want to save the mapping for the next time you import a SmartSymbol, check **Save mapping**.

---

- 11 The SmartSymbol is imported and appears on the canvas.



## Restrictions for SmartSymbol Import

When you import an InTouch SmartSymbol, the following configuration is imported:

- InTouch graphics
- Graphical animations
- Scripts
- References

### Importing InTouch Graphics

The following tables shows you InTouch graphics that:

- Can be imported without any problem.
- Can be imported but are changed in their functionality or lose some functionality in the process.
- Cannot be imported.

The following InTouch graphics can be imported without any problem:

InTouch Graphic	ArchestrA Graphic Element	Notes
Rectangle	Rectangle	
Rounded Rectangle	Rounded Rectangle	
Ellipse	Ellipse	
Line	Line	
H/V Line	Line	Smart Symbols convert H/V lines to Lines. Therefore, ArchestrA can only generate lines.
Polyline	Polyline	
Polygon	Polygon	
Text	Text	
Bitmap	Bitmap	
Cell	Group	ArchestrA property "Treat as Icon" = false.
Symbol	Group	ArchestrA property "Treat as Icon" = true.
Button	Button	

The following InTouch graphics can be imported, but are changed in their functionality or lose some functionality in the process:

InTouch Graphics	ArchestrA Graphic Element	Notes
Wizard	Elements	When grouped in a SmartSymbol, it appears as a group of elements.
SmartSymbol	Elements	When grouped in another Smart Symbol, it is broken down into a cell, losing its SmartSymbol properties.

The following InTouch graphics cannot be imported, as they cannot be added to a SmartSymbol:

InTouch Graphic	ArchestrA Graphic Element	Notes
RealTime Trend	n/a	Cannot be added to a SmartSymbol.
Historical Trend	n/a	Cannot be added to a SmartSymbol.
ActiveX Controls	n/a	Cannot be added to SmartSymbol. This would include all ActiveX alarm controls (Alarm DB View, Alarm Viewer, and so on)

## Importing Graphical Animation

When you import an InTouch SmartSymbol, all data configured in InTouch animations is imported to ArcestraA animations. InTouch animations and ArcestraA animations often have a different name, but perform the same function.

The following table shows you which animations correspond to each other:

<b>InTouch Animation Link</b>	<b>ArcestraA animation</b>
User Inputs - Discrete	User Input
User Inputs - Analog	User Input
User Inputs - String	User Input
Sliders - Vertical Slider	Vertical Slider
Sliders - Horizontal Slider	Horizontal Slider
Touch Pushbuttons - Discrete Value	Pushbutton
Touch Pushbuttons - Action	Action Scripts
Touch Pushbuttons - Show Window	not supported
Touch Pushbuttons - Hide Window	not supported
Line Color - Discrete	Line Style
Line Color - Analog	Line Style
Line Color - Discrete Alarm	Line Style
Line Color - Analog Alarm	Line Style
Fill Color - Discrete	Fill Style
Fill Color - Analog	Fill Style
Fill Color - Discrete Alarm	Fill Style
Fill Color - Analog Alarm	Fill Style
Text Color - Discrete	Text Style
Text Color - Analog	Text Style
Text Color - Discrete Alarm	Text Style
Text Color - Analog Alarm	Text Style
Object Size - Height	Height
Object Size - Width	Width
Location - Vertical	Location Vertical
Location - Horizontal	Location Horizontal

<b>InTouch Animation Link</b>	<b>ArchestrA animation</b>
Percent Fill - Vertical	% Fill Vertical
Percent Fill - Horizontal	% Fill Horizontal
Miscellaneous - Visibility	Visibility
Miscellaneous - Blink	Blink
Miscellaneous - Orientation	Rotation
Miscellaneous - Disable	Disable
Miscellaneous - Tooltip	Tooltip
Value Display - Discrete Value	Value Display
Value Display - Analog Value	Value Display
Value Display - String Value	Value Display

## Importing Action Scripts

When you import a SmartSymbol, all action scripts associated with objects in SmartSymbol are imported as well. An action script in a SmartSymbol becomes a script animation in an ArchestrA Symbol.

Most of the predefined InTouch functions (QuickScripts) are imported .

### Mathematical Functions

The following mathematical functions in InTouch WindowMaker are supported by the ArchestrA Symbol Editor:

`Abs, ArcCos, ArcSin, ArcTan, Cos, Exp, Int, Log, LogN, Pi, Round, Sgn, Sin, Sqrt, Tan, Trunc`

### String Functions

The following string functions in InTouch WindowMaker are supported by the ArchestrA Symbol Editor:

`Dtext, StringASCII, StringChar, StringCompare, StringCompareNoCase, StringFromGMTTimeToLocal, StringFromIntg, StringFromReal, StringFromTime, StringFromTimeLocal, StringInString, StringLeft, StringLen, StringLower, StringMid, StringReplace, StringRight, StringSpace, StringTest, StringToIntg, StringToReal, StringTrim, StringUpper, Text, wwStringFromTime`

### System Functions

The following system functions in InTouch WindowMaker are supported by the ArcestrA Symbol Editor:

`ActivateApp`

### Miscellaneous Functions

The following miscellaneous functions in InTouch WindowMaker are supported by the ArcestrA Symbol Editor:

`DateTimeGMT`, `LogMessage`, `SendKeys`, `WWControl`,  
`WWExecute`, `WWPoke`, `WWRequest`, `IOSetRemoteReferences`,  
`IOSetAccessName`, `IOReinitialize`, `IOReInitAccessName`,  
`IOForceFailOver`, `IODisableFailover`

## Importing References

When you import a SmartSymbol, the following changes are made to tags and references:

InTouch SmartSymbol	ArcestrA Symbol	Example
Local Tags	Prefixed with “InTouch:” keyword	Real Memory Tag “TankLevel1” is converted to “InTouch:TankLevel1”
Local Tags with dotfields	Prefixed with “InTouch:” keyword	Discrete Memory Tag “TankLevel1.InAlarm” is converted to “InTouch:TankLevel1.InAlarm”
SuperTags	Prefixed with “InTouch:” keyword. You need to manually enclose the expression by the following syntax: <code>attribute(“...”);</code>	Real Supertag member “Reactor1\Level” is converted to “InTouch:Reactor1\Level”. You need to change the expression manually as follows: <code>attribute(“InTouch:Reactor1\Level”);</code>
I/O References	Prefixed with “InTouch:” keyword	Integer I/O Tag “Testprot:i00” is converted to “InTouch:Testprot:i00”
Galaxy References	“Galaxy:” prefix is removed	Galaxy Reference “galaxy:Pump1.Valve1” is converted to “Pump1.Valve1”



The following items are imported with functional change:

InTouch SmartSymbol	ArchestrA Symbol	Example
Galaxy:ObjectTagname. Property.#VString	“Galaxy:” prefix is removed but #VString is not supported. Applies also for #VString1, #Vstring2, #VString3 and #VString4	“Galaxy:Tank.PV.#VString4” is converted to “Tank.PV”
Galaxy:ObjectTagname. Property.#ReadSts	“Galaxy:” prefix is removed but #ReadSts is not supported	“Galaxy:Tank.PV.#ReadSts” is converted to “Tank.PV”
Galaxy:ObjectTagname. Property.#WriteSts	“Galaxy:” prefix is removed but #WriteSts is not supported	“Galaxy:Tank.PV.#WriteSts” is converted to “Tank.PV”
Galaxy:ObjectTagname. Property.#EnumOrdinal	“Galaxy:” prefix is removed but #EnumOrdinal is not supported	“Galaxy:Selection.Sel1.#EnumOrdinal” is converted to “Selection.Sel1”



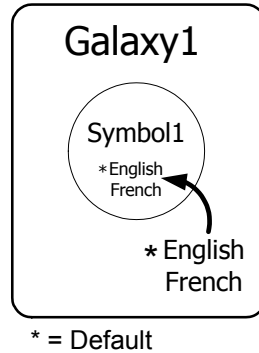
# Chapter 14

## Switching Languages for Graphic Elements

This section describes how to switch the language shown for ArcestrA symbols, the effects of different language settings between the Galaxy and an individual ArcestrA symbol, and language switching behaviors for certain features, such as embedded symbols, custom properties, and string substitution.

## About Language Switching for ArcestrA Graphics

The language settings of the Galaxy control which languages are available to symbols. You cannot add a language at the symbol level. Languages are only added at the Galaxy level using the ArcestrA IDE.



Using the Symbol Editor, you select which of the Galaxy-configured languages you want to show for your symbol. By default, symbol text is shown in the default language and font for the Galaxy.

You export symbol text for translation using the IDE.

## Graphic Elements that Support Translation

The following table describes the graphic elements that support translation. Element fonts are fonts set using the Property Editor and Symbol Editor tools.

Graphic Element	Type of Translation Supported
Text	Element Text, Font
TextBox	Element Text, TextFormat, Font
Button	Element Text, TextFormat, Font
Radio Button Group	Element Font
CheckBox	Element Font
EditBox	Element Font
ComboBox	Element Font
ListBox	Element Font
Calendar	Element Font

Graphic Element	Type of Translation Supported
DateTime Picker	Element Font
Embedded Symbol	String substitutions and overridden custom properties defined as static strings
Client Control	Element Font
Base Symbol	Custom Properties defined as static strings

## Animations that Support Translation

The following table lists the user translation support for animations. When you switch the language at design time or run time, the following portions of the animation change accordingly.

Animation	Type of Translation Supported
ValueDisplay (Boolean Type)	Static string in the True Message of the Boolean type. Static string in the False Message of the Boolean type.
UserInput	Static string in the Message to User for all types. Static string in the True Prompt for Boolean type. Static string in the False Prompt for Boolean type. Static string in the True Message for Boolean type. Static string in the False Message for Boolean type.
ToolTip	Only the static string in the tooltip animation can be translated.
ShowSymbol	Only the static string in the title of the ShowSymbol animation can be translated when the <b>Use Symbol Name for Window Title</b> option is unchecked.
PushButton (String Type)	Only the static strings in the Value1 and Value2 of the String type Pushbutton animation can be translated. No other types of PushButton animations can be translated.

Animation	Type of Translation Supported
Radio Button	Only the captions of individual radio button items in the Static type, Array type, or Enum type RadioButton animations can be translated when the <b>Use Values as Captions</b> option is unchecked.
CheckBox	Only the caption string of the CheckBox animation can be translated when the <b>Override the caption with the following expression</b> option is unchecked.
EditBox	The configuration text of the EditBox animation can be translated only when the EditBox data source is left empty. If a data source is provided, the EditBox text cannot be exported for translation.
ComboBox	Only the captions of individual combo box items in the Static type, Array type, or Enum type ComboBox animation can be translated when the <b>Use Values as Captions</b> option is unchecked.
ListBox	Only the captions of the individual list box items in the Static type, Array type, or Enum type ListBox animation can be translated when the <b>Use Values as Captions</b> option is unchecked.

## Selecting the Language for a Symbol

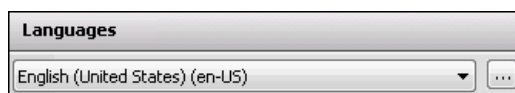
When you select a language for a symbol, all graphic elements show the translated text associated with the selected language, if it is available. You can switch languages even if you open the symbol in read-only mode.

You can only select languages that are currently configured for the Galaxy.

**Note** The current element creation font does not change when you switch to a different language.

### To select the language for a symbol

- 1 Open the symbol in the Symbol Editor.
- 2 In the **Languages** panel, select the language from the list.



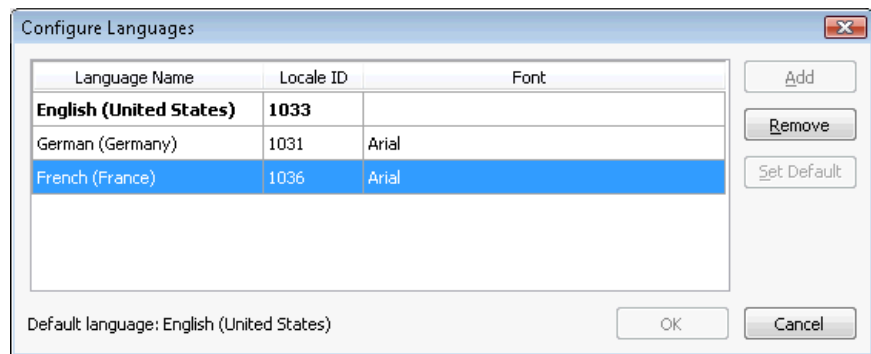
## Removing a Language for a Symbol

If you remove a language from the Galaxy, the language is still available for a symbol until you specifically remove it.

You cannot remove a language from a symbol if the language is configured at the Galaxy level.

### To remove the language for a symbol

- 1 Open the symbol in the Symbol Editor.
- 2 On the **Special** menu, point to **Configure** and then click **Locales**. The **Configure Languages** dialog box appears.



- 3 Select the language to remove and click **Remove**.
- 4 Click **OK**.

## Creating Elements When Multiple Languages are Defined for a Galaxy

You must select the default language for the Galaxy before you create a new graphic element. You cannot create an element if you have a secondary language selected in the Symbol Editor. This includes:

- Duplicating an element
- Pasting an element
- Embedding a symbol
- Grouping or ungrouping elements
- Combining paths
- Breaking paths
- Import an InTouch SmartSymbol
- Convert a symbol to a group

## Moving Symbols to Galaxies with Different Language Settings

If you import a symbol into a different Galaxy, any translation that already exists for the symbol is retained; however, only languages that are configured for the new Galaxy are available for use.

If you open a symbol that does not contain the default language of the Galaxy, then the default language is added to the list of languages for that symbol and is set as the new default language of the symbol. The text strings from the last saved default language of that symbol are transferred over to the new default language of the symbol.

For example, if a symbol only has the French language configured and is opened in a Galaxy where English is the default language, the French strings are transferred over to the English language when the symbol is opened, and English is made the default language for the symbol. The French language still exists for the symbol, but it will no longer be the default language.

If a previously imported symbol has English and French configured with French as the default language and the symbol is opened in a Galaxy with English as the default language, the default language for the symbol is set to English.

If you open a symbol in read-only mode, no changes are made to the symbol language settings to reflect the language settings of the Galaxy.

We recommend that you validate templates and instances that have symbols after migrating them to a new Galaxy and before exporting localizations.



## How Fonts are Applied at Design Time

When you create a graphic element that supports visible text, it is created in the default language of the Galaxy. The font used is the last font persisted in the editor. However, if you provide a specific translation for an element in a secondary language, the Galaxy-configured font for the language is applied to the element.

For example:

- 1 You configure three languages: English (Default, Font = Arial), French (Font = Courier New) and German (Font = Times New Roman).
- 2 You open a symbol S1 in the English language. The editor default is Arial. You create a textbox in English. It is created with the Arial font.
- 3 You switch to German and translate the text. The font changes to Times New Roman, which is the font configured for German. The original font size and style remains the same.
- 4 You switch to French. The font is Arial because the text in French is not translated yet.

## Language Switching for Embedded Symbols

When you embed a symbol into another symbol, any translations are also loaded for the embedded symbol. Switching the language for a symbol also switches the language for all embedded symbols.

For example, a symbol S1 contains a text graphic with the text "English String." English is the default language. The text is set to "French String" for the French language. The following steps describe language switching for the embedded symbol:

- 1 You embed the symbol S1 into the symbol S2.  
You see the text showing the "English String."
- 2 You switch the language to French in the editor of S2.  
The text in the embedded symbol S1 switches to "French String."

If you convert an embedded symbol to a group, any translations defined for the embedded symbol are migrated to the new elements created in the symbol. If the embedded symbol supports languages that are not defined in the base symbol, those translations are removed during the conversion.

If you open a symbol containing one or more embedded symbols and the current language of the Symbol Editor is not available in the embedded symbols, then the embedded symbols use the Galaxy-configured default language, if available. If the Galaxy-configured default language is not available, then the embedded symbols use the last saved default language of that symbol.

For example:

- 1 You create symbol S1 with French and German languages. The default language for the Galaxy is German.
- 2 You embed S1 inside S2 and then save and close S2.
- 3 You change the Galaxy default language to French.
- 4 You add English as another language.
- 5 If you open the symbol S2 in the English language, then the symbol S1 is shown using the French text. This is because French is the Galaxy-configured default language.

Embedded symbols support translations for custom properties. For more information on translating custom properties, see [Translating String Custom Properties](#) on page 428.

Embedded symbols also support translations for string substitutions. For more information on translation and string substitutions, see [String Substitutions and Language Switching](#) on page 426.

## String Substitutions and Language Switching

You can substitute strings for textual elements within the symbol. For general information about string substitution, see [Substituting Strings](#) on page 164.

The following design time and run-time rules are applied during a language switch to properly update an embedded symbol with the current substitutions for the language:

- 1 Apply the string substitutions from the default language.
- 2 Apply the string substitutions from the secondary language, if switching to a secondary language.

For example, an embedded symbol contains a text graphic with the text "English String." English is the default language. The following steps describe how changing the language affects string substitution for the embedded symbol:

- 1 While editing in the default language, you select an embedded symbol and open the string substitution dialog box.  
You can see the old column with a value of "English String."
- 2 You replace the "English String" with "New English String."
- 3 You close the string substitution dialog box.
- 4 You switch to the French language.
- 5 You open the string substitution dialog box and see the string "New English String" in the old column.
- 6 You now replace the "New English String" with "New French String."
- 7 You close the string substitution dialog box.
- 8 You switch to the German language.
- 9 You open the string substitution dialog box and see the string "New English String" in the old column.
- 10 You now replace the "New English String" with "New German String."
- 11 You close the string substitution dialog box.
- 12 You switch to the French language.
- 13 You open the string substitution dialog box and see the string "New French String" in the old column.

If you select an alternate symbol, the string substitutions made on the initial symbol are reapplied to the new symbol across all languages.

The behavior of ArcestrA client control string substitutions are the same as the embedded symbols.

## Translating String Custom Properties

You can translate custom properties that are defined as static strings.

If the custom property is configured with a reference, then that reference applies across all languages in the symbol. If you change that reference in any language to a static string, that string is set for all languages, and you can provide specific translations in the other languages.

For example, you create a custom property CP1 of type string with a default value of "Hello." You can now translate this custom property. You switch to another language in the editor and modify the default value of CP1 to UD1.str1 (changed from string to reference). Now CP1 cannot be translated. If you go back and change CP1 from a reference to a string, you can translate it again. The value you place in the default value is the value shown for all other languages if you do not specify a different string in that language.

When the custom property dialog box opens, it shows the appropriate translated values for the constant string custom properties, as determined by the translation precedence rules. For more information on these rules, see [Precedence Rules for Showing the Language and Font on page 437](#).

## Translating Custom Properties for a Base Symbol

New custom properties you create can be translated specifically for the symbol, provided that:

- They are configured as string custom properties.
- The default language value of the custom property is a static string (as opposed to a reference).

For example:

- 1 While editing the default language, you create a new custom property called CP1 with a data type of string.
- 2 You enter the value "English String" as the custom property value.
- 3 You close the custom property dialog box to save the changes.
- 4 You switch to the French language.
- 5 You open the custom property dialog box and see the string "English String."
- 6 You can now enter a specific string for French in the CP1 custom property. You cannot enter a reference in the French language.

## Translating Custom Properties for an Embedded Symbol

You can translate custom properties exposed in an embedded symbol if the overridden value is a constant string. This also applies if the base value is a reference and is overridden to a constant string. All the translations for a single overridden custom property must be constant string values. The first localized overridden value in one local is propagated to all the other locales as the default overridden value.

For example:

- 1 While editing the default language, you select an embedded symbol and open the custom property dialog box.  
You can see a constant string custom property called "CP1" with a value of "English String."
- 2 You close the custom property dialog box.
- 3 You switch to the French language.
- 4 You open the custom property dialog box and see the string "English String."
- 5 You can now enter, for example, "Overridden French String" string for French in the CP1 custom property. You cannot enter a reference in the French language.
- 6 If you switch to English Language, you see "Overridden French String" as the default overridden value.

## Translation Support for Client Controls with Satellite Assemblies

A satellite assembly is a .NET Framework assembly that contains resources specific to a given language. You can place the resources for different languages in different assemblies, and the correct assembly is loaded into memory if the run-time user views the application in that language.

If you import a client control with satellite assemblies, then the satellite assemblies are also imported.

The translation support for client controls with satellite assemblies is as follows:

Scenario	Behavior of Client Control
Design Time - Symbol Editor	The ArcestrA Symbol Editor only supports English. The application locale has no effect on how the client controls are shown.
Design Time - WindowMaker	The localized version of the InTouch HMI being used determines how the client controls are shown.
Run Time - WindowViewer	The localized version of the InTouch HMI being used determines how the client controls are shown. The application locale has no effect on how the client controls are shown.

When you export text for translation, the base font information for client controls is not included. You need to use satellite assemblies to control this information.

## Translation Support for ArcestrA Client Controls

ArcestrA client controls, such as the ArcestrA Alarm Control, support the following types of translation:

- ArcestrA client controls with satellite assemblies work similar to other types of client controls.
- ArcestrA client controls support translated string substitutions. For more information on translated string substitution, see *Switching Languages and String Substitutions at Run Time* on page 439.

The number and type of strings used by the ArcestrA client control is specific to the control.

When you export text for translation, the base font information for ArcestrA client controls is not included. You need to use satellite assemblies to control this information.

## Importing InTouch SmartSymbols that Have Translated Data

You can import InTouch SmartSymbols having translated data into the Symbol Editor. All the language data in a SmartSymbol is imported into the ArcestrA symbol, including:

- Configured languages.
- Translated content for graphic elements (text and font)
- Translated animation links.

After you import, you can view the language data in the ArcestrA Symbol Editor at design time or in InTouch WindowViewer at run time.

For example, you import a SmartSymbol with language data for French and German. After the import, the French and German languages are added to the ArcestrA Symbol, if they do not already exist.

The language switching behavior at run time matches the behavior of the original SmartSymbol. Any variation in the default font usage at run time between ArcestrA Symbols and native InTouch graphics is resolved during the SmartSymbol import. During the import, elements that have translations but not a translated font are detected and their font set to the Galaxy-configured font for that language.

## Support for Empty Strings

You cannot substitute an empty string in the primary locale. You must use space characters. When you set an empty string for a primary locale, the empty string is propagated to all other locales that do not have translations.

Performing a first time substitution of an empty string in a secondary locale puts a space character in the primary and the current locale. The remaining locales will match the primary value if they do not already have a specific value.

If a primary locale contains an empty string, it will be exported for translation.

If you substitute an empty string for a secondary locale, the element shows as empty. However, if you switch to the primary locale and then back to the secondary locale, the element shows the primary string substitution again.

## Language Switching Example

The following table describes the effects of language switching on the various parts of the system. In this example:

- The Galaxy is configured with two languages: English and French.
- The default language in the Galaxy is English.
- There are various symbols configured in the Galaxy, some of which contain partial or mismatched language configurations compared to the Galaxy's configured languages.

Action	Effect on the Languages Configured for the Symbol	Effect on Elements that contain English Translations	Effect on Elements that contain French Translations
You open a new symbol.	English language is added to the symbol.	None	None
You open an existing symbol that only has English defined.	None	None	None
You open an existing symbol that only has the French language defined.	English language is added to the symbol.	French strings are transferred into the English language.	French strings are marked as specific translations for French.
You open an existing symbol that only has the German language defined.	English language is added to the symbol.	German strings are transferred into the English language. German strings are marked as specific translations for German.	None
You change a text string in the English language.	None	New string is set for the English language.	None
You switch to the French language for the first time in a symbol that only had the English language.	French language is added to the symbol.	None	The default language strings are shown unless a specific French translation exists.



Action	Effect on the Languages Configured for the Symbol	Effect on Elements that contain English Translations	Effect on Elements that contain French Translations
You change a string while viewing the French language.	None	None	The new string is set as the specific translation for the French language and used for display.
You create a new text element in a symbol that has English only while viewing the English language.	None	The new element's string value is saved for translation.	None
You delete an element with translations in English and French.	None	English translations are removed.	French translations are removed.
You copy animations from an element.	None	English animation translation strings are put into the clipboard.	French animation translation strings are put into the clipboard.
You paste animations to an element.	None	English animation translation strings are put into the destination animations.	French animation translation strings are put into the destination animations.
You clear animations for an element.	None	English animation strings are removed.	French animation strings are removed.
You copy and paste elements from a German-only symbol into a symbol containing English and French.	None	German strings are placed in the English language.	None
You copy and paste elements from an English and German symbol into a symbol containing English and French.	None	English strings from the source symbol are copied during the paste. The German strings are dropped.	None

Action	Effect on the Languages Configured for the Symbol	Effect on Elements that contain English Translations	Effect on Elements that contain French Translations
You export the French language, having never switched to the French language.	The French language is added to the symbol for the purposes of the export. The language is not saved back to the symbol during the export.	English strings are exported in the "Phrase" XML attribute field.	If specific strings exist for French, they are exported in the "Translation" XML field.
You import the French language.	The French language is added.	None	Any translations provided in the import are marked as specific translations for the French language. If the translation is empty, the default language value is shown.
You convert a symbol to a group.	Same logic as copy/paste.	Same logic as copy/paste.	Same logic as copy/paste.
You delete the German language.	German is removed.	None	None

## Overriding Translated Strings for ArcestrA Symbols in WindowMaker

After you embed an ArcestrA symbol into an InTouch window, you can override the translations for:

- Strings on the substitutable graphic elements within the ArcestrA symbol.
- String type custom properties on the ArcestrA symbol.

To translate these overrides, you must export and import the strings for the managed InTouch application.

### Overriding Translated String Substitutions

After you export the managed InTouch application using the IDE, all the string substitution overrides are exported into an InTouchViewApp dictionary file.

After you import the translated InTouchViewApp dictionary file, you can view the translated string overrides in InTouch WindowViewer at run time.

### Overriding Translated Custom Properties

After you export the managed InTouch application using the IDE, all the custom property overrides are exported into an InTouch ViewApp dictionary file.

After you import the translated InTouchViewApp dictionary file, you can view the translated custom properties in InTouch WindowViewer at run time.

## Language Switching at Run Time

At run time, languages can be switched in the following ways:

- Using the InTouch script function `SwitchDisplayLanguage()` from an InTouch script only.
- Setting the value of the `$Language` system tag within an InTouch or ArcestrA script.
- Selecting the language from the **Special** menu. For managed InTouch applications, the list of languages shown is based on the languages configured in the Galaxy.

## How Languages are Shown in WindowViewer

At run time, ArcestrA symbols are shown in the language set for InTouch WindowViewer.

If the ArcestrA symbol's language does not match the WindowViewer language setting, the language shown is the language that was the default language for the symbol when it was last saved.

For example, an InTouch Window has three different embedded ArcestrA symbols: S1, S2, and S3.

- S1 has a Text graphic with French and German languages configured.
- S2 has Textbox graphic with only German language configured.
- S3 has a Button graphic with English and Spanish languages configured. S3 has two client scripts. The first script switches the language to Chinese, and the second script switches the language to Spanish.

English is the default language. The following steps show how the translated symbols are shown:

- 1 Set the language to French in the WindowViewer.  
The Text in S1 shows the "French String."  
The Textbox in S2 shows the "German String."  
The Button in S3 shows the "English String."
- 2 Run the first client script on symbol S3.  
The Button in S3 still shows "English String."
- 3 Run the second client script on symbol S3.  
The Button in S3 shows "Spanish String."
- 4 Set the language to German in the WindowViewer.  
The Text in S1 shows the "German String."  
The Textbox in S2 still shows the "German String."  
The Button in S3 shows the "English String."

## Precedence Rules for Showing the Language and Font

All elements and animations that can be translated follow these precedence rules for showing the translated text:

- 1 The default language holds the master language text for the translation.
- 2 The secondary languages use the default language's text if no specific translation exists for the secondary language.
- 3 If a specific translation value exists for a secondary language, that text is shown instead of the default language's text.

The precedence rules for which font to use for a graphic element that supports visual text are as follows:

- 1 If there is no text translation for an element, the element shows the text in the default language.
- 2 If there is a specific translation in the secondary language and no translated font specified for the element, the element shows text using the font specified in the Galaxy for that language.
- 3 If there is a specific translation in the secondary language and a specific font set for the element, the element shows the text using the specific font.

For example:

- 1 You create a TextBox element on a symbol with the text "English String." The Galaxy-configured languages are English with Arial font and German with Tahoma font.
- 2 You switch over to the German Language. The TextBox element still shows "English String" with the Arial font.
- 3 You change "English String" to "German String." Because there is now a translation for this string, the TextBox shows "German String" with the Tahoma font.

## Default Language Fonts at Run Time

InTouch WindowViewer uses the default font for a given language if an element has a translation but no translated font specified. For ArcestrA symbols, the font element of the default Galaxy language is used.

## Switching Languages for Custom Properties at Run Time

Embedded ArcestrA symbols and ArcestrA symbols embedded in the InTouch HMI can contain translated custom properties.

When you switch to a language for the first time, the translated overrides of the custom properties are loaded from the dictionary file of the ArcestrA symbol's overrides.

After switching to a new language in the InTouch WindowViewer, the translated custom properties are updated based on the switched language. All the animations/scripts that subscribe to a custom property are notified of the value change and the animation/script values are updated accordingly.

For example:

- Symbol S1 has a constant string custom property CP1.
- S1 has two languages configured: English and French. English is the default language. The English is "English CP1."
- CP1 has the French translation, "French CP1." S1 has two graphic elements: Textbox and Button.
- Each graphic has a value display animation that subscribes to CP1.
- An InTouch Window has two embedded symbols of S1.
- The second instance of S1 has the translated CP1 override in French ("French Override CP1") and German ("German Override CP1").

The following steps describe how the translated custom properties are shown.

- 1 Set the language to French in WindowViewer.  
The value display animation of Textbox and Button in the first instance of S1 shows "French CP1."  
The value display animation of Textbox and Button in the second instance of S1 shows "French Override CP1."
- 2 Set the language to German in the WindowViewer.  
The value display animation of Textbox and Button in the first instance of S1 shows "English CP1."  
The value display animation of Textbox and Button in the second instance of S1 shows "German Override CP1."

## Switching Languages and String Substitutions at Run Time

An embedded ArcestrA symbol and ArcestrA symbol embedded into InTouch can contain translated string substitutions. All the graphic elements that contain the translated string substitutions are updated based on the current language setting.

For example:

- Symbol S1 has two graphic elements: Textbox and Button. S1 has two languages configured: English and French. The strings are “English String” and “French String.” English is the default language.
- The Textbox element has a French string substitution, “French Sub String.” The Button element has an English substitution, “English Sub String.”
- An InTouch window has two embedded symbols of S1. The second instance of S1 has the translated string substitution overrides for Textbox and Button in French (“French Override Sub String”) and German (“German Override Sub String”).

The following steps describe how the translated string substitutions are shown.

- 1 Set the language to French in WindowViewer.
  - The text of Textbox and Button in the first instance of S1 shows "French Sub String" and "English Sub String," respectively.
  - The Text of Textbox and Button in the second instance of S1 shows "French Override String."
- 2 Set the language to German in the WindowViewer.
  - The Text of Textbox and Button in the first instance of S1 shows "English String" and "English Sub String," respectively.
  - The Text of Textbox and Button in the second instance of S1 shows "German Override String."

## Language Settings for Popup Symbols

At run time, all popup symbols are shown using the current language setting for WindowViewer. The popup symbol always tries to show the current language, even if the symbol opening the popup does not support the language.

For example, a symbol S1 is configured with French and German languages. The following steps show how the current language setting updates:

- 1 Switch the language from French to German in WindowViewer.  
The current language setting is to German.
- 2 Open a popup symbol S2 from the symbol S1.  
The symbol S2 is shown using the German language. If symbol S2 does not have German language configured, then it is shown using its last saved default language.
- 3 Close the popup symbol S2.
- 4 Changes the Language property of the symbol S1 to French. The current language setting of the InTouch WindowViewer is now German, and the language setting of the symbol S1 is French.
- 5 Open a popup symbol S2 from the symbol S1.  
The symbol S2 is shown using the French language. If symbol S2 does not have French language configured then it is shown using its last saved default language.

### Dynamic Propagation of Language Changes to Popup Symbols

When you change the language setting of WindowViewer, the change dynamically propagates to all popup symbols. This includes both modal and modeless popup symbols.

For example, symbol S1 is configured with French and German languages. WindowViewer is set to show the French language. The following steps show how the language is updated for the popup symbols:

- 1 You open a modeless popup symbol S2 from the symbol S1.
- 2 You open a modeless popup symbol S3 from the symbol S2.



- 3 You open a modeless popup symbol S4 from the symbol S1.

The symbols S1, S2, S3, and S4 are shown using the French language.

- 4 You change the language to German.

The symbols S1, S2, S3 and S4 now show the German language. If the popup symbol does not support the German language, it is shown using the last saved default language of that popup symbol.

## Language Settings and Data Types

All data and symbols are shown by ArchestrA as English formatted data types, regardless of the language being shown, the operating system on which the InTouch HMI is installed, or the translated version of the InTouch HMI you are using.



# Appendix A

## List of Element Properties

This section shows you the properties of elements, the canvas, element groups, and embedded symbols.

Each property has a purpose, a category it belongs to, where it is used if it can be used in scripting at run time, and where to find more information on how to use it.

The first part of this section contains an alphabetical list of all properties, the second part shows a table for each category of properties.

## Alphabetical List of Properties

The following table contains a list of properties used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

Asterisk (\*) marks properties that are specific to only one type of element or the canvas, a group or an embedded symbol.

Property	Purpose, category, usage and further information
AbsoluteAnchor*	<p><b>Purpose:</b> Defines the absolute anchor point of the source symbol. By default, this is the center point of all elements on the canvas but can be changed.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>
AbsoluteOrigin	<p><b>Purpose:</b> Defines an X, Y location relative to the top, left (0, 0) origin of the symbol or window.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Embedded Symbol, Group, Path, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box.</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Points of Origin in the Properties Editor on page 130</p>
Alignment	<p><b>Purpose:</b> Controls the location of the text relative to the bounding rectangle of the element.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Button, Text, Text Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Alignment on page 162</p>

Property	Purpose, category, usage and further information
AnchorFixedTo	<p><b>Purpose:</b> Determines if the anchor point is fixed to the canvas when you resize, delete, or add elements (Absolute), or if the anchor point is recalculated relative to the element sizes and positions (Relative).</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Embedded Symbol, Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>
AnchorPoint*	<p><b>Purpose:</b> Defines the anchor X, Y location of the embedded symbol.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>
Angle	<p><b>Purpose:</b> Defines the current angle of rotation of the element. 0 is always the top of the element relative to the canvas. Angle is always determined relative to the top of the element and rotates in a clockwise direction.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Group, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Rotating Elements by Changing the Angle Property on page 127</p>
AutoScale	<p><b>Purpose:</b> If this property is set to True then the text is stretched horizontally and vertically (larger or smaller) to fit the bounding rectangle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting Auto Scaling and Word Wrapping for a Text Box on page 196</p>

Property	Purpose, category, usage and further information
ButtonStyle*	<p><b>Purpose:</b> Determines if the button appears as a standard button or as an image.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring Buttons with Images on page 203</p>
CalendarColumns*	<p><b>Purpose:</b> Defines the number of columns the calendar object has.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Number of Calendar Month Sheets on page 230</p>
CalendarRows*	<p><b>Purpose:</b> Defines the number of rows the calendar object has.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Number of Calendar Month Sheets on page 230</p>
Caption*	<p><b>Purpose:</b> Defines the text shown on the Check Box at design time and at run time when the caption property is not bound to a reference in the checkbox animation panel.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Check Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Caption Text of a Check Box Control on page 224</p>
Checked*	<p><b>Purpose:</b> Sets or gets the value of check box. This is the initial value of the check box when the control is not connected to a reference and is overridden at run time with value of reference.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Check Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Default State of a Check Box Control on page 224</p>

Property	Purpose, category, usage and further information
.Color1	<p><b>Purpose:</b> Color1 is a sub-property of a FillColor, UnfilledColor, LineColor or TextColor property. It is used to change the first color of the fill, unfill, line or text style if applicable.</p> <p><b>Category:</b> Depends on its source property</p> <p><b>Used by:</b> Depends on its source property</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Enabling and Disabling Elements for Run-Time Interaction on page 187</p>
.Color2	<p><b>Purpose:</b> Color2 is a sub-property of a FillColor, UnfilledColor, LineColor or TextColor property. It is used to change the second color of the fill, unfill, line or text style if applicable.</p> <p><b>Category:</b> Depends on its source property</p> <p><b>Used by:</b> Depends on its source property</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Enabling and Disabling Elements for Run-Time Interaction on page 187</p>
.Color3	<p><b>Purpose:</b> Color3 is a sub-property of a FillColor, UnfilledColor, LineColor or TextColor property. It is used to change the third color of the fill, unfill, line or text style if applicable.</p> <p><b>Category:</b> Depends on its source property</p> <p><b>Used by:</b> Depends on its source property</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Enabling and Disabling Elements for Run-Time Interaction on page 187</p>
ControlStyle	<p><b>Purpose:</b> Defines the control style as Flat or 3D.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Radio Button Group, Check Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Layout of the Radio Button Group Options on page 223 and Setting the 3D appearance of a Check Box Control on page 225</p>

Property	Purpose, category, usage and further information
Count	<p><b>Purpose:</b> Indicates how many items there are in a list.</p> <p><b>Category:</b> not available at design time</p> <p><b>Used by:</b> Radio Button Group, Combo Box, List Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Using Radio Button Group-Specific Properties at Run Time on page 223, Using Combo Box-Specific Properties at Run Time on page 229 and Using List Box-Specific Properties at Run Time on page 237</p>
CustomFormat*	<p><b>Purpose:</b> Defines the format to be used in the DateTime Picker control for input of a date or time.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> DateTime Picker</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring DateTime Picker Controls on page 234</p>
CustomProperties	<p><b>Purpose:</b> The collection of CustomProperties defined by the symbol.</p> <p><b>Category:</b> Custom Properties</p> <p><b>Used by:</b> Canvas, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Using Custom Properties on page 239</p>
Description*	<p><b>Purpose:</b> Contains a meaningful description of the symbol.</p> <p><b>Category:</b> Graphic</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
DefaultValue	<p><b>Purpose:</b> The default time value to use for the control.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar, DateTime Picker</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Default Value of the Calendar Control on page 233 and Configuring DateTime Picker Controls on page 234</p>



Property	Purpose, category, usage and further information
DownImage*	<p><b>Purpose:</b> Defines the image that is rendered in the button element when it is clicked or held down.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring Buttons with Images on page 203</p>
DropDownType*	<p><b>Purpose:</b> Defines the type of combo box: simple, drop-down or drop-down list.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Type of Combo Box Control on page 227</p>
DropDownWidth*	<p><b>Purpose:</b> Defines the width of the drop-down list.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Width of the Drop-Down List on page 228</p>
DynamicSizeChange*	<p><b>Purpose:</b> Determines if the embedded symbol propagates the size changes from the source symbol.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Enabling or Disabling Dynamic Size Change of Embedded Symbols on page 401</p>

Property	Purpose, category, usage and further information
Enabled	<p><b>Purpose:</b> When set to True enables the element at run time and allows the user to interact with it. If the property is set to False the user cannot use the mouse or keyboard to interact with the element. Data changes as a result of an animation or script still execute.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Enabling and Disabling Elements for Run-Time Interaction on page 187</p>
End	<p><b>Purpose:</b> Defines the end of a line or H/V line as X, Y location.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Line, H/V Line</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Start or End Points of a Line on page 157</p>
EndCap	<p><b>Purpose:</b> Defines the cap used at the end of the line of an open element.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Line End Shape and Size on page 195</p>
FillBehavior	<p><b>Purpose:</b> Determines how the Fill (Horizontal, Vertical or Both) should be applied to the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Fill Behavior on page 155</p>

Property	Purpose, category, usage and further information
FillColor	<p><b>Purpose:</b> Defines the fill style used for the filled portion of the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, List Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Fill Style on page 153 and Changing Background Color and Text Color of Windows Common Controls on page 220</p>
FillOrientation	<p><b>Purpose:</b> Determines the orientation of the fill when the element orientation is any value other than 0.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Fill Orientation on page 155</p>
FirstDayOfWeek*	<p><b>Purpose:</b> Defines the first day of the week used for the display of the columns in the calendar.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the First Day of the Week on page 231</p>
Font	<p><b>Purpose:</b> Defines the basic text font as defined by the operating system.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Button, Text, Text Box, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Font on page 161</p>

Property	Purpose, category, usage and further information
Format*	<p><b>Purpose:</b> Defines the format of the reference values. This is only available for array mode.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> DateTime Picker</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring DateTime Picker Controls on page 234</p>
HasTransparentColor*	<p><b>Purpose:</b> Indicates whether or not the image applies a transparent color. If True the image is rendered transparent wherever a color in the image matches the TransparentColor property.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Image Color Transparency on page 200</p>
Height	<p><b>Purpose:</b> Defines the height of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Resizing Elements by Changing Size Properties on page 122</p>
HorizontalDirection	<p><b>Purpose:</b> Determines the horizontal direction of the fill for the element. Can be “Right” or “Left”.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Horizontal Fill Direction and Percentage on page 156</p>

Property	Purpose, category, usage and further information
HorizontalPercentFill	<p><b>Purpose:</b> Determines the percentage of horizontal fill for the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting Horizontal Fill Direction and Percentage on page 156</p>
HorizontalScrollbar	<p><b>Purpose:</b> Determines if a horizontal scroll bar appears on a list box control to allow the user to scroll the list box items horizontally.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> List Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Using a Horizontal Scroll Bar in a List Box Control on page 236</p>
Image*	<p><b>Purpose:</b> Defines the image that is rendered in the element. Any image format supported by the application can be used.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Selecting a Different Image on page 202</p>
ImageAlignment*	<p><b>Purpose:</b> Controls the location of the image relative to the bounding rectangle of the graphic. This property is only applicable when the ImageStyle is set to Normal.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Image Alignment on page 199</p>
ImageStyle	<p><b>Purpose:</b> Defines how the image is rendered relative to its bounding rectangle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Image Display Mode on page 198</p>

Property	Purpose, category, usage and further information
IntegralHeight	<p><b>Purpose:</b> Determines if the List Box size is an integral multiple of the Font Size so that a finite number of items fit in it without being clipped.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Avoiding Clipping of Items in the Simple Combo Box Control on page 228</p>
Language	<p><b>Purpose:</b> Defines the current language of the symbol.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Selecting the Language for a Symbol on page 422.</p>
LanguageID	<p><b>Purpose:</b> Defines the current language ID of the symbol.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Selecting the Language for a Symbol on page 422.</p>
Layout*	<p><b>Purpose:</b> Defines the way the radio buttons are arranged in the group (Horizontal or Vertical).</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Radio Button Group</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Layout of the Radio Button Group Options on page 223</p>
LineColor	<p><b>Purpose:</b> Defines the color and affects of the line or border.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Line Style on page 159</p>

Property	Purpose, category, usage and further information
LinePattern	<p><b>Purpose:</b> Defines the pattern of the line or border.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Line Pattern on page 158</p>
LineWeight	<p><b>Purpose:</b> Determines the weight of the element's line or border. A value of 0 means that there is no line or border.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Text Box, Path</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Line Weight on page 158</p>
Locked	<p><b>Purpose:</b> Locks or unlocks the element's size, position, orientation and origin. Other properties that can have an affect on element size, position, orientation and origin are also locked. These are element-specific.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Locking and Unlocking Elements on page 132</p>
MaxDropDownItems*	<p><b>Purpose:</b> Defines the maximum number of items the drop-down list shows.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Maximum Number of Items to Appear in the Combo Box Drop-Down List on page 229</p>

Property	Purpose, category, usage and further information
Multiline*	<p><b>Purpose:</b> Determines if the control shows several lines of text that automatically wrap up when reaching the right border of the control.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Edit Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring the Text to Wrap in an Edit Box Control on page 226</p>
MultiplePopupsAllowed*	<p><b>Purpose:</b> If False, ShowSymbol animations only show within a single dialog window no matter how many animations are invoked and regardless of how the animations are configured. If True, ShowSymbol animations show in separate dialog windows.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
Name	<p><b>Purpose:</b> Gives the element a meaningful unique name.</p> <p><b>Category:</b> Graphic</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Embedded Symbol, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
NewIndex	<p><b>Purpose:</b> Returns the index of the last value added to the list. This is provided for migration of InTouch Windows common controls.</p> <p><b>Category:</b> not available at design time</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Using Combo Box-Specific Properties at Run Time on page 229 and Using List Box-Specific Properties at Run Time on page 237</p>



Property	Purpose, category, usage and further information
OwningObject*	<p><b>Purpose:</b> Used as the ArchestrA object reference to replace all "Me." references in expressions and scripts. Everywhere there is a "Me." reference this object reference is used instead. The object name can be set either using a tag or hierarchical name of an AutomationObject.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Detecting and Editing the Containing AutomationObject Instance on page 404</p>
Radius*	<p><b>Purpose:</b> Defines the radius of the corners of the Rounded Rectangle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rounded Rectangle</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
ReadOnly*	<p><b>Purpose:</b> Determines if the user can type data into the edit box.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Edit Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring the Text to be Read-Only in an Edit Box Control on page 226</p>
RelativeAnchor*	<p><b>Purpose:</b> Relative anchor point of the source symbol. By default, this is 0,0.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>

Property	Purpose, category, usage and further information
RelativeOrigin	<p><b>Purpose:</b> Defines the relative origin as X, Y location. The location is relative to the center point of the element (0, 0).</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Points of Origin in the Properties Editor on page 130</p>
Scripts*	<p><b>Purpose:</b> Defines a collection of scripts configured for the symbol.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Adding and Maintaining Symbol Scripts on page 353</p>
SelectedValue	<p><b>Purpose:</b> Reads the value of the selected item, or selects the item with that value if it exists.</p> <p><b>Category:</b> not available at design time</p> <p><b>Used by:</b> Radio Button Group, List Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Using Radio Button Group-Specific Properties at Run Time on page 223 and Using List Box-Specific Properties at Run Time on page 237</p>
ShowToday*	<p><b>Purpose:</b> Determines if today's date is shown on the calendar control.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Showing or Hiding Today's Date on a Calendar Control on page 232</p>

Property	Purpose, category, usage and further information
Smoothing*	<p><b>Purpose:</b> When False the graphics are rendered normally, when True graphics are rendered with anti-aliasing which produces a smoother appearing graphic.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
Start	<p><b>Purpose:</b> Defines the start of a line or H/V line as X, Y location.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Line, H/V Line</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Start or End Points of a Line on page 157</p>
StartAngle	<p><b>Purpose:</b> Defines the starting angle of an Arc, Pie or Chord. 0 is always the top of the graphic relative to its orientation. A positive number is clockwise from 0 and a negative number is counter clockwise from 0. If a negative number is used to set the property it is automatically converted to a positive value.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Angles of Arcs, Pies and Chords on page 207</p>
StartCap	<p><b>Purpose:</b> Defines the cap used at the start of the line of an open graphic.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Line End Shape and Size on page 195</p>

Property	Purpose, category, usage and further information
SweepAngle	<p><b>Purpose:</b> Defines the ending angle of the Arc, Pie or Chord. This angle is always measured from the start angle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Angles of Arcs, Pies and Chords on page 207</p>
SymbolReference*	<p><b>Purpose:</b> Contains the exact location that the Embedded Symbol is linked to. This can help the user in locating the original definition for editing purposes.</p> <hr/> <p><b>Note</b> This property is always disabled.</p> <hr/> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Detecting the Source Symbol of an Embedded Symbol on page 398</p>
TabOrder	<p><b>Purpose:</b> Defines the tab order for the element. The tab order is only used when navigating by the keyboard. This property is valid only when the TabStop property is set to true.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Editing the Tab Order of an Element on page 188</p>

Property	Purpose, category, usage and further information
TabStop	<p><b>Purpose:</b> Determines if the element can be navigated to and can receive focus at run time.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Editing the Tab Order of an Element on page 188</p>
Tension	<p><b>Purpose:</b> Specifies how tightly the curve bends through the control points of the curve.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Closed Curve, Curve</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Changing the Tension of Curves and Closed Curves on page 206</p>
Text	<p><b>Purpose:</b> Defines the unicode text that is shown by the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text, Text Box, Edit Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Displayed Text on page 160</p>
TextColor	<p><b>Purpose:</b> Defines the color and affects applied to the text.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Button, Text, Text Box, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Color on page 161 and Changing Background Color and Text Color of Windows Common Controls on page 220</p>

Property	Purpose, category, usage and further information
TextFormat	<p><b>Purpose:</b> Defines the formatting string that is applied to the text when it is shown.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text, Text Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Text Display Format on page 160</p>
TitleFillColor*	<p><b>Purpose:</b> Determines the background solid color in the title bar of the calendar control.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Title Fill Color and Text Color on a Calendar Control on page 232</p>
TitleTextColor*	<p><b>Purpose:</b> Determines the text solid color in the title bar of the calendar control.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Title Fill Color and Text Color on a Calendar Control on page 232</p>
TopIndex*	<p><b>Purpose:</b> Returns the index of the top most item in the list. This is provided for migration of InTouch Windows common controls.</p> <p><b>Category:</b> not available at design time</p> <p><b>Used by:</b> List Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Using List Box-Specific Properties at Run Time on page 237</p>
TrailingTextColor*	<p><b>Purpose:</b> Determines the text solid color of the text for the trailing days. The trailing days are days outside the current month.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Color for Trailing Dates in a Calendar Control on page 233</p>

Property	Purpose, category, usage and further information
Transparency	<p><b>Purpose:</b> Defines the transparency of the element. A value of 0 means fully opaque and a value of 100 means fully transparent.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Transparency Level of an Element on page 185</p>
.Transparency	<p><b>Purpose:</b> Transparency is a sub-property of a FillColor, UnfilledColor, LineColor or TextColor property. It is used to change the transparency of the fill, unfill, line or text style if applicable. The transparency acts in addition to the transparency of the element.</p> <p><b>Category:</b> Depends on its source property</p> <p><b>Used by:</b> Depends on its source property</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Enabling and Disabling Elements for Run-Time Interaction on page 187</p>
TransparentColor*	<p><b>Purpose:</b> Defines the RGB color value that is used as the transparent color.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Image Color Transparency on page 200</p>

Property	Purpose, category, usage and further information
TreatAsIcon	<p><b>Purpose:</b> If this property is set to False, the animations defined on the graphics within the group or embedded symbol take precedence over an animation defined on the group or embedded symbol. If there are no animations or the user clicked on an area of the group or embedded symbol that does not have an animation, then the group or embedded symbol animation executes.</p> <p>If the property is set to True, only the animation on the group or embedded symbol is executed. The interactive animations within the group or embedded symbol never execute.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Group, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Editing the Embedded Symbol on page 395</p>
UnFilledColor	<p><b>Purpose:</b> Determines the element's unfilled area appearance.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Unfilled Style on page 154</p>
UpImage*	<p><b>Purpose:</b> Defines the image that is used in the button element when it is un-clicked or released.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring Buttons with Images on page 203</p>
Value	<p><b>Purpose:</b> Reads the value of the selected item, or selects the item with that value if it exists. Its data type depends on the control.</p> <p><b>Category:</b> not available at design time</p> <p><b>Used by:</b> Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Reading and Writing the Selected Value at Run Time on page 221</p>



Property	Purpose, category, usage and further information
VerticalDirection	<p><b>Purpose:</b> Defines the vertical direction of the fill. Can be “Top” or “Bottom”.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Vertical Fill Direction and Percentage on page 156</p>
VerticalPercentFill	<p><b>Purpose:</b> Determines the percentage of vertical fill for the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting Vertical Fill Direction and Percentage on page 156</p>
Visible	<p><b>Purpose:</b> Determines the visibility of the element. This property is configured at design time and used only at runtime. At design time all elements are visible irrespective of this setting.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Changing the Visibility of Elements on page 187</p>

Property	Purpose, category, usage and further information
Width	<p><b>Purpose:</b> Defines the width of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Resizing Elements by Changing Size Properties on page 122</p>
WordWrap	<p><b>Purpose:</b> When set to True, the text in the button or text box is formatted to fit as much text on a single line within the horizontal bounding area of the element and then continued to the next line. This continues as long as there is vertical space.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Wrapping Text in Buttons on page 203</p>
X	<p><b>Purpose:</b> Defines the left position of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Moving Elements on page 109</p>

Property	Purpose, category, usage and further information
Y	<p><b>Purpose:</b> Defines the top position of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Moving Elements on page 109</p>

## List by Functional Area

Each property of the elements, the canvas, element groups and embedded objects belongs to one of the following property categories:

- Graphic
- Appearance
- Fill Style
- Line Style
- Text Style
- Runtime Behavior
- Custom Properties

## Graphic Category Properties

The following table contains a list of properties in the Graphic property category used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

It shows their purpose, where they are used and where to find more information on how to use them.

Asterisk (\*) marks properties that are specific to only one type of element or the canvas, a group or an embedded symbol.

Property	Purpose, category, usage and further information
Description*	<p><b>Purpose:</b> Contains a meaningful description of the symbol.</p> <p><b>Category:</b> Graphic</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
Name	<p><b>Purpose:</b> Gives the element a meaningful unique name.</p> <p><b>Category:</b> Graphic</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Embedded Symbol, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>

## Appearance Category Properties

The following table contains a list of properties in the Appearance property category used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

It shows their purpose, where they are used and where to find more information on how to use them.

Asterisk (\*) marks properties that are specific to only one type of element or the canvas, a group or an embedded symbol.

Property	Purpose, category, usage and further information
AbsoluteAnchor*	<p><b>Purpose:</b> Defines the absolute anchor point of the source symbol. By default, this is the center point of all elements on the canvas but can be changed.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>
AnchorFixedTo	<p><b>Purpose:</b> Determines if the anchor point is fixed to the canvas when you resize, delete, or add elements (Absolute), or if the anchor point is recalculated relative to the element sizes and positions (Relative).</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Embedded Symbol, Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>

Property	Purpose, category, usage and further information
AbsoluteOrigin	<p><b>Purpose:</b> Defines an X, Y location relative to the top, left (0, 0) origin of the symbol or window.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Embedded Symbol, Group, Path, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box.</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Points of Origin in the Properties Editor on page 130</p>
AnchorPoint*	<p><b>Purpose:</b> Defines the anchor X, Y location of the embedded symbol.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>
Angle	<p><b>Purpose:</b> Defines the current angle of rotation of the element. 0 is always the top of the element relative to the canvas. Angle is always determined relative to the top of the element and rotates in a clockwise direction.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Group, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Rotating Elements by Changing the Angle Property on page 127</p>
AutoScale	<p><b>Purpose:</b> If this property is set to True then the text is stretched horizontally and vertically (larger or smaller) to fit the bounding rectangle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting Auto Scaling and Word Wrapping for a Text Box on page 196</p>

Property	Purpose, category, usage and further information
ButtonStyle*	<p><b>Purpose:</b> Determines if the button appears as a standard button or as an image.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring Buttons with Images on page 203</p>
CalendarColumns*	<p><b>Purpose:</b> Defines the number of columns the calendar object has.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Number of Calendar Month Sheets on page 230</p>
CalendarRows*	<p><b>Purpose:</b> Defines the number of rows the calendar object has.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Number of Calendar Month Sheets on page 230</p>
Checked*	<p><b>Purpose:</b> Sets or gets the value of check box. This is the initial value of the check box when the control is not connected to a reference and is overridden at run time with value of reference.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Check Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Default State of a Check Box Control on page 224</p>
ControlStyle	<p><b>Purpose:</b> Defines the control style as Flat or 3D.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Radio Button Group, Check Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Layout of the Radio Button Group Options on page 223 and Setting the 3D appearance of a Check Box Control on page 225</p>

Property	Purpose, category, usage and further information
CustomFormat*	<p><b>Purpose:</b> Defines the format to be used in the DateTime Picker control for input of a date or time.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> DateTime Picker</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring DateTime Picker Controls on page 234</p>
DefaultValue	<p><b>Purpose:</b> The default time value to use for the control.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar, DateTime Picker</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Default Value of the Calendar Control on page 233 and Configuring DateTime Picker Controls on page 234</p>
DownImage*	<p><b>Purpose:</b> Defines the image that is rendered in the button element when it is clicked or held down.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring Buttons with Images on page 203</p>
DropDownType*	<p><b>Purpose:</b> Defines the type of combo box: simple, drop-down or drop-down list.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Type of Combo Box Control on page 227</p>
DropDownWidth*	<p><b>Purpose:</b> Defines the width of the drop-down list.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Width of the Drop-Down List on page 228</p>



Property	Purpose, category, usage and further information
DynamicSizeChange*	<p><b>Purpose:</b> Determines if the embedded symbol propagates the size changes from the source symbol.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Enabling or Disabling Dynamic Size Change of Embedded Symbols on page 401</p>
End	<p><b>Purpose:</b> Defines the end of a line or H/V line as X, Y location.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Line, H/V Line</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Start or End Points of a Line on page 157</p>
FirstDayOfWeek*	<p><b>Purpose:</b> Defines the first day of the week used for the display of the columns in the calendar.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the First Day of the Week on page 231</p>
Format*	<p><b>Purpose:</b> Defines the format of the reference values. This is only available for array mode.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> DateTime Picker</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring DateTime Picker Controls on page 234</p>
HasTransparentColor*	<p><b>Purpose:</b> Indicates whether or not the image applies a transparent color. If True the image is rendered transparent wherever a color in the image matches the TransparentColor property.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Image Color Transparency on page 200</p>

Property	Purpose, category, usage and further information
Height	<p><b>Purpose:</b> Defines the height of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Resizing Elements by Changing Size Properties on page 122</p>
HorizontalScrollbar	<p><b>Purpose:</b> Determines if a horizontal scroll bar appears on a list box control to allow the user to scroll the list box items horizontally.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> List Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Using a Horizontal Scroll Bar in a List Box Control on page 236</p>
Image*	<p><b>Purpose:</b> Defines the image that is rendered in the element. Any image format supported by the application can be used.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Selecting a Different Image on page 202</p>
ImageAlignment*	<p><b>Purpose:</b> Controls the location of the image relative to the bounding rectangle of the graphic. This property is only applicable when the ImageStyle is set to Normal.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Image Alignment on page 199</p>

Property	Purpose, category, usage and further information
ImageStyle	<p><b>Purpose:</b> Defines how the image is rendered relative to its bounding rectangle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Image Display Mode on page 198</p>
IntegralHeight	<p><b>Purpose:</b> Determines if the List Box size is an integral multiple of the Font Size so that a finite number of items fit in it without being clipped.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Avoiding Clipping of Items in the Simple Combo Box Control on page 228</p>
Layout*	<p><b>Purpose:</b> Defines the way the radio buttons are arranged in the group (Horizontal or Vertical).</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Radio Button Group</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Layout of the Radio Button Group Options on page 223</p>
Locked	<p><b>Purpose:</b> Locks or unlocks the element's size, position, orientation and origin. Other properties that can have an affect on element size, position, orientation and origin are also locked. These are element-specific.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Locking and Unlocking Elements on page 132</p>

Property	Purpose, category, usage and further information
MaxDropDownItems*	<p><b>Purpose:</b> Defines the maximum number of items the drop-down list shows.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Combo Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Maximum Number of Items to Appear in the Combo Box Drop-Down List on page 229</p>
Multiline*	<p><b>Purpose:</b> Determines if the control shows several lines of text that automatically wrap up when reaching the right border of the control.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Edit Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring the Text to Wrap in an Edit Box Control on page 226</p>
Radius*	<p><b>Purpose:</b> Defines the radius of the corners of the Rounded Rectangle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rounded Rectangle</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
ReadOnly*	<p><b>Purpose:</b> Determines if the user can type data into the edit box.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Edit Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring the Text to be Read-Only in an Edit Box Control on page 226</p>
RelativeAnchor*	<p><b>Purpose:</b> Relative anchor point of the source symbol. By default, this is 0,0.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Size Propagation and Anchor Points on page 47</p>

Property	Purpose, category, usage and further information
RelativeOrigin	<p><b>Purpose:</b> Defines the relative origin as X, Y location. The location is relative to the center point of the element (0, 0).</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Points of Origin in the Properties Editor on page 130</p>
ShowToday*	<p><b>Purpose:</b> Determines if today's date is shown on the calendar control.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Showing or Hiding Today's Date on a Calendar Control on page 232</p>
Smoothing*	<p><b>Purpose:</b> When False the graphics are rendered normally, when True graphics are rendered with anti-aliasing which produces a smoother appearing graphic.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
Start	<p><b>Purpose:</b> Defines the start of a line or H/V line as X, Y location.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Line, H/V Line</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Start or End Points of a Line on page 157</p>

Property	Purpose, category, usage and further information
StartAngle	<p><b>Purpose:</b> Defines the starting angle of an Arc, Pie or Chord. 0 is always the top of the graphic relative to its orientation. A positive number is clockwise from 0 and a negative number is counter clockwise from 0. If a negative number is used to set the property it is automatically converted to a positive value.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Angles of Arcs, Pies and Chords on page 207</p>
SweepAngle	<p><b>Purpose:</b> Defines the ending angle of the Arc, Pie or Chord. This angle is always measured from the start angle.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Changing Angles of Arcs, Pies and Chords on page 207</p>
Tension	<p><b>Purpose:</b> Specifies how tightly the curve bends through the control points of the curve.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Closed Curve, Curve</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Changing the Tension of Curves and Closed Curves on page 206</p>
Text	<p><b>Purpose:</b> Defines the unicode text that is shown by the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text, Text Box, Edit Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Displayed Text on page 160</p>

Property	Purpose, category, usage and further information
TextFormat	<p><b>Purpose:</b> Defines the formatting string that is applied to the text when it is shown.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text, Text Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Text Display Format on page 160</p>
Transparency	<p><b>Purpose:</b> Defines the transparency. A value of 0 means fully opaque and a value of 100 means fully transparent.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Transparency Level of an Element on page 185</p>
TransparentColor*	<p><b>Purpose:</b> Defines the RGB color value that is used as the transparent color.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Image</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Image Color Transparency on page 200</p>
UpImage*	<p><b>Purpose:</b> Defines the image that is used in the button element when it is un-clicked or released.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Configuring Buttons with Images on page 203</p>

Property	Purpose, category, usage and further information
Width	<p><b>Purpose:</b> Defines the width of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Resizing Elements by Changing Size Properties on page 122</p>
WordWrap	<p><b>Purpose:</b> When set to True, the text in the button or text box is formatted to fit as much text on a single line within the horizontal bounding area of the element and then continued to the next line. This continues as long as there is vertical space.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Button, Text Box</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Wrapping Text in Buttons on page 203</p>
X	<p><b>Purpose:</b> Defines the left position of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Moving Elements on page 109</p>



Property	Purpose, category, usage and further information
Y	<p><b>Purpose:</b> Defines the top position of the element.</p> <p><b>Category:</b> Appearance</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Status, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Moving Elements on page 109</p>

## Fill Style Group Properties

The following table contains a list of properties in the Fill Style property category used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

It shows their purpose, where they are used and where to find more information on how to use them.

Asterisk (\*) marks properties that are specific to only one type of element or the canvas, a group or an embedded symbol.

Property	Purpose, category, usage and further information
FillBehavior	<p><b>Purpose:</b> Determines how the Fill (Horizontal, Vertical or Both) should be applied to the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Fill Behavior on page 155</p>

Property	Purpose, category, usage and further information
FillColor	<p><b>Purpose:</b> Defines the fill style used for the filled portion of the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, List Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Fill Style on page 153 and Changing Background Color and Text Color of Windows Common Controls on page 220</p>
FillOrientation	<p><b>Purpose:</b> Determines the orientation of the fill when the element orientation is any value other than 0.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Fill Orientation on page 155</p>
HorizontalDirection	<p><b>Purpose:</b> Determines the horizontal direction of the fill for the element. Can be “Right” or “Left”.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Horizontal Fill Direction and Percentage on page 156</p>
HorizontalPercentFill	<p><b>Purpose:</b> Determines the percentage of horizontal fill for the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting Horizontal Fill Direction and Percentage on page 156</p>

Property	Purpose, category, usage and further information
TitleFillColor*	<p><b>Purpose:</b> Determines the background solid color in the title bar of the calendar control.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Title Fill Color and Text Color on a Calendar Control on page 232</p>
UnFilledColor	<p><b>Purpose:</b> Determines the element's unfilled area appearance.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Unfilled Style on page 154</p>
VerticalDirection	<p><b>Purpose:</b> Defines the vertical direction of the fill. Can be “Top” or “Bottom”.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Vertical Fill Direction and Percentage on page 156</p>
VerticalPercentFill	<p><b>Purpose:</b> Determines the percentage of vertical fill for the element.</p> <p><b>Category:</b> Fill Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Button, Text Box, Path</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting Vertical Fill Direction and Percentage on page 156</p>

## Line Style Group Properties

The following table contains a list of properties in the Line Style property category used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

It shows their purpose, where they are used and where to find more information on how to use them.

Asterisk (\*) marks properties that are specific to only one type of element or the canvas, a group or an embedded symbol.

Property	Purpose, category, usage and further information
EndCap	<p><b>Purpose:</b> Defines the cap used at the end of the line of an open element.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Line End Shape and Size on page 195</p>
LineColor	<p><b>Purpose:</b> Defines the color and affects of the line or border.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Line Style on page 159</p>
LinePattern	<p><b>Purpose:</b> Defines the pattern of the line or border.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Text Box, Path</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Line Pattern on page 158</p>

---

Property	Purpose, category, usage and further information
LineWeight	<p><b>Purpose:</b> Determines the weight of the element's line or border. A value of 0 means that there is no line or border.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Text Box, Path</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Setting the Line Weight on page 158</p>
StartCap	<p><b>Purpose:</b> Defines the cap used at the start of the line of an open graphic.</p> <p><b>Category:</b> Line Style</p> <p><b>Used by:</b> Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Line End Shape and Size on page 195</p>

---

## Text Style Group Properties

The following table contains a list of properties in the Text Style property category used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

It shows their purpose, where they are used and where to find more information on how to use them.

Asterisk (\*) marks properties that are specific to only one type of element or the canvas, a group or an embedded symbol.

Property	Purpose, category, usage and further information
Alignment	<p><b>Purpose:</b> Controls the location of the text relative to the bounding rectangle of the element.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Button, Text, Text Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Alignment on page 162</p>
Caption*	<p><b>Purpose:</b> Defines the text shown on the Check Box at design time and at run time when the caption property is not bound to a reference in the checkbox animation panel.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Check Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Caption Text of a Check Box Control on page 224</p>
Font	<p><b>Purpose:</b> Defines the basic text font as defined by the operating system.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Button, Text, Text Box, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Font on page 161</p>

---

Property	Purpose, category, usage and further information
TextColor	<p><b>Purpose:</b> Defines the color and affects applied to the text.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Button, Text, Text Box, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Color on page 161 and Changing Background Color and Text Color of Windows Common Controls on page 220</p>
TitleTextColor*	<p><b>Purpose:</b> Determines the text solid color in the title bar of the calendar control.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting Title Fill Color and Text Color on a Calendar Control on page 232</p>
TrailingTextColor*	<p><b>Purpose:</b> Determines the text solid color of the text for the trailing days. The trailing days are days outside the current month.</p> <p><b>Category:</b> Text Style</p> <p><b>Used by:</b> Calendar</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Text Color for Trailing Dates in a Calendar Control on page 233</p>

---

## Runtime Behavior Group Properties

The following table contains a list of properties in the Runtime Behavior property category used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

It shows their purpose, where they are used and where to find more information on how to use them.

Asterisk (\*) marks properties that are specific to only one type of element or the canvas, a group or an embedded symbol.

Property	Purpose, category, usage and further information
Enabled	<p><b>Purpose:</b> When set to True enables the element at run time and allows the user to interact with it. If the property is set to False the user cannot use the mouse or keyboard to interact with the element. Data changes as a result of an animation or script still execute.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Enabling and Disabling Elements for Run-Time Interaction on page 187</p>
Language	<p><b>Purpose:</b> Defines the current language of the symbol.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Selecting the Language for a Symbol on page 422.</p>



Property	Purpose, category, usage and further information
LanguageID	<p><b>Purpose:</b> Defines the current language ID of the symbol.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Selecting the Language for a Symbol on page 422.</p>
MultiplePopupsAllowed*	<p><b>Purpose:</b> If False, ShowSymbol animations only show within a single dialog window no matter how many animations are invoked and regardless of how the animations are configured. If True, ShowSymbol animations show in separate dialog windows.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Setting the Radius of Rounded Rectangles on page 194</p>
OwningObject*	<p><b>Purpose:</b> Used as the ArchestraA object reference to replace all "Me." references in expressions and scripts. Everywhere there is a "Me." reference this object reference is used instead. The object name can be set either using a tag or hierarchical name of an AutomationObject.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Detecting and Editing the Containing AutomationObject Instance on page 404</p>
Scripts*	<p><b>Purpose:</b> Defines a collection of scripts configured for the symbol.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Canvas</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Adding and Maintaining Symbol Scripts on page 353</p>

Property	Purpose, category, usage and further information
SymbolReference*	<p><b>Purpose:</b> Contains the exact location that the Embedded Symbol is linked to. This can help the user in locating the original definition for editing purposes.</p> <hr/> <p><b>Note</b> This property is always disabled.</p> <hr/> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Detecting the Source Symbol of an Embedded Symbol on page 398</p>
TabOrder	<p><b>Purpose:</b> Defines the tab order for the element. The tab order is only used when navigating by the keyboard. This property is valid only when the TabStop property is set to true.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Editing the Tab Order of an Element on page 188</p>
TabStop	<p><b>Purpose:</b> Determines if the element can be navigated to and can receive focus at run time.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> No</p> <p><b>Info:</b> Editing the Tab Order of an Element on page 188</p>

Property	Purpose, category, usage and further information
TreatAsIcon	<p><b>Purpose:</b> If this property is set to False, the animations defined on the graphics within the group or embedded symbol take precedence over an animation defined on the group or embedded symbol. If there are no animations or the user clicked on an area of the group or embedded symbol that does not have an animation, then the group or embedded symbol animation executes.</p> <p>If the property is set to True, only the animation on the group or embedded symbol is executed. The interactive animations within the group or embedded symbol never execute.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Group, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Editing the Embedded Symbol on page 395</p>
Visible	<p><b>Purpose:</b> Determines the visibility of the element. This property is configured at design time and used only at runtime. At design time all elements are visible irrespective of this setting.</p> <p><b>Category:</b> Runtime Behavior</p> <p><b>Used by:</b> Rectangle, Rounded Rectangle, Ellipse, Polygon, Closed Curve, 2 Point Pie, 3 Point Pie, 2 Point Chord, 3 Point Chord, Line, H/V Line, Polyline, Curve, 2 Point Arc, 3 Point Arc, Button, Text, Text Box, Image, Radio Button Group, Check Box, Edit Box, Combo Box, Calendar, DateTime Picker, List Box, Group, Path, Embedded Symbol</p> <p><b>Can be read by script at run time:</b> Yes</p> <p><b>Info:</b> Changing the Visibility of Elements on page 187</p>

## Custom Properties Group Properties

The following table contains a list of properties in the Custom Properties property category used by the:

- Elements.
- Canvas.
- Element groups.
- Embedded symbols.

It shows their purpose, where they are used and where to find more information on how to use them.

The Custom Properties group contains also any other custom property you define.

---

<b>Property</b>	<b>Purpose, category, usage and further information</b>
CustomProperties	<b>Purpose:</b> The collection of CustomProperties defined by the symbol. <b>Category:</b> Custom Properties <b>Used by:</b> Canvas, Embedded Symbol <b>Can be read by script at run time:</b> No <b>Info:</b> Using Custom Properties on page 239

---

# Appendix B

## Windows Common Control List Methods

You can use the methods of the Windows common controls to manipulate the controls at run time by using them in scripting.

## Overview of Windows Common Control List Methods

The following table contains a list of methods you can use in scripting to:

- Load and save the contents of the Edit Box control from and to a file.
- Manipulate items in the lists of the List Box control and Combo Box control.

Method	Purpose, syntax and information
AddItem	<p><b>Purpose:</b> Add an item (coerced to String) to the list. If the list is sorted, then the new item is inserted at the right position and selected if the list is unsorted, the item is added to the bottom of the list.</p> <p><b>Used by:</b> Combo Box, List Box</p> <hr/> <p><b>Note</b> This function does not work when using an Enum or Array to populate the List Box.</p> <hr/> <p><b>Syntax:</b> <i>ControlName.AddItem(CaptionString);</i></p> <p><b>Info:</b> Adding and Inserting Items into a List on page 365</p>
Clear	<p><b>Purpose:</b> Removes all items from the List. If the list is bound, it clears the bound reference (array or enum) in ArchestraA.</p> <hr/> <p><b>Note</b> This function does not work when using an Enum or Array to populate the List Box.</p> <hr/> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.Clear();</i></p> <p><b>Info:</b> Deleting Items from a List on page 366</p>
DeleteItem	<p><b>Purpose:</b> Accepts an index as a parameter and removes that item from the list. The first item in the list has an index of 0.</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.DeleteItem(Index);</i></p> <p><b>Info:</b> Deleting Items from a List on page 366</p>

Method	Purpose, syntax and information
DeleteSelection	<p><b>Purpose:</b> Delete the currently selected item from the list.</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.DeleteSelection();</i></p> <p><b>Info:</b> Deleting Items from a List on page 366</p>
FindItem	<p><b>Purpose:</b> Accepts a string as a parameter and returns the index of the first item that matches the string. The first item in the list has an index of 0.</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.FindItem(SearchString);</i></p> <p><b>Info:</b> Finding an Item in a List on page 367</p>
GetItem	<p><b>Purpose:</b> Returns the item associated with an index supplied as a parameter to this function. The first item in the list has an index of 0.</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ItemCaption = ControlName.GetItem(Index);</i></p> <p><b>Info:</b> Reading the Caption of a Selected Item in a List on page 367</p>
InsertItem	<p><b>Purpose:</b> Inserts the supplied string after the current selection in the List. Does not work if list is sorted.</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.InsertItem(String);</i></p> <p><b>Info:</b> Adding and Inserting Items into a List on page 365</p>
SetItemData	<p><b>Purpose:</b> Associates a value with an item in the list which index is provided to the function. The first item in the list has an index of 0.</p> <hr/> <p><b>Note</b> This function only works when <code>UseValuesAsItems</code> is set to false. It does not work when using an Enum or Array to populate the List Box control.</p> <hr/> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.SetItemData(Index, Value);</i></p> <p><b>Info:</b> Associating Items with Values in a List on page 368</p>

---

Method	Purpose, syntax and information
GetDataItem	<p><b>Purpose:</b> Returns the value associated with the item in the list which index is supplied to the function. The first item in the list has an index of 0.</p> <hr/> <p><b>Note</b> This function only works when UseValuesAsItems is set to false. It does not work when using an Enum or Array to populate the List Box control.</p> <hr/> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>Value = ControlName.GetDataItem(Index);</i></p> <p><b>Info:</b> Associating Items with Values in a List on page 368</p>
LoadList	<p><b>Purpose:</b> Loads a list of strings from a file which name is passed as parameter to the function. The default location for files is the users folder, for example: c:\documents and settings\username.</p> <hr/> <p><b>Note</b> The LoadList method does not work when using an Enum or Array to populate the List Box control.</p> <hr/> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.LoadList(FileName);</i></p> <p><b>Info:</b> Loading and Saving Item Lists on page 368</p>
LoadText	<p><b>Purpose:</b> Loads a text from a file into the Edit Box control. The default location for files is the users folder, for example: c:\documents and settings\username.</p> <p><b>Used by:</b> Edit Box</p> <p><b>Syntax:</b> <i>ControlName.LoadText(FileName);</i></p> <p><b>Info:</b> Configuring Edit Box Methods on page 364</p>
SaveList	<p><b>Purpose:</b> Save a list to a file which name is passed as parameter to the function. The default location for files is the users folder, for example: c:\documents and settings\username.</p> <p><b>Used by:</b> Combo Box, List Box</p> <p><b>Syntax:</b> <i>ControlName.SaveList(FileName);</i></p> <p><b>Info:</b> Loading and Saving Item Lists on page 368</p>

---



Method	Purpose, syntax and information
SaveText	<p><b>Purpose:</b> Saves the current text in the Edit Box control to a file. The default location for files is the users folder, for example: c:\documents and settings\username.</p> <p><b>Used by:</b> Edit Box</p> <p><b>Syntax:</b> <i>ControlName.SaveText(FileName);</i></p> <p><b>Info:</b> Configuring Edit Box Methods on page 364</p>

---



# Index

## A

- aaHistClientDatabase.dll 385
- .aaPKG file
  - exporting client controls in 373, 388
  - exporting symbols to 78
  - importing a previously exported 376
  - importing client controls from 373, 376
  - importing symbols and graphic toolsets 77
  - importing symbols from 77
- absolute point of origin 130
- AbsoluteAnchor property 400, 444, 469
- AbsoluteOrigin property 444, 470
- action scripts
  - importing with a SmartSymbol 415–416
  - starting with a key combination 62
  - triggers 316–317
  - using 63–64
  - using to add an item to a Combo Box control 365
  - using to configure an element with animation 317–318
  - using to load text into an Edit Box control 364
- ActiveFactory 372
- ActiveFactory TagPicker client control 375, 380
- AddItem() method 365, 494
- Alignment property 444, 486
- anchor points
  - changing position 400
  - hiding 401
  - setting 400
  - showing 401
- AnchorFixedTo property 445, 469
- AnchorPoint property 445, 470
- Angle property 127, 445, 470
- Animation Summary 250
- animations
  - adding to graphic element 250–251
  - associating with client control contain properties 387
  - comparing InTouch and ArchestraA 414–415
  - comparison between InTouch and ArchestraA Symbol Editor 60
  - configuration parameters by type 263
  - configuring action script 316–318
  - configuring analog value displays 292
  - configuring analog value pushbutton 313–314
  - configuring blink 277–278
  - configuring Boolean fill style 265–266
  - configuring Boolean value displays 291–292
  - configuring Boolean value pushbutton 311–312

- configuring Calendar Control 342–343
- configuring Check Box 335–336
- configuring Combo Box 337–341
- configuring DateTime Picker 343–345
- configuring disable 298
- configuring Edit Box 336–337
- configuring fill style 264–268
- configuring height 287–288
- configuring hide symbol 327
- configuring horizontal location 284
- configuring horizontal slider 307–309
- configuring line style 269–273
- configuring List Box 345–349
- configuring name displays 296–297
- configuring orientation 288–290
- configuring percent fill horizontal 279–281
- configuring percent fill vertical 281–283
- configuring pushbutton 314–316
- configuring Radio Button Group 330–335
- configuring show symbol 319–326
- configuring status element 328–330
- configuring string value displays 293
- configuring string value
  - pushbutton 314–316
- configuring text style 273–277
- configuring time value displays 294–296
- configuring tooltip 297
- configuring truth table fill style 266–268
- configuring user input 299–307
- configuring value display 291–297
- configuring vertical location 285
- configuring vertical slider 309–310
- configuring visibility parameters 264
- configuring width 286–287
- connecting to data sources 260–261
- connecting to element properties 257–258
- connecting to element property
  - references 258–259
- connecting with InTouchEventViewApp
  - attributes 260
- cutting, copying, and pasting 350
- data type state 41
- element-specific 328

- enabling and disabling 254–255
- fill direction 279, 281
- fill orientation 279, 281
- hiding list 252
- importing from a SmartSymbol to
  - ArchestrA graphic 414–415
- language switching 421
- managing 262
- removing from a graphic element 253
- resetting default configuration
  - values 256
- reviewing graphic element
  - assignment 251
- showing list 252
- sorting the list 262
- states 41
- switching between graphic
  - elements 263
- truth table state 42
- types 249
- unfill color 279, 281
- validating configuration 255

- anti-aliasing filter 190
- application scripts 62
- ArchestrA package file
  - exporting client controls 388
  - exporting symbols to 78
  - importing previously exported client
    - controls 376
  - importing symbols and graphic
    - toolsets 77
- ArchestrA Symbol Editor
  - adding animations from Animation
    - Summary 250
  - adding custom properties to a
    - symbol 54
  - animation replication 53
  - animation summary 25
  - canvas 25
  - Canvas drawing area 55
  - combining elements into a group 53
  - comparison of animations to
    - InTouchEvent 60
  - comparison of supported data types to
    - InTouchEvent 58
  - configuring the grid 90–91
  - converting data from InTouchEvent 58
  - creating graphics 54–56
  - data sources 57

- description 17, 22
  - differences between InTouch WindowMaker 51–54
  - differences between WindowMaker 51–54
  - disabling animations 254
  - editing the source of an embedded symbol 399
  - element positioning 53
  - elements 26–31
  - Elements List 24, 24
  - embedding source symbols contained in an AutomationObject instance 394
  - embedding source symbols contained in AutomationObject template 393–394
  - enabling animations 254
  - miscellaneous enhancements 54
  - opening symbols in read-only mode 19
  - panning and zooming the canvas 84–89
  - Properties Editor 25
  - redoing a specified number of undone changes 134
  - removing an animation 253
  - reviewing animations assigned to graphic elements 251
  - setting color of drawing area 55
  - setting style ??–185
  - showing and hiding panels 84
  - starting 83–84
  - style replication 53
  - supported InTouch mathematical functions 415
  - supported InTouch string functions 415
  - supported InTouch system functions 416
  - tools and palettes 23
  - Tools panel 24
  - types of animations 60–61
  - types of scripts 62
  - undoing a single change 132, 133
  - undoing a specified number of changes 133
  - usability enhancements 52
  - using animations 57–61
  - using to configure predefined symbol scripts 356–357
  - using to review which animations are assigned to elements 251
  - viewing properties of embedded client controls 388
  - Windows common controls 29–30
  - arcs
    - changing sweep angle 208
    - setting starting point 207
    - setting sweep angle with handles 105
  - arrowhead line end 195
  - attributes
    - connecting animations 256–257
    - connecting animations with data sources 256
    - InTouchViewApp 260
    - reference mode 261
    - referencing a SuperTag 260
    - static mode reference 261
    - writing to attributes with Secured Write or Verified Write security 354
  - auto image display mode 198
  - AutomationObject instances
    - creating a copy containing an embedded symbol 405
    - creating symbols 66, 70–71
    - inheriting symbols from a template 21
    - location source of embedded symbol 404
    - selecting alternative embedded symbol 403
    - using for a single object 18
  - AutomationObject templates
    - creating symbols 68–69
    - managing symbols 19
    - re-using symbols 66
    - using for multiple instances of symbols 18
  - AutomationObjects
    - creating an instance by embedding an ArcestrA symbol 22
    - creating ArcestrA symbols in 19
    - hosting symbols 46
    - using me keyword to reference attributes 32
  - AutoScale property 445, 470
- ## B
- basic objects 55
  - bell-shaped color distribution 177
  - BindTo() method 246
  - blink behavior 215, 218

- Boolean data type 57
  - Boolean fill style animation 265–266
  - Boolean line style animation
    - configuring 269–270
    - setting line style and thickness 270
    - using default properties 271
  - Boolean text style animation
    - configuring 274–275
    - setting default text style 275
    - using default style 275
  - Brick pattern property 181
  - buttons
    - automatically scaling text 202
    - configuring with images 203
    - description 202
    - down image 203
    - DownImage property 203
    - drawing 96
    - up image 203
    - using 202
    - wrapping text 203
  - ButtonStyle property 446, 471
  - buttonsUpImage property 203
- C**
- Calendar controls
    - changing colors 345
    - configuring 230–233
    - configuring animations 342–343
    - data type 221
    - description 29
    - description of date properties 230
    - resizing restrictions 121
    - setting date for first time view 233
    - setting first day of week 231
    - setting number of calendar month sheets 230–231
    - setting text color for trailing dates 233
    - setting title text color 232–233
    - showing or hiding current date 232
    - ShowToday property 458, 477
    - TitleFillColor property 462, 483
    - TitleTextColor property 462, 487
  - CalendarColumns property 446, 471
  - CalendarRows property 446, 471
  - Canvas
    - aligning elements to the grid 91
    - changing grid size 90
    - changing grid subdivision size 90
    - description 55
    - dragging elements 98
    - setting background color 90
    - setting grid color 90
    - show or hiding the grid 91
  - Caption property 446, 486
  - Check Board pattern property 181
  - Check Box animations 335
  - Check Box control
    - configuring 224
    - ControlStyle property 225
    - data type 221
    - setting 3D appearance 225
    - setting caption text 224
    - setting default state 224
  - Checked property 446, 471
  - chords
    - changing start and sweep angle 208
    - setting sweep angle 207
  - circle line end 195
  - circular reference 44
  - Clear() method 366, 494
  - client controls
    - ActiveFactory TagPicker 375
    - adding animation 387
    - animating 387
    - binding properties to attributes or element references 381–383
    - configuring event logging messages 386
    - configuring event scripts 384–386
    - description 32, 372
    - embedding 377–378
    - embedding a symbol into an InTouch application 32
    - embedding into an ArcestrA symbol 377
    - exporting 388
    - exporting directly as ArcestrA package files 388
    - import failure message 374
    - importing 373–375
    - importing a previously exported ArcestrA package 376
    - importing previously exported client controls 376
    - installing ActiveFactory TagPicker 375
    - organizing 376

- properties of element container 379
- securing 388
- translating 429, 430
- viewing additional information 388–389
- viewing and changing properties 379–381
- viewing assemblies 389
- viewing class name, vendor, and version 389
- viewing objects and referencing symbols 390
- color bar 168
- color disk 168
- Color Picker 169
- Color1 property 154, 447
- Color2 property 154, 447
- Color3 property 154, 447
- colors
  - adding to custom palette 170
  - analog alarm animation 61
  - analog fill animation 61
  - analog line animation 60
  - analog text animation 61
  - bell-shaped distribution 177
  - Canvas grid background 90
  - changing gradient variant 176
  - changing with element style 186
  - discrete alarm animation 60, 61
  - discrete fill animation 61
  - discrete line animation 60
  - discrete text animation 61
  - falloff distribution 178
  - gradient direction properties 174
  - gradient properties 171
  - hue property 168
  - locking focus scales width and height 180
  - luminance 168
  - removing from custom palette 170
  - RGB 168
  - saturation property 168
  - setting background of a Windows common control 220
  - setting center point of radial or point based gradient 179
  - setting custom gradient angle 176
  - setting distribution 178–179
  - setting fill gradient 154
  - setting focus scales 179–180
  - setting focus scales height 180
  - setting focus scales width 180
  - setting for text 161
  - setting from color disk and bar 168
  - setting from the custom palette 169
  - setting horizontal gradient 174
  - setting image transparency 200
  - setting image transparency color 200
  - setting number in gradient 172–174
  - setting pattern background 182
  - setting pattern foreground 182
  - setting point based gradient 175
  - setting radial gradient 175
  - setting solid from the Standard Palette 166–167
  - setting the drawing area 55
  - setting transparency 185
  - setting vertical gradient 174
  - setting with Color Picker 169
  - setting with value input boxes 168–169
  - standard palette 167
  - triangular distribution 177
  - using for text animation 43
- Combo Box animations
  - configuring array and captions 339–340
  - configuring enum 341
  - configuring static 338–339
  - types 337
- Combo Box controls
  - clipping items 228
  - configuring 227–229
- Count property 229
- data types 221
- description 29
- loading an item list 368
- MaxDropDownItems property 229
- NewIndex 229
- preventing clipping of items 228
- saving an item list 368
- setting maximum number of items in drop-down list 229
- setting type 227
- setting width of drop-down list 228
- using properties at run time 229
- condition scripts 63
- Confetti pattern property 181

- control points
    - adding or removing 205
    - changing the tension of curves 206
    - description 204
    - editing within a path graphic 144
    - moving 204
  - ControlStyle property 225, 447, 471
  - Count property 229, 237, 448
  - Cross pattern property 181
  - curves
    - adding control points 205
    - changing shape with control points 205
    - changing tension 206
    - deleting control points 205
    - drawing 95
    - moving control points 204
  - custom palette
    - adding colors to 170
    - description 169
    - loading 171
    - removing colors from 170
    - saving 170
  - custom properties
    - adding 241
    - adding to a symbols 54
    - configuring 242–243
    - connecting animations 258–259
    - deleting 241
    - description 32, 99, 239
    - Edit Custom Properties dialog box 240
    - external linking to InTouch tags 245
    - group properties 491–492
    - language switching 428, 438
    - linking to external sources 245
    - managing 240
    - moving to a group 398
    - overriding 245
    - overriding source symbol 396
    - public 44
    - renaming 244
    - resetting configuration to default values 244
    - restoring default value and description 397
    - reverting to original 246
    - translating 429
    - translations 428
    - using binding 246–247
    - using to extend functionality of a symbol 33
    - using to reference InTouch tag from ArchestrA symbol 259
    - validating 244
  - CustomFormat property 448, 472
  - CustomProperties property 448, 492
- ## D
- data change scripts 63
  - DataStatus animations
    - configuring 328–329
    - restrictions 330
  - dates
    - setting customized format in DateTime Picker control 234
    - setting long format of DateTime Picker control 234
  - DateTime Picker animations 343–345
  - DateTime Picker controls
    - AbsoluteOrigin() property 444
    - changing background color 220
    - configuring 234, 235
    - CustomFormat() property 448
    - data type 221
    - DefaultValue() property 448
    - description 30
    - Enabled() property 450
    - Font() property 451
    - resizing restrictions 121
    - setting date formats 234
    - setting default value 235
    - time formats 234
  - DefaultValue property 448, 472
  - DeleteItem() method 366, 494
  - DeleteSection() method 366
  - DeleteSelection() method 495
  - Description property 448, 468
  - Diagonals pattern property 181
  - Diamon pattern property 181
  - diamond line end 195
  - discrete data type 57
  - documentation conventions 15
  - down image 203
  - DownImage property 203, 449, 472
  - DropDownType property 449, 472
  - DropDownWidth property 449, 472



dynamic size change 399, 401  
 DynamicSizeChange property 449, 473

## E

Edit Box animations  
   configuring 336–337

Edit Box controls  
   configuring 225–226  
   data type 221  
   description 29  
   Editline property 226  
   LoadText() method 364, 496  
   SaveText() method 364, 497  
   setting default text 226  
   setting text to read-only 226  
   setting text word wrap 226  
   using methods to save and load text at run time 363

elapsed time  
   data type 57  
   selecting as custom property 242

element mode 138

Elements List 152  
   adding elements to group 136  
   description 24  
   selecting elements from 100, 103  
   using to change z-order of elements 124

element-specific animations  
   description 249  
   supported elements 328

ellipses  
   basic element 26  
   closed element 26  
   drawing 94  
   editing line properties 157

embedded symbols  
   appearance 44  
   changing 44  
   controlling size propagation 399  
   converting to a group 398  
   creating an new AutomationObject instance 405  
   description 21, 44  
   editing 395–396  
   editing source 399  
   embedding from Graphic Toolbox 392–393

  enabling dynamic size change 401–402  
   hiding anchor point 401  
   instantiating 45  
   language switching 425, 439  
   locating AutomationObject instance that contains source symbol 404  
   overriding custom properties 396–397  
   propagating changes 46–48  
   renaming source symbols 395  
   restoring to original size of source symbol 397  
   selecting alternate AutomationObject instances 404  
   selecting alternate from AutomationObject instance 403  
   showing anchor point 401  
   translating custom properties 429  
   viewing the source 398

Enabled property 450, 488

End property 450, 473

EndCap property 450, 484

## F

falloff color distribution 178

falloff gradient  
   description 178

FillBehavior property 155, 450, 481

FillColor property 451, 482

FillOrientation property 451, 482

fills  
   blink behavior 215  
   overriding appearance to indicate a change in quality 215  
   setting behavior 155  
   setting orientation 155  
   setting status override appearance for 211  
   setting style 153, ??–159, ??–162  
   setting unfilled style 154

FillStyle property 154

FindItem() method 367, 495

FirstDayOfWeek property 451, 473

float  
   custom property 243  
   data type 57  
   setting as a text display 160  
   tension values 206  
   user input animation 299

- focus scales
    - description 179
    - locking width and height 180
    - setting height 180
    - setting width 180
  - font
    - language font at run time 437
    - language switching 425
  - Font property 161, 451, 486
  - format painter
    - copying element formats to target elements 189
    - copying format of an element to another 189
    - copying the format of an element in repetitive mode 190
    - description 189
  - Format property 452, 473
  - functions
    - importing from
      - aaHistClientDatabase.dll 385
    - InTouch miscellaneous 416
    - InTouch string 416
    - InTouch system 416
    - InTouch types that can be imported to ArchestrA Symbol Editor 415–416
    - IOSetReferences() 247
- G**
- Galaxy Browser 223, 229
    - browsing for List Box control 237
    - browsing properties in the List Box control 237
    - using to browse methods of the Combo Box control 337
    - using to browse methods of the Edit Box control 336
    - using to browse methods of the List Box control 345
    - using to connect animations to attribute references 256–257
    - using to connect animations to element properties 257
    - using to connect animations to element property references 257–258, 258–259
    - using to connect animations to InTouch tags 260
    - using to embed a client control into an ArchestrA symbol 377–378
      - using to embed symbols from the Graphic Toolbox 393–394
      - using to select an alternate instance 404
      - using to select an alternate symbol 403
      - using to show and select properties of elements or symbols 362
  - GetItem() method 367, 495
  - GetItemData() method 368, 496
  - gradients
    - changing variant 176
    - falloff 178
    - peak 177
    - properties 172
    - setting bell-shaped 178
    - setting custom angle 176
    - setting direction 174
    - setting focus scales 179–??
    - setting horizontal direction 174
    - setting number of colors 172
    - setting peak with one or two colors 178
    - setting point based 175
    - setting transparency 185
    - setting vertical direction 174
    - triangular 177
    - using triangular 178
  - graphic elements
    - absolute point of origin 130
    - adding animations 250–251
    - adding control points 205
    - adding outlines to indicate non-good status or quality 217–218
    - adding to an existing group 136
    - adding to path graphics 148–??
    - adjusting space between 115–120
    - adjusting the z-order 124–125
    - aligning 110–114
      - aligning by centers 113
      - aligning by points of origin 114
      - aligning horizontally 111
      - aligning to Canvas grid 91
      - aligning vertically 112
    - appearance properties 469–481
    - basic 26
    - binding client control properties 382–383
    - bringing one level forward 125
    - bringing to front 125

- changing angles of arcs, pies, and chords 207–208
- changing properties with scripts 362–363
- changing start and sweep angle of an arc, pie, or chord 208
- changing tab order 188
- changing the tension of curves 206
- changing visibility 187
- changing z-order within path graphic 147
- closed 26
- configuring 328
- configuring analog value display animations 292
- configuring analog value pushbutton animations 313–314
- configuring analog value user input animations 301–302
- configuring animation fill style 264–268
- configuring animation line style 269–273
- configuring animation text style 273–277
- configuring blink animation 277–278
- configuring Boolean value display animations 291–292
- configuring Boolean value pushbutton animations 311–312
- configuring disable animations 298
- configuring discrete value user input animations 299–300
- configuring elapsed time value user input animations 306–307
- configuring height animations 287–288
- configuring hide symbol animations 327
- configuring horizontal location animations 284
- configuring horizontal slider animations 307–308
- configuring name display animations 296–297
- configuring orientation animations 288–290
- configuring percent fill horizontal animations 279–281
- configuring percent fill vertical animations 281–283
- configuring show symbol animations 298, 324–326
- configuring string value display animations 293
- configuring string value pushbutton animations 314–316
- configuring string value user input animations 302–303
- configuring time value display animations 294–295
- configuring time value user input animations 304–306
- configuring tooltips 297
- configuring vertical location animations 285
- configuring vertical slider animations 309–310
- configuring width animations 286–287
- connecting animations to ArchestraA attributes 256–257
- connecting animations to custom properties 258–259
- connecting animations to element properties 257–258
- connecting animations to InTouch tags 259–261
- control points 204–205
- copying and pasting animations 350
- copying locked and grouped 106
- copying, cutting, and pasting 106–108
- creating a group 135
- custom properties 491–492
- cutting and pasting 107
- cutting and pasting animations 350
- decreasing space between 119–??
- deleting 107
- deleting control points 205
- description 52, 93–94
- distributing 116
- dragging 98
- dragging and drawing 94–98
- duplicating 108
- editing controls points within path graphic 144
- editing fill properties 152–156
- editing line properties 157–159
- editing properties 98–99
- editing start or sweep angle within path graphic 143
- editing the name 152

- editing the tab order 188
- enabling and disabling animations 254
- enabling or disabling for run-time interaction 187
- fill style properties 481–483
- flipping 131
- flipping horizontally 131
- flipping vertically 131
- graphic properties 468
- handles 100
- hiding animation list 252
- hiding at run time 187
- increasing space between 118
- inline editing 104–??, 104, ??–105
- InTouch types that can be imported with restrictions 413
- language switching 420
- line style properties 484–485
- locking 132
- locking and unlocking 132
- moving 109–110
- moving by specifying X and Y properties 109
- moving control points 204
- moving points of origin with mouse 129
- moving the origin 129–130
- moving with keyboard 110
- moving with mouse 110
- moving within path graphic 141
- open 26
- overriding appearance depending on quality and status attributes 212–219
- overriding fill appearance to indicate non-good status or quality 215
- positioning with ArchestrA Symbol Editor 53
- properties 444–467
- reference mode 261
- relative point of origin 130
- removing all space between 120–??
- removing animations 253
- removing from a group 136
- removing from path graphic 149
- resetting default animation values 256
- resizing 121–124
- resizing by changing size properties 122
- resizing proportionally 122
- resizing to same height 123
- resizing to same size 124
- resizing to same width 123
- resizing with mouse 121
- resizing within path graphics 142
- reviewing assigned animations 251
- rotating 126–??
- rotating 90 degrees 128
- rotating by changing Angle property 127
- rotating with mouse 126
- run-time behavior properties 488–491
- selecting 100–104
- selecting all with Select All function 102
- selecting by lasso 102
- selecting by mouse click 101
- selecting with Elements List 103
- sending one level backward 125
- sending to back 125
- setting colors and transparency of a gradient 186
- setting equal space between 117
- setting fill behavior 155
- setting fill gradient 154
- setting fill orientation 155
- setting line end shape and size 195–196
- setting overlap when duplicated 108
- setting text alignment 162–163
- setting text auto scaling 196
- setting text color 161
- setting text font 161
- setting text properties 160–166
- setting text word wrap 196
- setting the radius of rounded rectangles 194
- setting transparency level 185
- setting unfilled style 154
- showing animation list 252
- showing at run time 187
- static mode reference 261
- substituting references 351–352
- supported InTouch types that can be imported to ArchestrA 411–412
- swapping end points within path graphic 145–146
- switching between animations 263
- task list 93–94
- text style properties 486–487

- ungrouping 135
- unlocking 132
- unselecting 103–104
- using the format painter 189–190
- using undo and redo 133–134
- validating animation configuration 255
- working with groups 134–137
- Graphic Toolbox
  - creating copies of symbols 75
  - creating generic symbols 66
  - creating Graphic Toolsets 73
  - creating symbols 66–67
  - customizing Graphic Toolsets 76
  - deleting Graphic Toolsets 75
  - editing hosting symbols 46
  - embedding source symbols 393
  - managing symbols 18
  - moving Graphic Toolsets 76
  - moving symbols to different Graphic Toolsets 74
  - renaming Graphic Toolsets 75
  - Toolsets folder 18
  - using to store ArcestrA symbols 18
  - viewing class name, vendor, and version of a client control 389
  - viewing client control assemblies 389
  - viewing read-only symbols 82
  - viewing referencing AutomationObjects and symbols of client controls 390
- Graphic Toolsets
  - creating 73
  - customizing 76
  - deleting 75
  - moving 76
  - moving symbols 74
  - renaming 75
- Grid pattern property 181
- groups
  - adding elements to 136
  - converting embedded symbols 398
  - creating 134–135
  - deleting 107
  - description 56
  - editing components within 137
  - moving custom properties 398
  - properties of 34
  - removing elements from 136
  - selecting elements from 105
  - ungrouping 135
- H**
- H/V lines
  - adding to path graphic 148
  - drawing 94
  - editing 157
  - setting start and end points 157
- handles
  - description 100
  - primary 100
  - secondary 100
- HasTransparentColor property 197, 452, 473
- Height property 121, 452, 474
- hide symbol animations 327
- horizontal alignment 111
- Horizontal pattern property 181
- horizontal scroll bar
  - configuring 236
  - displaying with HorizontalScrollBar property 453
- HorizontalDirection property 452, 482
- HorizontalPercentFill property 453, 482
- HorizontalScrollBar property 453, 474
- HorizontalScrollbar property 236
- I**
- IDE
  - configuring security 80
  - creating a new symbol 66
  - deriving an instance from a template 21
  - description 15
  - symbol management tasks 65
  - symbol management tools 65
- Image property 202, 453, 474
- ImageAlignment property 199, 453, 474
- images
  - adding editing application 201
  - alignment positions 199
  - auto sizing 198
  - configuring for buttons 203
  - description 197
  - editing 200
  - enabling color transparency 200
  - importing 96

- placing on canvas 197
- properties 197
- selecting replacement 202
- setting alignment 199
- setting color transparency 200
- setting display mode 198
- setting editing application 200, 201
- setting frame size 198
- setting status override appearance for 211–212
- setting transparency color 200
- stretching 198
- supported file formats 183, 197
- tiled 198
- ImageStyle property 453, 475
- inline editing 104
- InsertItem() method 365, 495
- IntegralHeight property 228, 236, 454, 475
- interaction animations
  - description 249
  - types 263
- InternationalizedString data type 57
- InTouch
  - animations 59–61
  - basic objects 55
  - binding run-time behavior with animations 37
  - comparing script types to ArcestrA scripts 62–64
  - comparison of animations to ArcestrA Symbol Editor 60
  - comparison of supported data types to ArcestrA Symbol Editor 58
  - configuring data source 57
  - connecting element animations to tags 259–261
  - connecting tags to ArcestrA symbols 57
  - converting data to ArcestrA Symbol Editor 58
  - data types 57
  - differences between WindowMaker and ArcestrA Symbol Editor 51
  - editing a symbol embedded in a window 72
  - embedded symbol containing a client control 32
  - embedding symbol into a managed application 383
  - embedding symbols 20
  - functions supported by ArcestrA Symbol Editor 416
  - graphic objects not included in ArcestrA Symbol Editor 54
  - importing functions to ArcestrA Symbol Editor 415–416
  - importing SmartSymbol animation links 414–415
  - importing SmartSymbols into an ArcestrA symbol 56
  - importing SmartSymbols into ArcestrA symbols 408–411
  - instantiating a symbol 22
  - keyword 40, 416
  - linking symbol custom properties to tags 245
  - mathematical functions supported by ArcestrA Symbol Editor 415
  - overview of ArcestrA integration 17
  - reference syntax to link a symbol to a tag 245
  - restrictions to importing SmartSymbols to ArcestrA Symbol Editor 411–413
  - starting ArcestrA Symbol Editor 83
  - string functions supported by ArcestrA Symbol Editor 415
  - supported graphic types that can be imported to ArcestrA 411–413
  - system functions supported by ArcestrA Symbol Editor 416
  - types of animations 60–61
  - types of scripts 62
  - updating embedded symbols in WindowMaker 46
  - using custom properties to reference a tag 33
  - using custom properties with tags 239
  - using tag values in a symbol 40
  - wizards 56
- InTouch wizards 54
- InTouchEventViewApp
  - connecting animations with attributes 260
  - connecting attributes to animations 260
  - managed InTouchEventViewApp application object 259
- IOSetReferences() function 247

**K**

key scripts 62

keywords

InTouch 40, 416

self-reference me 32

**L**

language

removing for a symbol 423

selecting for a symbol 422

Language property 454, 488

language switching

ArchestrA symbols 420

custom properties 428, 438

data types 441

embedded symbols 425, 439

example 432

font 425

graphic elements 420

overriding translated custom properties 435

overriding translated string substitutions 435

overriding translations in WindowMaker 435

popup symbols 440

precedence rules 437

propagation of language setting changes 440

run time 435

SmartSymbols 431

string substitution 426

LanguageID property 454, 489

lasso 100, 102

Layout property 454, 475

LineColor property 454, 484

LinePattern property 455, 484

lines

ArchestrA Symbol Editor basic element 22

blink behavior 218

element of a symbol 26

end shapes 195

overriding appearance to indicate a change in quality 209

overriding appearance to indicate non-good status or quality 216–217

selecting tool from Tools panel 24

setting end shape and size 195–196

setting size of arrowheads 196

setting status override appearance for 211

used in path graphics 30

LineWeight property 455, 485

List Box animations

configuring array and captions 347–348

configuring enum 348–349

configuring static 346–347

types 345

List Box controls

adding an item 365

associating items with values in a list 368

avoiding clipping of items 236

configuring 236–237

Count property 237

data type 221

deleting a selected item from a list 366

deleting all items from a list 366

description 29

finding an item in a list 367

getting item data 368

inserting an item in a list 365

loading the item list 368

NewIndex property 237

reading the caption of a selected item in a list 367

saving the item list 368

SelectedValue property 237

setting item data 368

TopIndex property 237

using horizontal scroll bar 236

using properties at run time 237

LoadList() method 368, 496

LoadText() method 364, 496

Locked property 455, 475

**M**

managed application 260

MaxDropDownItems property 229, 455, 476

message data type 57

messages

edit symbol confirmation 72

Not Found after deleting embedded symbol 79

security after user attempts to export a symbol without appropriate permissions 80  
warning when overwriting colors in a custom palette 171

#### methods

AbsoluteOrigin property 444  
AddItem() 365, 494  
BindTo() 246  
Clear() 366, 494  
configuring Combo Box and List Box 365–369  
configuring Edit Box 364  
DeleteItem() 366, 494  
DeleteSection() 366  
DeleteSelection() 495  
FindItem() 367, 495  
GetItem() 367, 495  
GetItemData() 368, 496  
InsertItem() 365, 495  
LoadList() 368, 496  
LoadText() 364, 496  
SaveList() 369, 496  
SaveText() 364, 497  
SetItemData() 368, 495  
using in scripting 363  
Multiline property 226, 456, 476  
MultiplePopupsAllowed property 456, 489

## N

Name property 456, 468

#### named scripts

removing from a symbol 361  
renaming 360

#### names

changing in Properties Editor 152  
changing in the Elements List 152  
NewIndex property 229, 237, 456  
normal image display mode 198

## O

#### outlines

adding to elements to indicate non-good status or quality 217–218  
description as quality status indicator 209

replacing group handle 105  
resetting to default 219  
OwningObject property 404, 457, 489

## P

#### pan

description 89  
using mouse 89

#### path graphics 31

adding elements 148–??  
breaking the path 140–??  
changing 140–147  
changing element z-order 146–147  
creating 139  
description 52, 138–??  
editing element control points within 144  
editing start or sweep angle of element within 143  
moving elements within 141  
removing elements 149  
resizing element within 142  
swapping element end points within 145  
viewing 138

#### path mode 138

#### patterns

properties 181  
setting 182  
setting transparency 185

#### peak gradient 177

#### Percent pattern property 181

#### pies

changing sweep angle 208  
setting starting point 207

#### point based gradient 175, 179

#### points of origin 114

#### polylines

basic element of ArchestrA Symbol Editor 26  
drawing 95  
inline editing 104  
moving control points 204  
part of path graphics 31  
shaping with control points 105

#### popup symbols

language switching 440

#### pop-ups 191



- properties
  - AbsoluteAnchor 400, 444, 469
  - AbsoluteOrigin 470
  - Alignment 444, 486
  - AnchorFixedTo 445, 469
  - AnchorPoint 445, 470
  - Angle 127, 445, 470
  - AutoScale 445, 470
  - binding client control attributes 381–??
  - ButtonStyle 446, 471
  - CalendarColumns 446, 471
  - CalendarRows 446, 471
  - Caption 446, 486
  - changing element tab order with 188
  - Checked 446, 471
  - client control container 379, 387
  - Color1 154, 447
  - Color2 154, 447
  - Color3 154, 447
  - configuring fill style with 154
  - ControlStyle 225, 447, 471
  - Count 229, 237, 448
  - CustomFormat 448, 472
  - CustomProperties 448, 492
  - DefaultValue 448, 472
  - Description 448, 468
  - DownImage 203, 449, 472
  - DropDownType 449, 472
  - DropDownWidth 449, 472
  - DynamicSizeChange 449, 473
  - Enabled 450, 488
  - End 450, 473
  - EndCap 450, 484
  - FillBehavior 155, 450, 481
  - FillColor 451, 482
  - FillOrientation 451, 482
  - FillStyle 154
  - FirstDayOfWeek 451, 473
  - Font 161, 451, 486
  - Format 452, 473
  - HasTransparentColor 197, 452, 473
  - Height 121, 452, 474
  - HorizontalDirection 452, 482
  - HorizontalPercentFill 453, 482
  - HorizontalScrollBar 453, 474
  - HorizontalScrollbar 236
  - Image 202, 453, 474
  - ImageAlignment 199, 453, 474
  - ImageStyle 453, 475
  - IntegralHeight 228, 236, 454, 475
  - Language 454, 488
  - LanguageID 454, 489
  - Layout 454, 475
  - LineColor 454, 484
  - LinePattern 455, 484
  - LineWeight 455, 485
  - Locked 455, 475
  - MaxDropDownItem 455
  - MaxDropDownItems 229, 476
  - Multiline 226, 456, 476
  - MultiplePopupsAllowed 456, 489
  - Name 456, 468
  - NewIndex 229, 237, 456
  - OwningObject 404, 457, 489
  - Radius 457, 476
  - ReadOnly 226, 457, 476
  - RelativeAnchor 400, 457, 476
  - RelativeOrigin 458, 477
  - Scripts 458, 489
  - SelectedValue 237, 458
  - selecting replacement image with 202
  - setting fill behavior 155
  - setting fill gradient with 154
  - setting text display format with 160
  - setting text font with 161
  - ShowToday 458, 477
  - Smoothing 459, 477
  - Start 459, 477
  - StartAngle 459, 478
  - StartCap 459, 485
  - SweepAngle 460, 478
  - SymbolReference 398, 460, 490
  - TabOrder 460, 490
  - TabStop 188, 461, 490
  - Tension 461, 478
  - Text 461, 478
  - TextColor 461, 487
  - TextFormat 160, 462, 479
  - TitleFillColor 462, 483
  - TitleTextColor 462, 487
  - topIndex 462
  - TrailingTextColor 462, 487
  - Transparency 463, 479
  - TransparentColor 197, 200, 463, 479
  - TreatAsIcon 464, 491
  - UnFilledColor 464, 483

UpImage 203, 464, 479  
 using Combo Box control at run time 229  
 using List Box controls at run time 237  
 Value 221, 464  
 VerticalDirection 465, 483  
 VerticalPercentFill 465, 483  
 Visible 465, 491  
 Width 121, 466, 480  
 WordWrap 203, 466, 480  
 X 109, 466, 480  
 Y 109, 467, 481  
 Properties Editor  
   changing absolute point of origin 130  
   changing element names 152  
   changing relative point of origin 130  
   description 25  
   editing properties with 98  
 pushbutton animations  
   analog value 313–314  
   Boolean value 311–312  
   string value 314–316

## Q

quality  
   adding outlines to elements to indicate status or non-good quality 217–218  
   changing status to run named scripts 354  
   including Status element to show icon 26  
   monitoring symbol 209  
   overriding element appearance depending on attributes 212  
   overriding element fill appearance to indicate non-good status 215  
   overriding line appearance to indicate non-good quality or status 216–217  
   overriding text appearance to indicate non-good status 213–214  
   previewing all override appearances 218  
   previewing override appearances 218  
   resetting override appearances to default 219  
   showing 49  
   showing by overriding animations 49  
   showing for elements 24  
   showing symbol status 209

  using DataStatus animation 328  
   using Status element to create icon 209  
   using Status element to show status 27–28  
   using status to add outlines to elements 217–218  
   using status to override appearance of lines 216–217  
   using status to override fill appearance of elements 215  
   using status to override text appearance of elements 213

## R

radial gradients 175, 179  
 Radio Button Group  
   configuring 222–223  
   data types 221  
 Radio Button Group animations  
   configuring array and captions 332–334  
   configuring enum 334–335  
   configuring static 331  
   types 330  
 Radio Button Group controls  
   configuring 222  
   setting 3D appearance 222  
   setting layout of options 223  
   using properties at run time 223  
 Radius property 457, 476  
 ReadOnly property 457, 476  
 Readonly property 226  
 real  
   data type 57  
   imported local tag types 416  
   supported data type for Windows common controls 221  
   used in SuperTags 416  
 rectangles  
   basic object 24  
   closed object 26  
   drawing 94  
   elements of an ArcestrA symbol 22  
   fill style 184  
   selecting as basic object from ArcestrA Symbol Editor 55  
 references  
   animation 24, 36  
   AutomationObjects 78

- circular 44
  - element 32
  - keyword 32
  - substituting in elements 351–352
  - to InTouch tag 33
  - to InTouch tags 57
  - relative point of origin 130
  - RelativeAnchor property 400, 457, 476
  - RelativeOrigin property 458, 477
  - RGB color properties 168
  - rounded rectangles
    - description 194
    - enlarging radius 194
    - reducing radius 194
    - setting exact radius 194
- S**
- satellite assemblies
    - translations 429
  - SaveList() method 369, 496
  - SaveText() method 364, 497
  - scripts
    - action triggers 316–317
    - adding to a symbol 358–359
    - application 62
    - binding 246
    - changing properties of graphic elements 362
    - condition 63
    - configuring animations 316–318
    - configuring for client controls 384–386
    - configuring predefined of a symbol 356–357
    - configuring to symbols 353
    - data change 63
    - description 62–64, 353
    - editing for symbols 360
    - error handling 355
    - execution order 354
    - importing from a SmartSymbol 415–416
    - importing functions from aaHistClientDatabase.dll 385
    - key 62
    - named 354
    - predefined 354
    - removing from an ArcestrA symbol 361
    - renaming a named script 360
    - security 354
    - setting time-out period for symbol script 355
    - substituting attribute references 361
    - using 62–64
    - using methods 363
    - using with Combo box methods 365–369
    - using with Edit Box methods 364
    - using with List Box methods 365–369 While Showing 357
  - Scripts property 458, 489
  - Secured Write 354
  - security
    - configuring for symbols 80–81
    - enforcing by user role and permissions 80
    - IDE user permissions 80
    - in client controls 388
    - in scripts 354
    - warning message after attempting to export symbol without appropriate permissions 80
  - SelectedValue property 237, 458
  - SetItemData() method 368, 495
  - show symbol animations
    - configuring 324–326
    - element attributes 319–324
  - ShowToday property 458, 477
  - SmartSymbols
    - importing action scripts 415–416
    - importing into an ArcestrA symbol 56, 408–411
    - importing tags and references 416–417
    - importing tags and references to ArcestrA symbols 416–417
    - restrictions to importing 411–417
    - translating 431
  - Smoothing property 459, 477
  - snap-to-grid option 91
  - square line end 195
  - standard palette 167
  - Start property 459, 477
  - StartAngle property 459, 478
  - StartCap property 459, 485
  - Status element
    - using to indicate quality 27–28

- Status elements
  - associating with attributes and animated elements 210
  - configuring DataStatus animations 328–329
  - description 49, 52, 209
  - drawing 97
  - drawing icon on canvas 209
  - overriding element blink behavior 215
  - overriding element fill appearance 215
  - previewing all override appearances 218
  - resetting to default appearance 219
  - restrictions of DataStatus animations 330
  - setting appearance 210
  - setting default appearance 210–212
  - using 209–212
- stretch image display mode 198
- string
  - comparison between InTouch and ArchestrA Symbol Editor animation 60
  - data type 57, 221
  - finding 165
  - language switching 426
  - matching 166
  - replacing 165
  - substituting 164
  - substituting in text properties of an element 160
- SweepAngle property 460, 478
- symbol scripts
  - actions against symbols 353
  - adding named type to a symbol 358–359
  - configuring predefined 356–357
  - editing 360
  - error handling 355
  - error messages from syntax errors 355
  - execution order 354
  - named 354
  - predefined 354
  - removing from a symbol 361
  - renaming named type 360
  - running when a condition becomes true 63
  - security 354
  - setting time-out period 355
  - writing to attributes with Secured Write or Verified Write security 354
- SymbolReference property 398, 460, 490
- symbols
  - adding custom properties with ArchestrA Symbol Editor 54
  - adding named scripts 358–359
  - adding Windows common controls 219
  - allowing multiple pop-ups 191
  - animations 37–43
  - appearance of embedded 44
  - changing embedded properties 44
  - changing position of an anchor point 400
  - configuring predefined scripts 356–357
  - configuring security 80–81
  - controlling size propagation of embedded 399
  - creating in AutomationObject instances 70–71
  - creating in AutomationObject templates 68–69
  - creating in Graphic Toolbox 66–67
  - creating with the ArchestrA Symbol Editor 22
  - creation methods 66
  - data types 57–59
  - deleting 79
  - description 17, 65
  - editing contained in an AutomationObject 72
  - editing description 191
  - editing embedded into an InTouch window 72
  - editing general properties 190–191
  - editing scripts 360
  - embedded description 44
  - embedding 392–394
  - embedding client controls 377–378
  - exporting to a .aaPKG file 78
  - hosted by AutomationObject 46
  - hosted by the Graphic Toolbox 46
  - importing from a .aaPKG file 77
  - importing SmartSymbols 408–411
  - importing SmartSymbols tags and references 416–417
  - inherited 19

- instantiating embedded 45
  - instantiating with an
    - AutomationObject template 22
  - language settings for Galaxies 424
  - language switching 420
  - managed by Graphic Toolbox 18
  - managing 18
  - managing in AutomationObjects 19
  - monitoring and showing quality and status 209–212
  - monitoring quality and status 209
  - moving between Graphic Toolsets 74
  - opening for editing 71
  - organizing into Toolsets 18
  - placement in ArcestrA
    - environment 18
  - propagating changes 46
  - properties 32–36
  - removing languages 423
  - removing scripts 361
  - renaming 74
  - renaming scripts 360
  - renaming source and hosting
    - AutomationObjects 395
  - restrictions to importing
    - SmartSymbols 411–417
  - re-using 20–21
  - script triggers 62
  - selecting a language 422
  - setting anchor point 400
  - setting the radius of rounded rectangles 194
  - showing quality and status 49
  - showing quality and status by overriding 49
  - substituting strings 164–??
  - supported file formats 197
  - types of animations 59–61
  - using anti-aliasing filter 191
  - using buttons 202
  - using custom properties 33
  - viewing in read-only mode 82
  - viewing references to a client control 390
- T**
- tab order 188
    - description 188
    - editing 188
  - TabOrder property 460, 490
  - TabStop property 188, 461, 490
  - tags
    - attribute reference to a SuperTag 260
    - binding animations to ArcestrA
      - attributes 37
    - browsing for in a managed application 260
    - connecting to element animations 259–261
    - importing from SmartSymbols 416–417
    - linking to symbol custom properties 245
    - SuperTags 260
    - using values in an ArcestrA symbol 40
  - technical support 16
  - Tension property 461, 478
  - text
    - autoscaling 196
    - drawing boxes 97
    - overriding appearance to indicate a change in quality 209
    - overriding appearance to indicate non-good status or quality 213–214
    - placing on the canvas 97
    - scaling size for buttons 202
    - setting Check Box control caption 224
    - setting color 161
    - setting color for Calendar control 233
    - setting color for trailing dates of Calendar control 233
    - setting color in Windows common controls 220
    - setting default in Edit Box controls 226
    - setting font 161, 162–163
    - setting text display format 160
    - setting to display 160
    - setting to read-only in Edit Box controls 226
    - setting word wrap in Edit Box controls 226
    - substituting strings 164–166
    - word wrapping 196
    - wrapping for buttons 203
  - Text property 461, 478
  - TextColor property 461, 487
  - TextFormat property 160, 462, 479

- textures
    - description 183
    - setting 183
    - setting transparency 185
  - the 80
  - time
    - Calendar property 221
    - elapsed 242
    - formats for DateTime Picker control 234
    - setting customized in DateTime Picker control 234
    - setting display in DateTime Picker control 234
  - title image display mode 198
  - TitleFillColor property 462, 483
  - TitleTextColor property 462, 487
  - tooltips
    - configuring animations 297
  - TopIndex property 462
  - TrailingTextColor property 462, 487
  - translating
    - custom properties 429
  - transparency level 185
  - Transparency property 463, 479
  - TransparentColor property 197, 200, 463, 479
  - TreatAsIcon property 464, 491
  - triangular color distribution 177
  - triangular gradient 177
  - truth table fill style animation
    - adding a condition 267
    - changing condition processing order 268
    - configuring element 266–267
    - deleting a condition 268
    - setting default 267
    - using default in a truth table 267
  - truth table line style
    - changing condition processing order 273
    - configuring 271–272
    - deleting a condition 273
    - setting a condition 272
    - setting default properties 272
    - using default properties 272
  - truth table text style animation
    - adding a condition 276
    - changing the condition processing order 277
    - configuring 275–276
    - deleting a condition 277
    - setting default properties 276
    - using default properties 276
- U**
- UnFillColor property 464, 483
  - up image 203
  - UpImage property 203, 464, 479
- V**
- Value property 221, 464
  - Verified Write 354
  - vertical alignment 112
  - vertical gradients 174
  - Vertical pattern property 181
  - vertical slider animations 309–310
  - VerticalDirection property 465, 483
  - VerticalPercentFill property 465, 483
  - Visible property 465, 491
  - visualization animations
    - description 249
    - hiding 264
    - showing 264
- W**
- While Showing script 357
  - Width property 121, 466, 480
  - WindowMaker
    - deriving an AutomationObject instance by embedding an ArchestrA symbol 21
    - differences between ArchestrA Symbol Editor 51–54
    - marking embedded symbols for an update 46
    - setting time-out period for scripts 355
  - Windows common controls
    - changing color of background and text 220
    - configuring to write data 349
    - data types 221
    - description 219
    - drawing 98
    - methods 494–497
    - provided types 29–30
    - reading and writing selected value at run time 221

- reading and writing selected values at run time 221
- types 219
- value property 221
- Windows controls
  - drawing 98
  - listing 56
- WindowViewer
  - viewing languages for symbols 436
- wizards
  - InTouch 54, 56
  - using 56
- WordWrap property 203, 466, 480

## X

- X property 109, 466, 480

## Y

- Y property 109, 467, 481

## Z

- zoom
  - description 89
  - selected element 87
  - specified area 88
  - specifying value of 88
  - to a specified point 87
  - to default value 87
  - using mouse 89
- z-order
  - adjusting element order 124
  - changing tab order 188
  - changing within path graphic 146–147
  - linking path graphic elements by 31

