Dashboards and Presentation Design User Guide

SAP

Table of Contents

1   Document History

The following table provides an overview of the most important document changes:

Version	Date	Description
SAP BusinessObjects Dashboards 4.0	November 2010	First release of this document.
SAP BusinessObjects 4.0 Feature Pack 3	March 2012	Documentation added for hierarchical support, OLAP and BEx queries, and Report-Report Interface. • *Using selector components* [page 79] • *To add or edit queries* [page 218] • *Report-Report Interface* [page 220]
SAP BusinessObjects 4.0 Support Package 4	June 2012	Documentation updated for filter prompts. • *To set options on filter prompts* [page 229]
SAP BusinessObjects 4.0 Support Package 5	September 2012	Documentation added for mobile support. Documentation updated for the components and object browsers, previews, and saving to platform. • *Mobile support* [page 267] • *Using the Components browser* [page 17] • *Using the Object Browser* [page 20] • *To preview models* [page 170] • *To save models to BI platform folders* [page 246]
SAP BusinessObjects 4.0 Support Package 7	August 2013	Documentation updated for mobile support.

2   Introduction to Dashboards and Presentation Design

Dashboards is data visualization software that allows you to create and export interactive dashboards from Excel spreadsheets or other external data sources. These dashboards contain various components, such as charts, gauges, and dials, that are bound to data sources. The components display the data in a compact and visual manner.

Presentation Design is a presentation tool that enables users to create and export interactive dashboards from Excel spreadsheets.

Both products have much in common and are documented in this single user guide. Both products use the term *model* to describe the dashboard that is being designed.

Both products also include a fully functioning Excel spreadsheet. Data and supported Excel formulas can be imported or entered directly into the embedded spreadsheet and then modified as required. In addition, Dashboards includes the ability to link external data sources, such as data stored in SAP BusinessObjects Business Intelligence platform (the *BI platform*) repositories, directly to the embedded spreadsheet or components in the model.

2.1   Product editions

This section lists the features included in each product edition. All editions have the same basic functionality; the main differences between editions are the formats that models can be exported to, the ability to access and configure external data connections, and the number of external data connections that are supported.

- SAP Crystal Presentation Design enables users to create models with Microsoft Excel spreadsheets. These models cannot be exported to Flash.
- SAP Crystal Dashboards, personal edition includes all the features of SAP Crystal Presentation Design, plus it enables users to create models and connect them to a maximum of two external data sources. These models can also be exported to Flash.
- SAP Crystal Dashboards, departmental edition includes all the features of the personal edition, plus it enables users to create models with an unlimited number of data sources and deploy them as a standalone model, or to SAP Crystal Reports Server.
- SAP BusinessObjects Dashboards enables users to create models with an unlimited number of data sources and deploy them as standalone models or to SAP BusinessObjects Business Intelligence platform (the BI platform), SAP BusinessObjects Edge Series, or SAP NetWeaver BW environments.

Throughout this guide, these four product versions are collectively referred to as Dashboards.

The following table summarizes the components available in each edition of the software:

Component	SAP Crystal Presentation Design	SAP Crystal Dashboards, personal edition	SAP Crystal Dashboards, departmental edition	SAP BusinessObjects Dashboards
Basic data presentation components	☑	☑	☑	☑
Themes and color schemes	☑	☑	☑	☑
Play Control, Play Selector, and Accordion Menu		☑	☑	☑
Interactive Calendar, Panel Set, History, Trend Analyzer		☑	☑	☑
Reporting Services button			☑	☑
Slide Show, Connection Refresh button, SWF Loader		☑	☑	☑
Query Refresh button, Query Prompt Selector			☑	☑

The following table summarizes miscellaneous features available in each edition of the software:

Feature	SAP Crystal Presentation Design	SAP Crystal Dashboards, personal edition	SAP Crystal Dashboards, departmental edition	SAP BusinessObjects Dashboards
Data connectivity	None	Web Service (2 connections maximum) XML Data (2 connections maximum) Flash Variables Crystal Reports FS Command External Interface	Personal Edition connections plus: Web Service (unlimited) XML Data (unlimited) Portal Data LCDS Connections Live Office	Departmental Edition connections plus: Query as a Web Service SAP NetWeaver BW
BI platform connectivity	Not available	Not available	SAP Crystal Reports Server 2011	SAP BusinessObjects

Feature	SAP Crystal Presentation Design	SAP Crystal Dashboards, personal edition	SAP Crystal Dashboards, departmental edition	SAP BusinessObjects Dashboards
				Business Intelligence platform 4.0
Viewing license required for connected dashboards	Not available	Not available	SAP Crystal Dashboard Viewing option	Xcelsius Interactive viewing license
Limitations	Does not support external data connections	Maximum of 2 Web Service or XML Data connections total in any one model	Maximum of 100 named users can view a given dashboard

SAP BusinessOne is the only supported SAP application

Cannot be used with SAP BusinessObjects Business Intelligence platform or SAP Edge BI | None |
| Mobile support | Not available | Not available | Not available | Models can be viewed on supported mobile devices |

2.2 File formats and object types

Dashboards models conform to the Adobe Flash file format, SWF. SWF is the vector-based graphics format designed to run in the Adobe Flash Player. Because a Flash file is vector-based, its graphics are scalable and play back smoothly on any screen resolution, and the file usually has a smaller file size than other animation formats.

When saved locally, Dashboards files are stored in a zipped format with an XLF extension. The XLF file format contains the model information and its associated embedded spreadsheet file. This file must be opened in the designer (Dashboards or Presentation Design).

> **i Note**
>
> Newer Abode Flash versions have security features that may prevent models from connecting to an external data source when running them outside the designer. If you cannot connect to an external data source, see *Adobe Flash Player Security Restrictions* [page 264]for more information.

When saved to a BI platform repository, Dashboards models are saved as Dashboards objects. These objects contain both the design document (XLF) and the Flash file (SWF) in a single object.

If the object is opened in Dashboards or in the translation management tool and modified, the SWF cached in the Dashboards object is automatically updated to reflect the changes. You do not need to re-export the Flash object separately for users to receive the latest changes.

> **i Note**
>
> The BI platform repository may also contain Xcelsius objects and Flash objects created in earlier versions of the software. For more information about using and importing these documents to Dashboards, see *Importing files created in previous versions of the software* [page 32].

Related Links

Adobe Flash Player Security Restrictions [page 264]

Importing files created in previous versions of the software [page 32]

To import XLF files created in Xcelsius 4.5 or Xcelsius 2008 [page 33]

2.3   About this document

Since SAP BusinessObjects Dashboards, SAP Crystal Dashboards, and SAP Crystal Presentation Design share many of the same features, they are documented together in this user guide. If a feature or function is available only in a specific product edition, a note at the beginning of the procedure indicates those restrictions. In addition, the table in the product edition section outlines the features and functions available in each edition.

This document includes procedures for creating models including: adding and configuring components, linking components to data, configuring external data sources, and exporting models to different formats. Use this document as a reference for creating basic and advanced models.

Locate information quickly

- Access online help from the program's *Help* menu.
- Access a PDF version of the Dashboards and Presentation Design documentation on the SAP Help Portal at *http://help.sap.com/*, under ▶ *SAP BusinessObjects* ▶ *All Products* ◢ .

Related Links

Product editions [page 6]

3 Getting Started

This section provides general information regarding: the Dashboards workspace; file extensions; sample models to help you learn the program's features; and notes on backwards compatibility.

3.1 Understanding the workspace

Before you begin designing models, take a moment to familiarize yourself with the interface. The workspace and its major features are illustrated below:
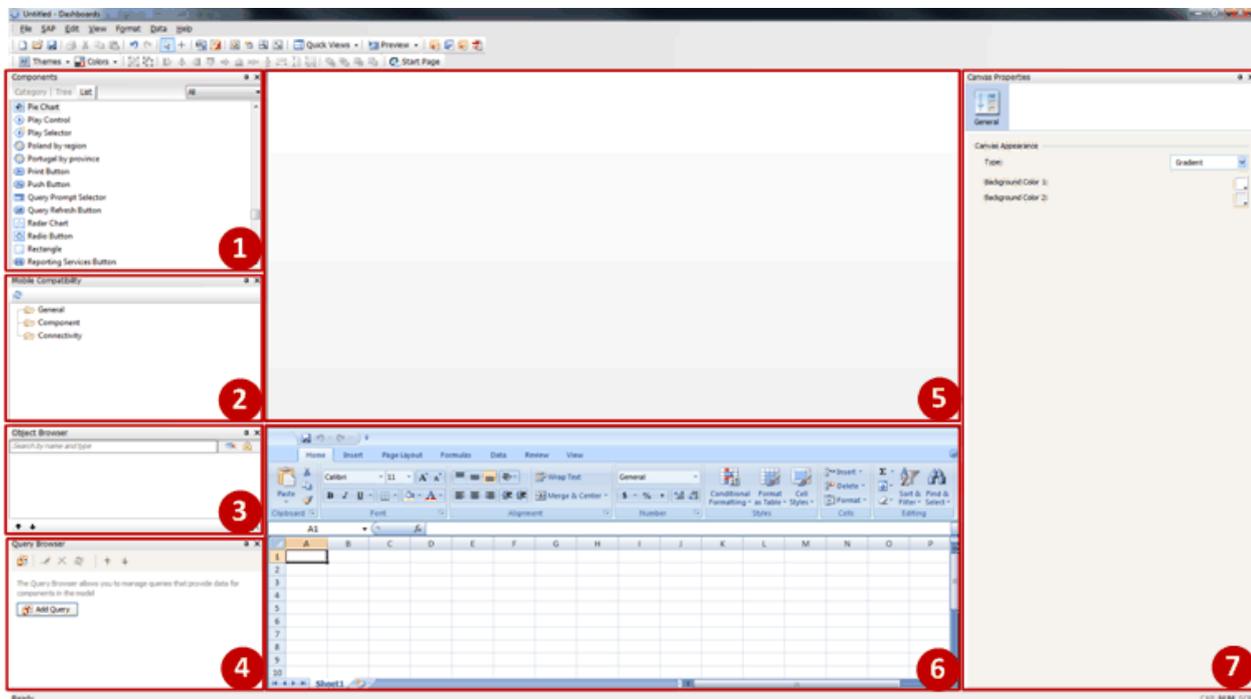
1. *Components Browser*
   For more information, see *Using the Components browser* [page 17].
2. *Mobile Compatibility*
   For more information, see *Using the Mobile Compatibility panel* [page 267].

   > ℹ Note
   >
   > This is not available in Presentation Design.

3. *Object Browser*
   For more information, see *Using the Object Browser* [page 20].
4. *Query Browser*
   For more information, see *To build a query* [page 221].

   > ℹ Note

> This is not available in Presentation Design.

5. *Canvas*
   For more information, see *Using the Canvas* [page 23].
6. *Embedded spreadsheet*
   For more information, see *Understanding the embedded spreadsheet* [page 173].
7. *Properties* panel
   Available tabs and options on the Properties panel vary based on what component is currently selected. For general information on the Properties panel, see *Using the Properties panel* [page 36]. For a comprehensive list of the options that are available for each component, see the individual component categories.

> **i Note**
>
> The title of this panel reflects the component you currently have selected.

3.1.1 Toolbars and buttons

The software includes five toolbars that you can use when working with models: standard, theme, export, format, and start page.

To show or hide toolbars

- To toggle between showing and hiding a toolbar, click ▶ *View* ❯ *Toolbars* ❯ and click the toolbar name. A checkmark beside a toolbar name indicates that the toolbar is currently showing.

Standard toolbar

The standard toolbar contains buttons for completing general tasks. The following table describes the available buttons and their keyboard shortcuts, if any:

Icon	Description	Keyboard Shortcut
	New Document	CTRL + N
	Open File	CTRL + O
	Save	CTRL + S
	Print	CTRL + P
	Cut	CTRL + X
	Copy	CTRL + C

Icon	Description	Keyboard Shortcut
	Paste	`CTRL` + `V`
	Undo	`CTRL` + `Z`
	Redo	`CTRL` + `Y`
	Selection Tool	
	Component Tool	
	Import Spreadsheet	`CTRL` + `Shift` + `I`
	Manage Connections	`CTRL` + `M`
	Increase Canvas	
	Decrease Canvas	
	Fit Canvas to Components	
	Fit Canvas to Window	
	Quick Views	
	Toggle Preview Mode	`CTRL` + `Enter`

Themes toolbar

The themes toolbar contains the following buttons for working with themes:

Icon	Description
	Change Theme
	Change Color Scheme

Export toolbar

The export toolbar contains the following buttons for exporting your model to specific formats:

Icon	Description
	Export to PowerPoint

Icon	Description
	Export to Microsoft Word
	Export to Microsoft Outlook
	Export to Adobe PDF

Format toolbar

The format toolbar contains buttons for adjusting components on the canvas. The following table describes the available buttons and their keyboard shortcuts, if any:

Icon	Description	Keyboard Shortcut
	Group Components	`CTRL` + `G`
	Ungroup Components	`CTRL` + `Shift` + `G`
	Align Left	
	Center Horizontally	
	Align Right	
	Align Top	
	Center Vertically	
	Space Evenly Across	
	Space Evenly Down	
	Make Same Width	
	Make Same Height	
	Make Same Size	
	Bring to Front	`CTRL` + `NumPad +`
	Send to Back	`CTRL` + `NumPad -`
	Bring Forward	`NumPad +`
	Send Backward	`NumPad -`

Start Page toolbar

The start page toolbar only contains one button, which is used to toggle the start page. The start page provides shortcuts to create new files or to open existing files, and it also displays any online news relevant to Dashboards.

Icon	Description
	Open Start Page

3.1.2   To set document properties

You can set various document properties when designing a model. Document properties are unique to the model currently being worked on.

1. Click ▶ *File* ▶ *Document Properties* ◢.
2. In the *Document Properties* dialog box, set the following options:

Property	Description

Canvas size in pixels	Select one of the following:
	
	○ *Preset Size*
	   Select one of the options from the drop-down menu to set the canvas size.
	○ *Custom Size*
	   Manually enter the desired height and width of the canvas in the appropriate text boxes.
	
	For more information about resizing the canvas, see *Changing the canvas size* [page 25].

Use Global Font	When this option is enabled, you can set a global font that is used on all components. This option is disabled by default. For more information, see *To set global font options* [page 29].
Show Loading Status	When this option is enabled, more detailed text is displayed above the loading bar when the model is being previewed or loaded. This option is disabled by default.
Description	Enter a description of the model. It can be useful to write a description for a complex model, or a model that is regularly updated by more than one designer.
Spreadsheet	When *Treat Zero as Blank Cell* is enabled, the embedded spreadsheet, and components bound to the embedded spreadsheet, will treat cells with the number 0 as blank. This option is disabled by default. ℹ Note If your model uses numeric Excel formulas, it is recommended that you disable this option.

3. Click *OK*.

3.1.3   To set preferences

You can set various preferences when working in Dashboards. Certain preferences apply only to the model currently being worked on, while other preferences apply to the product as a whole and apply to all models.

1. Click ▶ *File* ❯ *Preferences* ◢ .
2. In the *Preferences* dialog box, there are several different tabs:

Option	Description
Document	On this tab, set the following options: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*Canvas size in pixels*</td><td>Select one of the following: ○ *Preset Size*. Select one of the options from the drop-down menu to set the canvas size. ○ *Custom Size* Manually enter the desired height and width of the canvas in the appropriate text boxes. For more information about resizing the canvas, see *Changing the canvas size* [page 25].</td></tr><tr><td>*Components*</td><td>Set the following options: ○ *Default Theme*. The selected theme will automatically be applied to all new models. ○ *Default Color Scheme* The selected color scheme will automatically be applied to all new models. For more information about themes and color schemes, see *Using themes* [page 167] and *Using color schemes* [page 167].</td></tr><tr><td>*Queries*</td><td>When designing a model with BI platform data, you can connect to the platform and retrieve sample data. When *Save Query Result with Document* is enabled, the sample data is stored in the saved XLF file.</td></tr></table>
Grid	On this tab, set the following options: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*Show Grid*</td><td>When this option is enabled, a grid is displayed on the canvas.</td></tr><tr><td>*Snap to Grid*</td><td>When this option is enabled, components automatically align to the nearest gridlines.</td></tr><tr><td>*Width*</td><td>Specify the spacing between grid columns in pixels.</td></tr></table>

Option	Description

Option	Description
Height	Specify the spacing between grid rows in pixels.

For more information about the grid, see *To display a grid on the canvas* [page 24].

Option	Description
Open	On this tab, you can select the default folder that is displayed when you click ▶ *File* ❯ *Open* ◣: ○ *Last Folder Open* ○ *Folder* Manually select the filepath of the default folder.
Languages	On this tab, set the following options:

Option	Description
Current Language	Select Dashboards' language from the drop-down menu. Only installed languages are displayed.
Default Currency	Set the following options: ○ *Negative Values* Specify the format to use for negative currency values. ○ *Decimal Places* Select the number of decimal places to display for currency values. ○ *Prefix* If selected, type the symbol to appear before the numbers in currency values. ○ *Suffix* If selected, type the symbol to appear after the numbers in currency values.

For more information about currency settings, see *To set default currency formats* [page 250].

Option	Description
Excel Options	On this tab, set the following options:

Option	Description
Live Office Compatibility	When this option is enabled, it allows you to work with Live Office enabled spreadsheets within Dashboards.
Maximum Number Of Rows	Set the maximum number of rows that can be referenced by a formula or a component binding.
Optimize Spreadsheet	When this option is enabled, the exported model performs better at runtime, but takes longer to load.
Ignore Excel Formula Errors	When this option is enabled, cells containing invalid Excel formulas are treated as an empty cell.

For more information about Excel options, see *To set preferences for the embedded spreadsheet* [page 174].

3. Click *OK*.

3.1.4 Using the Components browser

The *Components* browser lists all the components that are available to add to your model. From this list you can drag components onto the canvas to add them to your model.

You can view the list of components grouped into functional categories (*Category* or *Tree* view) or as an alphabetical list (*List* view). Components are grouped into the following functional categories:

Category	Description
Favorites	A list of components you added to this list. **ⅈ Note** The *Favorites* category is available only in *Category* and *Tree* views. • To add components to the favorites list, in *Category* or *Tree* view, right-click the component and select *Add to Favorites*. • To remove components from the favorites list, in the favorites list, right-click the component and select *Remove From Favorites*.
Charts	Chart components allow you to create visual representations of data to make it easy for dashboard consumers to see comparisons, patterns, and trends.
Containers	Container components group and display other components. **ⅈ Note** Container components are not available in Presentation Design.
Selectors	Selector components allow the dashboard consumer to dynamically select data at runtime.
Single Value	Single-value components can be linked to a single cell in the spreadsheet and can either allow dashboard consumers to change the value in that cell or display the product of a formula from that cell.
Maps	Map components provide geographical representations that allow you to display data by region.
Text	Text components can be used to add labels to your model, or to allow the dashboard consumer to enter text at runtime.
Other	The Other category includes various components that can enhance the model, such as calendars, trend icons, panel sets, and print buttons.
Art & Backgrounds	Art & Backgrounds components allow you to visually enhance your model by adding images and backgrounds.
Web Connectivity	Web connectivity components allow data retrieval at runtime to by linking models to the Internet.
Universe Connectivity	Universe connectivity components allows users to interact with Universe connections such as refreshing data and entering prompt values. **ⅈ Note** Universe Connectivity components are not available in Presentation Design.

Category view

In *Category* view, the components are grouped in a sliding accordion menu. To navigate to the components, click a category title or scroll down to open each category in sequence. Each category displays a visual representation of the components.
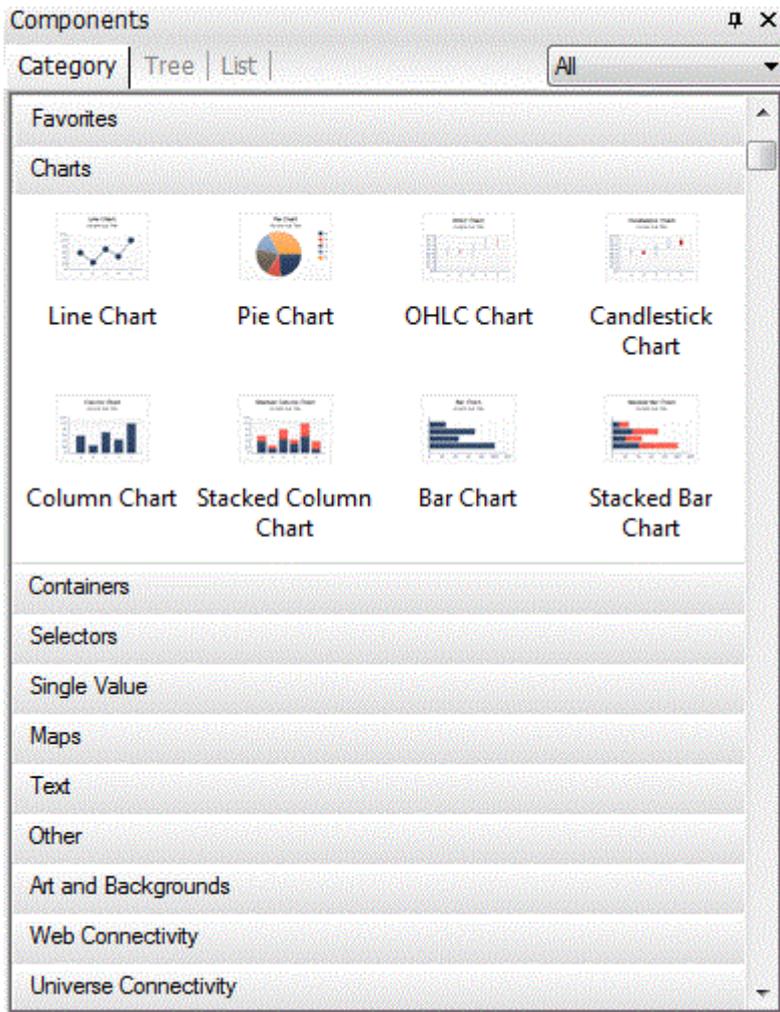
Tree view

In *Tree* view, the components are organized in folders according to their category. To navigate to components, click the folders to expand the categories and see the list of available components.
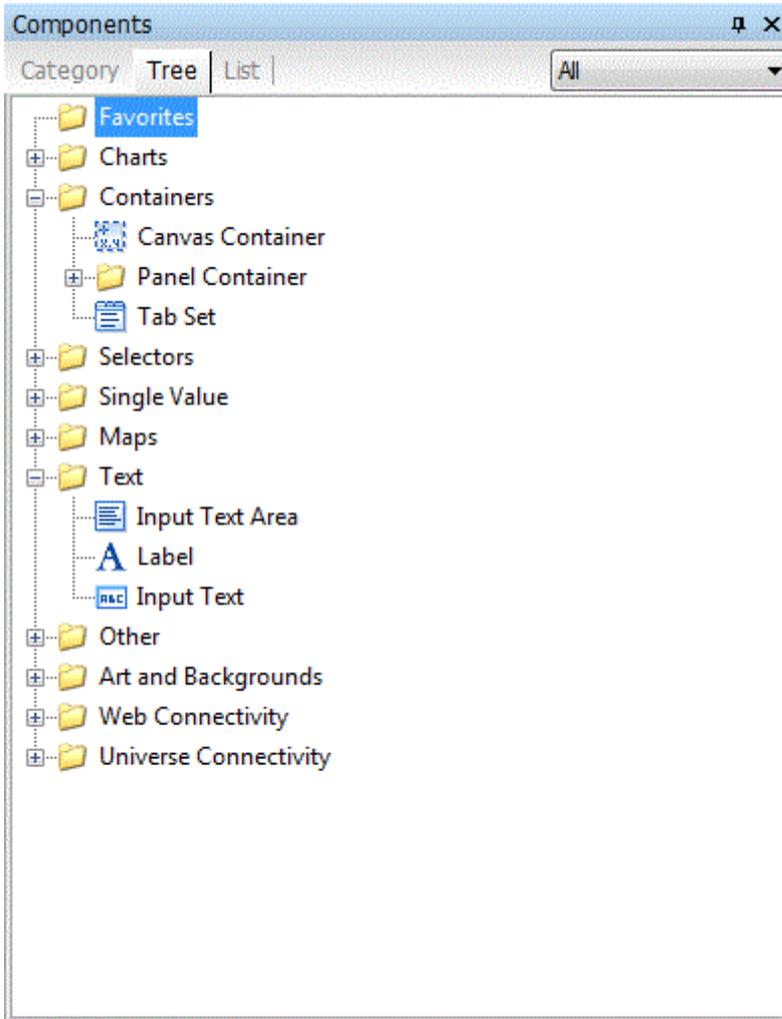
List view

In *List* view, all components are organized alphabetically and are not grouped by functionality.

To navigate to components in this view, click anywhere within the window, and type the first letter of the component or scroll up or down to find the desired component.
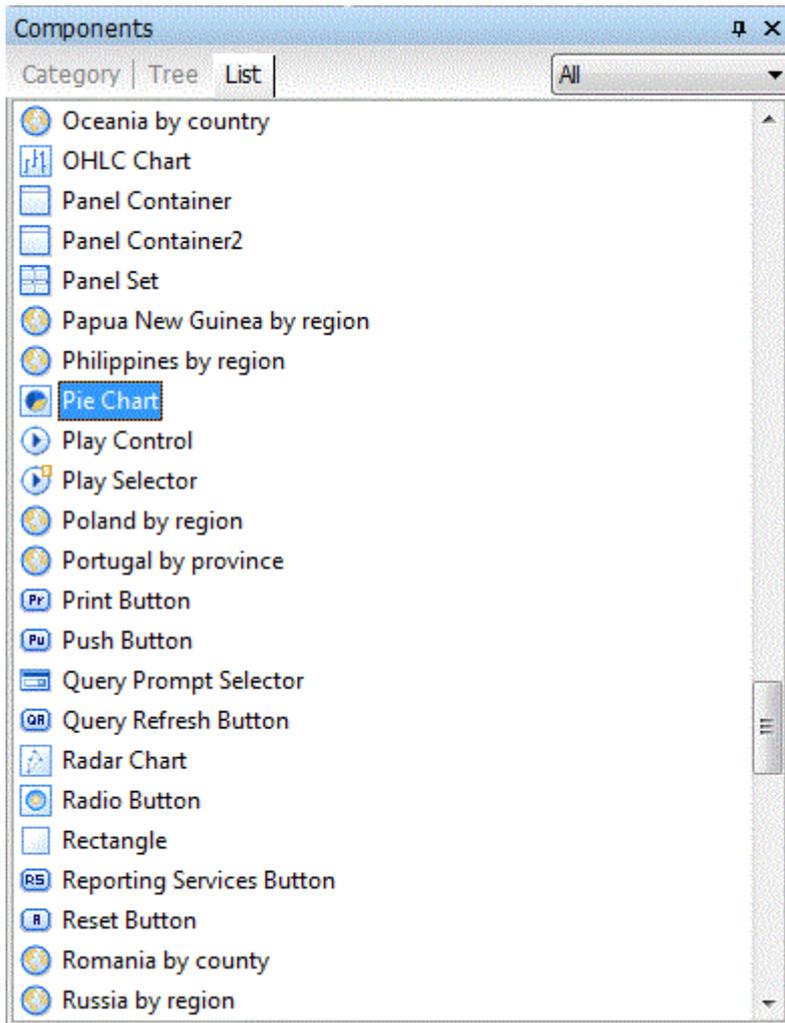
> **i Note**
>
> To view only components supported on mobile devices, click the drop-down arrow in the top right corner of the *Components* browser and select *Mobile Only*. For more information about creating dashboards for mobile devices, see *Mobile support* [page 267].

3.1.5 Using the Object Browser

The *Object Browser* provides a list of the components that are currently part of your model. You can use the *Object Browser* to do the following:

- Search for components.
- Select, group, and ungroup components.
- *Lock* or *Hide* components on the canvas.
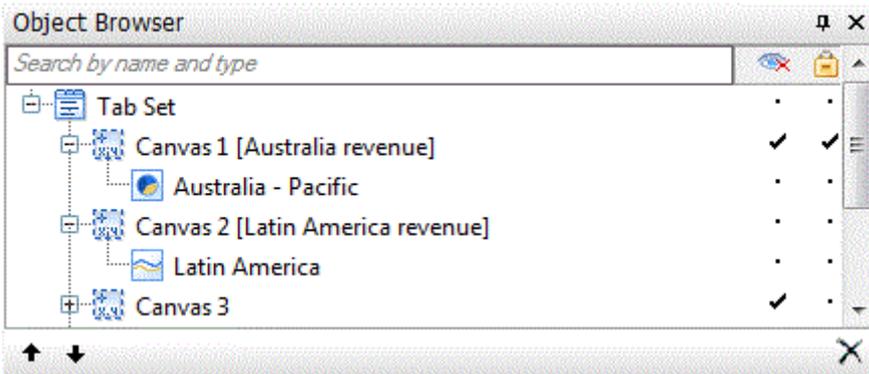
Related Links

To group and ungroup components [page 28]

3.1.5.1   Selecting components

To select a component in the *Object Browser*, click the component name or icon to make it the active component on the canvas.

To select multiple components, click a component name or icon, and then hold down the *CTRL* key while you click additional components.

Select all components by selecting any component, then pressing *CTRL+A*.

> **i** Note
>
> On the canvas, you cannot select components that are either placed completely off the canvas, or marked as hidden or locked. You can only select those from the *Object Browser*.

3.1.5.2   Searching for components

You can search for a component in the *Object Browser* either by its type, such as *Tab Set* or *Pie Chart*, or by its assigned name, such as *Tab Set 1* or Latin American sales.

To search for a component in the *Object Browser*, click the search field, type the component type or assigned name, and press `Enter`.

3.1.5.3   Renaming components

When you place a component on the canvas, it is automatically assigned a name. This name appears in the title bar of the *Properties* panel and in the *Object Browser*. You can rename components as required.

1. In the *Object Browser*, click the component you want to rename.

2. Click the name of the component.
   The name is highlighted and the cursor is in the text box.

3. Type a name for the component and press ⌷Enter⌷.

3.1.5.4 Hiding components

Components and groups can be hidden on the canvas to prevent any interaction with them. This can be helpful if you have many components on the canvas, and you want to hide some while you modify others. Hidden components are still part of the model and retain all their formatting, but they are not visible in design mode.

> ℹ Note
>
> If you preview or export the model, the exported file shows hidden components with their stacking order preserved.

To hide or show components on the canvas, in the *Object Browser*, do any of the following:

To	Do this
Hide a component or group	Beside the component or group, click the dot in the column under the *Hide* icon ( ).
Hide all components and groups	In the *Object Browser* toolbar, click *Hide* ( ).
Show a component or group	Beside the hidden component or group, click the check mark in the column under the *Hide* icon ( ).
Show all components and groups	If all items are hidden, in the *Object Browser* toolbar, click *Hide* ( ). If not all items are hidden, in the *Object Browser* toolbar, click the *Hide* icon to hide all components and then click it again to show all components.

3.1.5.5 Locking Components

Components and groups can be locked on the canvas to prevent any interaction with them. This can be helpful when there are many components laid out on the canvas, or if you have a large group of components and want to modify newly added components. The component or group will be selectable only in the *Object Browser*.

To lock or unlock components on the canvas, in the *Object Browser* do any of the following:

To	Do this
Lock a component or group	Beside the component or group, click the dot in the column under the *Lock* icon ( ).

To	Do this
Lock all components and groups	In the *Object Browser* toolbar, click *Lock* (🔒).
Unlock a component or group	Beside the locked component or group, click the check mark in the column under the *Lock* icon (🔒).
Unlock all components and groups	If all items are locked, in the *Object Browser* toolbar, click *Lock* (🔒). If not all items are locked, in the *Object Browser* toolbar, click *Lock* (🔒) to lock all components and then click it again to unlock all components.

3.1.6   Using the Canvas

The canvas area is the main work area where you place and manipulate components to create models.

You can change the color of the canvas background and add a grid to help you arrange items on the canvas. You can also adjust the size of the canvas to give you more space to work or to reduce the overall size of a saved model.

3.1.6.1   To set the canvas background

Depending on the current theme, the canvas has a solid or gradient background.

> i Note
>
> If you want a transparent background, you must set the *Canvas Appearance Type* to **None**.

1. Open or create a model.
2. If the *Canvas Properties* panel is not open, click the canvas and then click ▶ *View* ❯ *Properties* ◣.
3. On the *Canvas Properties* panel, set the following options:

Option	Description
Type	Select one of the following types of background: | Option	Description
Solid	A single solid color
Gradient	Two colors that fade together from top to bottom	

Option	Description		
	Option		**Description**
	Image		A JPG or SWF file. Select the file to import as the background
	None		A transparent background
Background Color	If you selected the solid or gradient type, click the *Background Color* box to set the colors to use in the background. For solid backgrounds, select a single color. For gradient backgrounds, in *Background Color 1*, select the color to use at the top of the model and in *Background Color 2*, select the color to use at the bottom.		
Import	If you selected the image type, click *Import* to select the image file to use. **ⓘ Note** The selected image is stretched to fit the canvas. You can manually adjust the canvas size to avoid distortion of the image.		

Related Links

3.1.6.2 To display a grid on the canvas

When you are arranging components on the canvas, it can be helpful to line items up along grid lines. You can use the grid manually to position each component or you can have components snap to the closest gridlines automatically when they are placed on the canvas.

1. Click ▶ *File* ❯ *Preferences* ◢.
   The *Preferences* dialog box opens.
2. On the left side of the dialog box, click *Grid*.
3. In the *Grid* area, set the following options:

Option	Description
Show Grid	Select to display a grid on the canvas.
Snap To Grid	Select to have components automatically align to the nearest gridlines.
Width	Specify the spacing between grid columns in pixels.
Height	Specify the spacing between grid rows in pixels.

4. Click *OK*.

Grid lines are not displayed when you preview or export the model.

24 **Getting Started**

3.1.6.3 Changing the canvas size

You can change the size of your work area by changing the canvas size. The software provides the following methods of changing the canvas size:

● Set a specific size in pixels
● Fit the canvas to the window or to components
● Increase or decrease the canvas size incrementally

When you are working on a model, you might want to expand the canvas size as much as possible to allow for more room to add and arrange the various components. When your model is complete, you might want to reduce the canvas size around the components before publishing the model to reduce the overall size of the file.

> **i Note**
>
> When you export your model, the software embeds the entire canvas in the exported file. The canvas size might affect your ability to size and position your embedded model, as well as the time required to load the model.

3.1.6.3.1 To set the canvas size

If you want to set a specific size for the canvas, adjust the document properties. Setting a specific size is useful if you want a specific display ratio for your model, or if you are using an image as your canvas background.

1. Click ▶ *File* ▶ *Document Properties* ◢.
   The *Document Properties* dialog box opens.
2. In the *Canvas Size in Pixels* area of the dialog box, do one of the following:
   ○ Select *Preset Size* then, from the list, select one of the available sizes.
   ○ Select *Custom Size*, then in the *Width* box, enter the desired canvas width in pixels and in the *Height* box, enter the desired canvas height in pixels.
3. Click *OK*.

3.1.6.3.2 To fit the canvas to the components or the window

When you are working on your model, it can be useful to expand the canvas as much as possible to allow for more room to add and arrange components. To stretch the canvas to the current workspace window, do one of the following:

● On the *Standard* toolbar, click *Fit Canvas to Window* ().
● Click ▶ *View* ▶ *Canvas Sizing* ▶ *Fit Canvas to Window* ◢.

After you complete your model, you can fit the canvas tightly around all the components - this reduces the file size and loading time of the exported model. To shrink the canvas to the current components, do one of the following:

● On the *Standard* toolbar, click *Fit Canvas to Components* ().
● Click ▶ *View* ▶ *Canvas Sizing* ▶ *Fit Canvas to Components* ◢.

Related Links

To increase or decrease the canvas size [page 26]

3.1.6.3.3 To increase or decrease the canvas size

You can fine-tune the size of your canvas by increasing or decreasing its size incrementally. This can be useful if you want to add a border around your model after you fit it to the components.

- To increase the canvas width and height by 10 pixels each, do one of the following:

  - On the *Standard* toolbar, click *Increase canvas* ( ).
  - Click *View* *Canvas Sizing* *Increase Canvas* .

- To decrease the canvas width and height by 10 pixels each, do one of the following:

  - On the *Standard* toolbar, click *Decrease canvas* ( ).
  - Click *View* *Canvas Sizing* *Decrease Canvas* .

Related Links

To fit the canvas to the components or the window [page 25]

3.1.6.4 Arranging components on the canvas

The software offers several options for aligning and positioning components relative to the page or to other components. You can also change the stacking order of components on the canvas if some components overlap.

If you have numerous components on the canvas, you can group them and work with those components as though they were a single item.

3.1.6.4.1 To place components on the canvas

To place a component on the canvas, do one of the following:

- From the *Components* browser, drag the component onto the canvas. An outline of the component's default size is shown as you drag it onto the canvas.
- In the *Components* browser, click the component, and then click the location on the canvas where you want to place the component. Note that the cross-hairs (+) represent the top left corner of the component.

> **i Note**
>
> If you set a grid on the canvas and have selected the *Snap to Grid* option, components are automatically aligned with the nearest grid lines when you place them on the canvas.

Related Links

To display a grid on the canvas [page 24]

3.1.6.4.2   To align multiple components

You can align the edges or centers of components relative to the first component in the selection.

1. Select a component.
2. Hold the `CTRL` key and select the components you want to align with the first component.
3. Click ▶ *Format* ❯ *Align* ◣ and then click one of the following options:

Option	Description
⊫ Left	Aligns the left edge of the selected components with the left edge of the component that was selected first.
✛ Center	Aligns the center of the selected components vertically with the center of the component that was selected first.
⊣ Right	Aligns the right edge of the selected components with the right edge of the component that was selected first.
⊤ Top	Aligns the top edge of the selected components with the top edge of the component that was selected first.
⊹ Middle	Aligns the center of the selected components horizontally with the center of the component that was selected first.
⊥ Bottom	Aligns the bottom edge of the selected components with the bottom edge of the component that was selected first.

These options are also located on the *Formatting* toolbar.

3.1.6.4.3   To space components evenly on the canvas

You can distribute multiple components on the canvas to have the same amount of space between their center points either vertically (down) or horizontally (across).

> **i Note**
>
> The Space Evenly option adjusts the space between the center points of components, not between their edges.

1. Select a component.
2. Hold the `CTRL` key and select the components you want to space evenly with the first component.
3. Click ▶ *Format* ❯ *Space Evenly* ◣ and then click one of the following options:

Option	Description	
⊡ Across	Adjusts the component positions so there is an equal amount of horizontal space between their center points.	
⊟ Down	Adjusts the component positions so there is an equal amount of vertical space between their center points.	

These options are also located on the *Formatting* toolbar.

3.1.6.4.4   To copy size settings to other components

To quickly adjust the sizes of multiple components, you can copy the height and width settings of one component to other components on the canvas.

1. Select a component.
2. Hold the `CTRL` key and select the components you want to make the same size as the first component.
3. Click ▶ *Format* ▶ *Make Same Size* ◀ and then click one of the following options:

Option	Description
⊢⊣Width	Sets the width of each of the selected components to the width of the first component.
Height	Sets the height of each of the selected components to the height of the first component.
Both	Sets both the height and the width of each of the selected components to the height and width of the first component.

These options are also located on the *Formatting* toolbar.

3.1.6.4.5   To center components on the canvas

You can center components or groups of components on the canvas horizontally, vertically, or both.

1. Select one or more components. To select multiple components, hold the `CTRL` key and select the components you want to center on the canvas.
2. Click ▶ *Format* ▶ *Center in Canvas* ◀ and then click one of the following options:

Option	Description
Vertically	Centers the selected components between the top and bottom margins of the canvas.
Horizontally	Centers the selected components between the left and right margins of the canvas.
Both	Centers the selected components on the canvas.

3.1.6.4.6   To group and ungroup components

When you are working on a complex model with multiple components, you may want to group some components together so they are not accidentally separated as you design the layout. Grouped components act as a single item and retain their relative positions when moved.

1. Select the components you want to group or ungroup.
2. Do one of the following:

   ○ Click ▶ *Format* ▶ *Group* ◀. You can also click *Group components* (⊡) on the *Formatting* toolbar.

   ○ Click ▶ *Format* ▶ *Ungroup* ◀. You can also click *Ungroup components* (⊡) on the *Formatting* toolbar.

28   Dashboards and Presentation Design User Guide
Getting Started

The selected components are either grouped so a single set of boundary markers surround all the grouped components and the group can be moved as a single item, or ungrouped so each component has boundary markers around themselves and can be moved independently of the other components.

3.1.6.4.7   To change the stacking order of components

On the canvas, components and grouped components are positioned in layers. Components on a front layer cover or overlap any components that are on the layers behind. When you are designing your model, you may want to move components from back to front so you can focus on each one.

1. Select a component.
2. Click ▷ *Format* ▷ *Order* ◿ and then click one of the following options:

Option	Description
Bring to Front	Moves the selected components to the top layer.
Bring Forward	Moves the selected components one layer closer to the top from its current position.
Send Backward	Moves the selected components one layer closer to the bottom layer from its current position.
Send to Back	Moves the selected components to the bottom layer.

These options are also located on the *Formatting* toolbar.

3.1.7   To set global font options

By default, you can specify different fonts for different components. However, you can set up a global font so the same font is used for all components. The global font can be based on device fonts (fonts available on your machine) or on embedded fonts (fonts embedded in the exported SWF file).

When device fonts are used, the exported file is smaller, but the text cannot be rotated and does not use anti-aliasing. When fonts are embedded, you can rotate text and the text uses anti-aliasing, but the exported file size is larger and the model takes longer to load.

> **i Note**
>
> If you set different fonts for different components, the fonts used are all device fonts. So, if the specified fonts are not available when the model runs (on a different computer), another font is substituted for the missing font and the model may appear different than when it was created.

1. Click ▷ *File* ▷ *Document Properties* ◿.
   The *Document Properties* dialog box opens.
2. In the *Document Properties* dialog box, select *Use Global Font*.
   The options for setting up global fonts are enabled.
3. From the list, select the font you want to use for all components.
4. Select one of the following options:

Option	Description
Use Device Fonts	Fonts are not embedded in the exported file. Device fonts use the Flash player to render the specified TrueType fonts. If a specified font is not available at the time the model runs, another similar font is used. By default, the software uses device fonts. ➡ Tip If a model requires the extended character set defined by Unicode, it is recommended that you use device fonts.
Use Embedded Fonts	Fonts used in models will always be available when the model runs, text can be rotated, and anti-aliasing is used. ℹ Note Embedding the fonts increases the file size and the amount of time required to load the model. To change the text characters that are embedded in the exported file, click *Advanced*, select the characters you want to embed, and click *OK*. ⚠ Restriction Asian characters sets cannot be embedded.

5. Click *OK*.

3.1.8   Using Quick Views

You can use *Quick Views* to quickly access different configurations of the workspace.

Access Quick Views by clicking ▶ *View* ❯ *Quick Views* ◤. The Quick Views are as follows:

○ *My Workspace*
Select this Quick View to view the workspace as you last configured it.
○ *Canvas Only*
Select this Quick View to view the canvas only. This is best suited to arrange components on the canvas during the early stages designing your model.
○ *Spreadsheet Only*
Select this Quick View to view the spreadsheet only. This is best suited to verify the correctness of the spreadsheet data, to create blank cells, or to put formulas in cells for later use with components. This view can also make it easier to find specific cells if your spreadsheet is very large.
○ *Canvas and Spreadsheet*
Select this Quick View to view the canvas and spreadsheet only. This gives the most space for you to select components and bind them to spreadsheet cells if you open the *Properties* panel after switching to this view.

> ➡ Remember
>
> Select a component and press `Alt` + `Enter` to open the *Properties* panel.

You can also access these views by clicking *Quick Views* (🖳) on the *Standard* toolbar.

3.1.9 Using Auto Hide

The *Auto Hide* feature automatically collapses panels when they are not being used, providing more screen space. The Auto Hide function is available for the *Properties*, *Object Browser*, *Components*, and *Query Browser* panels.

Click *Auto Hide* () at the top of a panel to enable auto hide and collapse the panel. Expand a collapsed panel by clicking or pointing to the panel name on the left or right side of the workspace. When Auto Hide is enabled for a panel, the panel will automatically collapse when you click or point outside the panel.

3.1.10 Docking and undocking panels

You can move the *Components* browser, *Object Browser*, *Query Browser*, and the *Properties* panel from their default locations and dock them in other areas.

To dock a panel in another area of the workspace, click the top of the panel and drag it onto one of the docking icons (for example,) that appear. When you move your cursor over a docking icon, an area is highlighted that shows where the panel will dock. Each icon has a unique docking location.

To undock a panel so it is in a window that is separate from the workspace, drag the panel to any place away from a docking icon.

3.2 Sample models

As you are learning how Dashboards works and what it can do, it can be helpful to examine sample models as they appear during runtime, as well as the underlying configuration and bindings of their components.

The software includes several sample models that illustrate how some of the components and functions work. Each sample includes its own embedded spreadsheet data, so you can see how the components are bound to the data in the spreadsheet. Use these samples to help you understand how to configure similar functions in your own model, or even use them as a starting point and customize them for your own needs.

> **i Note**
>
> Since samples include their own sample spreadsheet data, opening a sample file overwrites any data currently in the embedded spreadsheet.

Related Links

To view the sample models [page 32]

3.2.1 To view the sample models

1. Click ▶ *File* ❯ *Samples* ❯.
   The *Samples* dialog box opens.
2. In the *Category* list, click *User Guide Samples*.
   A list of the available samples opens in the *Items* list.
3. In the *Items* list, click a sample.
   A fully interactive preview of the model is displayed in the *Preview* pane.
4. Click *OK*.
   The sample file opens.
5. To view the model as the dashboard consumer would see it, click *Preview*.

3.3 Importing files created in previous versions of the software

If you have models (XLF files) created in Crystal Xcelsius 4.5 or Xcelsius 2008, you can import them into Dashboards or Presentation Design and take advantage of new features and functionality.

You cannot import files created in Xcelsius 3.0, 3.5, and 4.0. To use these files in Dashboards or Presentation Design, you must first import them into Crystal Xcelsius 4.5 and then import the files into Dashboards or Presentation Design.

The product minimizes the impact of migrating models from earlier versions; however, some models or components may appear to be different than they were in the original file.

The following differences are the most common:

- Component size. In most cases, the size difference is a few pixels.
- Custom colors may appear to be slightly different.
- The *Spreadsheet Table* component size and formatting.

In addition, if you are importing an XLF file created in Xcelsius 4.5 and it contains an external data connector (*XML Data Button*, *Excel XML Map*, *Web Service Connection*, *Live Office*, *FS Command*, or *Web service query (Query as a Web Service)*), the following actions occur for each connector or button:

- A connection is created in the *Data Manager* with the previous *Refresh Options* and *Load Status* settings.
- A *Connection Refresh Button* is added to the canvas and the *Trigger Behavior* settings are mapped to it.

In addition, for *Web Service* and *Web service query (Query as a Web Service)* connections, refresh options, and trigger behaviors are mapped as described in the following table:

Option in Xcelsius 4.5	Setting in Dashboards
Name	In the Data Manager, the connection name for migrated connections is set as follows: • For Web Services: SOAP • For Web Service query: Query as a Web Service

32   Dashboards and Presentation Design User Guide
Getting Started

Option in Xcelsius 4.5	Setting in Dashboards
Refresh on Load	• For versions prior to Xcelsius 2008 SP3, *Refresh on Load* is selected on the *Usage* tab in the *Data Manager* for the connection. • For Xcelsius 2008 SP3 and later, *Refresh Before Components are Loaded* is selected on the *Usage* tab in the *Data Manager* for the connection.
Refresh on Interval	*Refresh Every* is selected on the *Usage* tab in the *Data Manager* for the connection.
Trigger on Change Only	In the *Properties* panel for the *Connection Refresh Button*, the value in *Trigger Cell* maps to *Trigger Cell* on the *Behavior* view, and *When Value Changes* is selected.
Trigger On Any Insert	In the *Properties* panel for the *Connection Refresh Button*, the value in *Trigger Cell* maps to *Trigger Cell* on the *Behavior* view, and *When Value Changes* is selected. At runtime, when a new value is inserted into the trigger cell, the data refreshes.

3.3.1   To import XLF files created in Xcelsius 4.5 or Xcelsius 2008

⚠ Caution

Backup your current XLF, JPEG, and external SWF files before upgrading your model.

1. Click ▶ *File* ❯ *Open* ◣.
2. Navigate to the file you want to import and click *Open.*
   The software detects that the file is from the previous version of the product and a warning dialog box opens.

   ⚠ Caution

   After you import and save your model with Dashboards or Presentation Design, the model cannot be opened using an earlier version of the product.

The model opens on the canvas. Some components in the model may look slightly different from the original file.

If the components appear to be formatted differently, select the component and open the *Properties* panel. Re-bind any properties that are tied to the cells in the spreadsheet.

3.4 Customizing with add-ons

A variety of model components and feature extensions are available for all Dashboards versions. These items are called add-ons and have been created by SAP partners and customers. You can download charts, maps, and other model and feature extensions and use them to customize your models.

3.4.1 To download add-ons

Before you can install an add-on, you must download it to your system.

1. Click ▶ *File* ❯ *Manage Add-Ons...* ◢.
   The *Add-On Manager* dialog box opens.
2. Click *Get More Add-Ons* and, in the security dialog box, click *OK*.
   The *SAP EcoHub* website opens.
3. Under *Find an Add-on*, click *Visit the Marketplace to View All Add-ons.* Alternatively, you can click *this link* to go directly to the EcoHub.
   The website lists the available add-ons. Click the one you want to download.
4. Click *Check Out*.
   The *Sign In* dialog box opens.
5. In the *Sign In* dialog box, do one of the following:

   ○ Enter your sign-in information and click *Sign In*.

   ○ To register, click *Create New Account* and follow the instructions.
6. Read the terms of the offering and select *I agree to the Terms of Use* checkbox. Click *Checkout* and follow the online instructions to download the file.

The add-on is available for installation.

If the add-on was in a zipped folder, extract the files, and then install the downloaded add-on.

3.4.2 To install add-ons

After you download an add-on, you must install it before you can use it.

1. Click ▶ *File* ❯ *Manage Add-Ons...* ◢.
   The *Add-On Manager* dialog box opens.
2. Click *Install Add-On*.
3. Navigate to the location where the add-on files were saved and extracted. Select the XLF file and click *Open.*

The add-on is installed and available for use.

3.4.3   To remove add-ons

1. Click ▶ *File* ❯ *Manage Add-Ons...* ◣.
    The *Add-On Manager* dialog box opens.
2. From the list on the left, select the add-on you want to remove. Click *Remove Add-On.*
    The *Confirm Add-on Removal* dialog box opens.
3. Click *Remove.*

The add-on is removed from the software.

4 Working with Components

Components are the visual elements of your dashboard. They are designed to display or select large amounts of data in an efficient manner. Components are categorized by function: there are charts, selectors, maps, and several other categories to choose from. You can decide on the best component to use based on the type of data you are working with.

Components can be linked to either a query, or to cells on the embedded spreadsheet. These relationships can operate in both directions: while almost all components can receive data from a query or cells in the embedded spreadsheet, many others can also return data to the query, or write data to cells.

4.1 Using the Properties panel

The *Properties* panel contains the setting and format options for the selected component. After you place a component on the canvas area, you can access the *Properties* panel is several different ways:

- Double-click the selected component.
- Right-click the component and select *Properties*.
- Press `Alt+Enter`.
- In the *Object Browser*, right-click the component and select *Properties*.
- In the *Object Browser*, select the component and press `Alt+Enter`.

In general, the *Properties* panel has the same structure for all components, but the options available are specific to each individual component.

The *Properties* panel may include the following views:

View	Description
	Use the *General* view to set the basic configuration for components, such as titles, labels, and the location of the source and destination data.
	The *Insertion* view is available for some charts, and for the hierarchical table. Use this view to configure charts to act as selectors, so that clicking a chart element inserts more detailed information into the embedded spreadsheet, which can then be used by another component to create drilldown behavior. For information about configuring chart drilldown behavior, see *Configuring drilldown behavior*.
	Use the *Behavior* view to set up how the component works within the model. For example, you can set limits, interactivity, and visibility.
	Use the *Appearance* view to format the look of components, including font size, title locations, legends, colors, and so on.

View	Description
	Use the *Alerts* view to set up visual notifications regarding the data values. Alerts help identify values that are acceptable and ones that require attention, for instance. You can turn alerts on, set the target values, and configure colors for different alert levels. For information about using alerts, see *Understanding Alerts*.

4.1.1 Binding components to data sources

Binding components to data sources is an essential part of creating a dashboard model. Bindings are necessary for displaying numerical data in charts, maps, tables, and so on. In addition, you can create destination bindings that allow users to insert values to data sources at runtime.

You can also bind component parameters to data sources to set a component's titles, labels, chart scales, colors, visibility, and so on.

Two types of bindings are available. You can bind components to cells in the embedded spreadsheet, or to query data.

Data type	*Bind* button	Source bindings	Destination bindings
Embedded spreadsheet		Most component data sets and many component parameters can be linked to source data in the embedded spreadsheet. To provide source data for component parameters or data sets using the embedded spreadsheet, you specify the cells that contain data for each parameter or data set. You can add data to these cells manually, or connect the spreadsheet to an external data source.	Some components can also write data to the embedded spreadsheet. You use the same procedure to bind the component to destination cells in the embedded spreadsheet.
Query		Query data from BEx or universe queries can be used as source data for many components. You can choose the query and the result objects that you want to bind to the component. Hierarchical result objects can be linked to two types of components: *Query Prompt*	Some components can write data to query prompts. If you use a query prompt as a destination for component selections, the component allows users to restrict the results of the query at runtime.

Data type	Bind button	Source bindings	Destination bindings
		Selectors and Hierarchical Tables .	
		In most cases, you cannot bind query data directly to parameters for component appearance, behavior, or alerts.	

Related Links

Linking components and spreadsheet data [page 176]

4.1.1.1   To bind components to the embedded spreadsheet

When you link a component to cells in the embedded spreadsheet, you set a reference to those cells on the component's *Properties* panel. Each field that can be bound to the embedded spreadsheet has a *Cell Selector* () to its right.

> **i Note**
>
> By default, you can select a maximum of 512 rows in a range. To change this value, click ▶ *File* ❯ *Preferences* ❯, click *Excel Options* on the left, and enter your desired maximum number of rows. However, note that very large selections can negatively affect model performance.

1. On the canvas, double-click the component that you want to link to the embedded spreadsheet.
2. On the *Properties* panel, click the tab that contains the field that you want to bind.
3. Click the *Cell Selector* () to the right of the field to open the *Select a Range* dialog box.

   If the *Bind* button shows a query (), click the arrow next to the button and choose *Spreadsheet*.
4. On the embedded spreadsheet, click a single cell or click and drag to select a range of cells that you want to bind to the field.
5. In the *Select a Range* dialog box, click *OK*.

The field is linked to the range that you selected, and the range appears in the field.

> **i Note**
>
> To remove a binding, click the *Cell Selector* () and delete the range in the *Select a Range* dialog box.

Related Links

Understanding the embedded spreadsheet [page 173]
Linking components and spreadsheet data [page 176]

4.1.1.2   To bind components to queries

When you link a component to a query, you set a reference to query result objects or a query prompt on the component's *Properties* panel. A *Bind* button with a drop-down arrow (⬚ or ⬚) appears to the right of each field that can be bound to query data.

1. On the canvas, double-click the component that you want to link to the embedded spreadsheet.
   The *Properties* panel for the component opens.
2. On the *Properties* panel, click the tab that contains the field that you want to bind.

   Fields that can be linked to query data usually appear on the *General* tab and the *Insertion* tab.
3. Click the arrow next to the *Bind* button and click *Query Data* for source data or *Query Prompt* for destination data.
4. In the *Select from Query* dialog box, click the appropriate query in the left pane.
5. Select the binding in the right pane:

   ○ For source data bindings, select one or more objects in the *Result Objects* list. For some components, you can select only one object. If you are configuring a *Hierarchical Table* for a query with more than one hierarchical result object, you must choose one hierarchy to display in the table.

   > ℹ **Note**
   >
   > *Hierarchical Table* and *Query Prompt Selector* components are the only components that can display data in a hierarchy. If you bind hierarchical result objects to other types of components, they display all of the values on the same level, not in a multi-level hierarchy.

   ○ For destination bindings, select a prompt in the *Prompts* list. You cannot select prompts that have hierarchical lists of values.
6. Click *OK*.

The *Bind* button shows a query (⬚) and the field is linked to the query data that you selected.

> ℹ **Note**
>
> If you remove or edit a query, your changes are reflected in any components that are bound to the query.

> ℹ **Note**
>
> To remove a binding, click the arrow next to the *Bind* button and click *Clear Binding*.

Related Links

4.1.1.2.1   Unused result objects in queries

When you bind query data to a component, unused result objects are not removed from the query results. That is, the query data remains the same regardless of the result objects that you bind to the component.

Query measures are always aggregated by the most specific dimension in a query. As a result, a component may not function as intended if you do not include the most specific dimension.

To avoid this issue, when creating a query, include only the result objects that will be bound to a component. You may need to create several similar queries to show different levels of data aggregation in different components.

> ➡ Remember
>
> Each query uses a separate connection to the data source, and multiple queries can affect the performance of your model.

> ❖ Example
>
> If you create a query that includes Country and Region dimensions and a Sales Revenue measure, the query results will show sales revenue by region only. If you bind the query results to a chart with the Sales Revenue measure as the data values and the Country dimension as the labels, the chart does not display sales revenue by country. Instead, sales revenue values for each region are still displayed in the chart and labels for each country are repeated multiple times.
>
> To display sales revenue by country properly, you must create a query with the Country dimension and the Sales Revenue measure only. You can create this query in addition to the first query if you also want to display sales revenue by region in your model.

Related Links
Universes and BEx queries [page 215]
To add or edit queries [page 218]

4.1.2　Setting properties for multiple components

By selecting multiple components on the canvas, you can simultaneously edit properties for the selected components. To access properties for multiple components, select the components and double-click the selection. You can also access the properties by right-clicking the selection and selecting *Properties* from the context menu.

If the components are of the same type, you can edit properties specific to that component type. Any changes made to the properties apply to all components in the selection.

With different types of components selected, the *Properties* panel is limited to the dynamic visibility parameters on the *Behavior* view.

When a number of components are grouped, you can access properties for the individual components by right-clicking components in the *Object Browser*. For more information, see *Using the Object Browser*.

4.1.3　Setting colors for component elements

You can select specific colors for almost every element of every component in a dashboard. Colors are specified on the *Properties* panel for each component on the *Appearance* view.

> **i Note**
>
> The *Appearance* view is not available for some components, such as backgrounds. Color settings for these components can be found on the *General* view.

A number of common color definitions are available. You can also create your own color definitions.

To make your dashboard more visually engaging, you can bind the color of an element or alert range to a cell in the embedded spreadsheet. The color is then set dynamically during runtime.

Colors can be applied to some elements through alerts, too, and you can customize the color of each alert range.

Related Links

Understanding alerts [page 158]

4.1.3.1 To specify element colors

To set the color of a component element, you can select from a number of predefined colors.

1. On the *Properties* panel for the appropriate component, click *Appearance*.

2. On the *Color* tab, click the *Color Selector* ( ) for the appropriate element.

3. Choose a color for the element by performing one of the following tasks:

   ○ Click a color in one of the sections of the *Color* dialog box:

Section	Description
Theme Colors	These are the colors of the current theme.
Standard Colors	These are a group of basic colors.
Recent Colors	These are colors that you have recently used for another element.

   ○ Click *More Colors* to choose a color from a larger color wheel. When you have selected the color you want, click *OK*.

4.1.3.2 To create a custom color for elements

If you want to assign a specific color to component elements, you can specify the RGB or HSL values to define the color precisely. Custom colors can then be assigned to component elements in the same way predefined colors are assigned.

1. Select the component and open the *Properties* panel.

2. On the *Color* tab, click the *Color Selector* ( ) for the appropriate element.
   The *Color* dialog box opens.

3. Click *More Colors*.

4. On the *Custom* tab, in the *Color Model* list, select *RGB* or *HSL*.

5. Set the color setting levels, that is, *Red*, *Green*, and *Blue* levels for RGB or *Hue*, *Sat* (Saturation), and *Lum* (Luminosity) levels for HSL.

6. To apply the custom color to the selected element, click *OK*.

> ➡ Tip
>
> To apply the custom color to other elements, you can access it in the *Recent Colors* section of the *Color* dialog box.

4.1.3.3 To dynamically set colors at runtime

If you want to configure an element to change colors at runtime based on user input or a setting for another component, you can bind the color setting to a cell in the embedded spreadsheet.

When the color is bound to a spreadsheet cell, the cell's value sets the color. You can then configure the value in the cell to be set by user input, another component, an external data source, and so on.

The following color formats are supported:

* HEX (For example, FFFFFF, #FFFFFF, 0xFFFFFF)
* RGB (For example, (255,255,255))
* VGA Color names (For example, Aqua, Black, Blue, Fuchsia, Gray, Green, Lime, Maroon, Navy, Olive, Purple, Red, Silver, Teal, White, Yellow)
* Decimal Color Equivalent (For example, F00FF becomes 16711935)

> ℹ Note
>
> If the value in the specified cell is not a valid color setting or the cell is empty, the component element color is set to black.

> ℹ Note
>
> Colors that are bound to a cell are not affected by changes to the color theme applied to the model.

1. On the *Properties* panel for the appropriate component, click *Appearance*.

2. On the *Color* tab, click the *Color Selector* () for the appropriate element.

3. On the *Color* dialog box, click *Bind to a Color*.
   The *Select a Range* dialog box appears.

4. Select the cell you want to bind the color to and click *OK*.

The element color is updated based on the value in the selected cell and the *Color Selector* () changes to a *Bound Color* icon ().

4.1.4   Formatting text for component elements

For each component element (titles, labels, button text, mouse-over text, legends, and so on), you can choose to display or hide the text. If the text is displayed, you can specify how the text will look and, for numbers, what format they will use.

If you save the model to the platform, you can also select default options that are updated at runtime to use the regional format associated with the user's viewing locale.

> ℹ **Note**
>
> Models saved locally use the text format defined when the model is created and are not changed at runtime.

Related Links
Translating and globalizing models on the BI platform [page 248]

4.1.4.1   To display text elements in a model

1. On the *Properties* panel for the component, select the *Appearance* view.
2. Click the *Text* tab.
   The *Text* tab shows a list of text elements available for the selected component. Text elements that are currently set to display in the model have a checkmark in the left column. If a text element is listed but cannot be selected, that element may not be enabled for the component.
3. Do either of the following:

   o To display a text element in the model, select the element.
   o To hide a text element in the model, deselect the element.

4.1.4.2   To format text elements

1. On the *Properties* panel for the component, select the *Appearance* view.
2. Click the *Text* tab.
   The *Text* tab shows a list of text elements available for the selected component. Text elements that are currently set to display in the model have a checkmark in the left column.
3. Click a text element that is currently displayed in the model.
4. In the *Format Selected Text* section, select the font, font size, format, justification, and text color settings.

   > ℹ **Note**
   >
   > If a global font is set, the font type cannot be set in the component properties. For more information, see *To set global font options*.

5. If available, in the *Position* section, select where the label should be in relation to the component and, in *Offset*, adjust the label position by setting a distance to move it relative to its position.

6. In the *Number Format* section, select one of the following number formats:

Option	Description
General	The data has no specific number format.
Numeric	The data is treated as numbers only. Additional settings are available for setting the number format.
Currency	The data is displayed as monetary values. Additional settings are available for setting the currency format.
Percent	The data is displayed as a percentage.
Date	The data is displayed as a date value. Additional settings are available for setting the date format.
Time	The data is displayed as a time value. Additional settings are available for setting the time format.
Boolean	The data is displayed as a two value option (on/off, true/false, yes/no).
Text	The data is displayed as text only.

Depending on the selection, additional formatting options may be available.

7. If additional formatting options are available, set them. The following options might be available:

Option	Description
Default Number Format	Select this option to use the regional number format associated with the user's locale. If this option is selected, other number formatting options are not available. This option applies to *Numeric* elements when the model is saved to the platform.
Default Currency Format	Select this option to use the regional currency format associated with the user's locale. If this option is selected, other currency formatting options are not available. This option applies to *Currency* elements when the model is saved to the platform.
Negative Values	Specify the format to use for negative values. This option applies to *Numeric* and *Currency* elements.
Decimal Places	Specify the number of decimal places to display. This option applies to *Numeric*, *Currency*, and *Percent* elements.
Prefix	Select this option to specify the currency symbol to show before the number. This option applies to *Currency* elements.
Suffix	Select this option to specify the currency symbol to show after the number. This option applies to *Currency* elements.
Use 1000 Separator	Select this option to insert a comma between the hundreds and thousands numerals. This option applies to *Numeric* and *Currency* elements.
Type	Specify the type of date or time format to use from a list of common formats. This option applies to *Date* and *Time* elements. ℹ **Note** For models saved to the platform, you can also select default options to use a regional format associated with the user's locale.

Related Links

4.2   Using components

4.2.1   To configure component properties

Choose a component and add it to the canvas before configuring the settings in the *Properties* panel.

Not all tabs are available for each component, and the options in each tab vary depending on the component.

1. Select the component and open the *Properties* panel.
2. If the *General* tab is available on the *Properties* panel, click it and configure the appropriate options. This tab contains options for titles, labels, display, destination, and source data, and so on.
3. If the *Insertion* tab is available on the *Properties* panel, click it to configure data insertion options for the component.

    Components with data insertion can insert information into cells in the embedded spreadsheet or into a query prompt when a user interacts with the component during runtime.

    > **i Note**
    >
    > Map components and selector components, except the *Hierarchical Table*, have data insertion options on the *General* tab.

4. If the *Behavior* tab is available on the *Properties* panel, click it to set the options for how the component functions in the model including handling variable amounts of data, scaling, animations, and dynamic visibility.
5. If the *Appearance* tab is available on the *Properties* panel, click it to set the visual characteristics of the component, including chart legend options, layout options, text formatting, and colors.
6. If applicable, click the *Alerts* tab to set up conditional formatting based on data values. Alerts help users identify which values are within acceptable or unacceptable ranges.

The component is configured. Any changes that you make in the appearance of the chart and the data that it displays is updated in the canvas. You can see how the component functions during runtime by clicking *Preview*.

Related Links

4.2.2   Using chart components

Charts allow you to represent data visually to make it easy for users to see comparisons, patterns, and trends. The software includes the following types of charts.

Area Chart
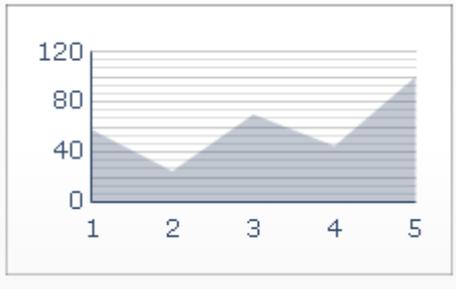
The *Area Chart* is a standard chart with vertical and horizontal axes. Each point along the horizontal axis represents a data point. The values for each data point are plotted along the vertical axis. For each series, colored areas are created by connecting the plotted points against the horizontal axis. This chart is typically used in models that emphasize a trend line, such as stock prices or revenue history.

Bar Chart and Column Chart
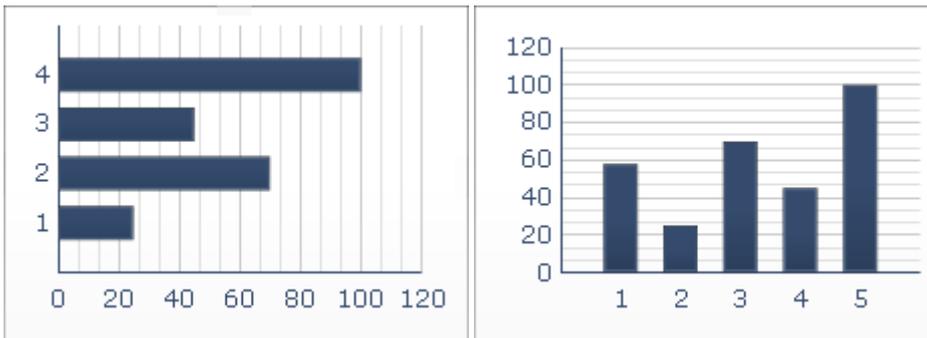
These are multi-bar charts that show and compare one or more items over a period of time or in a specific range of values.

You could, for example, use this chart in models that contain the quarterly headcount by region.

> ℹ **Note**
>
> *Bar Charts* and *Column Charts* are functionally identical; they display the same information. The only difference is the orientation of the bars.

Bubble Chart
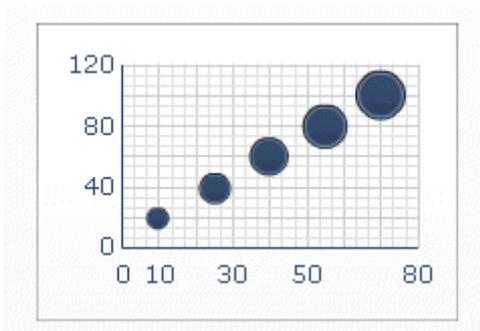
The *Bubble Chart* is one of the most powerful analytic tools available. It lets you compare a group or series of items based on three different parameters. It has an X-axis and Y-axis to represent the item location over the chart area, and a Z-value to represent the item size.

You could, for example, use this chart to represent the market composition with the X-axis representing the return on investment (ROI) by industry type, the Y-axis representing the cash flow, and the Z-value representing the market value.

Bullet Chart
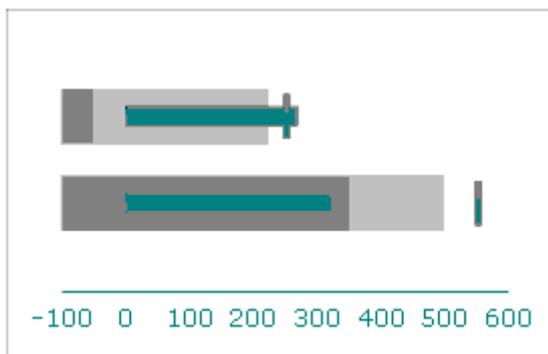
The *Bullet Chart* provides a method to display information in a condensed, less-distracting format and is available with either a vertical or horizontal orientation.

These charts have a single primary measure (for example, current year-to-date revenue) that is displayed as a vertical or horizontal bar. A marker indicates a target or performance goal and changing color hues behind the bar indicate qualitative ranges of performance (for example, poor, satisfactory, and good).

You can configure the chart to display multiple series and can configure the scale, axis, and ticks separately for each series. For example, the chart can be configured to show revenue in thousands of dollars, profit in dollars, and order size as a count.

Combination Chart
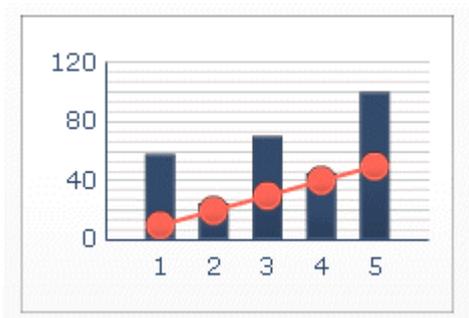
The *Combination Chart* is ideal for displaying a range of values and a trend line for those values. This chart is typically used in models examining stocks. A line series might show the historical stock price over the year, and a column chart might show the volume of trading for that stock.

The *Combination Chart* can also have alert notifications when it has multiple data series. For more information on alerts, see *Understanding alerts*.

Line Chart
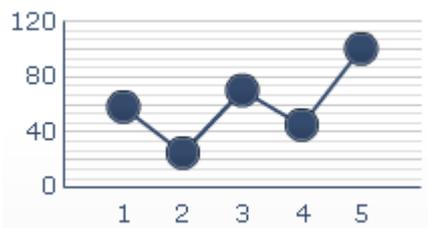
The *Line Chart* is ideal for showing tendency over a period of time. This chart is typically used in models that emphasize a trend or a continuing sequence of data, such as stock prices or revenue history.

OHLC Chart and Candlestick Chart
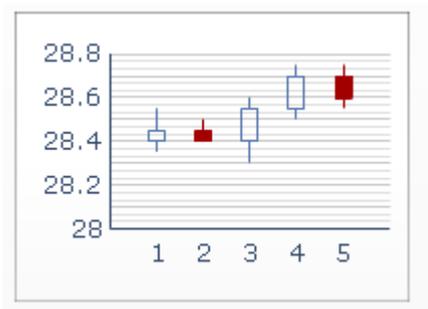
These charts are typically used to display stock data. Each marker corresponds to the values, which are represented as lines attached to the marker on the *OHLC Chart* and as colors on the *Candlestick Chart*. The open

value displays the opening price of the stock. The high value displays the highest price the stock achieved on that day. The low value displays the lowest price of the stock on that day. The close value displays the closing price of the stock.

Pie Chart
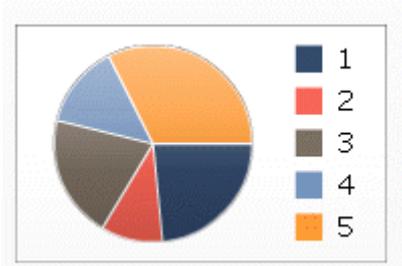
The *Pie Chart* is a circular chart that represents the distribution or participation of each item (represented by a slice) of a certain total that is represented as the overall pie value. You could, for example, use this chart to display revenue contribution by product. The overall pie size would represent the total revenue, and each slice would represent a different product.

Radar Chart and Filled Radar Chart
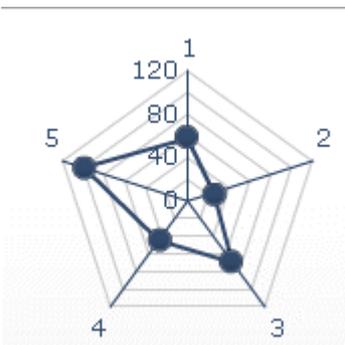
The *Radar Chart* and *Filled Radar Chart* have axes that radiate outwards from the center of the chart. These charts can have several axes and are useful for plotting multi-dimensional sets of data. In the *Filled Radar Chart*, the shape created by connecting the points along each axis is filled in with color. These charts are typically used to compare aspects of stocks. One axis could display the price, another the volume, another the price to earnings ratio, and any other relevant data.

Scatter Plot
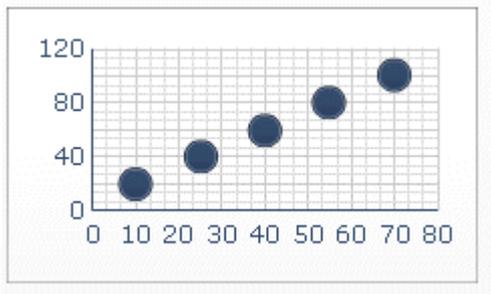
This chart displays data that encompasses two dimensions. The *Scatter Plot* shows each data point as a result of the intersection of X values and Y values. You can, for example, use this chart in models that compare ROI (on the X axis) against market value (on the Y axis) for a group of companies.

> **i** Note
>
> This component is named *XY Chart* in previous versions of Dashboards.

Sparkline Chart
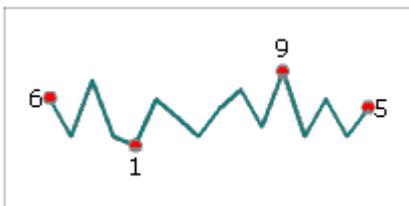
The *Sparkline Chart* presents trends and variations associated with some measurement, such as stock market activity, in a simple and condensed way.

You can set key data points (start value, end value, low value, and high value) that are displayed as markers on the chart, and also define a normal range that is displayed as a colored background behind the chart.

Stacked Area Chart
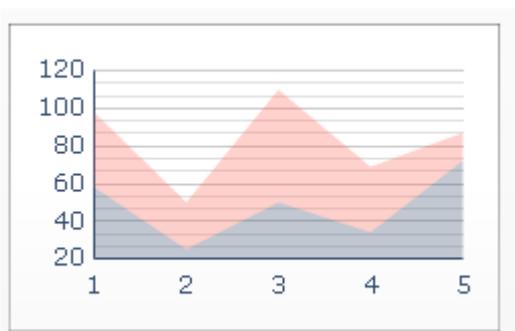
The *Stacked Area Chart* has vertical and horizontal axes. Each point along the horizontal axis represents a data point. The actual values for the data points are plotted against the vertical axis, with each series adding to the total value. You could, for example, use this chart to compare the revenue for multiple products as well as the combined revenue of all the products and the contribution of each product to the combined revenue.

Stacked Column Chart and Stacked Bar Chart
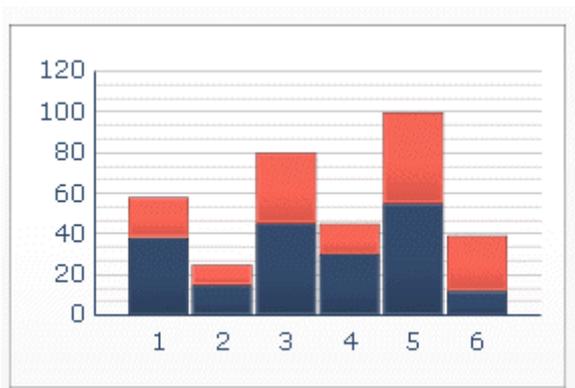
These charts compare one or more variables, with each series adding to the total value. These charts are typically used to compare several variables over a period of time, such as marketing cost and administrative cost. Each one of the cost components is presented in a different color and each bar represents a different time period. The total bar size represents the total cost.

Tree Map

The *Tree Map* displays data in an area. Each data point is represented by a rectangle. The *Tree Map* displays two parameters, represented by size and color intensity, and can be used to compare two sets of data.

You could, for example, use a *Tree Map* to represent the size and interest rate of loans. You can set the size of the rectangle as the loan amount; a large loan would be represented by a larger rectangle. The color of the rectangle would represent the interest rate; a higher interest rate would be represented by a brighter color value.

> ➡ Tip
>
> To see an example of a model with a *Tree Map*, click ▶ *File* ❯ *Samples* ◤ and open the sample file *Tree Map*.

Waterfall Chart
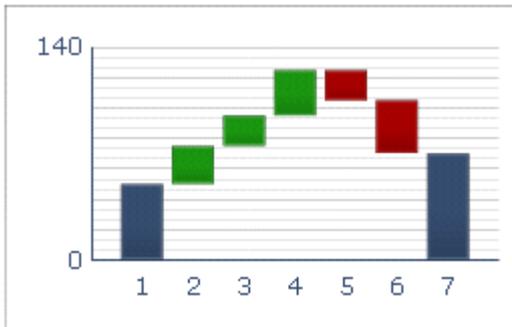
The *Waterfall Chart* is typically used to display how an initial value is affected by a series of sequential positive or negative values. The initial and final values are represented by full columns, while the intermediate values are represented by partial columns. The columns are color-coded to distinguish between positive and negative values.

You could, for example, use this chart to represent the inventory level of goods that a retail business sells. Purchases of additional inventory are represented by positive values, and sales of goods are represented by negative values.

4.2.2.1 Chart general properties

Use the following table as a guide for setting properties on the *General* view for chart components. Some options are available only for specific chart types.

Option	Description
Titles area	
Chart	Specify the main title for the chart.
Subtitle	Specify a secondary title for the chart.
Category (X) Axis	Specify the label for the data categories displayed along the X axis.
Category (Y) Axis	Specify the label for the data categories displayed along the Y axis.
Horizontal (X) Axis	Specify the label for the data along the X axis.
Vertical (Y) Axis	Specify the label for the data along the Y axis.
Value (Y) Axis	Specify the label for the values displayed along the Y axis.
Value (X) Axis	Specify the label for the values displayed along the X axis.
Secondary Value (Y) Axis	Specify the label for the secondary Y axis displayed on the right of the chart. i Note

52

Dashboards and Presentation Design User Guide
Working with Components

Option	Description
	This option is available only if *By Series* is selected and *Plot Series On* is set to *Secondary Axis*.
Secondary Value (X) Axis	Specify the label for the secondary X axis displayed at the top of the chart. ⓘ **Note** This option is available only if *By Series* is selected and *Plot Series On* is set to *Secondary Axis*.
Data area	
By Range	Select this option to display a range of values for a single data series. Select one of the following options: • If the data series is from the columns of the worksheet, select *Data in Columns*. • If the data series is from the rows of the worksheet, select *Data in Rows*.
By Series	Select this option to display multiple series of values. To add a series to the chart, click *Add* (⊞). Then, for standard charts, select each series and set the following options: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*Name*</td><td>Specify the name for the series.</td></tr><tr><td>*Values*</td><td>Specify the X or Y data points from the spreadsheet or query. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (the *Bind* button shows a spreadsheet ▦▾) or *Query Data* to bind the values to query results (the *Bind* button shows a query ▤▾).</td></tr><tr><td>*Plot Series On*</td><td>Specify the axis you want to plot the series on.</td></tr><tr><td>*Category Labels*</td><td>Specify the label displayed on the X axis to identify each category in the series.</td></tr></table>
By Series	For *OHLC Charts* and *Candlestick Charts*, select this option to display multiple series of values. For each of the value options, you can link the value to a spreadsheet cell or to query data objects. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (▦▾) or *Query Data* to bind the values to query results (▤▾). Set the following options:

Option	Description	

Option	Description
Series Name	The name of the series. For reference only.
Open	The source data for the opening value.
High	The source data for the high value.
Low	The source data for the low value.
Close	The source data for the closing value.
Category Label	The label displayed on the X axis to identify each category in the series.

Option	Description
By Series	For *Tree Maps*, select this option to display multiple series of values. For each of the value options, you can link the value to a spreadsheet cell or to query data objects. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (the *Bind* button shows a spreadsheet) or *Query Data* to bind the values to query results (the *Bind* button shows a query). To add a series to the chart, click *Add* ( ). Then, select each series and set the following options:

Option	Description
Display Labels	The name of the series. For reference only.
Values (size)	The range of values represented by the size of squares on the chart.
Values (color intensity)	The range of values represented by the color intensity on the chart.

Option	Description
By Series	For *Bullet Charts*, select this option to display multiple series of values. For each of the value options, you can link the value to a spreadsheet cell or to query data objects. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (the *Bind* button shows a spreadsheet) or *Query Data* to bind the values to query results (the *Bind* button shows a query). To add a series to the chart, click *Add* ( ). Then, select each series and set the following options:

Option	Description
Label	The name of the series. For reference only.
Sub-Label	A secondary label for the series that appears below the series label.
Performance Value	The range of values that represent actual performance.

Option	Description		
	Option		**Description**
	Comparative Value		The range of values for the target performance.
	Scale Value		The range of values that define the scale for performance, for example, poor, fair, good.
Values	Specify the range of values represented by the colors on a *Pie Chart*. You can link the values to a spreadsheet cell or to query data objects. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (the *Bind* button shows a spreadsheet) or *Query Data* to bind the values to query results (the *Bind* button shows a query). Select one of the following options: • If the data series is from a column of the worksheet, select *Data in Columns*. • If the data series is from a row of the worksheet, select *Data in Rows*.		
Labels	Specify the labels displayed beside each color in the key of a *Pie Chart*. You can link the value to a spreadsheet cell or to query data objects. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (the *Bind* button shows a spreadsheet) or *Query Data* to bind the values to query results (the *Bind* button shows a query).		
Header Labels	Specify the labels displayed beside each series in the chart. You can link the value to a spreadsheet cell or to query data objects. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (the *Bind* button shows a spreadsheet) or *Query Data* to bind the values to query results (the *Bind* button shows a query).		

Related Links

To configure component properties [page 45]

Binding components to data sources [page 37]

Linking components and spreadsheet data [page 176]

Chart behavior properties [page 59]

Chart insertion properties [page 55]

Chart appearance properties [page 65]

4.2.2.2   Chart insertion properties

Use the following table as a guide for setting properties on the *Insertion* view for charts that contain a series. Some options are available only for specific chart types.

Option	Description
Enable Data Insertion	Allows the chart to behave as a selector and insert data into a cell. This option is available only to charts that contain a series. For more information about setting up a series, see *Chart general properties*.
Series Name Destination	The cell in the embedded spreadsheet where the currently selected series name will be inserted. This option is used to identify the series of data that is inserted.
Insertion Type	This list determines the structure of the data that is inserted into the embedded spreadsheet when a selection is made. For more information, see *To configure charts to drilldown*.
Series	For each series the chart contains, select the series and then set the applicable bindings, including the *Source Data* to be inserted and the *Destination*. For pie charts, the *Series* list is not available. You must define only one set of *Source Data* and *Destination* values. With *Value* insertion using *Bubble Charts*, *XY Charts*, and *Tree Maps*, the *Value Set* option allows you to choose which set of values to use for insertion. For each destination value, you can link the value to a spreadsheet cell or to a query prompt. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet () or *Query Data* to bind the values to query results ().
Interaction Options	Specifies how selections are made on the component. Select either *Mouse Click* or *Mouse Over*.
Default Selection	Specifies the item or series that is selected by default when the model is loaded.

Related Links

To configure component properties [page 45]
Understanding insertion types [page 97]
Chart general properties [page 52]
Chart behavior properties [page 59]
Chart appearance properties [page 65]

4.2.2.2.1 Configuring drilldown behavior

Chart insertion capability allows charts to behave as selectors; when users click a part of the chart, the chart writes data to cells in the embedded spreadsheet or to a query prompt. Other components can use that data to display more detailed information so that users can drill down into the data.

For example, a *Pie Chart* that shows annual sales could be configured to insert the monthly sales totals into specified destination cells. Then, a *Bar Chart* could use that data to display the monthly totals each time a piece in the *Pie Chart* is clicked.

> ➡ **Tip**
>
> To see an example of a model with drilldown behavior, click ▶ *File* ❯ *Samples* ❯ and open the sample file *Chart - Drilldown*.

Related Links

Chart insertion properties [page 55]

To configure charts to drill down

Before performing this task, you must configure a chart and bind it to a data source. More detailed drilldown information must also be available in a spreadsheet or query.

> ℹ **Note**
>
> The following charts do not have insertion options and cannot be configured to drill down in this way:
>
> - *Area Chart*
> - *Stacked Area Chart*
> - *Horizontal Bullet Chart* and *Vertical Bullet Chart*
> - *Sparkline Chart*
> - *Radar Chart*
> - *Filled Radar Chart*

1. Select the chart you want to use to drill down and open the *Properties* panel.
2. On the *Insertion* tab, select *Enable Data Insertion*.

   > ℹ **Note**
   >
   > If the *Enable Data Insertion* checkbox is not available, verify that the chart is bound to a data source.

3. If the chart contains a series, in the *Series Name Destination* box, you can click the *Cell Selector* ( ) and select a cell in the spreadsheet where you want the name of the series to be inserted.
4. In the *Insertion Type* list, select one of the following options to specify the type of data that is inserted when a selection is made:

Option	Description
Position	When the user selects a value, the position of that value within the series is entered in the target cell.
	For example, for a *Bar Chart* with three values, when a user selects the first bar, a 1 is entered in the destination cell or query prompt. When a user selects the middle bar, a 2 is entered and for the final one, a 3 is entered.
	For spreadsheet insertion, you must select a single *Destination* cell for this insertion type.

Option	Description
Value	When the user selects a value, that value is entered in the destination cell or query prompt.
	For spreadsheet insertion, you must select a single *Destination* cell for this insertion type.
	Bubble Charts, *OHLC Charts*, *XY Charts*, *Candlestick Charts*, and *Tree Maps* have multiple values for each data point, so you must specify the value set that you want to be inserted. For example, for a *Tree Map*, you can use the values for size or for color intensity.
Row	When a selection is made, a row of source data values corresponding to the selection is entered in the destination row or query prompt.
	For example, for a *Pie Chart* with four values, you designate four rows of source data. When a user selects the first slice of the chart, the first row of source data is inserted into the destination. If the user then selects the second slice, the second row of source data is inserted, overwriting the previous data.
	For spreadsheet insertion, you must select several rows of *Source Data* for this insertion type, and one *Destination* row with the same number of columns.
	For query prompt insertion, select a single column of source data to insert one parameter to the query prompt each time the user makes a selection. If you are inserting data to a query prompt with an *In List* or *Not in List* operator, you can choose multiple columns of source data to insert several parameters at a time.
Column	When a selection is made, a column of source data values corresponding to the selection is entered in the destination column or query prompt.
	This insertion type is similar to *Row* insertion. You must select several columns of *Source Data* and one *Destination* column with the same number of rows.
Status List	When a selection is made, a value of 1 is entered in the cell in the range representing that selection and a value of 0 is entered in the other cells of the target range.
	For example, for an *OHLC Chart* with six values, you designate six cells in a row or column as the *Destination*. If the user selects the second data point in the chart, a 1 is inserted into the second cell in the destination row or column, and a 0 is inserted into the remaining five cells.

Option	Description
	For spreadsheet insertion, you must select a *Destination* row or column with one cell for every data point in the chart.

> ➜ Tip
>
> *Column* and *Row* insertion are often used to create drilldown effects.

5. Unless you are configuring a *Pie Chart*, in the *Series* list, select a series to configure.

6. If the *Insertion Type* is set to *Row* or *Column*, beside *Source Data*, click the *Cell Selector* (　) and select the cells in the spreadsheet that contain the source data.

> ⓘ Note
>
> If the *Insertion Type* is set to *Value*, *Position* or *Status List*, the source data is determined by the selection in the chart.

7. For *Bubble Charts*, *XY Charts*, *Candlestick Charts*, *OHLC Charts*, and *Tree Maps*, if *Insertion Type* is set to *Value*, in the *Value Set* list, select the value to be inserted.

8. Beside the *Destination* box, click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet or *Query Data* to bind the values to a query prompt.

9. Select the appropriate spreadsheet range or query prompt to complete the binding.

10. Under *Interaction Options*, in the *Insert On* list, select *Mouse Click* to insert the data when the user clicks the chart or *Mouse Over* to insert the data when the user moves the mouse pointer over the chart selection.

11. Under *Default Selection*, in the *Series* list, select the series and, in the *Item* list, choose the item that is selected by default when the model loads.

The chart is configured to insert data into a destination range when a selection is made. The data in the destination range can be used by another component in the model.

Related Links

Binding components to data sources [page 37]
Using chart components [page 46]
Chart insertion properties [page 55]

4.2.2.3　Chart behavior properties

Use the following table as a guide for setting properties on the *Behavior* view for chart components. Some options are available only for specific chart types.

Option	Description
Common tab	
Ignore Blank Cells	Specifies whether blank cells at the end of the data are displayed on the chart or ignored.

Option	Description
	Set any of the following options: • *In Series*: The chart does not display empty series past the last non-empty series in a range. • *In Values*: The chart does not display empty values past the last non-empty value in a series. For more information about using this option, see *Working with variable amounts of data*.
Normal Range Area	For a *Sparkline Chart*, select this option to enable an area of color behind the sparkline to indicate the range of normal values for the given data.
Auto	Select this option to set the normal range area based on the variance of the data.
Manual	Select this option to set the normal range area based on specific low and high values. Enter the low and high values or click the *Cell Selector* ( ) to select a source data location for these values.
Enable Run-Time Tools	Specifies whether users are able to adjust the chart scale during runtime. If this option is selected, when the model runs, a set of runtime tools is available and allows users to adjust the chart scale. You can also select which tools are available in the runtime tool set. For more information about using this option, see *Allowing users to adjust chart scales*.
Enable Range Slider	Adds a range slider to allow users to focus on a particular data set rather than viewing the entire chart at once. Users can change the size of the range by adjusting control handles on either side of the range slider. For more information about using this option, see *Adding Range Sliders to charts*.
Enable Sorting	Sorts the chart values according to specified data series or chart labels. Select one of the following options: { table below }
Dynamic Visibility	This option allows you to specify conditions that control when the chart is visible during runtime. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].

Inner table under *Enable Sorting*:

Option	Description
By Data	Sort according to the data values. If the chart has multiple series, in the *Series* list, select the series to sort on. In the *Order* list, select *Ascending* or *Descending*.
By Category Labels	Sort by category axis labels. Labels are sorted alphanumerically in ascending order. To sort in descending order, select the *Reverse Order* check box.

Option	Description
Scale tab	
Scale	Use these options to specify how the scale looks on the chart. For more information, see *Configuring Scales*.
Divisions	Configure how many dividing lines appear on the chart. For more information, see *Configuring Scales*.
Animations and Effects tab	
Enable Data Animation	When enabled, data animations add movement to chart components. For example, in *Bar Charts*, the bars gradually increase or decrease in size, and in point-based charts, the points slide to their new position as the data changes. When not enabled, the value markers jump to their new position when the data changes.
Entry Effect	Select how you want the container to appear on the canvas when the model opens or when the dynamic visibility of the chart is triggered. You can select a *Type* of entry effect and specify its *Duration*.

Related Links

To configure component properties [page 45]

Chart general properties [page 52]

Chart insertion properties [page 55]

Chart appearance properties [page 65]

4.2.2.3.1 Configuring scales

The scale controls how data is plotted on the chart and how the chart axes adjust as the data values change.

> **i Note**
>
> Scales do not apply to *Pie Charts*, *Sparkline Charts*, or *Tree Maps*.

You can allow chart scales to adjust automatically based on the data. However, depending on what the data is showing, it may be more appropriate to set the scale manually. If you set a manual scale, the scale does not change to accommodate the data and if data values exceed the chart range, they do not appear on the chart. For example, if your data is represented as a percentage of a specific quantity, it may not be appropriate to include 120%.

You can also specify the scale format as either linear (based on addition) or logarithmic (based on multiplication). In general, logarithmic scales are well-suited to larger number ranges and data that contains a wide variation of values.

> **i Note**
>
> If you want to allow users to adjust the scale at runtime, enable runtime tools on the ▶ *Behavior* ❯ *Common* ❯ tab of the *Properties* panel.

Related Links

Allowing users to adjust chart scales [page 73]
Chart behavior properties [page 59]

To configure chart scales

Follow these steps after placing a chart on the canvas and linking it to cells in the embedded spreadsheet.

1. Select the chart and open the *Properties* panel.

2. In the *Properties* panel, click ▶ *Behavior* ▷ *Scale* ◣.

3. If you are configuring a *Bullet Chart* and you want to configure a different scale for each series, select the *Configure Scale by Series* option. Then select the series that you want to configure from the list.

   Use this option when you need to display metrics that are not closely comparable within a single scale or when the series use different scale measures such as percent, count, and dollars.

4. Under *Scale*, select one of the following options:

   ○ *Manual Axis*
   Select this option to set fixed scale limits that will not change to accommodate the data values. If the data values exceed the specified scale range, the markers do not show the outlying values. Users can see the values when in a tooltip when they move the mouse over the marker. If you select this option, set the *Maximum Limit* and *Minimum Limit* for each scale axis.

   ○ *Auto Axis*
   Select this option to allow the scale limits to change to accommodate the data values, then set the following options:

Option	Description
Allow Zoom Out Only	Select this option if you want the scale to grow as data values increase, but not shrink when the values decrease. This option minimizes the number of times the chart must rescale and is useful to reduce the impact on performance when the chart is included in an animation.
Zoom Sensitivity	If *Allow Zoom Out Only* is selected, adjust the zoom sensitivity. This slider determines how much the axis scale increases as the chart values increase.

5. In the *Axis Scale* list for each chart axis, select one of the following options:

Option	Description
Linear	A linear scale increases by adding a set amount to each scale marker (for example, 1, 2, 3, 4). This is the most commonly used scale and works well when the data value range is fairly small.
Logarithmic	A logarithmic scale increases in multiples. This scale is helpful when the data value range is broad or includes some values that are much larger than others. For example, a chart with values like 10, 100, and 60,000 does not fit well on a linear scale and the smaller values would be lost. A logarithmic scale can reveal structured relationships over many orders of magnitude.

6. To lock the width of the axis labels and prevent them from being resized when the scale changes, select *Fixed Label Size*.

   With *Fixed Label Size* selected, axis labels can be abbreviated. For example, 12K would be displayed instead of 12,000.

7. If *Fixed Label Size* is selected and you want to define the characters displayed for label abbreviations (K for thousands, M for millions, and so on), click the *Manual Edit* icon () beside *Label Abbreviations*.

8. If the *Manual Axis* option is selected, under *Divisions*, select one of the following options:

   ○ *Number of Divisions*
   To set the number of scale markers to show on the axis. The value of the markers is determined by the range of the scale. If the scale changes, the values of the markers are adjusted to maintain the same number of markers.

   ○ *Size of Divisions*
   To set the values of the markers based on a specific value that separates each marker. The value of the markers is determined by adding this amount to the previous marker. If the scale changes, the number of markers may change, but the values of the markers remain the same.

   > **i Note**
   >
   > If a scale is defined for more than one chart axis, the division settings apply to all of them. You cannot set different scales for each axis.

9. In the *Minor Divisions* list, select the number of grid lines to show between chart markers.

   > **i Note**
   >
   > You can hide either set of gridlines on the ▶ *Appearance* ▶ *Axes* ◀ tab by clearing *Show Major Gridlines* or *Show Minor Gridlines*.

Related Links

Allowing users to adjust chart scales [page 73]
Chart behavior properties [page 59]
Chart appearance properties [page 65]

4.2.2.3.2 Adding range sliders to charts

When you add a range slider to a chart, a miniature version of the chart displays below or beside the chart. A slider on the miniature chart allows users to select a portion of the data to show in the main chart.
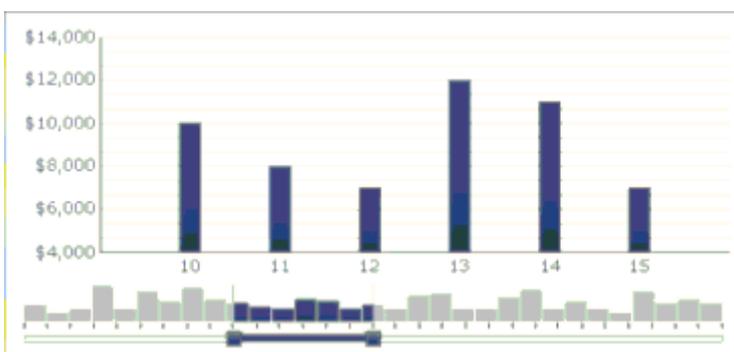
> **i Note**
>
> For *Bar Charts* and *Stacked Bar Charts*, the range slider appears to the left of the chart.

When you configure the range slider, you can specify the initial size and location of the range slider by setting the beginning and end range values. By default, these values are set to the minimum and maximum values, so the

entire chart is visible. You can set the values to a specific data position or, if category axis labels are defined for the chart, to specific category labels. You can also bind the range values to cells in the embedded spreadsheet that are set by another source, such as a selector component in the model.

> **i Note**
>
> The beginning range value must precede the end range value. Otherwise, the range slider displays the entire chart.

To change the data displayed in the main chart, users drag the range slider across the miniature chart. Users can also drag the control handles on either side of the range slider to view a greater range of data in the main chart.

Related Links

Using chart components [page 46]

To configure range sliders

1. Select the chart and open the *Properties* panel.
2. In the *Properties* panel, click ▌▶ *Behavior* ▶ *Common* ◤.
3. Select the *Enable Range Slider* checkbox.
   The range slider is displayed below or beside the chart. The color of the slider is determined by the color theme of the model.
4. In the *Beginning Range Value* list, select one of the following options:

   ○ *Position*: to specify a data position in the chart. For example, if 30 points are being plotted in the chart, setting the *Position* value to 2 would set the beginning of the range at the second data point.

   ○ *Category Label*: to specify a category axis label. For example, if category axis labels are set to the days of the month, you could specify March as the beginning range value.

   > **i Note**
   >
   > This option is available only if category axis labels are defined.

5. In the *Beginning Range Value* box, type a value or click the *Cell Selector* (🔲) to select a cell in the embedded spreadsheet.

   > **i Note**
   >
   > To have the value set by another component, select an empty cell and then configure another component to enter a value in that cell.

6. Repeat steps 4 and 5 to set the *End Range Value*.
7. Beside the *Range Labels* box, do one of the following:

   ○ Click the *Cell Selector* (🔲) to select cells in the spreadsheet.

   ○ Click *Manual Entry* (📝) to type in the names of the labels.

The range labels are distributed across the miniature chart. The placement of the labels depends on the number of data points and the total number of labels. For example, if there are 36 data points plotted on the chart and 4 labels, a range label is displayed every 8 data points.

4.2.2.3.3 Working with variable amounts of data

If the amount of data available for a chart can vary, you can configure the chart to display only the values that are available and to ignore blank cells at the end of a range.

For example, if data is entered for each day in a month, when the month has 31 days, you want the chart to include the data for each of those days. When a month has 30 days, the chart should not display an empty value for day 31. You must set the range to include 31 days, but if the last cells are empty, you do not want them to appear on the chart.

> **i Note**
>
> Although the chart does not display all the data within the data range, the size of the range still affects the model's performance.

To configure a chart for variable amounts of data

1. Select the chart and open the *Properties* panel.
2. Click the *Behavior* view.
3. On the *Common* tab, select *Ignore Blank Cells*.
4. To prevent the chart from displaying empty series past the last non-empty series in a range, select *In Series*.
5. To prevent the chart from displaying empty values past the last non-empty value in a series, select *In Values*.

4.2.2.4 Chart appearance properties

Layout tab

Use the following table as a guide for setting the *Appearance* properties for chart layout. Some options are available only for specific chart types.

Option	Description
Show Chart Background	Select this option to include a solid background behind the chart. Otherwise, the chart is transparent. If you include a background, set the following options:

Option	Description

Option	Description
Margin	Select the amount of space to allow between the edge of the background and the data.
	If you want to adjust the four margin settings individually, clear the *Same Margins on All Sides* checkbox.
Horizontal Gap	For a *Sparkline*, set the amount of space between the series labels and the plot area.
Vertical Gap	For a *Sparkline*, set the amount of space between each series of data.

> ➡ **Tip**
>
> You can set a color for the chart background on the color tab.

Option	Description
Plot Area	For the plot area (the area bounded by the chart axes), set the following options:

Option	Description
Show Fill	Applies a background to the plot area.
Show Border	Applies a border around the plot area.
Border Thickness	If *Show Border* is selected, select a thickness for the border.

Option	Description
Title Area	For the area behind the title, set the following options:

Option	Description
Show Fill	Applies a background to the title area.
Show Border	Applies a border around the title area.
Border Thickness	If *Show Border* is selected, select a thickness for the border.

Option	Description
Enable Legend	Select this option to display a legend for the chart. For more information about configuring legends, see *Adding legends to charts*.

Series tab

Use the following table as a guide for setting the *Appearance* properties for chart series. Some options are available only for specific chart types.

Option	Description
Series	Lists the series that have been defined for the chart.
	> ⓘ Note

Option	Description
	If, on the *General* view, the *By Series* option is not selected and set up for the chart, this column is empty.
Type column	Select the type of marker to use for this series: • *Column* • *Line*
Slices section	Select each data point and specify a color. This sets the colors of the wedges in a pie chart.
Shape column	Select one of the following marker symbols to appear on the series data points: • circle • diamond • star • triangle • X
Line column	Select a color for the line that connects the data points.
Fill column	Select a color to represent each series.
Performance Color column	Select a color to represent the actual performance values.
Comparative Marker Color column	Select a color to represent the target data values.
Scale High Color column	Select a color to represent high value marker color.
Scale Low Color column	Select a color to represent low value marker color.
High Color column	Select a color to represent the color for high values.
Low Color column	Select a color to represent the color for low values.
Line Thickness	Select a series and set the thickness for the lines that connect the data points. ⅈ Note For *Combination Charts*, this option is available only when *Line* is selected in the *Type* column for the selected series.
Enable Markers	Select this option to show data point markers on the line graph.
Enable Series Markers	Select this option to show the data point markers on the line graph.
Marker Size	Specify the size of the data point marker.

Option	Description
	i Note For *Combination Charts*, marker size can be set for both the *Column Settings* and the *Line Settings*.
Marker Gap	Specify the amount of space between the markers for each series.
Marker Overlap	Specify how much the bars or columns for each series overlap the bars or columns of the other series. If set to 100, they appear in the same bar or column space to make a single bar or column that changes color to show the second series.
Transparency	Set the visibility of objects behind the chart or chart markers. **i** Note For *Combination Charts* , transparency is set for both the *Column Settings* and the *Line Settings*.
Performance Marker Size	Specify the size of the marker that indicates the actual performance value.
Comparative Marker Size	Specify the size of the marker that indicates the target value.
Show Lines	Specify if lines should be visible between the sections that show different series values. If this option is selected, specify the line thickness.
Positive Color	Specify a color for positive values.
Negative Color	Specify a color for negative values.
Bar Size	Specify the width of the bars in the chart.

Axes tab

Use the following table as a guide for setting the *Appearance* properties for chart axes. Some options are available only for specific chart types.

Option	Description
Enable Vertical Axis	Select this option to include a line along the vertical axis. If selected, you can specify the thickness and line color, and whether to show ticks to indicate major and minor values.

Option	Description
Enable Horizontal Axis	Select this option to include a line along the horizontal axis. If selected, you can specify the thickness and line color, and whether to show ticks to indicate major or minor values.
Horizontal Gridlines	Choose whether to display gridlines for the horizontal axis. You can select *Show Major Gridlines* and *Show Minor Gridlines* and set the line color and thickness of the lines.
Vertical Gridlines	Choose whether to display gridlines for the vertical axis. You can select *Show Major Gridlines* and *Show Minor Gridlines* and set the line color and thickness of the lines.

Text tab

Use the following table as a guide for setting the *Appearance* properties for chart text. In the *Show* column, select the checkbox for each text item you want to display on the chart. Then, select the text item and format the text appearance. For more information about formatting text, see *Formatting text for component elements*.

Some options are available only for specific chart types.

Option	Description
Chart Title	Displays a main title for the chart.
Sub Title	Displays a secondary title for the chart.
Header Labels	Displays labels above the columns in the chart.
Series Labels	Displays the name for each data series.
Series Sub-Labels	Displays a secondary label beside the series label for each data set.
Start Value	Displays the first value in the data set to the right of the plot area.
Low Value	Displays the lowest value in the data set to the right of the plot area.
High Value	Displays the highest value in the data set to the right of the plot area.
End Value	Displays the last value in the data set to the right of the plot area.
Vertical Axis Title	Displays a title along the vertical axis.
Horizontal Axis Title	Displays a title along the horizontal axis.
Vertical Axis Labels	Displays value labels along the vertical axis.
Horizontal Axis Labels	Displays value labels along the horizontal axis.
Vertical (Value) Axis Title	Displays a title for the value scale along the vertical axis.

Option	Description
Vertical (Value) Axis Labels	Displays labels for values along the vertical axis.
Horizontal (Value) Axis Title	Displays a title for the value scale along the horizontal axis.
Horizontal (Value) Axis Labels	Displays labels for values along the horizontal axis.
Vertical (Category) Axis Title	Displays a category title along the vertical axis.
Vertical (Category) Axis Labels	Displays a label for categories along the vertical axis.
Horizontal (Category) Axis Title	Displays a category title along the horizontal axis.
Horizontal (Category) Axis Labels	Displays labels for categories along the horizontal axis.
Mouse-Over Values	Displays values when the user moves the mouse pointer over a data marker.
Mouse-Over Labels	Displays labels when the user moves the mouse pointer over an area on the Tree Map.
Legend	Displays a legend that defines the symbols and markers in the chart. For more information about using legends, see *Adding legends to charts*.
Data Labels	Displays data labels on or around the chart markers. If the chart contains multiple series, you can specify if data labels are displayed for each series. For more information about using data labels, see *Displaying data labels on charts*.
Range Labels	If the range slider is enabled on the *Behavior* view, this option displays labels on the range slider. For more information about using the range slider, see *Adding Range Sliders to charts*. i Note The range labels ignore duplicate, sequential values in the label data. For example, if the labels are bound to cells in the spreadsheet where year values for the data are inserted, on the range slider the first occurrence of a year appears in a label and subsequent occurrences of that year are ignored. The next year value is displayed and subsequent occurrences for that year are ignored.

Color tab

Use the following table as a guide for setting the *Appearance* properties for chart colors. For more information about using the *Color* dialog box to set component colors, see *Setting colors for component elements*.

Option	Description
Background Color	The color for the background area behind component elements.
Series	The color of the markers that represent the data. You can also set different colors for specific values for some charts, including the *Performance Color*, *Comparative Marker Color*, *Scale High Color* and *Scale Low Color* for bullet charts, the *High Color* and *Low Color* for tree map charts, and the *Initial Value Color*, *Final Value Color*, *Positive Value Color*, and *Negative Value Color* for waterfall charts.
Slices	The color of the markers that represent the data.
Line Colors	The color of the lines that represent each series of data for a sparkline chart. You can also set the *Line Thickness*.
Markers	In the *Markers* list, select the checkbox next to a marker to show an icon at that point on the sparkline chart. You can choose the shape and color of individual markers, and set the size and transparency of all the markers.
Title Area	The background color of the area around the title. You can set colors for the fill and border if the corresponding options are selected in the *Layout* tab.
Plot Area	The background color of the area where the data is plotted. You can set colors for the fill and border if the corresponding options are selected in the *Layout* tab.
Legend Area	The background color of the area around the legend. You can set colors for the fill and border if the corresponding options are selected in the *Layout* tab.
Axes and Gridlines	The color of the chart axes lines and gridlines that are displayed on the chart. You can set colors for the horizontal and vertical axes and gridlines, as well as for major and minor gridlines, if they are enabled.
Line Color	The color of the lines between the slices of a pie chart.

Related Links

4.2.2.4.1 Adding legends to charts

For a chart that shows multiple series of data, a legend can help to define the various markers used in the chart. When you add a legend, you can define the appearance and location of the legend. In addition, you can allow users

to hide or show the data for a specific series at runtime. With this ability, users can choose to view all the data on the chart at once or to simplify the chart and view only some of the series data.

To add a legend

1. Select the chart and open the *Properties* panel.
2. On the *Properties* panel, click ▶ *Appearance* ▶ *Layout* ◢.
3. Select *Enable Legend* and set the following options:

Option	Description
Position	Specify a position for the legend in the chart.
Vertical Offset	If you choose *Left* or *Right* as the *Position*, you can change this setting to shift the legend up or down.
Horizontal Offset	If you choose *Top* or *Bottom* as the *Position*, you change this setting to shift the legend left or right. Negative numbers shift the legend to the left and positive numbers shift it to the right.
Show Fill	Adds a solid background color behind the legend. Click the *Color Selector* (⬜▾) to specify a color for the background.
Show Border	Adds a border around the legend. Click the *Color Selector* (⬜▾) to specify a color for the border. You can also set the *Border Thickness*.
Enable Hide/Show Chart Series at Run-Time	Select this option to allow users to select specific series to show on the chart.
Interaction	If *Enable Hide/Show Chart Series at Run-time* is selected, select one of the following user interaction options: ○ *Checkbox*: adds a checkbox for each series label in the legend. Users click the checkbox to show or hide the series. ○ *Mouse Click*: users click the series name to show or hide the series data on the chart.

Related Links
Setting colors for component elements [page 40]

4.2.2.4.2 Displaying data labels on charts

Data labels allow you to display data values or labels directly on or around the data markers. To show data labels overlaid on the markers, you first mark them to be displayed on the ▶ *Appearance* ▶ *Text* ◢ tab. Then, you can configure their position, contents, and label separators as required.

To configure data labels

1. Select the chart component and open the *Properties* panel.

2. Click ▶ *Appearance* ❯ *Text* ◀.

3. In the *Show* column for *Data Labels*, select the checkbox.

   If the chart contains multiple series, you can enable and configure data labels for each series individually. For *Combination* charts, you must configure some settings individually since line and bar markers have different properties.

4. To set the default format for all data labels, select *Data Labels*. To set the format for the data labels of an individual series, click the arrow in the *Show* column to expand the list of series, and then select a series.

5. In the *Format Selected Text* area, set the text appearance properties.

6. In the *Position* list, select where you want the label to appear relative to the data marker.

   The options available depend on the type of chart. They include *Center*, *Inside End*, *Outside End*, *Best Fit*, *Left*, *Right*, *Above*, *Below*, *Inside Base*, and *Outside with Leader Lines* (Pie charts only).

   For Combination charts, if *Data Label* is selected, the *Position* list is not available. You must select an individual series to set the position of its data labels.

7. In the *Offset* lists, select values to adjust the label location more specifically by setting the X and Y offsets.

8. You can also specify what data to include in the labels for all charts except *Candlestick Charts* and *OHLC Charts*. The available data varies by chart, and can include the *Series Name*, *Category Name*, *Value*, *Percentage*, *X Value*, *Y Value*, *Bubble Size*, *Size Value*, and *Color Intensity*.

9. You can also choose a separator string for labels with multiple pieces of data. In the *Separator* list, select a value or select *Custom* and, in the *Custom* box, type the value to use.

4.2.2.5   Allowing users to adjust chart scales

When you design a chart based on dynamic data that can be entered or adjusted by users at runtime, it can be difficult to set an appropriate scale. For many charts, you can define a manual scale and then include a toolbar to allow users to adjust the scale values at runtime. These runtime tools allow users to easily create a custom scale.

> ℹ **Note**
>
> Runtime tools are not available for pie, tree map, horizontal and vertical bullet, or sparkline charts.

When runtime tools are enabled, a runtime tool button ( ) is available in the upper left corner of the chart. When users move the mouse pointer over this button, the runtime tool menu opens.

Depending on which options are selected, the menu can contain the following buttons:

Button	Description
Grow	Sets the scale to allow zoom out only. The scale grows as data values increase, but does not shrink when data values decrease. This option is useful for users who are inserting data that pushes the scale limits but do not want to reset the scale each time, or for animated charts.
Off	Turns off scaling so the chart uses the minimum and maximum data points as the minimum and maximum scale values. The chart continues using this scale even if the data changes.
Auto	Sets the scaling behavior to auto, so the minimum and maximum scale values automatically adjust to accommodate all data points.
⊚	When chart scaling is set to *Off*, the *Focus Chart Data* button forces the chart axes to rescale based on the current data. The chart then retains that scale even if the data subsequently changes.
↻	The *Reset Chart Scale* button sets the scale values back to the default settings from when the model was opened.

Related Links

To display the Run-time Toolbar menu on charts [page 74]

Using Chart components [page 46]

4.2.2.5.1 To display the runtime tools menu on charts

1. Click the chart component and open the *Properties* panel.
2. Click ▶ *Behavior* ▶ *Common* ◀.
3. Select the *Enable Run-Time Tools* checkbox.
4. To customize which buttons are available on the toolbar menu, select any of the following options:

Option	Description
Show Focus Button	Displays the *focus chart data* button.
Show Reset Scale Button	Displays the *reset scale* button.
Show Scale Behavior Options	Displays the *Grow*, *Off*, and *Auto* buttons.

> **ℹ Note**
>
> If you deselect all the tool options, the runtime tools button is not displayed in the model, even if *Enable Run-time Tools* is selected.

When the model runs, the runtime tools button is available in the upper right corner of the chart.

Related Links

Using the Properties panel [page 36]

4.2.3 Using container components

Containers group, frame, and display other components.

> **i Note**
>
> Container components are not available in Presentation Design.

Container components can be nested within each other to create multi-layer models. To optimize the model's performance, use only one layer of nesting—that is, avoid placing a container element within another container element.

Canvas Container
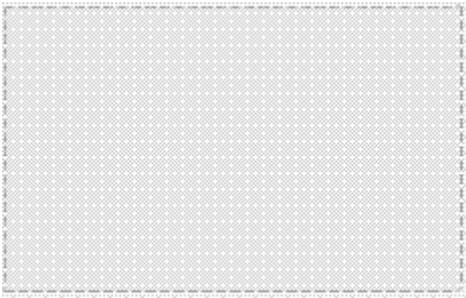
The *Canvas Container* acts as a smaller canvas within the main canvas and can hold one or more components. Components in the *Canvas Container* can be moved, added, changed or deleted as if they were placed on the canvas.

Canvas Containers do not have any art such as backgrounds, borders, heading bars, and so on. In design mode, these containers are visible to allow you to work with them, but when the model is run, the containers are invisible and components in the containers appear to be free floating. If the components in the canvas container extend beyond the container borders, scrollbars are added.

Panel Container

75

The *Panel Container* acts as a smaller canvas within the main canvas and can hold one or more components. Components in the *Panel Container* can be moved, added, changed or deleted. In the *Object Browser* panel, to see the list of components within a panel container, click *Expand* (⊞) next to the *Panel Container* name. The visual variant *Panel Container2* has the same functionality and only differs in appearance.

Tab Set

The *Tab Set* acts as a smaller canvas within the main canvas and contains multiple tab views. Each view can contain one or more components, which can be linked to different data sets. To display a view, users click its corresponding tab.

When configuring a *Tab Set*, you can increase or decrease the number of tab views by selecting the component then clicking the plus or minus icons above the row of tabs. When the model runs, the *Plus* and *Minus* icons are not visible.

> **i Note**
>
> The *Tab Set* functions as a single grouping of tabs. You cannot create subgroups of the tabs within the component.

To see an example of a model with a *Tab Set*, click ▶ *File* ❭ *Samples* ❭ and open the sample file *Tab Set*.

> **i Note**
>
> If you delete a container or remove a tab from a *Tab Set*, all components within the container or tab are also deleted. If you want to keep the child components, move them to another part of the canvas.

4.2.3.1 Container general properties

Use the following table as a guide for setting container component properties on the *General* view.

> **i Note**
>
> The *General* view is available for *Panel Containers* and *Tab Sets* only.

> **i Note**
>
> If you want to set options for a *Tab Set*, select one of the tabs. To set options for an individual *Tab Set* tab, select the tab and then click the canvas area.

Option	Description
Tab Alignment	For a *Tab Set*, this option specifies whether the tabs at the top of the component are aligned to the right, center, or left of the canvas.
Label	For a *Tab Set* tab, type a label for the tab or click the *Cell Selector* ( ) to link the label to a cell in the embedded spreadsheet.
Title	For a *Panel Container*, type a title for the component or click the *Cell Selector* ( ) to link the title to a cell in the embedded spreadsheet. i Note You can configure the title format on the *Appearance* view.

Related Links

Container behavior properties [page 77]
Container appearance properties [page 78]

4.2.3.2   Container behavior properties

Use the following table as a guide for setting the *Behavior* properties for containers. Some options are available only for specific container types.

Option	Description
Common Tab	
Selected Item	In the *Item (By Position)* list for a *Tab Set*, select the tab that opens by default when the model loads. This list identifies tabs by numerical values according to their position, so Item 1 corresponds to the first tab on the left, Item 2 corresponds to the second tab, and so on. You can also bind this setting to a cell in the spreadsheet.
Horizontal Scroll Bar	Specify when a horizontal scroll bar is available on the canvas. Choose *On* to display the scroll bar at all times, *Off* to disable it, or *Auto* to show the scroll bar if the contents of the canvas exceeds its width.
Vertical Scroll Bar	Specify when a vertical scroll bar is available on the canvas. Choose *On* to display the scroll bar at all times, *Off* to disable it, or *Auto* to show the scroll bar if the contents of the canvas exceeds its height.
Dynamic Visibility	This option allows you to specify conditions that control when the container is visible in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162]. i Note

Option	Description
	You cannot set dynamic visibility for individual canvases on a *Tab Set*.
Animations and Effects Tab	
Entry Effect	Select how you want the container to appear on the canvas when the model opens or when the dynamic visibility of the container is triggered. You can select a *Type* of entry effect and specify its *Duration*. **i** Note You cannot set entry effects for individual canvases on a *Tab Set*.

Related Links

Container general properties [page 76]
Container appearance properties [page 78]

4.2.3.3 Container appearance properties

Use the following table as a guide for setting the *Appearance* properties for containers. Some options are available only for specific container types.

Option	Description
Text Tab	
Text list	Lists the elements that can have text. To hide the text for an element, deselect it. **i** Note You cannot deselect the tab labels on the *Tab Set*.
Format Selected Text	The settings in this area allow you to format the text for the selected text element. **i** Note If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.
Color Tab	
Background Color	The color for the background area behind the *Canvas Container*.
Panel Color	The color for the frame around the *Canvas Container*.

Dashboards and Presentation Design User Guide
Working with Components

Option	Description
Tab Backgrounds	The color for tabs on a *Tab Set*. You can set the *Default Color* for when the tab is not active, the *Over Color* for when the mouse pointer moves over the tab, the *Selected Color* for when the tab is active.
Scroll Bar	The colors for the following parts of the scroll bar: • *Track Color* • *Thumb Color* • *Button Color* • *Button Symbol Color*
Labels	The colors for the tab labels of a *Tab Set*. You can set the *Default Color* for when the tab is not active, the *Over Color* for when the mouse pointer moves over the tab, the *Selected Color* for when the tab is active. > **i Note** > You can also set the label text color on the *General* view.

Related Links

4.2.4 Using selector components

Selector components add interactivity to your models by allowing users to pick different options during runtime. You can configure selectors to insert data such as the selected item's row, position, value, and label into the embedded spreadsheet. Other components can then retrieve that data to display more specific information to the user, for example.

> **i Note**
>
> If available, in the *Data Insertion* area for selector components, you can click *Information* ( ) to see a quick visual tutorial about the component.

Accordion menu
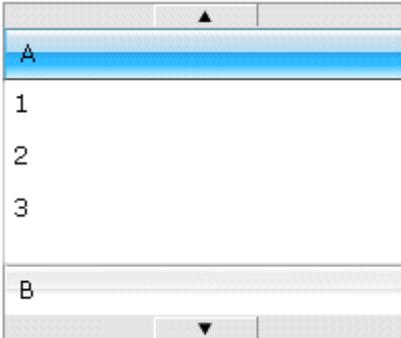
The *Accordion Menu* is a two-level menu that lets users select a category and then select from items within that category.

> **i Note**
>
> This component is not available in Presentation Design.

> **➜ Tip**
>
> To see an example of a model with an *Accordion Menu*, click ▌ *File* ❯ *Samples* ▐ and open the sample file *Accordion Menu*.

Check Box

The *Check Box* allows users to toggle between two states: checked and unchecked. You can assign specific values to the checked and unchecked states, or bind the value of each state to the embedded spreadsheet.

Combo Box
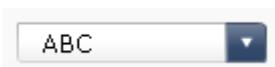
The *Combo Box* drops down a vertical list of items when it is clicked. Users can then select an item from the list.

Filter

The *Filter* looks at a range of cells with multiple fields of data and categorizes them by multiple sets of filter criteria. As users select different information in each filter list, the *Filter* sifts through the data range and inserts the corresponding data

> ➡ Tip
>
> To see an example of a model with a *Filter*, click ▌ *File* ❯ *Samples* ◣ and open the sample file *Filter*.

Hierarchical Table

Geography	Quantity Solc	Revenue
▾ ☐ ALL	11447982	348591052
► ☐ NORTH AM	8459421	257589369
► ☐ EUROPE	1287820	21184639
► ☐ ASIA_PAC	1243482	14921784
► ☐ MIDDLE_E.	457259	3658072

The *Hierarchical Table* allows users to view and select query data organized in a multiple-level hierarchy. The hierarchical list appears in the first column of the table. If a row in the hierarchy has one or more sub-entries, it displays a drop-down icon to its left that toggles displaying its sub-entries.

Users can select multiple items in a *Hierarchical Table*; for more information about the table's insertion properties, see *To configure insertion for a hierarchical table* [page 93].

Hierarchical Tables with BEx query data can also be configured as RRI (Report-Report Interface) senders. When RRI is enabled for a *Hierarchical Table*, users can open an action list for the BEx query at runtime and jump to external resources such as queries, reports, transactions, and websites. For more information, see *Report-Report Interface* [page 220].

> ℹ Note
>
> This component is not available in Presentation Design.

Fisheye Picture Menu

The *Fisheye Picture Menu* allows users to select from a set of pictures or icons. As the mouse is moved over each item in the menu, the item is magnified. The closer the mouse is to the center of the item, the more the item is magnified. This creates an effect similar to that of a fisheye lens.

> **i Note**
>
> This component is not available in Presentation Design.

> **➜ Tip**
>
> To see an example of a model with a *Fisheye Picture Menu*, click ▌ *File* ❯ *Samples* ◢ and open the sample file *Fisheye Picture Menu*.

Icon

The *Icon* can be used as a selector or display component. As a selector, it functions like the *Check Box* component.

You can also configure alert settings for an icon, so that it changes color depending on its value relative to a target value. For more information about alerts, see *Understanding alerts* [page 158].

> **i Note**
>
> To configure the *Icon* as a display-only component, click the *Appearance* tab and deselect *Show On/Off*.

Label Based Menu

The *Label Based Menu* lets users select items from a vertical or horizontal grouping of buttons.

List Box
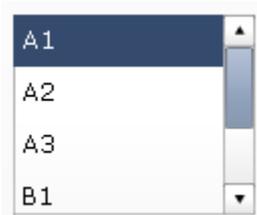
The *List Box* lets users select items from a vertical list.

List Builder
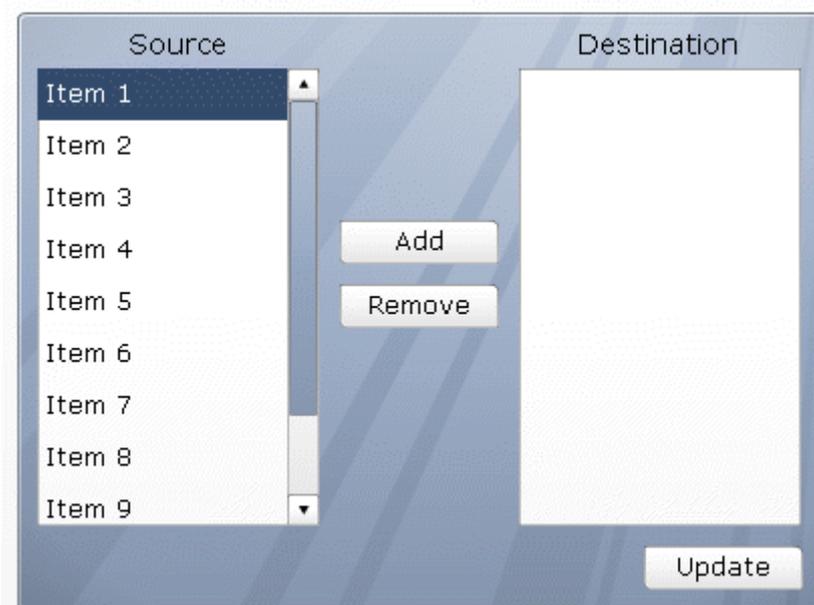
The *List Builder* provides a way for users to build a set of data from a large list of items. The *List Builder* consists of a *Source* list containing all possible selections, a *Destination* list containing the selected items, and an *Update* button. Users move items between the *Source* and *Destination* lists in one of the following ways:

- Double-clicking items in either the *Source* or *Destination* list to move the item to the other list.
- Dragging items from one list to another.
- Selecting an item and clicking *Add* or *Remove*.

When the *Update* button is clicked, the items in the *Destination* list are inserted into the destination range to be used by other components. To change the order of items in the *Destination* list, drag an item up or down in the list.

> **i Note**

This component is not available in Presentation Design.

> → Tip
>
> To see an example of a model with a *List Builder*, click ▶ *File* ❯ *Samples* ❯ and open the sample file *List Builder*.

List View

The *List View* allows users to view and sort a large set of data, and can also act as a selector to insert more detailed data to the embedded spreadsheet.

This selector has a scroll bar and its columns can be sorted and resized during runtime.

Play Selector
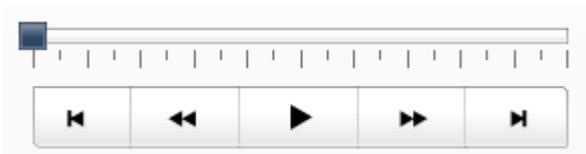
The *Play Selector* sequentially inserts one row or column after another from a defined range. You can link the destination cells to a chart so the chart data changes each time the *Play Selector* inserts a new row or column. This component can display a large amount of data with an animated feel, since users do not have to click each selection to view the data.

> ⓘ Note
>
> This component is not available in Presentation Design.

Push Button

The *Push Button* inserts specific data each time it is clicked. It can be used to allow users to restore single value components to their default values without resetting the values for the entire model, or to create a pop-up window that appears when the button is pushed.

You can configure the *Push Button* to set the values of one or multiple cells to specific values when users click the button. The values change only when the user clicks the button, so users have more control over when the data is copied to the destination cells.

Radio Button
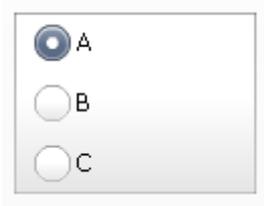
The *Radio Button* lets users select one item from a vertical or horizontal group of items.

Scorecard

The *Scorecard* allows you to display visual alerts in a grid. At runtime, users can click column headings to sort the grid according to the column's data. This component is similar to a *List View*, but it includes options to enable and configure individual sets of alerts for each column.

Alert levels can be displayed with cell background colors, font colors, or alert icons with or without values. For more information about alerts, see *Understanding alerts* [page 158].

Sliding Picture Menu

The *Sliding Picture Menu* allows users to select from a set of icons or pictures. The users can use arrows to scroll through the icons, or the menu can be configured to scroll through the icons as the user moves the mouse.

> **i Note**
>
> This component is not available in Presentation Design.

> **➡ Tip**
>
> To see an example of a model with a *Sliding Picture Menu*, click ▶ *File* ❯ *Samples* ◗ and open the sample file *Sliding Picture Menu*.

Spreadsheet Table

The *Spreadsheet Table* displays data exactly as it appears in the source range in the embedded spreadsheet. This selector functions like the *List View*, allowing users to view more detailed data by clicking a row.

> **i Note**
>
> The spreadsheet table component can be used as a display component rather than a selector component. To use the spreadsheet table as a display component only, deselect the *Row Selectability* for all rows on the ▶ *Behavior* ❯ *Common* ◗ tab.

Ticker

86

Dashboards and Presentation Design User Guide
Working with Components

The *Ticker* displays horizontally scrolling text in your model. Scrolling stops when the user moves the mouse pointer over the labels, and you can configure the component to insert data to the spreadsheet each time the user clicks a text item.

Toggle Button

| Off |

The *Toggle Button* lets users toggle between two states: on and off.

4.2.4.1   Selector general properties

Use the following table as a guide for setting the selector component properties on the *General* tab. Some options are available only for specific selector components.

Option	Description
Title	Specify the main title for the selector. For the *List Builder*, you can also specify titles for the *Source* list and *Destination* list. ➡ Tip You can change the location and format of the title on the *Appearance* view.
Labels	Set the labels that identify selector elements. Do one of the following steps: • To type the label text, click *Manual Edit* (🖉). • To retrieve the label text from the spreadsheet or query result data, click the arrow next to the *Bind* button and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet or select *Query Data* to bind the values to query results. For the *List Builder*, you can also specify the *Add Button*, *Remove Button*, and *Update Button* labels.
Value Labels	For a *Ticker*, this property sets the values associated with each label. Click the *Cell Selector* (📌) to select the value labels from the embedded spreadsheet.
Display Data	Set the range of data that appears in the component. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet or *Query Data* to bind the values to query results.
Number of Filters	For a *Filter*, this option specifies the number of drop-down filter lists.

Option	Description
Display Value	For an *Icon*, the display value appears in a tooltip when the user moves the mouse pointer over the component. Type a value or click the *Cell Selector* ( ) to select the label text from the embedded spreadsheet. i **Note** The value must be numeric. If a non-numeric value is entered, a zero is displayed.

Images	Set the image files used in the component. Embedding an image decreases loading time at the cost of increasing the model's file size. Select one of the following:
	
	• *Embedded*: copies of the image files are stored in the model. Click *Import* to select the file to embed.
	
	  i **Note**
	  The order that images appear in the menu is determined by the order in which they are imported. Import the images in the order you want them to appear in the menu.
	
	• *By URL*: the model references the specified URL to load the image files. Type the URL or click the *Cell Selector* ( ) to link the URL to a cell in the spreadsheet.

Data Insertion	The settings in this section define the type of data the selector controls, as well as the source and destination locations. When a selection is made, data is copied from the source location to the destination location. If the data in the source location changes after a selection is made, the data in the destination location is not automatically updated. i **Note** *Scorecards*, *Tickers*, *Icons*, and *Spreadsheet Tables* can act as display components. That is, you can disable the selector functionality in the *Behavior* view to prevent the user from interacting with the component at runtime. Some selector components can be configured to insert multiple data items with a single selection. For more information about multiple insertions, see *Configuring selectors for multiple insertions*. i **Note** The *Hierarchical Table* component has a separate *Insertion* view that contains data insertion options. *Hierarchical Tables* can display and insert data that is bound to a query only. For more information about data insertion with *Hierarchical Tables*, see *To configure insertion for a hierarchical table*.
Insertion Type	Specifies the structure of the data that is inserted when a selection is made. Select one of the following options: *Position*

Dashboards and Presentation Design User Guide
Working with Components

Option	Description
	Label
	Value
	Row
	Column
	Filtered Rows
	Status List
	The available options vary depending on the type of selector. For a description of each *Insertion Type*, see *Understanding insertion types*.
Source Data	Specify the location of the source data for insertion. Click the *Cell Selector* (⬚) to select the range in the spreadsheet. For *Check Box*, *Icon*, and *Toggle Button* components, click *Manual Edit* (⬚) to manually enter values. ℹ Note If the *Insertion Type* is set to *Position*, *Label*, or *Status List*, source data cells are not available.
Data	For *Hierarchical Tables*, click the *Bind* button (⬚) to choose hierarchical source data from a query. In the *Select from Query* dialog box, choose a query from the list, choose the hierarchy that you want to display in the table, and then select the additional result objects that you want to show. Click *OK*.
Configure Columns	For *Hierarchical Tables*, *List Views*, or *Scorecards*, click this button to open the *Configure Columns* dialog box and change the appearance and labels of the table columns. ℹ Note Some options may be unavailable, depending on the component selected or the current data binding. For example, disabling *Bind Column Headers* will make *Treat First Row as Header* unavailable. Click the up arrow for a column to move it to the left in the table, or click the down arrow to move it to the right. Click *Column Headers* or *Column Width* and type a value to change the header or size of the column. Select *Hide* for columns that you do not want to display in the table. To choose column headers from cells in the spreadsheet, select *Bind Column Headers* and pick the appropriate range in the spreadsheet.
Default Display	For *Hierarchical Tables*, choose how many levels of data appear expanded by default. You can choose from three options:

Option	Description
	Expand Level: to specify the number of levels that are expanded by default, select this option and type the number of levels in the box. *Expand All Levels*: select this option to expand all levels in the hierarchical table by default. *Collapse All Levels*: select this option to show only the first level of data in the hierarchical table by default. ⓘ Note These settings do not restrict users from accessing data in the table; users can expand and collapse items in the hierarchy at runtime.
Destination	Specify the location in the spreadsheet where the source data will be inserted. Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet or *Query Data* to bind the values to query results. ⓘ Note The required size of the destination range varies according to the structure selected in *Insertion Type*. For structures that use columns or rows of data, be sure to select a range that is large enough to contain your largest column or row of data.
Remove Destination When No Selected Item	Select this option to delete any existing data in the destination location when no component items are selected. If this option is not selected, the data remains in the destination location until another selection is made.
Insert On	Select one of the following options to specify when data is inserted into the destination location: • *Data Change and Interaction*: data is inserted whenever the data changes or when the user makes a selection. • *Interaction Only*: data is inserted only when the user makes a selection.
Orientation	Specifies the orientation for the selector. Select either *Horizontal* or *Vertical*.
Categories	Specifies the categories of an *Accordion Menu*. Click *Add* (⊞) to add categories and use the following options to configure each category.
Name	For *Accordion Menus*, this parameter specifies the category name for the item selected in the categories list. Type a name or click the *Cell Selector* (▣) to select the name from the spreadsheet.
Labels	For *Accordion Menus*, this parameter specifies the labels for items within the selected category. Type a label or click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet or *Query Data* to bind the values to query results.

Option	Description
Source Data	For *Accordion Menus*, this parameter specifies the source data associated with the labels for the selected category. Click the *Cell Selector* ( ) to select the label from the spreadsheet.
Category Name Destination	For *Accordion Menus*, this parameter specifies the target cell to insert the name of a category when it is selected. Click the *Cell Selector* ( ) to select the label from the spreadsheet.
RRI Sender	For *Hierarchical Tables* that use a BEx query as a data source, select *RRI Sender* to enable the action list. At runtime, when users point to a table cell, a button appears next to the corner of the cell. Users can click the button to open an action list. Destinations that have been defined for the BEx query appear in this list, and users can click the destinations to view external data. For more information, see *Report-Report Interface* [page 220].
Include Dynamic Jump Destinations	For *Hierarchical Tables* with *RRI Sender* selected, you can select *Include Dynamic Jump Destinations* to show Report-Report Interface destinations that change when users open the action list from different rows. When the user opens an action list with dynamic destinations, the software loads contextual information from the BEx query for those destinations. The contextual information changes based on the row that the user accessed the action list from. For example, a BEx query with sales figures for a number of companies might have a sender-receiver relationship with a website that shows current stock prices. Since this destination is dynamic, users point to the table row for a specific company to open a contextual action list, and the action list allows them to jump directly to online stock price information for that company. If this option is not selected, only static destinations appear in the action list, and the list does not need to be updated each time the user opens it. ⓘ Note The software does not send the parameters of query prompts or SAP variables to the action list or to Report-Report Interface destinations. Therefore, the list of actions and the information that appears in the destination may be less specific than the filtered query data that appears in the Dashboards model.

Related Links

To configure component properties [page 45]
Selector behavior properties [page 93]
Selector appearance properties [page 99]

4.2.4.1.1 Configuring selectors for multiple insertions

The following selectors can be configured to retrieve source data from multiple locations and insert the data into multiple destination locations with a single selection:

- *Combo Box*
- *Hierarchical Table*

- *Fisheye Picture Menu*
- *Sliding Picture Menu*
- *Label Based Menu*
- *List Box*
- *List Builder*
- *Radio Button*
- *Scorecard*
- *Ticker*
- *Spreadsheet Table*
- *Play Selector*

When you set up a selector for multiple insertions, you specify the source and destination for each chunk of data the same way you would set up the data source and destination for a single insertion. For each chunk, you can define the insertion type (row, column, label, position, status).

> **i Note**
>
> The *Filtered Rows* insertion type is not supported when a selector is configured for multiple insertions.

To set up multiple insertion behavior

1. Select the component and open the *Properties* panel if it is not already open.

2. On the *General* view, under *Data Insertion*, click *Add* (+) to add the number of items you want to insert with each selection.

   > **i Note**
   >
   > For *Hierarchical Tables*, this button is found on the *Insertion* view instead of the *General* view.

3. In the list of insertion items, select the first item and set the following options:

Option	Description
Name	Type a name for the insertion item.
Insertion Type	Select the type of data that is inserted for this item. For a description of each insertion type, see *Understanding insertion types*. > **i Note** > When multiple insertion is used, *Filtered Rows* is not available as an insertion type.
Source Data	Click the *Cell Selector* ( ) and select the source data location. > **i Note** > If the insertion type is set to *Label*, *Position*, or *Status List*, the source data is already set and this box is not available.

Option	Description
Destination	Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet or *Query Data* to use the values as parameters for a query prompt.

4. Repeat the previous step for the other insertion items in the table.

4.2.4.1.2   To configure insertion for a hierarchical table

Add a *Hierarchical Table* to the canvas and connect it to a query data source before configuring these insertion properties.

When the embedded spreadsheet is the destination, insertion behavior for a *Hierarchical Table* is different than other selector components. Users can select multiple items on a *Hierarchical Table*, and the data for each selected item is inserted if there is empty space in the destination range. Once the destination range is full, new selections overwrite old data.

Hierarchical Table can also insert data to a prompt for another query. They are capable of inserting multiple values for prompts that allow multiple selections.

1. Select the *Hierarchical Table* and open the *Properties* panel.
2. If you want to limit users to selecting only one table row at a time, on the *Behavior* view, deselect *Allow Multi-selection*.
3. On the *Insertion* view, click *Add* (⊞) to add the number of items you want to insert with each selection.
4. In the list of insertion items, select the first item and set the following options:

Option	Description
Name	Type a name for the insertion item.
Insertion Type	Select the insertion type for the item. Choose one of the following options: ○ *Value*: when a user selects a row, the table inserts the value of a single result object in that row. You specify the result object in the *Source Data* list. ○ *Row*: when a user selects a row, the table inserts the value of all result objects in the row.
Source Data	For a table with *Value* as the *Insertion Type*, choose the result object that is inserted.
Destination	Click the arrow next to the *Bind* button beside the box and select *Spreadsheet* to bind the values to cells in the embedded spreadsheet (the Bind button shows a spreadsheet) or *Query Prompt* to use the values as parameters for a query prompt (the Bind button shows a query). *Query Prompt* is available only if you selected *Value* as the *Insertion Type*.

5. Repeat the previous step for the other insertion items in the table.

4.2.4.2   Selector behavior properties

Use the following table as a guide for setting the *Behavior* properties for selectors. Some options are available only for specific selector types.

Option	Description
Common tab	
Insert On	Specify how the component responds when users interact with it. Select one of the following options: • *Mouse Click*: data is inserted when users click an item. • *Mouse Over*: data is inserted when users move the mouse over an item. For *Label Based Menu* components, the *Insert On* option is only available when the *Always Expanded* option is selected.
Slider Method	For a *Sliding Picture Menu*, this option specifies how the menu scrolls through the thumbnails. Select one of the following: • *Buttons*: user clicks buttons at each end of the menu to scroll. • *Mouse*: user moves the mouse across the thumbnails to scroll.
Slider Amount	For a *Sliding Picture Menu*, this option specifies how far the slider moves. Select one of the following options: • *One*: scrolls through thumbnails one at a time • *Page*: scrolls through thumbnails a page at a time. The number of thumbnails on a page is determined by the width of the component.
Scroll Speed	For a *Sliding Picture Menu*, this option adjusts the speed of the slide behavior as the user scrolls through the thumbnails.
Slide Speed	For an *Accordion Menu*, this parameter specifies how quickly categories expand when the user clicks the category label.
Play Time (seconds)	For a *Play Selector*, this parameter specifies the length of time that it takes to run through a complete sequence, displaying the whole range of source data. Select the number of seconds from the list.
Auto Play	If you select this option for a *Play Selector*, it plays automatically when the model opens.
Auto Rewind	If you select this option for a *Play Selector*, it rewinds itself automatically when the sequence has finished playing.
Auto Replay	If you select this option for a *Play Selector*, it automatically restarts the sequence when it finishes playing it.
Zoom Size	For a *Fisheye Picture Menu*, this option specifies the size of the popup effect when the mouse moves over an image. Slide the size slider to adjust the size.
Zoom Speed	For a *Fisheye Picture Menu*, this option specifies the speed that images popup on the selector when the mouse moves over the images. Slide the speed slider to adjust the speed.

Option	Description
Always Expanded	If you select this option for a *Label Based Menu*, the menu always displays the available labels. Otherwise, you must specify how users expand the menu and the direction that the labels expand towards. ℹ **Note** If this option is selected, the *Insert On* option is not available.
Expand On	For a *Label Based Menu*, specify how the component responds when users interact with it. Select one of the following options: • *Mouse Click*: the menu is expanded when users click a tab. • *Mouse Over*: the menu is expanded when users mouse over a tab.
Expand Direction	For a *Label Based Menu*, this option specifies the direction that the menu expands towards when it is opened. The available options depend on the *Orientation* setting on the *General* view. When the orientation is set to *Horizontal*, you can choose *Left* or *Right*. When the orientation is set to *Vertical*, you can choose *Center Down*, *Left Down*, *Right Down*, *Center Up*, *Left Up*, or *Right up*.
Open Animation	When this option is selected for a *Label Based Menu*, the menu opens in an animated fashion instead of expanding instantly.
Ticker Options	In this area, you can set options specific to the *Ticker*.
Item Separator	Specify the separator character used between items on a *Ticker*.
Scroll Direction	Specify the direction the data scrolls across a *Ticker*. Select *Left* or *Right*.
Scroll Speed	Specify how quickly the data scrolls across a *Ticker*.
Allow Column Sorting	Select this option to allow users to sort table rows at runtime. If selected, users can click a column heading to sort the data in that column alphabetically or by value.
Rows are Selectable	Select this option to use the table as a selector. When selected, users can click a row to insert the source data into a defined destination. With this option cleared, the table does not insert data during runtime even if you have configured *Data Insertion* options.
Selected Item: *Category*	For an *Accordion Menu*, this option specifies the category that is open by default when the model runs. Select the category based on its relative position in the list of categories. The first category is selected by default.
Selected Item: *Type*	Specify the type of data that will be matched to determine the default selection. If a match is not found, the selection is set to none. Choose one of the following options:

Option	Description	
	Option	**Description**
	Label	The data in the source location is matched to the component's labels.
	Position	The position of each selection item for the component is assigned a number value starting from 1.The data in the source location is matched to the position numbers.
	Dynamic	The position of each selection item is assigned a number starting from 0. The data in the source location is first matched to the component's labels. If there is no match, the data is matched to the position number. If a match is still not found, the first item is selected. This option is not available for the *List Builder*.
Selected Item: *Item*	Specify the item that is selected by default when the model runs. Select the item based on its name or relative position in the list of items. If you choose *No Selection*, an item is not selected when the model opens. **i Note** For the list builder, you must choose source data in the spreadsheet or click *Manual Edit* (⬚). The selected items are added to the *Destination* list, but are not added to the destination cells until the user clicks *Update*. To insert data for the selected items when the model opens at runtime, select the *Insert Items at Run-time* option.	
Selected Item: *Insert Selected Item*	Specify a location in the spreadsheet to store the selected item value. **➜ Tip** To reset a selector component to a "no selection" state, add another selector to set this value to None when *Insertion Type* is set to *Label* and -1 when it is set to *Position.* **i Note** If the type of the selected item is "Dynamic", *Insert Selected Item* is not available.	
Selected Item: *Insert Items at Runtime*	For a *List Builder*, select this option if you want to add items to the *Destination* list and insert the data from those items into the destination cells when the model opens at runtime. If you enable this option, your selections in the *Item* box are added to the *Destination* list and inserted into the destination cells in the embedded spreadsheet.	
Hide Empty Categories	For an *Accordion Menu*, select this option to hide categories that do not have any label data.	

Option	Description
Row Selectability	This option determines whether users can select specific rows in a table. By default, all rows are selectable. To prevent a row from being selected, clear the option beside the row number.
Rows are Selectable	This option determines whether users can select rows in the table. Clear this option to use the selector as a noninteractive display component.
Allow Multi-selection	For a *Hierarchical Table*, select this check box to allow users to select multiple items in the table.
Display Check Box	Select this option to show checkboxes for each row that help users identify which rows are selected.
Ignore Blank Cells	If this option is selected, the component does not display blank cells past the last non-empty label. This option is useful when the number of selections in the component varies.
Dynamic Visibility	This option allows you to specify conditions that control when the selector is visible in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].
Animations and Effects tab	
Drop-Down Animation	With this option selected, *Combo Boxes* drop down with an animated effect instead of expanding instantly.
Enable Sound	You can add a sound effect to some components. The sound effect is triggered when users interact with the component.
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.

Related Links

Selector general properties [page 87]
Selector appearance properties [page 99]

4.2.4.2.1   Understanding insertion types

When you configure a selector component, you set it up to insert specific data into the embedded spreadsheet. That data can then be used by other components. Depending on what you want to do with the data, you can insert different types of data to the target cells. The available options vary depending on the type of selector.

Insertion Type	Description
Position	The selected item's position in a list of items is written to the target cell. For example, if the second item in a list is selected, a value of 2 is written to the target cell.
Label	The label specified for the selected item is written to the target cell. For example, if B is selected from the options A, B, and C, the label "B" is inserted in the target cell. This insertion type is useful when labels can change at runtime and the labels are used for a subsequent data search.
Value	The value of the selected source cell is written to the target cell. The number of cells specified in the source data range needs to match the number of labels displayed in the component. **i Note** For *Hierarchical Tables*, multiple values can be selected and inserted. Each value that a user selects is inserted into the destination range until the range is full. At that point, the oldest data value is overwritten each time the user makes a new selection.
Row	The selected row from the source data is written to the target row. For example, if the user chooses the third item in a *Combo Box*, the third row of source data is inserted to the target row. **i Note** For *Hierarchical Tables*, multiple rows can be selected and inserted. Each row that a user selects is inserted into the destination range until the range is full. At that point, the oldest row of data is overwritten each time the user makes new selection.
Column	The selected column from the source data is written to the target column. This insertion type functions similarly to *Row* insertion.
Filtered Rows	This insertion type allows you to insert multiple rows when a single item is selected. The selector sorts rows of source data by their labels and displays each unique label. When the user selects an item, all the rows that share the selected label are

Insertion Type	Description
	written to the target destination. The inserted rows keep the same order as in the source data. If the *Destination* range does not have enough rows to contain the data, the lower rows are not inserted.
Status List	A 1 is entered in the target cell for the selected item and a 0 is entered in the target cells for all other items in the list.

Related Links

4.2.4.3   Selector appearance properties

Layout tab

Use the following table as a guide for setting the *Appearance* properties for selector components. Some options are available only for specific selectors.

Option	Description
Sizing Method	For selectors with images, this property specifies how images are sized to fit the thumbnail area. Select one of the following options: • *Scale*: images are resized, but they retain their aspect ratio and may not completely fill the available space. • *Stretch*: images are resized and may be stretched horizontally or vertically to fit within the thumbnail space exactly.
Width	For selectors with images, this property specifies the width of the thumbnail area in pixels.
Height	For selectors with images, this property specifies the height of the thumbnail area in pixels.
Horizontal Margin	For selectors with images, this property specifies the horizontal spacing between thumbnail images in pixels.
Vertical Margin	For selectors with images, this property specifies the vertical spacing between the thumbnail images and the area border in pixels.
Number of Labels Displayed	This property specifies the number of labels available in a selector list or menu. If more labels are defined for the component, users can scroll down to see the additional labels. You can display between 2 and 50 labels.
Filter Gap	For a *Filter*, this property specifies the amount of space between each list.

Option	Description
Marker Size	This property specifies the size of a *Radio Button*. Select a size between 1 and 300.
Horizontal Margin	This property specifies the size of the margins at the left and right edges of a *Radio Button*.
Vertical Margin	This property specifies the vertical distance between items on a *Radio Button*.
Enable Background	This property specifies whether the background behind *Radio Button* is visible. If selected, you can set the transparency level for the background.
Button Separation	For a *Label Based Menu*, this property specifies the amount of space between tabs.
Auto Replay	These options specify which elements are displayed for a *Play Selector*. Select the appropriate checkbox to display the following elements: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*Show Background*</td><td>Displays a background behind the controls.</td></tr><tr><td>*Show Progress Indicator*</td><td>Displays a progress slider.</td></tr><tr><td>*Show Rew/Fwd*</td><td>Displays fast forward and rewind buttons.</td></tr><tr><td>*Show Prev/Next*</td><td>Displays previous and next buttons. These buttons let users cycle through the values one by one.</td></tr></table>
Transparency	This property specifies whether objects behind the component are visible. A transparency of 0 means that users cannot see objects behind the component. A transparency of 100 means that the component is completely transparent and invisible to users. i **Note** For components with drop down lists, such as the *Combo Box*, this parameter is called *List Transparency*, and applies only to the list portion of the component.
Show Ticker Background	For a *Ticker*, this option specifies whether the background behind the labels is visible.
Show Button Background	For a *Push Button*, this option specifies whether the background behind the button label is visible. If selected, you can also set the transparency level by moving the slider to the right for more transparency and to the left for less transparency.
Hierarchy Level Indentation	For a *Hierarchical Table*, this option specifies the horizontal indentation of each hierarchical sub-level relative to its parent. The size is in pixels.

Option	Description
Expand / Collapse Icons	For a *Hierarchical Table*, this option specifies the style of the icons that expand or collapse each hierarchical level. The available choices are *Arrows* and *Plus/Minus* icons.
Show Gridlines	This parameter specifies whether gridlines are visible in a table component. For most tables, you can choose to select *Show Vertical Gridlines* and *Show Horizontal Gridlines*. If selected, you can click the *Color Selector* icon under the *Color* tab to specify the line color.

Text tab

Use the following table as a guide for setting the *Appearance* properties for selector text. In the *Show* column, select the checkbox for each text item you want to display on the selector. Then, select the text item and format the text appearance. For more information about formatting text, see *Formatting text for component elements*.

The available properties vary depending on the type of selector.

> **i** Note
>
> The *Text* tab is not available for spreadsheet table and play selector components.

Option	Description
Title	The main title for the component.
Labels	The labels on the component elements.
Source List Title	For a *List Builder*, select this option to display a title above the *Source* list.
Source List Labels	For a *List Builder*, click this option to format labels for items in the *Source* list. The labels cannot be disabled.
Destination List Titles	For a *List Builder*, select this option to display a title above the *Destination* list.
Destination List Labels	For a *List Builder*, click this option to format labels for items in the *Destination* list. The labels cannot be disabled.
Button Labels	For a *List Builder*, click this option to format labels on the buttons that move items from one list to the other. The labels cannot be disabled.
Display Values	For a *Ticker*, this option displays the data value for each item.
Header	A header row for a table component.
Mouse Over Label	For an *Icon*, select this option to display the component label in a tooltip when the user moves the mouse pointer over the icon.

Option	Description
Mouse Over Value	For an *Icon*, select this option to show the display value in a tooltip when the user moves the mouse pointer over the icon.
Category Names	For an *Accordion Menu*, click this option to format the category names. This option cannot be deselected.
Data	For a *Hierarchical Table*, select this option to display the data in all columns. To display or hide the text for individual columns, you can click the arrow icon to expand the *Data* list and select or deselect the checkboxes for the columns.
Format Selected Text	The settings in this area allow you to format the text for the selected text element. **i Note** If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.
Position	Sets the position of a text element relative to its component.
Offset	Adjusts the position of a text element by offsetting it relative to their set position.
Number Format	Sets the number format for numerical text elements.

Color tab

Use the following table as a guide for setting the *Appearance* properties on the *Color* tab for selectors. For more information about using the *Color* dialog box to set component colors, see *Setting colors for component elements*.

For many items, you can set three different colors:

- *Default Color*: applied when the item is not active.
- *Over Color*: applied when the mouse pointer is over the item.
- *Selected Color:* applied when the item is selected.

> **i Note**
>
> If alerts are enabled for an *Icon* component, the *Color* option is disabled.

Option	Description
Border Color	The color of the border around a table component.
Background Color	The color of the background area behind component elements.
Category Names	The text color on the category buttons of an *Accordion Menu*.
Category Background	The background color on the category buttons of an *Accordion Menu*.

Option	Description
Header Labels	The text color for table column headers. ℹ **Note** For hierarchical tables, this text item is called *Header Text*.
Header Background	The background color for table column headers.
Button	The color of a push button or toggle button. You can set a different color for when the button is on and when it is off.
Labels	The label text color.
Label Backgrounds	The background color for labels.
Label Header Color	The background color of the heading bar for a component with images. ℹ **Note** The label text color is set on the *Text* tab in Labels text formatting.
Drop-Down Button	The color of the button that opens or closes drop down lists. You can set different colors for the background (*Button Color*) and the drop down symbol (*Arrow Color*) when selected or not selected.
Display Values	The text color for the values in a *Ticker*.
Source List Labels	The text color of items in the source list for a *List Builder*.
Source List Backgrounds	The background color of the source list for a *List Builder*.
Destination List Labels	The text color of items in the destination list for a *List Builder*.
Destination List Backgrounds	The background color of the destination list for a *List Builder*.
Data	The text color of rows of data in a *Hierarchical Table*.
Thumbnails	The color of the thumbnail area behind images for a *Sliding Picture Menu* or a *Fisheye Picture Menu*.
Buttons	The color of buttons on the component. Depending on the component, you can set individual colors for different elements and states of buttons: • the button background for active, pressed, or disabled buttons • the symbol for active, pressed, or disabled buttons • the button label for default and pressed buttons

Option	Description
Gridlines	The color of lines between columns and rows in table components. For spreadsheet tables, this option is called *Gridline Color*.
Symbol Color	The color of the arrow symbol on a column header that shows the current sort direction. i Note The symbol is not visible in design mode.
Rows	The background color of rows in a spreadsheet. For some tables, you can also set a different color for alternating rows.
Scroll Bar	The colors for the parts of the scroll bar.
Unchecked Color	The color of a *Checkbox* when it is not selected.
Checked Color	The color of a *Checkbox* when it is selected.
Marker	The colors of the buttons of a *Radio Button*.
Progress Indicator	The color of the *Progress Bar* indicator. You can set a different color for the track and for the marker.

Related Links

4.2.5   Using single value components

Single value components are useful for displaying data from individual cells in the spreadsheet. Single value components can add user interactivity to your models, or you can also use them to focus attention on the values of important parameters.

i Note

Certain single value components have visually alternate versions of themselves, such as *Dual Slider* and *Dual Slider2*, or *Gauge* and *Gauge3*. These components function in the same manner and differ in appearance only.

Input or output

You can use any single value component as an input or output element in your model. As an input, a single value component permits user interaction. As an output, it displays the value of the cell or cells that it is linked to.

You can enable user interaction with single value components by selecting the *Enable Interaction* option on the *Behavior* view for the component.

> ℹ **Note**
>
> To access the *Enable Interaction* option, you must bind the component to a cell that does not contain a formula. If you link the component to a cell with a formula, user interaction is disabled and the component becomes a display component.

For example, if you link a *Gauge* to a cell that contains a static value, users can modify the cell value during runtime by dragging the gauge needle. If you have a *Gauge* linked to a cell that contains a formula, users cannot modify the value. The value is set by the formula in the cell.

Dial

The *Dial* can be linked to a single cell only, and is typically used as an input. Users click and drag the dial to change the value.

Horizontal Slider and Vertical Slider

The *Horizontal Slider* and *Vertical Slider* are linked to a single value, and are typically used as inputs. Users click and drag the slider to change the value.

Dual Slider

The *Dual Slider* is similar to the *Horizontal Slider*, but it represents two values: a low and high value. The low value cannot exceed the high value and vice versa. This component is typically used to represent the maximum and minimum values of a parameter.

Horizontal Progress Bar and Vertical Progress Bar

The *Horizontal Progress Bar* and *Vertical Progress Bar* are typically used as display components. As the value displayed by the progress bar increases, more and more of the progress bar area is filled in.

Gauge

The *Gauge* is typically used as a display component to reflect the value of a key metric. It can also be linked to multiple cells so that each value is represented by a marker in the *Gauge*.

If you bind a *Gauge* to more than one cell, it can act as a display component only. To allow user interaction, you must bind the *Gauge* to a single cell that does not contain a formula.

Value

The *Value* component displays a single value with a simple appearance. If you enable user interaction, users can click the component and drag up or down to change the value.

Spinner

The *Spinner* is usually used as an input component. Users can change the value by clicking the up and down arrows or by typing a value.

Play Control

The *Play Control* acts as an input component that incrementally increases the value of a cell in your spreadsheet. When a user clicks play, the value begins to increase by a predetermined increment until it reaches a maximum value. This component helps users visualize how the steady increase of a single variable affects other data.

Users can click the control buttons and also drag the slider to change the value. If you clear the *Enable Interaction* option, interaction is available through the control buttons only.

Related Links

4.2.5.1 Single value component general properties

Use the following table as a guide for setting the single value component properties on the *General* tab. Some options are available only for specific single value components.

Option	Description
Title	The title appears centered at the top of the component by default.
Data	Type a value to display it statically, or click the *Cell Selector* () to select a cell from the embedded spreadsheet. For most single value components, you must select a single cell. *Gauge* and *Dual Slider* components, however, can be linked to multiple cells.
By Range	For a *Gauge*, select the range of spreadsheet cells that the gauge displays or controls. You can link gauge components to multiple cells, but in this case it functions as a display component only. Users cannot manipulate a gauge with multiple markers during runtime. ➡ Tip The range is reflected in the *By Indicator* list, so selecting a range is a quick way to start configuring individual indicators.
By Indicators	Select each indicator in the list and set the following values: Option / Description *Name* — Marker name *Value* — Marker data *Type* — Marker appearance

Option	Description
Low Value	The default value for the left marker of a *Dual Slider*. Type a value to display it statically, or click the *Cell Selector* (⬛) to select a cell from the embedded spreadsheet.
High Value	The default value for the right marker of a *Dual Slider*. Type a value to display it statically, or click the *Cell Selector* (⬛) to select a cell from the embedded spreadsheet.
Manual	Select this option to define the *Minimum Limit* and *Maximum Limit* values for the component scale.
Minimum Limit	Type a value for the lower limit of the component scale, or click the *Cell Selector* (⬛) to select a cell from the embedded spreadsheet.
Maximum Limit	Type a value for the upper limit of the component scale, or click the *Cell Selector* (⬛) to select a cell from the embedded spreadsheet.
Auto	Select this option to have minimum and maximum scale limits calculated based on the component data. You can also choose a calculation method. The available options vary depending on the type of component: **Option** / **Description** *Value Based* — The limits form a range around the value. The size of this range depends on the size of the value itself. *Zero Based* — The upper limit remains the same as for a *Value Based* range, and the lower limit is set to zero. *Zero Centered* — The upper limit remains the same as for a *Value Based* range, and the lower limit is set as the negative of the upper limit. Thus, zero is at the center of the range. *Alert Based* — The range corresponds to the range that you configure in the *Alert Thresholds* table on the *Alerts* tab. You must set alerts *By Value* to use an alert-based scale.

Related Links

Single value component behavior properties [page 108]

Single value component appearance properties [page 111]

4.2.5.2 Single value component behavior properties

Use the following table as a guide for setting the *Behavior* properties for single value components. Some options are available only for specific single value components.

Option	Description	
Common Tab		
Progress Movement, *Slider Movement*, or *Needle Movement*	This option specifies how finely users can adjust the component value. Select one of the following options:	

Option	Description
Increment	Users can adjust the value by the specified increment. For example, if you set the increment to 10, users can set the component value to 10, 20, 30, and so on. Type the increment or click the *Cell Selector* () to select a cell from the embedded spreadsheet.
Major Ticks	Users can set values that correspond to the scale's major ticks only.
Major and Minor Ticks	Users can set values that correspond to the scale's major and minor ticks.

Option	Description
Object Elasticity	For a *Gauge*, this option determines how much the needle shakes back and forth when users move it to a new value. If you disable elasticity, the needle snaps to the new value immediately. If you add more elasticity, the needle settles into place with a dynamic animated feel.
Increment	This option specifies how much the component value changes every time it is adjusted. Type the increment or click the *Cell Selector* () to select a cell from the embedded spreadsheet.
Limits	This option specifies how limits are restricted when the model runs. For each of the *Minimum* and *Maximum* limits, select from the following options:

Option	Description
Fixed	The limit is fixed and users cannot set a value beyond the limit.
Adjustable	Users can adjust the limit at runtime.
Open	The limit is removed.

Option	Description
Enable Interaction	If this option is selected, users can change the value of the component. If you do not want the component to accept input from users at runtime, clear this option. **i Note** For a *Play Control*, users can click the control buttons whether *Enable Interaction* is selected or cleared.

Option	Description
Auto Resize	If this option is selected, the component automatically expands or contracts to fit the size of the entered value.
Mouse Sensitivity	This option determines how sensitive the component is to mouse movement. When the setting is high, the value changes by large increments as the mouse moves. When the setting is low, the value changes slowly as the mouse moves.
Mouse Tracking	For a *Gauge* or *Dial*, this option specifies how the user must drag the needle to change the value of the component. Select one of the following options: • *Vertical*: the user moves the mouse up and down to adjust the value. • *Radial*: the user moves the mouse pointer in a circular motion to adjust the value. **i Note** When the user starts to adjust the needle, the cursor changes to reflect this setting. For *Vertical* tracking, a vertical arrow is displayed, and for *Radial* tracking, a circular arrow is displayed.
Scroll Behavior	This option determines how users change the value. Select one of the following options: • *Manual*: the value corresponds directly to how high or low the user drags the mouse. The user must drag the mouse higher or lower to adjust the value. • *Auto*: the user drags the mouse above or below the component, and while the mouse button is pressed, the value continues to change without requiring further movement.
Enable Play Button	When this option is selected, a play button is available for the component. When the user clicks the *Play* button, the component cycles through the values from minimum to maximum in the increments set for the component. **i Note** For the *Horizontal Progress Bar* and *Vertical Progress Bar* components, if *Enable Play Button* is enabled, the component's progress bar cannot be interacted with while the component is cycling through values.
Auto Play	If this option is selected, the play sequence for the component automatically plays when the model is loaded.
Play Time	This option allows you to set the duration of the play sequence. Type the number of seconds from 1 to 100 in the box.
Auto Rewind	If this option is selected, when the play sequence reaches the maximum value, it automatically rewinds to the minimum value.
Auto Replay	If this option is selected, the play sequence automatically repeats.
Dynamic Visibility	This option allows you to specify conditions that control when the selector appears in the model.

Option	Description
	For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].
Animations and Effects Tab	
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.
Enable Sound	You can add a sound effect to some components. The sound effect is triggered when users interact with the component.

Related Links

Single value component general properties [page 107]
Single value component appearance properties [page 111]

4.2.5.3 Single value component appearance properties

Layout tab

Use the following table as a guide for setting the *Appearance* properties on the *Layout* tab for single value components. Some options are available only for specific single value components.

Option	Description
Enable Ticks	If this option is selected, small markers that indicate values (ticks) are visible on the component.
Auto Scale	This option automatically configures the number of ticks according to the size of the scale.
Manual	This option allows you to customize the appearance of the ticks. Select one of the following options: • *Number of Ticks*: allows you to specify how many major and minor ticks are included on the component. • *Size of Divisions*: allows you to set the increments between ticks. You can set the value for both *Major* and *Minor* ticks.
Radial Definition	For a *Gauge* or *Dial*, this option allows you to set the position of the minimum and maximum values of the scale. For both *Minimum Angle* and *Maximum Angle*, type the angle amount the needle can rotate to.

Option	Description
	> **i Note** If the *Minimum Limit* is set to *Open* on the *Behavior* view, you cannot set the *Minimum Angle* value. If the *Maximum Limit* is set to *Open*, you cannot set the *Maximum Angle*.
Show Background	For a *Value* component, select this option to display a background. Otherwise, the component is transparent.
Layout Options	This option determines which elements appear on a *Play Control* component. Select the elements you want to include: • *Rewind/Fast Forward* buttons • *Previous/Next* buttons • *Slider* (progress bar)

Text **tab**

Use the following table as a guide for setting the *Appearance* properties on the *Text* tab for single value components.

In the *Show* column, select the checkbox for each text item that you want to display. Then, select the text item and format the text appearance. For more information about formatting text, see *Formatting text for component elements*. The available options vary depending on the type of component.

Option	Description
Title	The title above the component.
Value	The current value. For the *Spinner* and *Value* components, you cannot clear this option.
Limits	The upper and lower limits for the component.
Format Selected Text	The settings in this area allow you to format the selected text element. > **i Note** If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.
Position	Sets the position for a text element relative to the component.
Offset	Adjusts the position of a text element by offsetting it relative to their set position.
Number Format	Sets the number format for numerical text elements.

Color tab

Use the following table as a guide for setting the *Appearance* properties on the *Color* tab for single value components. The available options vary depending on the type of component.

To set the color for a component element, click the *Color Selector* ( ) and from the *Color* dialog box, select the color you want to use.

Option	Description
Indicator table	The color of each marker for a gauge.
Needle Color	The color of the marker for a dial.
Frame Color	The color of the frame around the outside of a *Gauge*.
Grip Color	The color of the edge of the *Dial* face.
Center Color	The color of the center of a *Gauge* where the needles are mounted.
Background Color	The color of the background area behind component elements.
Play Button Color	The color of the *Play Button*.
Play Symbol Color	The color of the arrow symbol that appears on the *Play Button*.
Marker Color	The color of the marker on a *Horizontal Slider*, *Vertical Slider*, or *Play Control* bar. For progress bars, the *Marker Color* sets color for the part of the bar that is filled in.
Low Marker Color	The color of the left marker on a *Dual Slider*.
High Marker Color	The color of the right marker on a *Dual Slider*.
Track Color	The color of the track area on a *Slider* or *Progress Bar*.
Tick Color	The color of the tick markers on the component.
Limits Background Color	The color behind the text of adjustable limits for the component.
Up Button and *Down Button*	The colors of the buttons that raise or lower the value of a *Spinner*. You can set several different colors for each button: • *Button Color*: the background color of the button. • *Symbol Color*: the color of the up arrow symbol on the button. • *Pressed Button Color*: the background color of the button when a user clicks it. • *Pressed Symbol Color*: the color of the up arrow symbol on the button when a user clicks it.

Related Links

Single value component general properties [page 107]
Single value component behavior properties [page 108]

113

4.2.6   Using map components

Map components create models with geographical representations that can display data by region.

In addition to displaying data for each region, map components can be configured so that each region acts as a selector. By combining these two features, you can create a model that shows data for each region when users mouse over the region, and also allows users to click each region to explore more detailed information.

How does it work?

The software associates data with each region in the map by using region keys. The default region keys for each map are the full names of the regions. You can also enter your own region keys. When the map is bound to a range of data, the component searches either the first column or first row of that range for the region keys. The data in the row or column corresponding to the key is associated with the region.

You can edit the region keys associated with each region by clicking *Manual Edit* ( ) to type the new keys, or by clicking the *Cell Selector* ( ) beside the *Region Key* box to select the row or column containing the new keys.

> **i Note**
>
> In the spreadsheet you must enter the region keys and data in adjacent cells for both *Display Data* and *Source Data*.

> **➡ Tip**
>
> The software provides a spreadsheet that lists the default region keys for each map component. The file is located in the following location: `C:\Program Files\SAP BusinessObjects\Xcelsius 4.0\assets \samples\User Guide Samples\MapRegions.xls`

Related Links
To configure component properties [page 45]

4.2.6.1   Map general properties

Use the following table as a guide for setting the properties on the *General* view for a map component.

Option	Description
Title	The title is centered at the top of the component by default.
Region Keys	The region keys define the codes used for each region in the map component. These codes must be included in the embedded spreadsheet in the range selected for *Display Data* and *Source Data*. Each map has default region keys set up, which correspond

Option	Description
	to the full names of the regions. You can use the default values or customize the codes to correspond to your data. • To see and modify the region keys, click the *Manual Edit* button. • To assign codes from cells in the embedded spreadsheet, click the *Cell Selector* button.
Display Data	The *Display Data* is associated with a specific map region and appears on the component when the user clicks or mouses over the region. Click the *Cell Selector* ( ) to select the range from the embedded spreadsheet. When you select the data range, you must include the column or row that has the region keys. When the software searches for the region keys, it determines the number of rows and columns in the table and follows a set procedure to find the cells that contain the keys: • if the range has 2 columns and more than 2 rows, it looks for the keys in the first column. • If the range has 2 rows and more than 2 columns, it looks for the keys in the first row. • If the range has 2 rows and 2 columns, it looks in the first column and first row for the greatest number of keys.
Insertion Type	This option determines what type of data is inserted when the user clicks a region. You can choose to insert data from a *Row* or from a *Column*.
Source Data	This option specifies the data that is inserted into the table when a region is clicked. When you select the data range, you must include the column or row that has the region codes. The software uses the region code to associate the data with specific regions. Click the *Cell Selector* ( ) to select the range from the embedded spreadsheet.
Destination	This option specifies where the source data is inserted in the spreadsheet. The selected cells must match the *Insertion Type* and size of the source data. ⓘ Note The destination cells must be empty.

Related Links

To configure component properties [page 45]

Map behavior properties [page 116]

Map appearance properties [page 116]

4.2.6.2   Map behavior properties

Use the following table as a guide for setting the *Behavior* properties for maps.

Option	Description
Common Tab	
Insert On	Specifies how the user selects a region. Select either *Mouse Click* or *Mouse Over*.
Region	Specifies which region is selected when the model first opens, before the user makes a selection.
Dynamic Visibility	This option allows you to specify conditions that control when the map appears in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].
Animations and Effects Tab	
Enable Sound	You can add a sound effect to map components. The sound effect is triggered when users select a region.
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.

Related Links

To configure component properties [page 45]
Map general properties [page 114]
Map appearance properties [page 116]

4.2.6.3   Map appearance properties

Layout **tab**

Use the following table as a guide for setting the *Appearance* properties on the *Layout* tab for map components.

Option	Description
Transparency	Specifies how clearly objects behind the map can be seen. Set the slider to a value between 0 and 100 where 0 is not transparent at all and 100 is invisible.

Text **tab**

Use the following table as a guide for setting the *Appearance* properties on the *Text* tab for maps.

In the *Show* column, select the checkbox for each text item that you want to display. Then, select the text item and format the text appearance. For more information about formatting text, see *Formatting text for component elements*.

Option	Description
Title	The title for the map.
Mouse-Over Labels	The label of a region (that is, the region key) is displayed when users mouse over the region.
Mouse-Over Value	The display data for a region is displayed when users mouse over the region. If you select this option and *Mouse-Over Labels*, the label and value appear beside each other.
Format Selected Text	The settings in this area allow you to format the text for the selected text element. The same format applies to both *Mouse-Over Labels* and *Mouse-Over Values*. **i Note** If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.
Position	For the *Title*, this option sets the position of the text relative to the component.
Offset	For the *Title*, this option allows you to adjust the position of the text by offsetting it relative to its set position.
Number Format	For the *Mouse-Over Values*, this option allows you to set the number format.

Color tab

Use the following table as a guide for setting the *Appearance* properties on the *Color* tab for maps.

To set the color for a component element, click the *Color Selector* ( ) and from the *Color* dialog box, select the color you want to use.

Option	Description
Default Color	The color of regions that are not associated with source data in the spreadsheet and cannot be selected.
Over Color	The color of a selectable region when a user mouses over it.
Selected Color	The color of the selected region.
Selectable Color	The color of regions that are associated with source data in the spreadsheet and can be selected.

Option	Description
Border Color	The line color of borders in the map.

Related Links

To configure component properties [page 45]

Map general properties [page 114]

Map behavior properties [page 116]

4.2.7 Using text components

This category contains components for displaying labels and allowing users to type text into your model.

Text components let you customize your model. Not only can you insert static text to identify areas and elements of your model, but you can also link the text components to cells in the embedded spreadsheet to dynamically change the text displayed in the model based on the cell value or formula result. In addition, formulas can contain HTML code to increase formatting options.

The software includes the following text components:

Input Text Area

The *Input Text Area* can display multiple lines of text that the user can edit during runtime.

Label
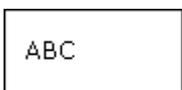
The *Label* component lets you add a box of text to the model. For example, you can use label components to create titles, subtitles, explanations, help, and more. Text in the *Label* cannot be changed by the user.

Input text

The *Input Text Component* can display a line of text that the user can edit during runtime. This component is typically used to insert text into a spreadsheet cell, while the *Label* is typically used to display text.

4.2.7.1   Text general properties

Use the following table as a guide for setting properties for text components on the *General* view. The available options vary depending on the type of component.

Option	Description
Link to Cell	Select this option to display text from a cell in the embedded spreadsheet. Changes to the data in the cell are reflected in the component. You can use this option to set dynamic labels by defining formulas that update based on data or user selections in the model.
Use Spreadsheet Formatting	For a *Label* linked to a cell, you can select this option to apply the formatting from the embedded spreadsheet to the text in the model.
Enter Text	Select this option to manually type text to display in the component.
Enable HTML Formatting	Select this option to enable the use of HTML codes for text formatting for a *Label* or *Input Text Area*. Several HTML codes are supported: • Anchor tag: `<a>` • Bold tag: `<b>` • Break tag: ` ` • Font tag: `<font>` • Image tag: `<img>` • Italic tag: `<i>` • List item tag: `<li>` • Paragraph tag: `<p>` • Text format tag: `<textformat>` • Underline tag: `<u>` **ℹ Note** If you intend to export your model to PowerPoint and include a link in the text with HTML formatting, you must include `target="_blank"` in your link tag so PowerPoint opens the link in a new window. For example, `<a href="http://www.google.com"target="_blank">`. If you do not include the target setting, users can open the link in PowerPoint only if they right-click the link and select *Open in new window*.
Insert Data on Load	Select this checkbox to insert the default text into the destination cell when the model first opens during runtime.
Destination	This option specifies where the text is inserted in the spreadsheet.

Option	Description
	i Note The destination cells must be empty.

Related Links

Text behavior properties [page 120]
Text appearance properties [page 122]

4.2.7.2 Text behavior properties

Use the following table as a guide for setting the *Behavior* properties for text components. The available options vary depending on the type of component.

Option	Description
Common Tab	
Treat All Input As Text	If this option is selected, the input is treated as text strings. Text is not available for use in mathematical formulas or calculations. If this option is not selected, numerical input for the component can be used for calculations or formulas.
Enable Password Protection	Select this option for an *Input Text* component where users type a password. Typed characters are displayed as asterisks (*).
Maximum Number of Characters	Use this option to limit the number of characters that users can enter.
Characters Allowed	This option allows you to control the set of characters that users can type in an *Input Text* component. By default, all characters are allowed. For more information about defining allowed characters, see *Restricting characters in text-based components*.
Horizontal Scroll Bar	If the text exceeds the width of the text area, a scroll bar allows users to scroll through the text. Select one of the following options: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*On*</td><td>The horizontal scroll bar is always visible.</td></tr><tr><td>*Off*</td><td>The horizontal scroll bar is never available.</td></tr><tr><td>*Auto*</td><td>The horizontal scroll bar appears only when the text exceeds the width of the text area.</td></tr></table>

Option	Description
Vertical Scroll Bar	If the text exceeds the height of the text area, a scroll bar allows users to move through the text. Select one of the following options: 	Option	Description	 |---|---| 	*On*	The vertical scroll bar is always visible.	 	*Off*	The vertical scroll bar is never available.	 	*Auto*	The vertical scroll bar appears only when the text exceeds the height of the text area.	
Dynamic Visibility	This option allows you to specify conditions that control when the text component appears in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].
Animations and Effects Tab	
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.

Related Links

Text general properties [page 119]
Text appearance properties [page 122]

4.2.7.2.1   Restricting characters in text-based components

By default, input text components allow users to enter any character. You can prevent users from entering certain characters in an input text component if necessary. For instance, if users need to type a name, you can disable numerical characters, punctuation, and special characters.

To limit the allowed characters:

1. On the *Properties* panel, click ▌▶ *Behavior* ▶ *Common* ◣.
2. In the *Characters Allowed* box, type the allowed and not allowed characters or click the *Cell Selector* button

   (   ) to select a cell in the spreadsheet that contains the values.
   When defining the set of allowed characters, use the following conventions:

Effect	Convention
Allow specific characters	Type the characters. For example: **abcABC**
Disallow specific characters	Precede the characters with a carat (^). For example: **^abcABC**

Effect	Convention
Allow a range of characters	Separate the first and last character with a dash (-). For example: `a-zA-Z`
Switch between allowed and disallowed characters	Surround the disallowed characters with carats (^), for example, `abc^def^gh` indicates that "abcgh" are allowed and "def" are not allowed.
Allow or disallow a carat (^), dash (-), or backslash (\) character	Precede these special characters with a backslash (\). For example, `^\^\-\\^` disables these three characters.
Use common character sets	Some common characters sets include: ○ Whole numbers including negative numbers: `\-0-9` ○ Real numbers including negative numbers: `\-0-9.` ○ Fractions: `\-/0-9` ○ Positive whole numbers: `0-9` ○ Positive real numbers: `0-9.` ○ Letters: `a-zA-Z` ○ Numbers and Letters: `0-9a-zA-Z`

Related Links

Text behavior properties [page 120]

4.2.7.3 Text appearance properties

Use the following table as a guide for setting the *Appearance* properties for text components. The available options vary depending on the type of component.

Option	Description
Layout Tab	
Wrap Text	If selected, lines of text that are longer than the width of the text area wrap to the next line.
Transparency	Specifies whether objects behind the component are visible. A transparency of 0 means that users cannot see objects behind the component. A transparency of 100 means that the component is completely transparent and invisible to users.
Draw Border On	Select the side or sides of a label to be bordered. You can choose *Top*, *Bottom*, *Left*, *Right*, or *All Sides*. You must also select *Show Border*, and choose a color and a *Border Thickness*.
Show Fill	If selected, the label has a solid background. You can choose the background color.
Text Tab	

Option	Description
Format Selected Text	The settings in this area allow you to format the text for the selected text element. For more information about formatting text, see *Formatting text for component elements*. i **Note** If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.
Number Format	Specify the formatting style for numbers.
Color Tab	
Background Color	The color of the background area behind the text.
Scroll Bar	The colors for parts of the scroll bar.
Fill Color	The background area behind the text of a label. i **Note** This option is available if *Show Fill* is selected on the *Layout* tab.

Related Links

Text general properties [page 119]
Text behavior properties [page 120]

4.2.8   Using other components

The *Other* category contains an assortment of components that you can use to enhance your models. Several components are included in this category:

Calendar

The *Calendar* is a selector that lets you incorporate date selection into your models.

> ℹ **Note**
>
> This component is not available in Presentation Design.

Local Scenario Button

The *Local Scenario Button* lets users save and load runtime configurations of a model on their local machine. Once saved, these configurations, known as scenarios, can be loaded again, even after closing the model. The user also has the option to set a saved scenario as the default scenario so it loads when the model opens. These scenarios are saved to the local machine, and are unavailable if the model is opened on a different machine.

Trend Icon

The *Trend Icon* shows an arrow that can indicate whether values are trending in a positive or negative direction. The icon is linked to a single cell.

- If the cell value is positive, the arrow points up.
- If the cell value is zero, the symbol is neutral.
- If the cell value is negative, the arrow points down.

History

The *History* component tracks changes in the value of a single cell and inserts the different values across a range of destination cells. This component helps visualize the effect of changes in worksheet data.

> ℹ **Note**
>
> This component is not available in Presentation Design.

> ℹ **Note**

This component is not visible at runtime.

Panel Set
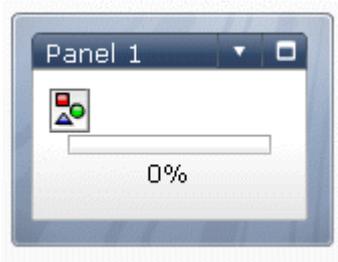
The *Panel Set* is a series of framing options that enables users to easily navigate between files in a presentation. You can embed JPEG or SWF files into the frames of the component and adjust a variety of formatting features to customize the look of the component in your model.

> **i** Note
>
> You cannot embed models saved as SWF files in this component.

> **i** Note
>
> This component is not available in Presentation Design.

Source Data

The *Source Data* component pushes data to destination cells when a *Selected Item Index* associated with the component changes. For example, this component can allow users to cycle through several rows of data by changing a single value selector linked to the *Selected Item Index*, such as a *Dial*.

> **i** Note
>
> This component is not available in Presentation Design.

> **i** Note
>
> This component is not visible at runtime.

Trend Analyzer

The *Trend Analyzer* combines data analysis and forecasting. The component analyzes spreadsheet data and inserts data points for a trend line into the spreadsheet, which can then be added to your model. You can also configure this component to forecast based on existing values.

> ℹ Note
>
> This component is not available in Presentation Design.

> ℹ Note
>
> This component is not visible at runtime.

Print Button

The *Print Button* allows users to easily print the model while it is running. The user can click the print button to print the SWF file.

Reset button

The *Reset Button* returns the model to its original state. Users can interact with the components and then click the button to clear their changes.

Grid

The *Grid* is a dynamic table that displays data from the spreadsheet. Users can enter new values into the component at runtime. *Grids* do not contain header rows or column labels, and are most useful when linked to numerical data only.

The *Grid* differs from the *Spreadsheet* and *List View* because the user can manually edit the values in the cells during runtime.

> i Note
>
> This component is not available in Presentation Design.

Related Links

To configure component properties [page 45]

4.2.8.1 Other component general properties

Use the following table as a guide for setting the properties on the *General* view for other components. The available options vary depending on the type of component.

Option	Description
Title	Type the title text or click the *Cell Selector* ( ) to select the title text from the embedded spreadsheet. The title is centered at the top of the component by default.
Label	Specify the text that appears on the button.
	Type the text in the box or click the *Cell Selector* ( ) to select the label text from the embedded spreadsheet.
Panel Titles	These titles appear on each frame of a *Panel Set*.
	Choose a *Layout* before configuring panel titles. Click the *Cell Selector* ( ) to select the title text from the embedded spreadsheet or click the *Manual Edit* button to type the title for each panel.
Layout	For a *Panel Set*, select the arrangement of panels from the list.
Panels	For a *Panel Set*, the number of panels in the list corresponds to the selected layout. Select each panel and set the content options.
Content Type	For a *Panel Set*, this option specifies the type of information provided in the selected panel. Select one of the following options:

Under *Content Type*:

Option	Description
None	The panel is blank.
Embedded JPEG or SWF	The image files for the panel are saved with the model. Click the *Import* button and browse to the location of the file.
By URL	The image files for the panel are loaded from an external URL location at runtime. Click the

Option	Description	
	Option	**Description**
		Cell Selector (⊞) to select the URL from the embedded spreadsheet or click the *Manual Edit* (⊞) to type the URL manually, and select an *Application Domain*.

Option	Description
Drop-Down Menu Labels	For a *Panel Set*, this option specifies the text that appears in the drop-down lists for each panel. If a panel contains several images, users can select the image to display using this list.

Select a panel in the *Panels* list and click the *Cell Selector* (⊞) button to select the label text from the embedded spreadsheet or click the *Manual Edit* (⊞) to type the label text.

If you create more labels than images for a panel, the extra labels do not appear in the drop-down list. |
| *Insertion Type* | This option specifies the type of information that is inserted when a selection is made. Select one of the following options:

For *Calendar* components:

- *Day*: inserts the number of the day of the month. For example, April 7, 2011 is inserted as "7".
- *Date*: inserts the number of the day, counting up from January 1, 1900. For example, April 7, 2011 is inserted as "40640".

For *Source Data* components:

- *Value*: inserts the value from a single cell.
- *Row*: inserts the values in a row of data.
- *Column*: inserts the values in a column of data. |
| *Source Data* | Specifies the location of the source data for selections. Click the *Cell Selector* (⊞) to select the range from the spreadsheet. |
| *Destination* | Specifies the location in the spreadsheet the component inserts data. Click the *Cell Selector* (⊞) to select the range in the spreadsheet.

> **ℹ Note**
>
> The required size of the destination range varies according to the structure selected in *Insertion Type*. If the component inserts columns or rows of data, select a range that is large enough to contain the largest column or row of data. |
| *Month Destination* | The location in the spreadsheet to insert the month value for the selected date. |
| *Year Destination* | The location in the spreadsheet to insert the year value for the selected data. |

Option	Description
Data	Specifies the source data that the component uses. Click the *Cell Selector* ( ) to select the range in the spreadsheet. For the *Grid* component, a value is displayed for each cell selected and values are arranged in the same order as in the spreadsheet. Changes to the spreadsheet values are reflected in the grid, and users can also change the spreadsheet values by editing the grid during runtime. For the *Trend Icon*, if the value in this cell is greater than 0, the component displays an up arrow. If the value is less than 0, it displays a down arrow, and if it equals 0, it displays a neutral symbol.
Minimum Limit and *Maximum Limit*	These options specify the upper and lower limit to the values that users can enter in the *Grid*. Type the limit in the box or click the *Cell Selector* ( ) to select the limit amount from the embedded spreadsheet.
Colors	For a *Trend Icon*, this option specifies the colors that represent different trends. Select a color for each option: | Option	Description
Positive Value Color	Icon color when the data value is greater than 0.
Zero Value Color	Icon color when the data value is equal to 0.
Negative Value Color	Icon color when the data value is less than 0.	
Data Destination	For a *History* component, this option specifies the location in the embedded spreadsheet where data is inserted. The number of cells in this range determines the number of historical values that are stored. When each cell in the range has a value, the oldest value gets overwritten each time a new historical value is inserted. For a range with a single column, new values are always inserted in the bottom cell, and the data is bumped up by a row each time a new value is inserted. A range consisting of a single row behaves similarly. New values are inserted in the rightmost cell. If the range has multiple rows and multiple columns, newer data is inserted in the bottom right cell. The data is bumped to the right until the bottom row is full, and then the next row is filled right-to-left, and so on.
When Value Changes	For a *History* component, select this option to record the cell history each time the value in the cell changes.
On Interval	For a *History* component, select this option to record the cell history at timed intervals. In the *Duration (seconds)* box, specify the interval duration.

4.2.8.2 Other component behavior properties

Use the following table as a guide for setting the *Behavior* properties for other components. The available options vary depending on the type of component.

Option	Description
Common Tab	
Default Date	For a *Calendar*, this option specifies the date that is first selected when the model loads. Select one of the following options: • *Use current date*: the calendar loads with the current date. • *Use custom date*: the calendar loads with a specific date that you choose. Enter values for *Day*, *Month*, and *Year*.
Enable Calendar Limits	For a *Calendar*, this option limits the date range available. If you select this option, enter values for *Start Month*, *Start Year*, *End Month*, and *End Year* to define the range of dates that the user can select.
Increment	For a *Grid* with *Enable Interaction* selected, you can restrict the numbers that users can enter during runtime. Users can only enter multiples of the *Increment*. For instance, if you type 12 as the *Increment*, users can set values in the grid to 0, 12, 24, 36, and so on. The *Increment* setting does not affect the initial data that appears in the grid when it loads.
Minimum Limit	For a *Grid* with *Enable Interaction* selected, you can set limits on the numbers that users can enter during runtime. This setting is based on the *Minimum Limit* on the *General* view. Select one of the following options: • *Open*: users can enter values that are less than the *Minimum Limit*. • *Fixed*: users cannot enter values that are less than the *Minimum Limit*.
Maximum Limit	For a *Grid* with *Enable Interaction* selected, you can set limits on the numbers that users can enter during runtime. This setting is based on the *Maximum Limit* on the *General* view. Select one of the following options: • *Open*: users can enter values that are greater than the *Maximum Limit*. • *Fixed*: users cannot enter values that are greater than the *Maximum Limit*.
Enable Interaction	Select this option to allow users to enter values in the *Grid* during runtime. ⓘ Note If a grid cell is linked to a spreadsheet cell that contains a formula, users cannot change the value of that cell even when *Enable Interaction* is selected.
Mouse Sensitivity	For a *Grid* with *Enable Interaction* selected, you can determine how sensitive the component is to click-and-drag mouse movements. When the setting is high, mouse movements change the value rapidly. When the setting is low, mouse movements change the value by small increments.

Option	Description
Scroll Behavior	For a *Grid* with *Enable Interaction* selected, this option determines the how the cell value changes when the user clicks and drags it. Select one of the following options: • *Manual*: the user clicks the cell and drags up or down to change the value. For each increment of change, the user must continue to drag the pointer. • *Auto*: the user clicks the cell and drags up or down to change the value. While the mouse button is pressed, the value continues to change without requiring further pointer movement.
Enable Maximize Button	For a *Panel Set*, you can select this option to enable a maximize button on each panel during runtime. This button allows users to expand a panel to fill the entire panel set. Users can click it again to restore the panel to its original size. This feature is useful for emphasizing different sets of data at different times during a presentation.
Zoom Speed	For a *Panel Set* with *Enable Maximize Button* selected, this option specifies how fast panels expand and contract when users click the minimize or restore button. Move the marker along the slider to set the speed.
Panel	For a *Panel Set*, you can set the image that appears in each panel when the model loads. Select a panel in the *Panel* list and select the image to display by default for the panel in the *Item* list.
Ignore Blank Cells	For a *Panel Set*, select *In Values* to prevent empty values past the last non-empty value from being displayed. This option allows a variable amount of data to be displayed.
Print Scale	For a *Print Scale*, this option specifies how the model scales when printing. Select one of the following options: • *Scale To*: scales to a specific percentage of the original dashboard size. Select the percentage from the list. • *Scale to Fit Page*: scales to fit on one page of the selected paper.
Selected Item Index	This option specifies the index for the *Source Data* component. Click the *Cell Selector* button ( ) to select a cell in the spreadsheet. When the value in this cell changes, the source data component inserts a new value from the source data to the destination range. For instance, if the *Selected Item Index* is 0, the source data component inserts the first cell, row, or column of source data to the destination range. If the *Selected Item Index* value changes to 1, the second cell, row, or column of data is inserted, overwriting the previous data.
Dynamic Visibility	This option allows you to specify conditions that control when the component appears in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].

Option	Description
Animations and Effects Tab	
Enable Sound	You can add a sound effect to some components. The sound effect is triggered when users make a selection.
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.

Related Links

Other component general properties [page 127]
Other component appearance properties [page 132]

4.2.8.3 Other component appearance properties

Layout tab

Use the following table as a guide for setting the *Appearance* properties on the *Layout* tab for other components. The available options vary depending on the type of component.

Option	Description
Show Button Background	Specifies whether the background behind a button component is visible. If you deselect this option, only the label text for the button appears.
Transparency	Specifies the visibility of the button. At 100% transparency, the button is completely invisible. Adjust the marker on the slider bar to change the transparency.
Show Panel Headers	For a *Panel Set*, select this option to display a header above each panel that shows the panel title, item list, and maximize or restore buttons.
Number of Labels Displayed	For a *Panel Set*, this setting specifies the number of items displayed in the drop-down list for each panel.
Background Transparency	For a *Panel Set*, this setting determines the visibility of the border area around the panels. Adjust the marker on the slider bar to change the transparency.
Vertical Margin	This setting changes the amount of space between each row in a *Grid*.
Horizontal Margin	This setting changes the amount of space between each column in a *Grid*.

Text tab

Use the following table as a guide for setting the *Appearance* properties on the *Text* tab for other components. The available options vary depending on the type of component.

Option	Description
Show	In the table that lists the text elements of the component, select or deselect the *Show* checkbox for each item to show or hide the text.
Format Selected Text	To format a text element, select it in the table and adjust the settings in the *Format Selected Text* area. For more information about formatting text, see *Formatting text for component elements*. **i** Note If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.

Color tab

Use the following table as a guide for setting the *Appearance* properties on the *Color* tab for other components. The available options vary depending on the type of component.

To set the color for a component element, click the *Color Selector* ( ) and from the *Color* dialog box, select the color you want to use.

Option	Description
Background Color	The color of the background area behind component elements.
Month Color	The color of the heading bar that contains the month name.
Day of the Week Color	The color of the heading bar that contains the days of the week.
Panel Background Color	The color of the background area on each panel.
Label Header Color	The color of the panel headers.
Labels	The text color for items in the drop-down lists of a *Panel Set*. You can set a different color for when the item is not active (*Default Color*), when the mouse pointer is over the item (*Over Color*), and when the item is selected (*Selected Color*).
Label Background	The background color for items in the drop-down lists of a *Panel Set*.

Option	Description
	You can set a different color for when the item is not active (*Default Color*), when the mouse pointer is over the item (*Over Color*), and when the item is selected (*Selected Color*).
Date Text	The text color for the dates in the month of a *Calendar*.
	You can set a different color for when the date is not active (*Default Color*), when the mouse pointer is over the date (*Over Color*), and when the date is selected (*Selected Color*).
Date Background	The background color for dates in a *Calendar*.
	You can set a different color for when the date is not active (*Default Color*), when the mouse pointer is over the date (*Over Color*), and when the date is selected (*Selected Color*).
Buttons	The colors of the buttons on a *Calendar* or *Panel Set* component. Depending on the component, you can set a different color for the following elements: • the button background for active, pressed, or disabled buttons • the button symbol for active, pressed, or disabled buttons • the button label for default and pressed buttons
Button	The color of a *Reset Button*, *Print Button*, or *Load Scenario Button*.
	You can set a different color for when the button is pressed and when it is not.
Labels	The text color of a *Reset Button*, *Print Button*, or *Load Scenario Button*.
	You can set a different color for when the button is pressed and when it is not.
Scroll Bar	The color for the parts of the scroll bar of a *Panel Set*.

Related Links

Other component general properties [page 127]

Other component behavior properties [page 130]

4.2.8.4 To configure trend analyzer components

Add source data to the embedded spreadsheet before configuring a *Trend Analyzer* component.

Use the *Trend Analyzer* component to analyze spreadsheet data and identify a mathematical trend in the data. You can also configure the *Trend Analyzer* to forecast future trends based on the existing data. The resulting trend and forecast data can then be added to a component in your model.

1. In the components pane, expand the *Other* folder, click *Trend Analyzer*, and then click the canvas to add the component to your model.
2. Select the *Trend Analyzer* and open the *Properties* panel.
3. On the *General* view, set the following options:

Option	Description
Data	The source data that the component analyzes. Click the *Cell Selector* ( ) to select a range in the spreadsheet that consists of a single row or column.
Trend/ Regression Type	This option specifies the pattern that best matches the data. Select an appropriate trend type from the list or select *Use Best Fit* to have the software pick the curve type. If you select *Polynomial*, in the *Order (Polynomial only)* list, select the desired order of the polynomial equation, from 2 to 6.
Analyzed Data Destination	Specifies the location in the spreadsheet to insert the analyzed data. Click the *Cell Selector* ( ) and select a single row or column. The analyzed data is made up of data points that match the best fit equation for the source data. To forecast subsequent values, you can select a larger range of cells as the destination.
Number of Forecast Periods	If the number of cells in the *Analyzed Data Destination* range is greater than the number of cells in the *Data* range, specify the number of periods you want to forecast. Type a number in the box or click the *Cell Selector* ( ) to bind this value to a cell in the spreadsheet. ➡ Tip By binding this value to a cell, you can let users control it with another component that specifies at runtime the number of periods to forecast.
Equation Type Destination	To see this option, click *More Options*. Specifies a location to insert the type of equation used to analyze the data.
Equation Destination	To see this option, click *More Options*. Specifies a location to insert the equation used to analyze the data.
R2 Value Destination	To see this option, click *More Options*. Specifies a location to insert the R-squared value for the analysis. ℹ Note R-squared is a value between 0 and 1 that shows the strength of trend. The more closely the source data matches the equation that the trend analyzer picks, the closer R-squared is to 1.
F Value Destination	To see this option, click *More Options*. Specifies a location to insert the F value for the analysis.

4. Click the *Behavior* view to set dynamic visibility options. For more information about dynamic visibility, see *Managing the Visibility of Components*.

4.2.9   Using art and background components

Art and background components provide visual enhancements for models. You can use them to enhance and customize your dashboards. The background components can also be used to import images or SWF files into your models.

The software includes the following art and background components:

Background

Background components are prebuilt artwork that you can add to your models. You can use them to create visual divisions or groups of related components. Several different types of backgrounds are available, but they all function in the same way. Unlike container components, backgrounds cannot contain or hold other components. They simply provide a visual background.

Ellipse

The *Ellipse* allows you to create visual divisions or outlines to define or contain sections or components.

Image
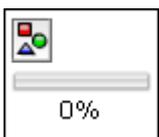
The *Image* component is used to display JPEG images or SWF files. The image component allows you to add your own logos or artwork to models. You can also import Flash files into this component to add videos, animations, and other interactive elements.

Horizontal Line and Vertical Line

The *Horizontal Line* and *Vertical Line* components allow you to create visual divisions or outlines to define or contain sections or components.

Rectangle

The *Rectangle* allows you to create visual divisions or outlines to define or contain sections or components.

4.2.9.1 To configure Background components

Use the *Background* component to change the appearance of your model. You can apply different backgrounds to different areas to visually group items.

> ➜ **Tip**
>
> You can also use the *Background* component to control user ability to interact with other components. To do that, place the *Background* component over the other item. Set the *Background* component to 100% transparency and enable *Block Mouse Events*. At runtime, users can see the underlying component, but they cannot click it or change its values. For more information, see *Disabling User Input*.

1. Select the *Background* component and open the *Properties* panel.
2. On the *General* view, set the following options:

Option	Description
Background Color	Click the *Color Selector* ( ) to select a color for the background. For more information about setting colors, see *Setting colors for component elements*.
Transparency	To allow components behind the background to be visible to users, move the slider to the right to increase the transparency of the component. To hide items behind the background, move the slider to the left.
Border Scale	Sets the thickness of the border around the background. Move the slider to the right to increase the border thickness.
Block Mouse Events	If the *Background* is placed over other components, selecting this option prevents users from using the mouse to interact with the underlying components. For more information, see *Disabling User Input*.

3. Click the *Behavior* view to set dynamic visibility options. For more information about dynamic visibility, see *Managing the Visibility of Components*.

4.2.9.2 To configure Image components

If you want to add images, videos, animations, or other interactive elements, you can import them into the *Image* component. This allows you to add your own logos and artwork to models. You can even insert another exported dashboard model.

> ℹ **Note**

The *Image* component supports standard RGB JPG file formats and SWF files. It does not support progressive JPGs or JPGs with CMYK color profiles.

When you add an image to a model, you can choose to embed the file in the SWF file or reference the external file. Files that are not embedded are saved to a folder with the same name and location as the exported model and are loaded as needed. For the model and external files to load correctly, the folder and external files must be in the same folder location as the exported model.

Inserted SWF files may have external assets such as other SWF files, images, or XML files. If these assets are referenced with relative paths, those paths must be available from the location where the host model is run. For example, a SWF file located at `z:\file.swf` loads an external image from `z:\images\myImage.jpg`, and you add that SWF file to your model at `c:\visualization.xlf`. When you open your model, the added SWF file attempts to load the referenced image from `c:\images\myImage.jpg`, which is the relative location of the image.

⚠️ **Caution**

If a model references an external file, when the model runs, Adobe security restrictions apply. When the model is run locally, the SWF file or the host application file must be made trusted. If the file is run on a web server, there must be a cross-domain policy file. For more information, see *Adobe Flash Player Restrictions*.

1. Select the *Image* component and open the *Properties* panel.
2. On the *General* view, before you import an image file, specify the *Resize Image to Component* option.

   This setting determines the size of the imported image. If you select *Resize Image to Component*, the image size is adjusted to fit the *Image* component. If this option is not selected, the *Image* component size is adjusted to fit the size of the image that you import. This setting does not apply to SWF files.
3. Click *Import*, navigate to the JPG or SWF file, and click *Open*.
4. Set the following options:

Option	Description
Embed File	Select this option to add a copy of the image file to the model. If you embed the image it becomes part of the exported SWF file and is always available. This option adds to the size of the SWF file and may affect performance.
	To reference an external file instead, deselect this option. The image is loaded when the user views it.
	ℹ️ **Note**
	You can reference external Dashboards SWF files, but you cannot embed them.
Transparency	Use this option to allow objects behind the image component to be seen through the image. Move the slider to the right to increase its transparency and move it to the left to reduce its transparency.
Block Mouse Events	If the *Image* component is placed over other components, selecting this option prevents users from using the mouse to interact with the underlying components. For more information, see *Disabling User Input*.
	ℹ️ **Note**
	For imported SWF files, mouse events are blocked even if this option is not selected.

138

Dashboards and Presentation Design User Guide
Working with Components

Option	Description
Hide SWF Background Color	For imported SWF files, select this option to make the SWF background invisible. If this option is not selected, the SWF background can be seen in the model.

5. Click the *Behavior* view to set dynamic visibility options. For more information about dynamic visibility, see *Managing the Visibility of Components*.

4.2.9.2.1 Embedding versus referencing external files

When you import an image or Flash file into an *Image* component, you can embed the file or reference it in an external location. The following table describes the differences between embedding external files in the model and referencing them.

Embedded Files	Referenced Files
The external file is built into the SWF file when the model is imported.	The external file is copied to a folder with the same name and location as the exported SWF file.
The model (either XLF or SWF) can be distributed as a single file.	The model (either XLF or SWF) must be distributed along with the external file folder, and the folder and external file must remain in the same relative location to the model.
The external file loads when the model loads, even if it is hidden through dynamic visibility.	The external file loads as needed. If it is hidden through dynamic visibility, the file is not loaded.
The model takes longer to load because it has to load the external file as well.	The model loads quickly because it does not have to load the external file immediately.
If dynamic visibility is configured to load external files, it works faster because the files are already loaded.	If dynamic visibility is configured to load external files, it takes longer to show hidden components because the external files must be loaded first.
When hidden with dynamic visibility, external files maintain their state.	When hidden with dynamic visibility, external files are closed and revert to their default state.
If the external file changes, the model must be re-exported.	If the external file changes, it can be replaced in the external file folder and the model will load it without being re-exported.

4.2.9.3 To configure Ellipse and Rectangle components

Use the *Ellipse* or *Rectangle* components to change the appearance of your model. You can add these shapes to different areas of the canvas to visually group items.

1. Select the component and open the *Properties* panel.
2. On the *General* view, under the *Enable Border* area, set the following options for the border:

Option	Description
Enable Border	Select this option to display an outline around the shape.
Color	If *Enable Border* is selected, click the *Color Selector* ( ) to set the border color.
Thickness	If *Enable Border* is selected, type a border thickness.
Transparency	If *Enable Border* is selected, to allow components behind the border to be visible, move the slider to the right to increase the transparency of the border. To hide items behind the border, move the slider to the left.

3. Under the *Fill* area, set the following options:

Option	Description
Type	From the list select one of the following options: ○ *None*: the shape has no fill and is completely transparent. ○ *Solid*: the shape is filled with a solid color. ○ *Linear (gradient)*: the shape color changes from one side to the other. ○ *Radial (gradient)*: the shape color changes from the center of the shape to the edge.
Color	Click the *Color Selector* ( ) to select the color for the shape or gradient.
Transparency	To make the fill transparent, move the slider to the right. To hide items that are behind the shape, move the slider to the left.
Gradient Preview	If *Type* is set to *Linear (gradient)* or *Radial (gradient)*, this panel illustrates the color transition for the gradient. To adjust the gradients in the shape, do any of the following: ○ To add another gradient section, mouse over the panel and click when a *Plus* icon appears. You can add several gradient sections. For each section, a marker appears below the *Preview* panel. ○ To change the position of a gradient section, drag the gradient marker that is below the *Preview* panel. ○ To change the color of a gradient section, double-click the gradient marker and select a color. ○ To change the transparency of a gradient, click the gradient marker and then move the marker on the *Transparency* slider to increase or decrease the visibility.
Rotation	If *Type* is set to *Linear (gradient)*, use this option to adjust the angle of the gradient axis. Enter a value between 0 and 360 to rotate the gradient axis.

4. Click the *Behavior* view to set dynamic visibility options. For more information about dynamic visibility, see *Managing the Visibility of Components*.

4.2.9.4 To configure Horizontal Line and Vertical Line components

Use the *Horizontal Line* and *Vertical Line* components to change the appearance of your model. You can add lines to different areas of the canvas to visually separate items.

1. Select the component and open the *Properties* panel.
2. On the *General* view, set the following options:

 Dashboards and Presentation Design User Guide
Working with Components

Option	Description
Line Color	Click the *Color Selector* (▢) to set the line color.
Thickness	Set the line thickness.

3. Click the *Behavior* view to set dynamic visibility options. For more information about dynamic visibility, see *Managing the Visibility of Components*.

4.2.10 Using the web connectivity components

This category contains a group of components that provides options for linking your models to the web.

> i **Note**
>
> Web connectivity is built into the software. In addition to using these components, you can use the *Data Manager* to configure web connectivity options.

Connection Refresh Button

Refresh

The *Connection Refresh Button* allows users to manually refresh the connections that you link it to.

> i **Note**
>
> This component is not available in Presentation Design.

> ➡ **Tip**
>
> To see an example of a model with a refresh button, click ▶ *File* ▶ *Samples* ◀ and open the sample file *Connection Refresh*.

Reporting Services Button

Reporting Services

The *Reporting Services Button* enables you to connect to the Reporting Services server and select a report to use to populate the model.

> i **Note**

This component is available only in SAP BusinessObjects Dashboards. In addition, the Reporting Services Gateway must be installed and configured. For more information, see the *Dashboards SharePoint Installation Guide*, available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ❯ *All Products* ◢.

Slide Show
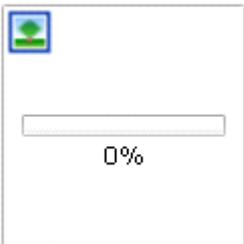
The *Slide Show* creates a slide show of URL-based images and SWF files.

ℹ Note

This component is not available in Presentation Design.

The *Slide Show* loads files from the URL specified in the URL data source. If the slide show URL changes, the slide show component attempts to load the image from the new location. Once the image is loaded, the component transitions from the current image to the one specified at the new URL data source.

ℹ Note

When an image is imported into normal image components, the software stores its own local copy of the image internally with the model. The original image file is no longer accessed or used in the model. Because the image is stored within the model, the model file can be moved or emailed and the image travels with it. However, the slide show points to a URL, which can change at runtime. Since the image or SWF file is not built into the model, the slide show component must be able to access the URL and the image or SWF file must be available to load at runtime.

⚠ Caution

If a model is connected to an external data source, when the model runs, Adobe security restrictions apply. When the model is run locally, the SWF file or the host application file must be made trusted. If the file is run on a web server, there must be a cross-domain policy file. For more information, see *Adobe Flash Player Restrictions*.

SWF Loader
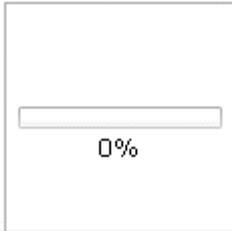
The *SWF Loader* is similar to the *Slide Show*: it loads SWF files from a specified URL so you do not have to import the files. *SWF Loaders* have better memory management than *Slide Shows*, but do not have transition effects and cannot block mouse events.

> **i Note**
>
> This component is not available in Presentation Design.

> **i Note**
>
> The Adobe Flash Player memory management method controls when inactive items are removed from memory. To avoid heavy memory consumption when using the SWF loader:
>
> - Reduce the number of *SWF Loader* and *Slide Show* components included in a single model.
> - Reduce the number of SWF files that the *SWF Loader* and *Slide Show* components in the model must load concurrently at runtime.

Loaded SWFs are not built into the model, so the *SWF Loader* must be able to access the URL and the file must be available to load at runtime. The *SWF Loader* can be configured to load other dashboard models that are saved on the BI platform using the DocumentDownload URL. For more information about using the DocumentDownload URL, see *Embedding BI platform documents in models* [page 244].

> **⚠ Caution**
>
> If a model is connected to an external data source, when the model runs, Adobe security restrictions apply. When the model is run locally, the SWF file or the host application file must be made trusted. If the file is run on a web server, there must be a cross-domain policy file. For more information, see *Adobe Flash Player Restrictions*.

URL Button

The *URL Button* links to a relative or absolute URL when users click it at runtime.

4.2.10.1 Web connectivity general properties

Use the following table as a guide for setting properties on the *General* tab. The available options vary depending on the type of component.

Option	Description
Label	The text that appears on a component button. Type the text in the box or click the *Cell Selector* (![icon]) to link it to a cell in the embedded spreadsheet.
Available Connections	This table lists the connections that can be linked to a *Connection Refresh Button*. Select the checkbox corresponding to the external data connections to be refreshed when the user clicks the button. To select all connections, select the checkbox at the top of the column.
Launch Data Manager	The items in the *Available Connections* list for a connection refresh button are based on connections configured in the *Data Manager*. Click this button to open the *Data Manager* to add or modify external data connections.
URL	The web address a *URL Button* links to. Type the URL in the box or click the *Cell Selector* (![icon]) to link it to a cell in the embedded spreadsheet. When this value is linked to the embedded spreadsheet, the button is updated whenever the value in the cell changes. You can then configure the source cell to be updated dynamically at runtime, allowing the URL button to call different pages. **i Note** You must include "http://" or "https://" in the web address.
Encode URL	For a *URL Button*, select this checkbox to encode special characters in the URL. This option ensures that there are no invalid characters.
Window Options	For a *URL Button*, this option specifies where the web page at the URL address opens. Select *New Window* to open the page in a separate browser window or *This Window* to open the page in the same window and replace the current contents in the window.
Select Report	For a *Reporting Services Button*, click this button to open the *Select Report* dialog box, connect to Reporting Services, and choose a source report for the button.
Display Parameters at Run Time	For a *Reporting Services Button*, select this option to display the report parameters when the model runs.
Use Service Values	For a *Reporting Services Button*, select a parameter in the list and select a service value from this list to include the values in the report.
Link to Cell	For a *Reporting Services Button*, select a parameter in the list and click the *Cell Selector* (![icon]) to use values from the embedded spreadsheet.

Option	Description
Data Mapping	For a *Reporting Services Button*, to map the report data to the model, select *Default* to use the default data mapping or select *Advanced* and then configure the following options:

Option	Description
Mapped Ranges	The list shows currently mapped ranges. Click *Plus* and *Minus* to add or delete ranges.
Selected Columns	The list shows the columns for the selected range. Select a column and click the *Up* and *Down* arrows to change the sort order. Click *X* to remove the selected column from the list.
Name	Type a name for the selected range.

Option	Description
Range	For a *Reporting Services Button*, this option defines the destination location for the report data. Type or click the *Cell Selector* ( ) to select cells in the embedded spreadsheet.
Use Report History	For a *Reporting Services Button*, select this option to enable use of the report's history.
URL (JPEG or SWF File)	The web address where a *Slide Show* retrieves a JPEG or SWF file. Type the URL in the box or click the *Cell Selector* ( ) to link it to a cell in the embedded spreadsheet. Once the image file is loaded, the slide show transitions to the image using the transition parameters defined on the *Behavior* view. ⓘ Note Progressive JPEGs and JPEGs with CMYK coloring are not supported.
Source URL	The web address where a *SWF Loader* retrieves a SWF file. Type the URL in the box or click the *Cell Selector* ( ) to link it to a cell in the embedded spreadsheet. You can also load another dashboard model that is stored in the BI platform by entering the DocumentDownload URL in this box. For more information about using the DocumentDownload URL, see *Embedding BI platform documents in models* [page 244].
Block Mouse Events	Select this option for a slide show if you want to prevent users from interacting with the component at runtime.
Application Domain	For a slide show or a *SWF Loader*, this option allows you to specify which domain the *Slideshow* or *SWF Loader* model will load into. The following three options are available:

Option	Description

Option	Description
New	The model loads into a child application domain. **i Note** If the model is connected using an LCDS connector, the data will not load.
Current	The model loads into the current application domain. The model uses only the first class definition; subsequent class definitions that are loaded in the domain (such as for the LCDS connector) are ignored. Use this option if the LCDS connection can share the main model's domain. **i Note** This option allows you to embed external codes to control the model, but if the LCDS connection is unable to share the main model's domain, data is not loaded.
Compatible	The model loads into a compatible bootstrap loader. If you need to run an LCDS model in a slideshow or SWF Loader, this option allows the model to share the classes defined in the bootstrap loader but use its own class definitions for everything else. For example, the bootstrap loader loads the Slideshow or SWF Loader model and the Slideshow or SWF Loader model loads the LCDS model. When the LCDS model is run in a slideshow or in the SWF Loader, the data is loaded correctly. **i Note** For more information about bootstrap loaders, see the *Adobe Flex documentation*.

Related Links

Web connectivity behavior properties [page 147]
Web connectivity appearance properties [page 148]

4.2.10.2 Web connectivity behavior properties

Use the following table as a guide for setting the *Behavior* properties for web connectivity components. Some options are available only for specific components.

Option	Description
Common Tab	
Refresh After Components are Loaded	When this option is selected for a *Connection Refresh Button*, connections are refreshed automatically after the model loads during runtime.
Trigger Cell	To configure an action within the model that triggers the component, you can select a cell in the embedded spreadsheet as the trigger cell. When the value in that cell meets the trigger condition, the component responds as if the user clicked it. Click the *Cell Selector* () and select a cell in the spreadsheet, and then set the trigger condition: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*When Value Changes*</td><td>The component is triggered when the value in the trigger cell changes.</td></tr><tr><td>*When Value Equals*</td><td>The component is triggered when the value in the trigger cell is set to a specific value. Enter the value or click the *Cell Selector* () to select a source location from the embedded spreadsheet.</td></tr></table>
Refresh Before Components are Loaded	When this option is selected for a *Reporting Services Button*, the data is refreshed at runtime before the components are loaded.
Refresh Every	The data is refreshed automatically at a set interval based on the time values set for this option.
Loading Message	For a *Reporting Services Button*, type a message to display while the data connection is loading, or click the *Cell Selector* () to select a source location for the message.
Idle Message	For a *Reporting Services Button*, type a message to display when the data connection is idle, or click the *Cell Selector* () to select a source location for a message.
Insert In	For a *Reporting Services Button*, click the *Cell Selector* () and select a destination to insert the text of the *Loading Message* and *Idle Message*.
Enable Load Cursor	For a *Reporting Services Button*, select this option to display a special cursor while the data connection is loading.
Disable Mouse Input on Load	For a *Reporting Services Button*, select this option to disable user interactivity while the data connection is loading.

Option	Description
Transition Type	For a *Slide Show*, select the animated effect for transitions from one slide to another.
Easing Type	For a *Slide Show*, this option specifies how the speed of the transition varies. Select from the following options:

Options	Description
Slow In	The transition begins slowly and accelerates as it progresses.
Slow Out	The transition begins quickly and slows down as it progresses
Slow In and Out	The transition begins slowly and increases its speed until the middle of the transition and then slows down as it finishes.

Option	Description
Transition Time	For a *Slide Show*, this option specifies how quickly the transition between images takes. ➡ Tip Slower transition times create smoother transitions.
Dynamic Visibility	This option allows you to specify conditions that control when the component is visible in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].
Animations and Effects Tab	
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.

Related Links

Web connectivity general properties [page 144]
Web connectivity appearance properties [page 148]

4.2.10.3 Web connectivity appearance properties

Use the following table as a guide for setting the *Appearance* properties for web connectivity components. Some options are only available for specific components.

Option	Description
Layout Tab	
Show Button Background	Specifies whether the background behind a button component is visible. If you deselect this option, only the label text for the button appears.

Option	Description
Transparency	Specifies the visibility of the button. At 100% transparency, the button is completely invisible. Adjust the marker on the slider bar to change the transparency.
Sizing Method	For *Slide Show* and *SWF Loader* components, this setting defines how images are sized within the bounds of the component. Select one of the following options: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*Original Size*</td><td>The images are not resized, but they are cropped to fit in the component.</td></tr><tr><td>*Stretch*</td><td>The image is resized to fit in the component, and can also be stretched horizontally or vertically.</td></tr><tr><td>*Scale*</td><td>The image is resized proportionally to fit the component.</td></tr></table>
Horizontal Alignment	For *Slide Show* and *SWF Loader* components, this setting defines how images are aligned horizontally within the component. Select one of the following options: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*Center*</td><td>The image is centered horizontally in the component.</td></tr><tr><td>*Left*</td><td>The image is aligned to the left edge of the component.</td></tr><tr><td>*Right*</td><td>The image is aligned to the right edge of the component.</td></tr></table>
Vertical Alignment	For *Slide Show* and *SWF Loader* components, this setting defines how images are aligned vertically within the component. Select one of the following options: <table><tr><th>Option</th><th>Description</th></tr><tr><td>*Middle*</td><td>The image is centered vertically in the component.</td></tr><tr><td>*Top*</td><td>The image is aligned to the top edge of the component.</td></tr><tr><td>*Bottom*</td><td>The image is aligned to the bottom edge of the component.</td></tr></table>
Text Tab	
Label	The text that appears on a button.
Format Selected Text	Click *Label* and adjust the settings in this area to format the label text. For more information about formatting text, see *Formatting text for component elements*. ℹ Note If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.
Color Tab	
Buttons	The color of the button.

Option	Description
	You can set a *Pressed Color* for when the button is clicked and a *Default Color* for when it is inactive.
Labels	The color of the label text.
	You can set a *Pressed Color* for when the button is clicked and a *Default Color* for when it is inactive.

Related Links

Web connectivity behavior properties [page 147]

Web connectivity general properties [page 144]

4.2.11 Using universe connectivity components

This category contains a group of components that can be used with queries on CSL universes to SAP BusinessObjects Business Intelligence platform and SAP Crystal Reports Server.

> ℹ **Note**
>
> These components are not available in Presentation Design.

Query Prompt Selector
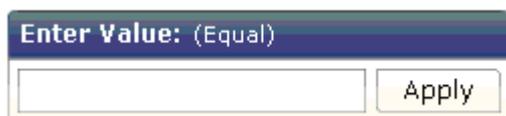
The *Query Prompt Selector* component allows users to choose values for BEx or universe query prompts at runtime.

Depending on the type of prompt that you link the component to, users can select a single value to use as a filter parameter, or choose a selection of result objects. For instance, you can configure a *Query Prompt Selector* to restrict query result objects to a small set chosen by users at runtime that can be represented in a chart.

Query Prompt Selectors are also capable of displaying hierarchical lists of query result objects that users can explore at runtime.

You can add a *Query Prompt Selector* from the component menu and then choose a source query prompt, or you can add a preconfigured *Query Prompt Selector* by dragging a query prompt from the *Query Browser* onto the canvas. For more information, see *To place a prompt on the canvas* [page 234].

Query Refresh Button

Query Refresh

The *Query Refresh Button* allows users to refresh the query data during runtime.

Related Links

To browse queries [page 214]

4.2.11.1 To set up Query Refresh Button components

If your model includes a universe or BEx query, you can add the *Query Refresh Button* to allow users to request a query refresh at runtime.

1. Select the *Query Refresh Button* and open the *Properties* panel.
2. On the *Properties* panel, click the *General* view and set the following options:

Option	Description
Label	Type the text to display on the button or click the *Cell Selector* (⬚) to retrieve the label text from a cell in the embedded spreadsheet.
Refresh Queries	The table lists the queries that have been defined for the model. Select the checkbox next to the queries that you want the button to refresh.

3. Click the *Behavior* view and set the following options:

Option	Description
Common Tab	
Refresh After Components Are Loaded	When this option is selected for a *Query Refresh Button*, connections are refreshed automatically after the model loads during runtime.
Trigger Cell	To configure an action within the model that triggers the *Query Refresh Button*, you can select a cell in the embedded spreadsheet as the trigger cell. When the value in that cell meets the trigger condition, the button responds as if the user clicked it. Click the *Cell Selector* (⬚) and select a cell in the spreadsheet, and then set the trigger condition:

Option	Description
When Value Changes	The button is triggered when the value in the trigger cell changes.
When Value Equals	The button is triggered when the value in the trigger cell is set to a specific value. Enter the value or click the *Cell Selector* (⬚) to select a source location from the embedded spreadsheet.

Option	Description
Dynamic Visibility	This option allows you to specify conditions that control when the component is visible in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].
Animations and Effects Tab	
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.

4. Click the *Appearance* view and set the following options:

Option	Description
Layout Tab	
Show Button Background	Specifies whether the background behind a button component is visible. If you deselect this option, only the label text for the button appears.
Transparency	Specifies the visibility of the button. At 100% transparency, the button is completely invisible. Adjust the marker on the slider bar to change the transparency.
Text Tab	
Label	If you want text to appear on the button, select the checkbox beside label, then under *Format Selected Text*, define what the text should look like. For more information about formatting text, see *Formatting text for component elements* [page 43].
Format Selected Text	Click *Label* and adjust the settings in this area to format the label text. For more information about formatting text, see *Formatting text for component elements*. i **Note** If a global font is set for the document, you cannot change the font type. For more information about global fonts, see *To set global font options*.
Color Tab	
Buttons	The color of the button. You can set a *Pressed Color* for when the button is clicked and a *Default Color* for when it is inactive.
Labels	The color of the label text. You can set a *Pressed Color* for when the button is clicked and a *Default Color* for when it is inactive.

4.2.11.2 To set up Query Prompt Selector components

If your model includes a universe or BEx query with a prompt, you can add the *Query Prompt Selector* component to your model to allow users to select parameters for the prompt at runtime.

> **i Note**
>
> You can also add *Query Prompt Selectors* to your model by dragging the prompt from the *Query Browser* to the canvas. *Query Prompt Selectors* that you add in this manner are configured according to the settings that you chose for the prompt in the *Add Query* dialog box. For more information, see *To place a prompt on the canvas* [page 234].

1. Select the *Query Prompt Selector* and open the *Properties* panel.
2. On the *Properties* panel, click the *General* view and set the following options:

Option	Description
Source Prompt	Select a prompt to set the label and the operator on the component, and the number and type of values that the user can choose at runtime. If you want to change the settings for the prompt, you must edit the prompt from the *Query Browser*. > **i Note** > If you delete or edit a prompt or query, the *Source Prompt* list is updated to reflect your changes.
Destination Prompt(s)	Select the query prompt that the selected prompt value is sent to. The *Source Prompt* that you selected is automatically selected in this table. If you have multiple queries, you can select prompts from other queries to send the value to these prompts too. For instance, you could configure a single *Query Prompt Selector* to choose the date for two queries. You cannot select more than one destination prompt for each query.
Insert Selected Items	To insert the captions of the selected prompt values in the spreadsheet, click the *Cell Selector* ( ) button and select a destination range in the spreadsheet. Select a single destination cell for prompts with single values, or a column for prompts that allow the user to select multiple values.
Refresh Queries	The *Refresh Queries* table lists the queries that have been defined for the model. Select the queries that the component refreshes.

Option	Description
Refresh When Selection Changes	Select this option to refresh the selected queries when the user selects a new prompt value.
Refresh On Button Click	Select this option to display a button that the user must click to refresh the selected queries.
Button Label	Type the text to display on the button or click the *Cell Selector* (🔲) to retrieve the label text from a cell in the embedded spreadsheet.

3. Click the *Behavior* view and set the following options:

Option	Description
Common tab	
Enable Search in List of Values	Select this option to add a search function to the component list that allows users to search for the prompt value.
Display "No Value" As:	Set the text that is displayed in the list of values when no value is set. The default text is *No Value*. ⓘ **Note** This property is disabled if the component is bound to a mandatory prompt.
Enable Display Menu	For a prompt with a hierarchical list of values, select this option to add a menu that allows users to customize how values are displayed.
Enable Subset Selection Options	For a prompt with multiple selection and a hierarchical list of values, select this option to add buttons to each value in the list of values at runtime. When users click the button for a parent value, a menu opens and allows them to select or deselect all children or descendants of the value. This option is useful for choosing values from large sets of hierarchical data.
Default display mode	For a prompt with a hierarchical list of values, specify how values appear in the list. You must select *Enable Display Menu* to configure these settings. If you leave *Enable Display Menu* selected, users can change these settings at runtime. If you deselect *Enable Display Menu*, the *Default display mode* that you choose cannot be changed at runtime.

Option	Description
	Select one of the following options for how each value is displayed:
	○ *Display caption*: only the caption is displayed.
	○ *Display unique name*: only the unique name is displayed. Values may be more difficult to identify using the unique name.
	○ *Display both caption and unique name*: the caption and the unique name are displayed.
	You can also select *Display number of children* to show the number of children for each value.
No Value	Select this option to treat empty selection values as having no value.
Empty String	Select this option to treat empty selection values as empty strings.
Dynamic Visibility	This option allows you to specify conditions that control when the component is visible in the model. For more information about using dynamic visibility, see *Managing the visibility of components* [page 162].
Animations and Effects tab	
Entry Effect	Configure an animated entry effect for the appearance of the component. This option is useful for components with dynamic visibility. Select an entry effect from the *Type* list and specify how long the effect lasts in the *Duration (seconds)* box.

4. Click the *Appearance* view and set the following options:

Option	Description
Layout Tab	
Display Options	Configure how the list of values appears. Select *Expanded* to display an expanded list that always stays open or *Collapsed* to display a drop-down list that can be opened or hidden by the user.
Hierarchy Levels	For a hierarchical list of values, specify which levels in the hierarchy appear expanded when the model loads. Users can expand or collapse individual levels at runtime.

Option	Description
	○ *Expand Level*: to specify the number of levels to expand, select this option and type a number in the box. ○ *Expand All Levels*: to expand all the levels in the list of values, select this option. ○ *Collapse All Levels*: to collapse all levels in the list of models, select this option.
Text Tab	
Prompt Text	Select this option to display the prompt label that you configured in the *Edit Prompt* dialog box. You can also format this text using the settings in the *Format Selected Text* area.
Selection Type	Select this option to display the operator or selection method for the component. You can also format this text using the settings in the *Format Selected Text* area.
Values	To format the text for the prompt values, click *Values* and configure the settings in the *Format Selected Text* area.
Apply Button Text	To format the text for the button that refreshes the query, click *Apply Button Text* and configure the settings in the *Format Selected Text* area.
Search Box Label	Select this option to display text that identifies the search box. This text disappears when the user clicks the search box. You can also format this text using the settings in the *Format Selected Text* area.
Search Box Text	To format the text that the user types in the search box, click *Search Box Text* and configure the settings in the *Format Selected Text* area.
Selection Information Text	To format the info text for the component, click *Apply Button Text* and configure the settings in the *Format Selected Text* area.
Color Tab	
Background Color	Set the color of the component background.
Text Color	Set the color of the text at the bottom of the component showing how many values in the list are selected.

156   Dashboards and Presentation Design User Guide
 Working with Components

Option	Description
Link Color	Set the color for the links at the bottom of the component to show the entire list of values or to show the selected values only.
Header	Choose the color for the following elements: ○ *Background Color*: the color of the header background ○ *Prompt Text Color*: the color of the text that you configured in the *Edit Prompt* dialog box. ○ *Selection Type Color*: the color of the operator or selection type text.
Values	Set the color of the values in the list, including the *Selected Members* list. You can set the *Default Color* for when values are inactive, the *Over Color* for when the user mouses over a value, and the *Selected Color* for values that are selected.
Value Backgrounds	Set the color of the background behind the values in the list, including the *Selected Members* list. You can set the *Default Color* for when values are inactive, the *Over Color* for when the user mouses over a value, and the *Selected Color* for a value that selected.
Buttons	Set the color of the component buttons. You can set different colors for the button backgrounds and for the color of the icons and text on the buttons. You can also set different colors for when the button is inactive (*Default Color*), for when the button is clicked (*Pressed Color*), and for when the button is disabled (*Disabled Color*).
Arrow Buttons	Set the *Button Color* and *Arrow Color* for the buttons that open drop-down lists.
Search Box	Set the *Text Color* for the text that the user types in the search box, and the *Label Color* for the label text in the search box.
Scroll Bar	Set the color for parts of the bars that let users scroll through the list of values. The settings that you can configure include *Track Color*, *Thumb Color*, *Button Color*, and *Button Symbol Color*.

4.2.12 Deleting components

To delete a component, select the component and press `Delete` or right-click it and choose *Delete*.

> ➡ Tip
>
> You cannot delete components in a group individually. You must either ungroup them to delete individual components, or delete the entire group.

Deleting a component can be undone pressing `CTRL` + `Z` or clicking *Undo* on the *Edit* menu.

4.2.13 Understanding alerts

Alerts call the attention of the user to a specific item or action that has reached a pre-established limit. These limits, often called targets, budgets, benchmarks, and so on, are a standard to which an item is compared.

When alerts are enabled, one or more colors appear on the component marking the levels of alerts. Typically, green represents a desired condition, yellow indicates that the situation is not ideal, and red marks an issue as critical. When you define the alert, you specify which values to monitor, as well as the number and value of alert thresholds, and whether it is preferable for the values to be higher than, lower than, or as close to the target value as possible. You can also define custom colors for different alert thresholds.

Alerts are available on the following components:

Component Type	Component
Charts	*Bar Chart*, *Bubble Chart*, *Column Chart*, *Combination Chart*, *Line Chart*, *Radar Chart*, *Stacked Bar Chart*, *Stacked Column Chart*, and *XY Chart* > ⓘ Note > > Alerts are unavailable for charts with multiple data series, except for combination charts.
Selectors	*Combo Box*, *Icon*, *Label Based Menu*, *List Box*, *List Builder*, *Scorecard*, *Spreadsheet Table*, *Ticker*
Single Value	All except *Play Control*
Maps	All
Other	*Grid*

4.2.13.1 To configure alerts

If you want a component to indicate when its data value reaches a specific range, you can configure and customize alerts.

1. Select the component and open the *Properties* panel.

2. Click the *Alerts* view.

3. Select *Enable Alerts*.

> **i Note**
>
> For charts with multiple data series, this option is only available with the *Combination Chart*.

> **i Note**
>
> For *Scorecard* and *Spreadsheet Table* components, you can configure different alert settings for each column. For each column that you want to show alerts for, select the checkbox in the *Show* column and then define the alert properties for the selected column.

4. For *Scorecard*, *Label-based Menu*, *List Builder*, *List Box*, *Combo Box*, and *Spreadsheet Table* components, in the *Alert Values* box, specify the values to be compared to the target values for alerts. Click the *Cell Selector* ( ) and select a source location in the spreadsheet for the values.

5. Select one of the following methods to use for the alerts:

Option	Description
As a Percent of Max Value	For *Single Value* components, select this option to base alerts on percentages of the maximum value. The maximum value is determined by the *Maximum Limit* setting on the *General* view.
As Percent of Target	Select this option to base the alerts on percentages of a target value that you specify. To set the target values by linking to cells in the embedded spreadsheet, click the *Cell Selector* ( ). To manually define the target values, click *Manual Edit* ( ). **i Note** You can set separate target values for each data point.
By Value	Select this option to base the alerts on values you set in the *Alert Thresholds* area.

6. For Bubble Charts, select an Alert Variable option:

Option	Description
Value Set	Select this option to base alerts on the X values, Y values, or bubble sizes.
Use a Range	Select this option to base alerts on a range of data in the Excel spreadsheet.

7. In the *Alert Thresholds* section, set the following options:

Option	Description
Use a Range	Select this option to select the threshold levels from cells in the embedded spreadsheet, then click the *Cell Selector* ( ) to select the cells.
Enter a Value	To manually define the threshold values, do the following: <table><tr><th>Task</th><th>Step</th></tr><tr><td>Edit an existing threshold values</td><td>Click the value in the *From* or *To* column and type a new value.</td></tr><tr><td>Add a threshold value to the list</td><td>Type a value in the box and click *Add*. This step creates a new alert range for the component.</td></tr></table>

Option	Description	
	Task	**Step**
	Delete a threshold value	Click the *X* icon beside the value. An alert range is removed from the component.
	Assign colors to threshold levels	Click the *Color Selector* beside each threshold value to set the color for the corresponding alert range. For more information about using the color selector, see *Setting colors for component elements*. ℹ **Note** If the *Enable Auto Colors* checkbox is selected, you cannot set colors for each threshold individually.

8. To automatically define the colors assigned to each threshold level, select *Enable Auto Colors*.

> ℹ **Note**
>
> To manually assign colors to each threshold level, deselect this option and select the *Color Selector* button beside each threshold level to set its color.

9. If *Enable Auto Colors* is selected, to set a different color gradient scheme, click *Auto-color* (   ).

10. In the *Color* dialog box, select a color scheme from the *Custom* or *Built-in* lists, or, to create a new color gradient, click *Create new gradient* and set the following options:

Option	Description
2-Color Gradient	Select this option to create a gradient for the alert scale that includes 2 colors.
3-Color Gradient	Select this option to create a gradient for the alert scale that includes 3 colors.
Fill	Click the colors in this column to set background colors for each part of the alert gradient, and for parts of a component with no data.
Font	Click the colors in this column to set text colors for each part of the alert gradient, and for parts of a component with no data.

11. Select a *Color Order* to specify which values are desirable. When you set the *Color Order*, the colors for the alert thresholds are rearranged so that the same colors always represent desirable values.

Option	Description
Low values are good	Use this option when the ideal value is lower than the target value. For example, select this option for a component that displays expenses that you want to minimize.
Middle values are good	Use this option when the ideal value is the target value. For example, select this option for a component that displays inventory quantities that you want to be as accurate as possible. ℹ **Note** This option is only available when using percentages.
High values are good	Use this option when the ideal value is higher than the target value. For example, select this option for a component that displays revenue values that you want to maximize.

12. For *Vertical Slider*, *Horizontal Slider*, *Horizontal Progress Bar*, *Dial*, *Dual Slider*, and *Gauge* components, select where you want the alert color to appear:

Option	Description
Background	The component background displays the alert color.
Marker	The marker, needle, and so on display the alert color.
Value	The alert color is added as a background to the text box that displays the current value.

Alerts are configured for the component.

4.2.13.2 To configure alert icons for a Scorecard

Before performing this task, configure a *Scorecard* and set the alert targets, threshold, and color order for each column as necessary.

Scorecards display alerts both as colors and as icons. The icons provide another visual clue about the state of values in the component; for instance, an acceptable value can be identified by a check mark, and a value that requires action can be marked by an exclamation point.

You can customize the icon settings by disabling the icons, choosing a preset group of icons, or choosing individual icon shapes and colors.

1. Select the *Scorecard* and open the *Properties* panel.
2. On the *Properties* panel, click the *Alerts* view and select the column that you want to configure alert icons for.
3. To configure icons individually, for each alert range, click the *Icon Selector* ( ), and in the *Icon* dialog box, select a shape and a color for the icon.
4. To select a preset group of icons, click *Auto Icons* ( ) and select an icon set.

   The icon sets are divided into groups based on the number of icons in the set. If possible, select a set with one icon for each alert threshold.
5. To create a custom icon set, click *Auto Icons* ( ) and click *Create New Icon Set*. Configure the options for the icon set and then click *Ok*. You can set the following options:

Option	Description
Number of Icons	Select the number of icons to include in the set. Icon sets are most effective when the number of icons matches the number of alert ranges.
Color	Click the colors in this column to set colors for each icon. The icons at the top of the column represent desirable values, and the lower icons represent less favorable values.
Shape	Click the *Icon Selectors* ( ) in this column to set shapes for each icon. The top icons represent desirable values, and the bottom icons represent unfavorable values.

6. Repeat these steps for other columns in the *Scorecard* as necessary.

The icons are configured. If you change the *Color Order*, the icons are rearranged so that the same icons represent desirable values.

4.2.14 Managing the visibility of components

When models contain too much information or too many visual components, users may have difficulty focusing on specific sets of information. If you need to display a lot of information, you can simplify the visual layout by displaying some information only when it is needed. To make your model easy to navigate you may want to reuse certain areas of the canvas and hide and show components at runtime.

Dynamic visibility allows you to control when a component is visible based on the contents of a spreadsheet cell. When the *Status* for the component matches the *Key* in the spreadsheet cell, the component is visible. Otherwise, it is hidden.

For example, you can add a *Toggle Button* that users can click to display a specific chart. You configure the button to alternately insert **<showChart>** and **<hideChart>** into its destination cell each time it is clicked. By linking the chart's *Status* to this destination cell and setting the chart's *Key* to **<showChart>**, the chart becomes visible when the *Toggle Button* inserts the value **<showChart>** and hidden when the *Toggle Button* inserts **<hideChart>**.

In a more complex model, you may want to allow users to switch between several different components. For example, you can create several charts and assign a unique key value to each one, such as **<chart1>**, **<chart2>**, and so on. Then, you can add a *List Box* component to the canvas and set the source data to match the chart keys and the destination as a blank cell in the spreadsheet. When you link each chart's status to that destination cell, selections in the list box at runtime make a single chart visible and hide the others.

When dynamic visibility is enabled, you can also add animation effects to control how each component appears and disappears from the model.

> ➡ **Tip**
>
> To see an example of a model with dynamic visibility, click ▌ *File* ❯ *Samples* ◣ and open the sample file *Dynamic Visibility*.

> **ⅈ** **Note**
>
> Add-on components do not support dynamic visibility. If you want to control the visibility of add-on components, you can place the component in a canvas container and apply the dynamic visibility settings to the canvas container to show or hide the add-on component it contains.

4.2.14.1 To set up dynamic visibility

1. Add a component to the canvas and open the *Properties* panel.

2. On the *Properties* panel, click ▌ *Behavior* ❯ *Common* ◣.

3. In the *Dynamic Visibility* area, beside the *Status* box, click the *Cell Selector* (▨) and select the cell in the spreadsheet that controls the component's visibility.

   > **ⅈ** **Note**
   >
   > Do not select a cell that is already bound to the component, such as a cell in its data range.

   The *Key* box becomes available when you set the *Status*.

4. In the *Key* box, type a value or click the *Cell Selector* ( ) and select a cell that contains the key value for this component.
   When the value in the *Status* cell matches the key value, the component becomes visible.

5. Add another component to the canvas to insert the value of the *Status* cell.

   Typically, you use a selector component such as a *Check Box* or a *Radio Button* to allow users to choose whether components are visible. You can also use a chart or a map component that is configured to insert data into the *Status* cell, however.

6. On the *General* view of the *Properties* panel for the selector, configure the *Source Data* so that it matches the *Key* values for the components it is controlling.

   > **i Note**
   >
   > If the selector is controlling only one component, you must enter a second value for hiding the component as well.

7. In the *Destination* box, click the *Cell Selector* ( ) and select the *Status* cell for the components with dynamic visibility.

At runtime, when the user clicks the selector component, it inserts a key value into the status cell. The dynamically visible component is visible when the inserted value matches its key value.

Related Links

Managing the visibility of components [page 162]
To configure entry effects for dynamic visibility [page 163]

4.2.14.2 To configure entry effects for dynamic visibility

When dynamic visibility is enabled, you can specify how components arrive in the model. For example, they can slide in from one side or slowly fade into few. You can also specify how long the effect lasts.

1. Select the component and open the *Properties* panel.
2. Click ▶ *Behavior* ▶ *Animations and Effects* ◀ .
3. Under *Entry Effect*, in the *Type* list, select one of the following options:

Option	Description
Fade-in	The component appears in a semi-transparent state and gradually becomes solid.
Wipe Right	The left side of the component appears and then the remainder of the component is gradually drawn in from left to right.
Wipe Right-Down	The top left corner of the component appears and the remainder of the component is gradually drawn in both down and toward the right.

> **i Note**
>
> If you select `None`, the component appears immediately and you cannot set the duration for its entrance.

4. In the *Duration (seconds)* list, select how long the effect lasts in seconds.

When the model is run and the component is triggered to appear, it is added to the screen using the selected entry effect and duration.

4.2.15 Disabling user input

Many components can be configured to allow users to modify values at runtime. If you do not want users to change data values from the model, you can often configure components to prevent interaction. However, if the design requires that the component accept user input, you can use a background component to block input.

> **i Note**
>
> For *Single Value* components, you can also prevent users from changing data values in two ways: bind the component to a cell that contains a formula or deselect the *Enable Interaction* option on the ▌ *Behavior* ❯ *Common* ❮ tab of the *Properties* panel.

> **i Note**
>
> If you want to block user input in specific situations, you can use dynamic visibility to only display the transparent background in those situations.

Related Links

4.2.15.1 To configure Background components to block user input

1. From the *Components* browser, drag a *Background* component onto the canvas and place it over the component for which you want to block user input.

2. Select the *Background* component and open the *Properties* panel.

3. On the *General* view, set the *Transparency* slider to 100%.

4. Select the *Block Mouse Events* option.

> **i Note**
>
> This option also disables effects that occur when users mouse over the component. For a bar chart, for instance, values are not displayed when a user places the mouse over each bar.

When the model runs, if users click the component that is behind the background, the mouse click is not registered.

164   Dashboards and Presentation Design User Guide
Working with Components

5 Working with Models

Models are visual representations of your data. A model is made up of components, and the underlying data to which the components are bound.

The basic process of creating a model can be divided into four steps:

1. Import or enter data into the embedded spreadsheet, or create a query from an external data source.
2. Add components to the canvas and link them to the spreadsheet cells or external data sources.
3. Modify the appearance, style, and layout of the components.
4. Preview and publish the model.

Dashboards provides several templates that have been designed for specific purposes. You can use these templates as they are or customize them for your specific needs.

In addition to using templates to quickly create models, you can create a specific look for all components in your model with themes and color schemes.

The software also includes sample models that illustrate some basic features of Dashboards. Each sample includes its own embedded spreadsheet data, so you can see how the components are bound to the data in the spreadsheet.

Related Links

Sample models [page 31]

5.1 Using templates

If you often create models with similar components or settings, you can use a template as a starting point for new models. Templates demonstrate assorted layouts and combinations of components. Some templates are bound to sample data to illustrate how the template works, while others, such as the layout templates, are not bound to any data. When you open a template, it is inserted on the canvas as a new model. You can then add, delete, or modify the template, and save it as a new model or as a modified or new template.

The software offers a set of templates you can use to create a new model. You can also create a model starting from a blank canvas and save it as a template.

> **i Note**
>
> Templates cannot be applied to existing models. To change the look of an existing model, you can apply a different theme or change the color scheme.

Related Links

Using themes [page 167]
Using color schemes [page 167]

5.1.1   To open templates

Rather than building your model from a blank canvas, you can use a template that contains the basic components that you want to use and customize it to suit your needs.

1. Click ▶ *File* ❯ *Templates* ◣.
   The *New From Template* dialog box opens. The *Category* area lists the categories that templates may be assigned to.

2. In the *Category* area, click a category.
   The *Items* area displays the available templates for that category.

3. In the *Items* area, click a template.
   An interactive preview of the template is shown in the *Preview* area and, if a description was written for the template, it appears in the *Description* area.

4. When you have selected the template you want to open, click *OK*.

The template opens on the canvas, and if the template has a sample spreadsheet, it is loaded into the embedded spreadsheet. Many of the templates' spreadsheets contain instructions for entering your own information into spreadsheet cells already bound to components.

Related Links

To import Excel spreadsheets [page 175]

Linking components and spreadsheet data [page 176]

5.1.2   To create a template

While Dashboards come with predefined templates, you can also create and save your own templates. For example, if you are creating a set of Portlets for use in your corporate portal, you may want to save your models as templates so you can re-use them when updates are necessary. To save one of your models as a template, do the following:

1. Create a model that you want to save as a template.

2. Click ▶ *File* ❯ *Save As...* ◣.

   The *Save As* dialog box opens.

3. Navigate to the folder where the Dashboards program is installed and then to the following location: `. . . \assets\template`.

   > **i Note**
   >
   > To create a new template category, create a folder under the \template directory.

4. In the *File name* field, type a file name for the template and click *Save*.

5. Click ▶ *File* ❯ *Export* ❯ *Flash (SWF)...* ◣.

6. Navigate to the templates folder where you saved the template.

7. In the *File Name* list, type the same file name for the SWF file that you entered for the template and click *Save*.

The model is saved as a template. The next time you open the templates window, your template will be available for preview and use.

166   Dashboards and Presentation Design User Guide
Working with Models

5.2 Using themes

Themes (formerly called skins) provide an easy way to customize your model and maintain a consistent look throughout the model. Themes control the style and properties applied to each component. Themes also include a color scheme, but you can further customize your model by changing the color scheme. If a new color scheme is applied, the components retain their style and property settings, but use the new color palette.

Related Links

Using color schemes [page 167]

5.2.1 To apply a theme

If you want to change the look of your model, you can apply a theme to set the style and property settings for all components.

1. Click ▶ *Format* ❯ *Theme* ◀.
2. From the *Theme* list in the left panel, select a theme.
   The *Preview* area shows how a small sample dashboard would look with that theme.
3. Click *OK.*
   The model is changed to use the styles and properties for the selected theme.

The new theme applies to all existing components and all new components. You can select individual components and change their properties.

> ℹ **Note**
>
> You can also set your model's theme by clicking *Themes* (🔳) on the *Themes* toolbar.

5.2.2 To set the default theme

If you frequently use the same theme when creating models, you can set it as the default so it is used each time you create a new model.

1. Click ▶ *File* ❯ *Preferences* ◀.
2. In the *Preferences* dialog box, click *Document*.
3. From the *Default Theme* list, select a theme.
4. Click *OK*.

5.3 Using color schemes

Color schemes (formerly called Styles) alter the color palette used for components in your model. You can apply a color scheme to your entire model or only to specific components. The software provides numerous predefined

color schemes that you can apply to your models, but you also can create and save your own color schemes. Color schemes do not affect the colors of fonts, text formatting, or alerts.

> **i Note**
>
> Color schemes are not transferred with XLF files. If the color scheme used by a model is not available at design time (for example, if you open your model on another computer), the custom color scheme continues to be used by the existing components but is not available for new components - you must copy the color scheme file from the original computer. See *To create a custom color scheme* [page 169] for the default filepath where color scheme files are saved.

5.3.1 To apply a color scheme

You can change the way your model looks by changing its color scheme. You can apply the new color scheme to new components only or choose to apply the change to existing components as well.

1. Open a model.
2. Click ▶ *Format* ❯ *Color Scheme* ◣.
3. In the *Color Schemes* dialog box, select the color scheme you want to apply.

   > **i Note**
   >
   > To change the color scheme back to its original settings, select *Current Theme Colors*.

4. If you want the color change to apply to components already in the model, click *Apply to Existing Components*. Otherwise, the color scheme is only applied to the canvas and any new components you place on the canvas.
5. Click *OK*.

The color scheme for the model changes. If you selected the *Apply to Existing Components* option, the colors of existing components change. Otherwise, the new color scheme applies when you place new components on the canvas.

5.3.2 To set the default color scheme

If you frequently use the same color scheme when creating models, you can set it as the default so it is used each time you create a new model.

1. Click ▶ *File* ❯ *Preferences* ◣.
2. In the *Preferences* dialog box, click *Document*.
3. From the *Default Color Scheme* list, select a color scheme.
4. Click *OK*.

5.3.3 To create a custom color scheme

You can create custom color schemes by modifying the parameters of an existing color scheme.

When you create a custom color scheme, it is saved to the following locations:

- In Windows XP: **<<Windows user>>**`\Application Data\XcelsiuscustomThemes\`
- In Windows 7: **<<Windows user>>**`\AppData\Roaming\\XcelsiuscustomThemes\`

where **<<windows user>>** is the path to the file location for the Windows user.

To share the color scheme with other work stations, copy the custom color scheme XML file to the same file location on the other system.

1. Click ▶ *Format* ▶ *Color Scheme* ◢.
   The *Color Schemes* dialog box opens.
2. In the *Color Schemes* dialog box, select a color scheme to modify and click *Create New*.
3. In the *Color Schemes* dialog box, set the following options:

Option	Description
Name	Type a name for the new color scheme.
Color bar	Click the color you want to change and, from the color selector, click the color you want to change it to.
Auto Match Colors	If you want the software to select a color palette based on a selected color, select the color you want to base the palette on and select this option. Move the slider to have the software adjust the palette to be more like the selected color or less like it.
Advanced	If you want to set the color for specific elements in components, click *Advanced*. In the *Advanced* area, click the tabs to view the elements for each component. Click the color box beside each element name and select the color you want to apply to that element.

4. Click *Save*.

The color scheme is saved, and is applied to the model. If you selected the *Apply to Existing Components* option, the colors of existing components change. Otherwise, the new color scheme is only applied to the canvas and any new components you place on the canvas.

> **ⓘ Note**
>
> You can also create a new color scheme by clicking *Colors* (▣) on the *Themes* toolbar and selecting *Create New Color Scheme*.

5.4 Previewing models

As you are designing your model, you may want to experiment with different combinations of components, colors, and layouts, and explore how the components interact with the data. When you preview a model, you see the dashboard as it would look if you exported it.

> **ⓘ Note**

If your model contains an SAP NetWeaver BW connection, you can preview the model configuration and test basic workflow, but the data from the SAP NetWeaver BW queries and query view results are not available. To preview a model with data from the connected SAP system, see *To preview Dashboards models with SAP NetWeaver BW data* [page 257].

5.4.1 To preview models

1. Open the model you want to preview.
2. Click ▶ *File* ❯ *Preview* ◢ and click one of the following:

   ○ *Desktop (Flash)*: how the model would look if viewed on a computer.
   ○ *Mobile (Fit to Screen)*: how the model would look if viewed on a mobile device with the *Fit to Screen* option selected. If the canvas size is not 1024x768, it will be scaled to fit the mobile device's resolution.
   ○ *Mobile (Original Size)*: how the model would look if viewed on a mobile device with the *Original Size* option selected. If the canvas size is larger than the mobile device's resolution, scrolling is supported.

   > ℹ **Note**
   >
   > You can also preview the model by clicking the drop-down arrow beside *Preview* (▣) on the *Standard* toolbar.

The software generates a preview of the model and displays it on the screen. This fully interactive preview lets you see the model as it would look if you exported it.

To close the preview and return to the working version of the model, click ▶ *File* ❯ *Preview* ◢ and click the preview type you selected.

> ℹ **Note**
>
> You can also close the preview by clicking *Preview* (▣) on the *Standard* toolbar.

Related Links
Exporting models [page 260]

5.4.2 To export model previews

If you would like to export an SWF of your model to view in a browser, you can export a preview of the model.

1. Open the model on the canvas.
2. Click ▶ *File* ❯ *Export Preview* ◢.

A progress bar opens to show the progress of the generation of the SWF for preview, and then a browser window launches with the preview model.

5.4.3 To take snapshots of models

While in *Preview* mode, you can interact with your model. If you find a view that you would like to keep, you can take a snapshot of the model in its changed state. A snapshot generates a SWF file and embeds it in a file format of your choice.

> **i Note**
>
> Since the snapshot feature uses the data in the embedded spreadsheet only, the generated SWF file may not reflect all changes you made while in *Preview* mode. If you want complete data retention, you should update the embedded spreadsheet to include the desired values, then export the model. For more information, see *Exporting models* [page 260].

1. On the toolbar, click *Preview* (▣).
   The software generates a SWF file for your model.

2. Interact with the model and when you want to take a snapshot, click ▶ *File* ❯ *Snapshot* ▎ and select one of the following options:

Option	Description
Excel spreadsheet	Generates an Excel spreadsheet (XLS) with the current data values. This option is best suited for diagnosing models that are not performing as desired.
Macromedia Flash (SWF)	Generates a Flash (SWF) file showing the current state of the components.
HTML	Generates an HTML file and a Flash (SWF) file that show the current state of the components.
PDF (Acrobat 6 or 9)	Generates an Adobe PDF file showing the current state of the components.
PowerPoint slide	Generates a Microsoft PowerPoint file with one slide that contains a Flash (SWF) file that shows the current state of the components.
Outlook	Opens a Microsoft Outlook email and attaches the Flash (SWF) file showing the current state of the components to a new message.

   The *Save As* dialog box opens.

3. Enter the location and filename for the snapshot and click *Save*.

There are limitations to export options based on product version. The following table summarizes snapshot option availability:

Snapshot option	SAP Crystal Presentation Design	SAP Crystal Dashboards, personal edition	SAP Crystal Dashboards, departmental edition	SAP BusinessObjects Dashboards
Excel spreadsheet	☑	☑	☑	☑
Macromedia Flash		☑	☑	☑
HTML			☑	☑
PDF (Acrobat 6 or 9)	☑	☑	☑	☑

Snapshot option	SAP Crystal Presentation Design	SAP Crystal Dashboards, personal edition	SAP Crystal Dashboards, departmental edition	SAP BusinessObjects Dashboards
PowerPoint slide	☑	☑	☑	☑
Outlook			☑	☑

6   Working with Data Sources

The software provides the following categories of data sources to connect to your components:

- The embedded spreadsheet
  This is best suited for small- to medium-sized business users who have most, if not all, of their business data in Excel spreadsheets. The embedded spreadsheet is available in all versions of Dashboards.
- External data connections
  These are best suited for small- to medium-sized business users who need a dynamic data source; these connections provide up-to-date data every time the model runs. Many of these connections are available in all versions of Dashboards - see the individual sections for more information.
- BEx and universe queries
  These are best suited for businesses running the BI platform. Queries are used to select specific result objects from the chosen data source, and, like external data connections, provide up-to-date data every time the model runs. However, you can link components to queries directly, without inserting data into the embedded spreadsheet. Queries are only available in SAP BusinessObjects Dashboards.

Related Links

The embedded spreadsheet [page 173]

External data connections [page 185]

BEx and universe queries [page 214]

6.1   Understanding the embedded spreadsheet

The embedded spreadsheet is a functional Excel spreadsheet that you can work with in the same way as you would in Excel. To associate your models with specific data, you can link components to cells in the embedded spreadsheet.

> ℹ **Note**
>
> The embedded spreadsheet does not support all Excel functions. It does not support macros, conditional formatting, linking to other external spreadsheets, and some formula functions. For a full list of supported functions, see *Supported Excel functions* [page 178].

The embedded spreadsheet includes the Excel toolbars. If you use Excel 2003, the embedded spreadsheet shows your default toolbars. Buttons that are not related to working with data are disabled: the Save, Open, and Print options are handled by Dashboards and are disabled on the embedded spreadsheet toolbar. If you use Excel 2007, the embedded spreadsheet displays the ribbon that appears at the top of the spreadsheet in. Any buttons or menus not related to working with data are disabled.

When saved locally, Dashboards uses the regional formatting settings defined on your computer, such as the settings for numerical separators, time separator, and date separator. To change the regional formatting settings, open the *Windows Control Panel*, double-click *Regional and Language Options*, adjust the settings, and click *OK*.

In SAP BusinessObjects Dashboards, if you save a model to the BI platform, the regional formatting settings are determined by the users' viewing locale. For more information about using regional formatting settings with the BI platform, see *Translating and globalizing models on the BI platform* [page 248].

While you are working on your model, you can display the embedded spreadsheet below the canvas. This arrangement allows you to see the data and bind components' input and output to cells on the spreadsheet. The embedded spreadsheet is completely independent from Excel: you can add, modify, and delete data directly within the embedded spreadsheet without having to import or re-import from Excel.

To insert data into the embedded spreadsheet, you can:

- Import a spreadsheet from Excel.
- Copy and paste data from another spreadsheet to the embedded spreadsheet.
- Manually enter data into the embedded spreadsheet.
- Link a query's result objects to cells in the embedded spreadsheet. This functionality is only available in Dashboards.

6.1.1   Preparing Excel spreadsheets

Before you import an Excel spreadsheet, make sure the spreadsheet provides a good structure for achieving the best performance for models. Consider the following guidelines:

- Use only supported Excel functions
  For a list of supported Excel functions, see *Supported Excel functions* [page 178].
- Use computationally intensive functions with small data sets only
  While the following Excel functions are supported in Dashboards, they can negatively affect the model's performance if used with larger data sets (over 100 rows):
  - SUMIF
  - COUNTIF
  - HLOOKUP
  - VLOOKUP
- Organize your data in a logical fashion
  To make your spreadsheet easier to understand, lay out your data in a logical fashion. Group related items together and use colors, labels, and borders to describe the structure and function of cells.
- Use colors, labels, and borders to identify data types (input and output)
  To make it easier to maintain your visual models, use colors, labels and borders to identify cells or ranges of cells in the spreadsheet and to describe their use. Adding a legend that describes what the colors represent helps maintain consistency and usability of the model.
- Place frequently used data and logic at the top of the spreadsheet
  To make it easier to select the data you want to bind to components and minimize the amount of scrolling required, place frequently used or common data and logic at the top-left of the tab in your spreadsheet.
- Use multiple tabs
  If your spreadsheet becomes large and requires a lot of scrolling to see your logic or data, consider separating data into multiple tabs (or worksheets).

6.1.2   To set preferences for the embedded spreadsheet

1. Click ▶ *File* ▶ *Preferences* ◢.

2. In the *Preferences* dialog box, from the list on the left, click *Excel Options*.
3. In the *Excel Options* area, set the following options:

Option	Description
Live Office Compatibility	Allows you to work with Live Office enabled spreadsheets within Dashboards. Enabling this option may affect the performance of other Microsoft Office programs. If this option is not enabled, you must set up and work with Live Office in a spreadsheet outside of Dashboards and then import the spreadsheet into Dashboards. Live Office is only available in SAP BusinessObjects Dashboards.
Maximum Number Of Rows	Use this option to set the maximum number of rows in the embedded spreadsheet that can be selected and bound to components. The software supports an unlimited number of rows in a range selection; however, linking components to large range selections can affect the performance of models. By default, you can select a maximum of 512 rows in a range.
Optimize Spreadsheet	Use this option to optimize the spreadsheet for runtime. When this option is enabled, the software calculates formulas in the model when you export the model and saves them with the SWF. It takes longer to generate the SWF, but since the calculations are already complete, the file performs better at runtime.
Ignore Excel Formula Errors	Use this option to avoid errors appearing in models because of formula errors in the spreadsheet. When selected, any cell that contains a formula error is treated as a blank cell. **i Note** If formula errors are ignored, functions that refer to cells with errors, such as ISERROR, will not work correctly in the embedded spreadsheet.

4. Click *OK*.

6.1.3 Synchronizing data between Excel and the embedded spreadsheet

The embedded spreadsheet is not connected to the source data imported from an Excel spreadsheet: changes made to the data in one location are not automatically updated in the other location. If you want to maintain synchronized copies of the data in the embedded spreadsheet and Excel, you can export the modified data back to Excel or copy and paste the changed cells to the source spreadsheet.

6.1.3.1 To import Excel spreadsheets

If you want to use an existing Excel spreadsheet as the source data for your model, you can import the spreadsheet into the embedded spreadsheet.

⚠ Caution

Importing an Excel spreadsheet overwrites any data currently in the embedded spreadsheet. If components have already been bound to cells in the embedded spreadsheet, when you import, the cell binding locations remain the same, but the data may change.

> **i Note**
>
> When the Excel file is imported the data is copied and is no longer connected to the original data. Editing the original source file will not change the data imported into the embedded spreadsheet.

1. Click ▶ *Data* ▶ *Import* ◣. You can also click *Import Spreadsheet* (📄) on the *Standard* toolbar.
2. In the warning dialog box, click *Yes*.
3. Select the spreadsheet you want to import and click *Open.*

6.1.3.2   To copy data between Excel and the embedded spreadsheet

If you want to add data from a Microsoft Excel spreadsheet to the embedded spreadsheet without importing the entire spreadsheet, you can copy the data you want in Excel and paste it into the embedded spreadsheet.

You can also open two instances of Dashboards and copy and paste data from the embedded spreadsheet in one instance of the software to other.

> **i Note**
>
> When you copy cells in an Excel spreadsheet, you can paste the values and formulas into the embedded spreadsheet, but any conditional formatting is not retained.

1. Open Dashboards on your desktop and load the model you want to add data to.
2. Open Excel or another instance of Dashboards on your desktop and load the spreadsheet or model you want to copy data from.
3. Select the cells you want to copy and press `Ctrl` + `C`.
4. In the embedded spreadsheet, select a cell and press `Ctrl` + `V`. Note that the cell you select will be the top left cell of the selection you copied.

6.1.4   Linking components and spreadsheet data

When you create models, you can bind or link component properties to specific cells in the embedded spreadsheet. These cells can contain numerical data, Excel formulas, or text (for example, labels or product names). In addition, certain components can output data to cells, and thus should be linked to empty cells.

If you add an external data connection, the external connection can also be configured to insert data into cells on the embedded spreadsheet. Components can then use data from those cells, or the aforementioned output cells, when the model runs.

> **i Note**

> Queries can link directly to most components without the need to update the embedded spreadsheet.

Once a component is bound to cells in the embedded spreadsheet, you can import updated data or modify the embedded spreadsheet and the changes are reflected by the component, except in the following situations:

- If you move a range after it is bound to one or more components, (for example, if you are adding a heading row on the spreadsheet) the software retains the new information and changes the reference to the new range. However, if you move only part of the range, the cell reference is not updated for the bound cells and you must open the *Properties* panel for the component(s) and re-bind it to the new location.

  > **i Note**
  >
  > After you move a range, insert cells to expand a range, or delete cells to decrease a range, the *Properties* panel does not display the changed range values; however, when you preview or publish the model, the component does use the new range values.

- If you change the sort order for data that is bound to components, the component updates its display when you preview or export the model. To force the component to display the new sort order in design mode, re-bind the component to the sorted cells.

If the data in your spreadsheet changes frequently, but the structure of the spreadsheet stays the same, you can export models based on the data in an external Excel spreadsheet. For example, if you need to generate the same model for multiple customers using the same spreadsheet layout with different values, then after you create your model and link the components to cells in the embedded spreadsheet, you can set an option to export the model using the external Excel file as the data source. You can then update the data in the external file for each customer and generate their model without having to re-upload the spreadsheet and re-link the components. For more information, see *To generate a SWF file using an external spreadsheet* [page 177].

6.1.5   To generate a SWF file using an external spreadsheet

If you need to generate the same model using the same spreadsheet structure but different values, you can export the model using an external spreadsheet.

> **i Note**
>
> This option does not update the data in the embedded spreadsheet: it only retrieves the data from the external spreadsheet to generate the SWF file. If you preview the model or export to any other format, Dashboards uses the data in the embedded spreadsheet. Also note that the structure of the external spreadsheet must be identical to the structure of the embedded spreadsheet. If the two spreadsheets are different, your model may not function properly since cells that components are bound to may be empty or contain incorrect values.

1. Click ▶ *File* ▶ *Export Settings* ◀.
   The *Export Settings* dialog box opens.
2. Select the *Use Another Excel File* option and in the list, click the file folder and select the spreadsheet file to use.

   > **i Note**
   >
   > The components in the model maintain their cell bindings, so ensure that the Excel file you are using has an identical structure.

3. Click *OK*.

4. Click ▶ *File* ❯ *Export* ❯ *Flash (SWF)* ◢.
    The *Save As* dialog box opens.

5. Enter a path and name for the SWF file and click *OK.*

6.1.6   To export data to Excel

If you want to view the data in the embedded spreadsheet in Excel, or synchronize the data between the embedded spreadsheet and the Excel source file, you can save the data in Excel format so you can open it using Microsoft Excel. This feature is available in all versions of Dashboards.

1. Click ▶ *Data* ❯ *Export* ◢.
    The *Save As* dialog box opens.

2. Enter a path and name for the Excel file and click *OK*.

6.1.7   Supported Excel functions

The software supports the following Excel functions:

A

*   ABS
*   ACOS
*   ACOSH
*   AND
*   ASIN
*   ASINH
*   ASSIGN
*   ATAN
*   ATAN2
*   ATANH
*   AVEDEV
*   AVERAGE
*   AVERAGEA
*   AVERAGEIF

B

- BETADIST

C

- CEILING
- CHOOSE
- COMBIN
- CONCATENATE
- COS
- COSH
- COUNT
- COUNTA
- COUNTIF

> ℹ **Note**
>
> If used with large data sets, this function could affect performance.

D

- DATE
- DATEVALUE
- DAVERAGE
- DAY
- DAYS360
- DB
- DCOUNT
- DCOUNTA
- DDB
- DEGREES
- DEVSQ
- DGET
- DIVIDE
- DMAX
- DMIN
- DOLLAR
- DPRODUCT
- DSTDEV
- DSTDEVP

- DSUM
- DVAR
- DVARP

E

- EDATE
- EFFECT
- EOMONTH
- EVEN
- EXACT
- EXP
- EXPONDIST

G

- GE
- GEOMEAN
- GT

H

- HARMEAN
- HLOOKUP

  > ℹ **Note**
  >
  > If used with large data sets, this function could affect performance.

- HOUR

I

- IF
- INDEX
- INT
- INTERCEPT
- IPMT
- IRR

- ISBLANK
- ISERR
- ISERROR
- ISEVEN
- ISLOGICAL
- ISNA
- ISNONTEXT
- ISNUMBER
- ISODD
- ISTEXT

K

- KURT

L

- LARGE
- LE
- LEFT
- LEN
- LN
- LOG
- LOG10
- LOOKUP
- LOWER

M

- MATCH
- MAX
- MEDIAN
- MID
- MIN
- MINUS
- MINUTE
- MIRR
- MOD
- MODE

- MONTH

N

- N
- NE
- NETWORKDAYS
- NORMDIST
- NORMINV
- NORMSINV
- NOT
- NOW
- NPER
- NPV

O

- ODD
- OFFSET
- OR

P

- PI
- PMT
- POWER
- PPMT
- PRODUCT
- PV

Q

- QUARTILE
- QUOTIENT

R

- RADIANS
- RAND
- RANGE_COLON
- RANK
- RATE
- REPLACE
- REPT
- RIGHT
- ROUND
- ROUNDDOWN
- ROUNDUP

S

- SECOND
- SIGN
- SIN
- SINH
- SLN
- SMALL
- SQRT
- STANDARDIZE
- STDEV
- SUM
- SUMIF

> ⓘ Note
>
> If used with large data sets, this function could affect performance.

- SUMPRODUCT
- SUMSQ
- SUMX2MY2
- SUMX2PY2
- SUMXMY2
- SYD

T

- TAN

- TANH
- TEXT
- TIME
- TIMEVALUE
- TODAY
- TRUE
- TRUNC
- TYPE

U

- UPPER

V

- VALUE
- VAR
- VDB
- VLOOKUP

> **i** Note
>
> If used with large data sets, this function could affect performance.

W

- WEEKDAY
- WEEKNUM
- WORKDAY

Y

- YEAR
- YEARFRAC

6.2 Using data connections and queries

In Dashboards, you can connect your models to an external data source. When the model runs, the data is updated from the external source so the model is based on current information rather than information that was available when the model was created.

There are two ways to connect models to external data sources:

* Configure an external data connection using the *Data Manager*.
* Create a query from a BI platform data source.

> **i Note**
>
> External data sources are not supported in Presentation Design, and only certain external data sources are available in Crystal Dashboards. For a detailed list of which product versions support which connections, see *Product editions* [page 6].

6.2.1 Managing data connections

The software supports several types of XML-compliant data connections. All data connections are added and managed through the *Data Manager*. To open the *Data Manager*, click ▶ *Data* ❯ *Connections* ◢.

> **i Note**
>
> The *Data Manager* is not available (since external data sources are not supported) in Presentation Design.

> **i Note**
>
> SAP Crystal Dashboards and SAP BusinessObjects Dashboards have maximum supported connections per model. For more information, see *Maximum number of data connections* [page 188].

The software supports the following types of connections:

Connection Type	Description
Web service query (Query as a Web Service)	*Query as a Web Service* is a tool that allows you to define a query from a universe and then publish that query as a standalone web service. You then create a *Web service query* connection to connect your model to that query. To create *Web service queries*, see the *Query as a Web Service* guide available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ❯ *All Products* ◢.
Web Service Connection	A *Web Service Connection* allows you to connect to an external data source using the HTTP protocol. When added to a model, this connection receives the data provided by the web service in real-time.

Connection Type	Description
SAP NetWeaver BW Connection	The *SAP NetWeaver BW Connection* retrieves query data from the Business Warehouse repository. An *SAP NetWeaver BW Connection* is considered an SAP data source: to configure it, see *To configure SAP NetWeaver BW connections* [page 252]. **i Note** If your data is deployed on the SAP BusinessObjects Business Intelligence platform, it is recommended that you create a query using the *Query Browser* instead of connecting through the *Data Manager*. For more information, see *To add or edit queries* [page 218].
XML Data	An *XML Data* connection allows you to connect to an external source using the HTTP protocol. When added to a model, this connection can send data to an external source to request specific information; for example, if a date is sent to a server script, the script could return the sales totals for that date.
Flash Variables	A *Flash Variables* connection allows you to pass data or variables from an HTML source to the _root level of a Flash movie. When you export your SWF as HTML, the Flash Variables are generated in the OBJECT and EMBED tags.
Portal Data	A *Portal Data* connection allows you to send and receive data from the SAP BusinessObjects BI workspace, IBM WebSphere, or Microsoft SharePoint. The connection can be set up to receive data from another web part (consumer), to provide data to another web part (provider), or to allow users to define the parameter values at runtime.
Crystal Reports Data Consumer	A *Crystal Reports Data Consumer* connection allows you to retrieve information from Crystal reports, and embeds metadata in the SWF file that Crystal Reports uses to bind Crystal Reports data with the Dashboards model.
FS Command	An *FS Command* connection allows you to include Flash FS Commands in your model. *FS Commands* allow Flash to run JavaScript statements in a web browser. **i Note** In Dashboards, *External Interface Connections* are the preferred connection for transferring data using JavaScript in a web browser. If you use FS Commands in your model, consider the Adobe Flash security restrictions. For more information, see *FS Command restrictions* [page 266].
LCDS Connections	A *LiveCycle Data Services (LCDS) Connection* allows you to stream real-time data through Adobe LiveCycle Data

Connection Type	Description
	Services. When added to a model, this connection receives the data provided by the services in real-time.
External Interface Connection	An *External Interface Connection* allows you to expose selected data ranges in the embedded spreadsheet to transfer data to the exported SWF file using JavaScript when the SWF file is embedded in an HTML web page.
Excel XML Maps	When you import an Excel spreadsheet that has an XML map, Dashboards retains the defined structure of the XML file and how it maps to the data within the spreadsheet. When you include Excel XML Map connections in a model, the data can be dynamically refreshed from the XML file each time the model loads, at designated intervals, or when the user clicks a refresh button.
Live Office Connections	A *Live Office Connection* allows you to connect your model to a *Live Office*-enabled spreadsheet; these spreadsheets are constructed from managed Crystal Reports or Web Intelligence data. When added to a model, this connection receives the data provided by the Live Office-enabled spreadsheet in real-time. A Live Office connection is considered an SAP data source; to configure a Live Office data connection, see *Using Live Office Data Connections* [page 237].
Web Dynpro Flash Island	A Web Dynpro connection allows you to embed your model in a Web Dynpro application as a Flash Island. The connection between your model and the WD application is bidirectional: changes made in the WD application context will update the model, and changes in the model will update the WD application.

Related Links

6.2.1.1   To add an external data connection

If you want the data in a model to update based on the data in an external source, use the *Data Manager* to add an external data connection.

> **i Note**
>
> If a model is connected to an external data source, then when the model runs, Adobe security restrictions apply. When the model is run locally, the SWF file or the host application file must be made trusted. If the file is run on a web server, there must be a cross-domain policy file. For more information, see *Adobe Flash Player Restrictions*.

1. Click ▶ *Data* ▶ *Connections* ◀ .
   The *Data Manager* dialog box opens.

2. In the *Data Manager* dialog box, click *Add* and select a connection type from the list.

3. Configure the connection. For more information about configuring data connections, see the instructions to configure specific connection types in this guide.

Certain connection types have a maximum number of supported connections per model.

6.2.1.1.1   Maximum number of data connections

This table lists the maximum number of connections per data connection type in each version of the product.

Data connection	Maximum connections per model in Crystal Dashboards, personal edition	Maximum connections per model in Crystal Dashboards, departmental edition	Maximum connections per model in SAP BusinessObjects Dashboards
Web service query (Query as a Web Service)	Not available	Not available	Unlimited
Web Service Connection	2 ℹ Note A web service connection is defined as a single method on a web service; calling two separate methods on the same web service counts as two connections.	Unlimited	Unlimited
SAP NetWeaver BW Connection	Not available	Not available	Unlimited
XML Data	2	Unlimited	Unlimited
Flash Variables	1	1	1
Portal Data	Not available	1	1
Crystal Reports Data Consumer	1	1	1
FS Command	Unlimited	Unlimited	Unlimited
LCDS Connections	Not available	Unlimited	Unlimited
External Interface Connection	1	1	1
Excel XML Maps	Unlimited	Unlimited	Unlimited
Live Office Connections	Not available	1	1
Web Dynpro Flash Island	Not available	1	1

6.2.1.2   To remove data connections

You can use the *Data Manager* to delete data connections from model.

1. Click ▶ *Data* ❯ *Connections* ▶.
    The *Data Manager* dialog box opens.

2. Select the connection you want to remove and click *Delete* (✖).

6.2.1.3   To configure Web service query (Query as a Web Service) connections

Before you configure a Web service query (Query as a Web Service) connection in SAP BusinessObjects Dashboards, you must create a web service using Web service query and have the URL for accessing the web service. For more information about the Web service query, see the *Query as a Web Service* guide, available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ❯ *All Products* ▶.

> **ℹ Note**
>
> This data connection is only available in SAP BusinessObjects Dashboards.

Use a Web service query (Query as a Web Service) connection to create a live connection to data in the SAP BusinessObjects Business Intelligence platform.

> **⚠ Caution**
>
> If you include external data connections in models, security restrictions in Adobe Flash Player may prevent users from accessing the data source. For more information, see *Adobe Flash Player Security Restrictions*.

1. Click ▶ *Data* ❯ *Connections* ▶.
    The *Data Manager* dialog box opens.

2. In the *Data Manager* dialog box, click *Add* and select *Web service query (Query as a Web Service)*.

3. On the *Definition* tab, set the following options:

Option	Description
Name	A name for the connection.
WSDL URL	Type the URL for the WSDL generated by the Web service query or select it from the list. Then, click *Import*.
	The model is linked to the Web service query. The Method, Web service URL, and Input, and Output values defined in the Web service query are displayed in the associated fields on the page.
Method	When you import the WSDL, this list displays the methods defined in the web service. Select the method you want to use for this connection. The Web service query may generate the following methods:
	For more information about these methods and the parameters available, see the *Web Services Administrator Guide*, available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ❯ *All Products* ▶.

Option	Description

Method	Description
RunQueryAsAService	This web service call is generated for index unaware prompts. This method requires an Enter_Year_As string as the prompt value for year.
RunQueryAsAServiceEx	This web service call is generated for index aware prompts. The input and output parameter sets are the same as for RunQueryAsAService method except for the input Prompt for Year values. This method requires an instance of LovValueIndex for year.
Valueof_	This method is generated only when the Web service query encloses at least one prompt and is intended to provide a list of values for the prompt it is named after.

Option	Description
Web Service URL	When you import the WSDL, this box displays the Web service URL defined in the Web service query. ➡ **Tip** If you want to control the Web service URL value through Flash Variables, you can copy the URL provided by the WSDL to a cell in the embedded spreadsheet. Then click the *Cell Selector* ( ) to link the *Web service URL* box to that cell. To point to a different server, you can then set up Flash Variables to update the cell with a different URL.
Input Values	When you import the WSDL, the required inputs and prompts defined for the web service are displayed in this list. To set a value for each prompt, select the Input object from the list then set the *Read From* option to map it to your model. ℹ **Note** Web service query Input objects include a user ID and password for the BI platform. Dashboards will generate a logon screen to prompt users for these values as required. The values and behavior of these prompts is based on the Authentication configuration defined in the Web service query. Unless you want to set these items to specific values, you generally do not need to specify values or map them to your model on this page. ⚠ **Caution** If you enter specific username and password values here, those values are stored in the Dashboards object without encryption. For security reasons, it is best not to enter specific values for these input items.

Option	Description
Read From	If the Input Values list includes prompts defined in the Web service query, select the Input item and then click the *Cell Selector* ( ) to select a cell from the embedded spreadsheet where the selected object will retrieve its value.
Output Values	When you import the WSDL, the output parameters for the Results objects in the Web Service query are displayed in this list in a tree structure, along with some additional parameters created by the Web service query. When using a Web service query, the result object's output is stored in the table node under the rows node. ⓘ Note To see the data type for results objects, move the mouse pointer over the object. To add the result values for a single object to your model, select an Output object from the list, then set the *Insert In* option to map it to your model. You can also map all objects in an enclosing folder in a single step. Click the enclosing folder to see the number of columns required for the objects within that folder. Then click the *Cell Selector* ( ) and select a range in the spreadsheet that is large enough to contain the specified number of columns. The number of rows required depends on the amount of data you expect to receive.
Insert In	For the selected Output value, click the *Cell Selector* ( ) to select a cell from the embedded spreadsheet to store the values retrieved by the Web service query. ⓘ Note By default, the software allows up to 512 rows to be retrieved and inserted into the embedded spreadsheet. To increase the number of rows allowed, see *To set preferences for the embedded spreadsheet*.

4. Click the *Usage* tab and set the refresh and message options. For more information, see *To set refresh options* and *To create load and idle messages*.

Once the data is available in Dashboards, you must map components to the data. Mapping a component to the Web service query is the same as mapping it to the embedded spreadsheet.

6.2.1.4 To configure Web Service Connections

Use a *Web Service Connection* to add a live data connection to a model.

➡ Tip

If you are connecting to SAP BusinessObjects Business Intelligence platform using a web service, use a *Web service query (Query as a Web Service)* connection instead of a generic web service connection. For more information on creating a *Web service query* connection, see *To configure Web service query (Query as a Web Service) connections* [page 189].

⚠ Caution

If you include external data connections in models, security restrictions in Adobe Flash Player may prevent users from accessing the data source. For more information, see *Adobe Flash Player Security Restrictions*.

1. Click ▶ *Data* ▶ *Connections* ◣.
   The *Data Manager* dialog box opens.
2. In the *Data Manager* dialog box, click *Add* and select *Web Service Connection*.
3. On the Definition tab, set the following options:

Option	Description
Name	Type a name for the connection.
WSDL URL	Type the URL for the WSDL generated by the web service, or select it from the list. Then, click *Import*.
	The model is linked to the web service and the methods, web service URL, and Input and Output values available in the Web Service are added to those lists.
Method	The available methods are defined by the WSDL specified in the WSDL URL box. If there is more than one method, select the method that you want to use for this connection.
Web Service URL	The available URLs are defined by the WSDL file specified in the WSDL URL box. Select the URL that you want to use for this connection. When the model runs, it will use this URL, not the WSDL URL.
	➡ **Tip** If you want to control the web service URL value through Flash Variables, you can copy the URL provided by the WSDL to a cell in the embedded spreadsheet. Then click the *Cell Selector* (🔲) to link the *Web Service URL* box to that cell. To point to a different server, you can then set up Flash Variables to update the cell with a different URL.
Input Values	When you import the WSDL, the required inputs defined for the web service are displayed in this list. To set a value for each item, select the Input object from the list then set the *Read From* option to map it to your model.
Read From	If the input values list includes items that require input, select the Input item and then click the *Cell Selector* (🔲) to select a cell from the embedded spreadsheet where the selected object will retrieve its value.
Output Values	When you import the WSDL, the output parameters for the results objects in the web service are displayed in this list.
	ℹ **Note** To see the data type for results objects, move the mouse pointer over the object.
	To add the result values for a single object to your model, select an output object from the list, then set the *Insert In* option to map it to your model.
	You can also map all objects in an enclosing folder in a single step. Click the enclosing folder to see the number of columns required for the objects within that folder. Then click the *Cell Selector* (🔲) and select a range in the spreadsheet that is large enough to contain the specified number of columns. The number of rows required depends on the amount of date you expect to receive.
	The available values are set by the WSDL file. Select an output object from the list then set the *Insert In* option to map it to your model.
	ℹ **Note**

Option	Description
	Repeat for each value you want to use in the model.
Insert In	For the selected output value, click the *Cell Selector* ( ) to select a cell from the embedded spreadsheet to store the values retrieved by the web service. **i Note** By default, the software allows up to 512 rows to be retrieved and inserted into the embedded spreadsheet. To increase the number of rows allowed, see *To set preferences for the embedded spreadsheet*.

4. Click the *Usage* tab and set the refresh and message options. For more information, see *To set refresh options* and *To create load and idle messages*.

5. If your Web Service connection requires information to be passed in the SOAP header, click the *Advanced* tab.

   Type the information in HTML format or click the *Cell Selector* ( ) to select cells in the embedded spreadsheet that contain the information.

   > **i Note**
   >
   > When you use the *Cell Selector* ( ), the <soap:Header> HTML tags are automatically added to the text values in the cell. For example, the selected cell would only need to contain "<username>joe</username> <password>1234</password>".

6.2.1.5   To configure XML Data connections

XML Data Connections allow you to refresh model data from an external source using HTTP and to send live data from models to external sources. For example, if a model has an XML data connection, users could use the model to submit information and collect data for forms. When you configure the XML data connection, you can specify how and when the data will be refreshed from the source.

> **⚠ Caution**
>
> If you include external data connections in models, security restrictions in Adobe Flash Player may prevent users from accessing the data source. For more information, see *Adobe Flash Player Security Restrictions*.

The XML Data connection sends and loads functionality simultaneously through one URL. The send ranges are sent to the URL prior to retrieving the load data. This lets you create scripts that examine the sent data and return data that corresponds to the sent values—much like a web request. For example, if a date was sent to a server script, the script could return the sales figures for that date.

Excel ranges are mapped to XML using a row-column structure. For example, a one-row, three-column range named "ExampleRange" would be represented in XML as:

```
<variable name="ExampleRange">
  <row>
    <column> Row1 Column1 value </column>
    <column> Row1 Column2 value </column>
```

```
      <column> Row1 Column3 value </column>
   </row>
</variable>
```

where "Row1 Column1 value" would be the actual value of the cell in the first row and first column of the range, and so on.

The entire group of ranges is encapsulated within <data> tags.

When the model loads, the XML is translated into Excel ranges using the reverse process.

> **i Note**
>
> XML Data connections support a simple row/column structure. If you need to access more complex XML data sources, use an XML Map connection.

1. Click ▌ *Data* ▶ *Connections* ▌.
   The *Data Manager* dialog box opens.
2. In the *Data Manager* dialog box, click *Add* and select *XML Data*.
3. On the *Definition* tab, set the following options:

Option	Description
Name	Type a name for the connection.
XML Data URL	Type the URL for the XML data or click the *Cell Selector* () to select cells from the embedded spreadsheet. > **i Note** > For hyperlinks, include "http://".
MIME Type	If more than one MIME type is available, select the type to use for this data.
Enable Load	Select this option to load (or receive) the XML data from the specified URL into the specified range when the connection is triggered. Select a range from the list or click *Add* () to add a new range and then set the Name and Range values for that range. Click *Remove* () to delete a range from the list. > **i Note** > Click *Preview XML* to see the structure of the XML that the model must have to receive data from the external data source.
Name	If you selected the *Enable Load* option, for each range in the list, type a name or use the *Cell Selector* () to select a name from the embedded spreadsheet.
Range	If you selected the *Enable Load* option, for each range in the list, type a range or use the *Cell Selector* () to select the range from the embedded spreadsheet where the retrieved data will be saved.
Enable Send	Select this option to send the XML data to the specified URL from the specified ranges when the connection is triggered. The translated ranges are sent in HTTP format using the POST method.

Option	Description
	Select a range from the list or click *Add* (⊞) to add a new range and then set the Name and Range values for that range. Click *Remove* (⊟) to delete a range from the list. ⓘ **Note** Click *Preview XML* to see the structure of the XML that the model must have to send data to the external data source.
Name	If you selected the *Enable Send* option, for each range in the list, type a name or use the *Cell Selector* (▣) to select a name from the embedded spreadsheet.
Range	If you selected the *Enable Send* option, for each range in the list, type a range or use the *Cell Selector* (▣) to select the range from the embedded spreadsheet.
Import Named Ranges	Select this option to import defined names from Excel.

4. Click the *Usage* tab and set the refresh and message options. For more information, see *To set refresh options* and *To create load and idle messages*.

6.2.1.6 To configure Flash variable connections

If you export your model to HTML, you can add a Flash Variable connection to pass values into your model as it loads. When you configure the Flash Variable connection, you specify a name and define where the values are stored in the spreadsheet.

You can use Flash Variables to send global session variables that you want your model to consume and re-use. For example, you can send session tokens, user IDs, Web Service URLs, and parameters passed from other embedded scripts or controls. Flash Variable connection data can also be used as a data source for other connections. For example, data retrieved from a Flash Variable connection can be used as the input values of a Query as a Web Service connection, or to answer a universe query prompt.

ⓘ **Note**

Flash Variables are not secure. Do not send sensitive or confidential information through Flash Variables, for example, passwords.

When you run the SWF in a web page, the Adobe Flash player enables it to consume data directly from the HTML. If you are dynamically generating your HTML page, you can pass variables into the SWF when it loads. You can also use JavaScript to dynamically change the variables during runtime using an External Interface connection, which allows other HTML embedded controls to pass data into the SWF without reloading the HTML page.

The Adobe Flash player also enables a SWF to consume data directly from defined parameters within a browser or path for which it is loaded. This method allows one SWF to pass parameters to another SWF that is loaded on a separate HTML page.

> ⚠️ **Caution**
>
> If you include external data connections in models, security restrictions in Adobe Flash Player may prevent users from accessing the data source. For more information, see *Adobe Flash Player Security Restrictions*.

1. Click ▶ *Data* ❯ *Connections* ◣.
   The *Data Manager* dialog box opens.
2. In the *Data Manager* dialog box, click *Add* and select *Flash Variables*.
3. On the *Definition* tab, set the following options:

Option	Description
Name	Type a name for the connection.
	Use only numbers, letters, and underline. Do not start the name with a number and do not include spaces in the name.
Variable Format	Select whether to send the variables to the SWF in XML or CSV format.
Variable Data	Select a range from the list or click *Add* (⊞) to add a new range and then set the Name and Range values for that range.
	Click *Remove* (⊟) to delete a range from the list.
Import Named Ranges button	Use this button to import named ranges from the spreadsheet.

6.2.1.6.1 Flash Variable restrictions in mobile dashboards

The following variables are not supported in mobile dashboards:

- CELogonToken
- CESerializedSession

6.2.1.7 Configuring Portal Data connections

Portal Data connections allows SWF-to-SWF communications when models are deployed in SAP BusinessObjects Business Intelligence platform, IBM WebSphere or Microsoft SharePoint environments. The portal data connection can be configured as a provider (sends data to another SWF portlet, also known as a web part in SharePoint) or as a consumer (receives data from another portlet). You can also set up the portal data connection to allow user input during run-time. For more information about deploying models in a portal environment, see the *Dashboards SharePoint Installation Guide* or the *Dashboards WebSphere Portlet User Guide* available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ❯ *All Products* ◣.

> ℹ️ **Note**
>
> Communication with third-party web parts is not supported.

> ⚠ **Caution**
>
> If you include external data connections in models, security restrictions in Adobe Flash Player may prevent users from accessing the data source. For more information, see *Adobe Flash Player Security Restrictions*.

6.2.1.7.1   To configure Portal Data connections

If you want a model to provide data to another portlet, set up the portal data connection as a provider. If you want it to receive data from another portlet, set up the portal data connection as a consumer. Only SAP BusinessObjects Dashboards portlets can consume information from a model that is using the portal provider connection.

1. Click ▶ *Data* ▶ *Connections* ◀.
2. In the *Data Manager* dialog box, click *Add* and select *Portal Data*.
3. On the *Definition* tab, set the following options:

Option	Description
Connection Type	To configure this connection to send data to another portlet, set this option to *Provider*. To configure it to receive data from another portlet, set it to *Consumer*.
Range Name	Type a name for the range. ℹ **Note** This name is displayed when you connect portlets together in portals. If you don't specify a name, the connection is not available for connecting portlets.
Range Type	Select from the following options to define the amount of data to pass: ○ Cell -- a single cell from another web part. ○ Row -- a single row with multiple columns. ○ Table -- multiple rows and columns. Cell and row are usually used for passing data between models. In SharePoint, tables are used to pass data from a Portal list.
Range	Type or click the *Cell Selector* (🔲) to select the area on the embedded spreadsheet where the incoming data will be retrieved from (provider) or written to (consumer).
Parameters	Define the parameter ranges that users can set from within the portal. To add a range, click *Add* (⊞), then select each range and set its name, range, and type values.
Name	Type a name for the parameter or click the *Cell Selector* (🔲) to link the name to a cell in the embedded spreadsheet. ℹ **Note** The entered name will appear in the *Property* dialog box for the model in SharePoint. If you do not enter a name, the parameter will not be listed as a parameter in SharePoint.

Option	Description
Range	Type the area in the spreadsheet where the parameters are set or click the *Cell Selector* () to select the range from the spreadsheet.
Type	Select one of the following parameter types: | Options	Description
Text	To set text-based data within the XLF, for example, chart titles.
Number	To set numeric-based data, for example, alert levels.
Check Box	To toggle the value of a cell between zero and one. This option can be useful for showing or hiding components using dynamic visibility.
Date	To define a date in the XLF. It sets the format to Date and uses a true date value within the XLF, for example, users can set date ranges for data access or for the amount of data to display.
List Box	To build user selection options into the parameters. This option allows you to specify set values that users can select to ensure their entry is compatible with the model. Text, Number, Check Box, and Date types do not restrict user entries.	
Entries	Available only when List Box is selected in the *Type* list. Set the values users are able to select from the list box. Click the *Cell Selector* () to select the values from the embedded spreadsheet.
Default Selection	Available only when List Box is selected in the *Type* list. Specify the values to use if the user doesn't make a selection in the list box.

4. If you are defining a provider connection, click the *Usage* tab to set the refresh options. For more information about setting refresh options, see *To set refresh options*.

6.2.1.7.2 To configure a Portal Data connection for user input

To define parameters that users can customize from within a portal, set up a portal data connection with the *Connection Type* set to "None".

1. Click ▶ *Data* ▶ *Connections* ▶.
2. In the *Data Manager* dialog box, click *Add* and select *Portal Data*.
3. On the *Definition* tab, set the following options:

Option	Description
Connection Type	To configure this connection for user input, set this option to "None".

Option	Description
Parameters	Define the parameter ranges that users can set from within the portal. To add a range, click *Add* (⊞), then select each range and set its name, range, and type values.
Name	Type a name for the parameter or click the *Cell Selector* ( ) to link the name to a cell in the embedded spreadsheet. The entered name will appear in the *Property* dialog box for the model in SharePoint. If you do not enter a name, the parameter will not be listed as a parameter in SharePoint.
Range	Type the area in the spreadsheet where the parameters are set or click the *Cell Selector* ( ) to select the range from the spreadsheet.
Type	Select one of the following parameter types: {{TABLE}}
Entries	Available only when List Box is selected in the *Type* list. Set the values users are able to select from the list box. Click the *Cell Selector* ( ) to select the values from the embedded spreadsheet.
Default Selection	Available only when List Box is selected in the *Type* list. Specify the values to use if the user doesn't make a selection in the list box.

Nested Type table:

Option	Description
Text	Sets text-based data within the XLF, for example, chart titles.
Number	Sets numeric-based data, for example, alert levels
Check Box	Toggles the value of a cell between zero and one. This option can be useful for showing or hiding components using dynamic visibility.
Date	Defines a date in the XLF. It sets the format to Date and uses a true date value within the XLF, for example, users can set date ranges for data access or for the amount of data to display.
List Box	Builds user selection options into the parameters. This option allows you to specify set values that users can select to ensure their entry is compatible with the model. Text, Number, Check Box, and Date types do not restrict user entries.

6.2.1.8 To configure Crystal Reports Data Consumer connections

If you want to embed a Dashboards model in Crystal Reports, and receive data from Crystal Reports for the model, you can add a *Crystal Reports Data Consumer* connection.

> ℹ **Note**
>
> Before you add a *Crystal Reports Data Consumer* connection, you need to set up the embedded spreadsheet and components in your model to correspond to the Crystal Reports data you will be connecting to.

1. Click ▶ *Data* ❯ *Connections* ◢.
2. In the *Data Manager* dialog box, click *Add* and select *Crystal Reports Data Consumer*.
3. On the Definition tab, set the following options:

Option	Description
Row Headers	Click the *Cell Selector* (▦) to select the row header labels from the embedded spreadsheet.
Column Headers	Click the *Cell Selector* (▦) to select the column header labels from the embedded spreadsheet.
Data	Click the *Cell Selector* (▦) to select the data range from the embedded spreadsheet.
Description	Type an optional description for the connection.

Export the model in Flash (SWF) format and embed the SWF in the Crystal Reports file. When you open the SWF file with the Flash Data Expert in Crystal Reports, a Dashboards tab will be available in the Flash Data Binding Expert so you can connect the SWF to the Crystal Reports data.

6.2.1.9   To configure FS Command connections

In Dashboards, External Interface connections are the preferred connection for transferring data in a web browser using JavaScript when the SWF is embedded in an HTML web page.

For detailed information about FS Command, see the Adobe Flash support site at *http://www.adobe.com/ support/flash/action_scripts/actionscript_dictionary/actionscript_dictionary372.html*

1. Click ▶ *Data* ❯ *Connections* ◢.
2. In the *Data Manager* dialog box, click *Add* and then select *FS Command*.
3. On the Definition tab, set the following options:

Option	Description
Name	Type a name for the connection.
Command	Type a command for the FS Command, or click the *Cell Selector* (▦) and select a command from the spreadsheet.
Parameters	Type a parameter value for the FS Command, or click the *Cell Selector* (▦) and select a value from the spreadsheet.

6.2.1.10   To configure LCDS Connections

To add an LCDS connection, the Dashboards LiveCycle Data Services (LCDS) must be installed on your system. For more information, see the *Dashboards LiveCycle Data Services Gateway Installation Guide* available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ❯ *All Products* ◢. In addition, a model and embedded spreadsheet must be set up and ready to connect to LCDS.

To provide up-to-the-second views of your business data using Adobe LiveCycle Data Services, a component of Adobe LiveCycle Enterprise Suite software, add an LCDS connection to your model. The LCDS connection streams real-time data into your models without users having to request a data refresh.

1. Click ▶ *Data* ❯ *Connections* ◣.
2. In the *Data Manager* dialog box, click *Add* and then select *LCDS Connections*.
3. On the Definition tab, set the following options:

Option	Description
Host	In the list, select the LCDS host connection to associate with this connection or click *Host* to add a new host. Then, in the *Manage Hosts* dialog box, click *Add* and enter a name for the host and the URL for the LCDS server using the following format: `http://<<server name>>:<<port>>/<<LCDS context path>>/xcelsius` where `<server name>` is the name of the LCDS server, `<port>` is the server port number, and `<LCDS context path>` is the context root for the virtual directory. For example, `http://localhost:8700/flex/xcelsius`
Query	To retrieve information about the available feed destinations from the selected host, click *Query*.
Available Data Destinations	Select one of the available feed destinations to bind to the model.
Advanced	To expand the settings available for the selected feed destination, click *Advanced*.
Current Destination ID	Available when you click *Advanced*. The identification number of the LCDS feed destination.
Current Destination Channel Type	Available when you click *Advanced*. The type of channel (or communication protocol) being used.
Current Destination Channel URL	Available when you click *Advanced*. The URL associated with the selected feed destination.
Destination Data	This section lists the data elements available from the selected host, their data type (number, text, or true/false), and their shape or structure (singleton - a single value, !D - a column or row of values, or 2D - a table of values). Select a data element, and then click the *Cell Selector* ( ) at the bottom right to select the range in the embedded spreadsheet where the data will be stored.

6.2.1.11   To configure External Interface Connections

External Interface Connections allow you to expose selected data ranges in the embedded spreadsheet to an external source so data can be copied into and out of the exported model. It also provides built-in event notification so the model instantly recognizes when the data source changes. When you configure the external interface connection, you specify a name and define where the values are stored in the spreadsheet.

> ⚠️ Caution
>
> If you include external data connections in models, security restrictions in Adobe Flash Player may prevent users from accessing the data source. For more information, see *Adobe Flash Player Security Restrictions*.

1. Click ▶ *Data* ❯ *Connections* ◣.
   The *Data Manager* dialog box opens.

2. In the *Data Manager* dialog box, click *Add* and select *External Interface*.

3. On the Definition tab, set the following options:

Option	Description
Range Name	Type a name for the range. Use only numbers, letters, and underline. Do not start the name with a number and do not include spaces in the name.
Range Type	Select one of the following options: | Option	Description
Cell	The range consists of a single cell.
Row/Column	The range consists of a single row or column.
Table	The range consists of more than one column or row.	
Range	Click the *Cell Selector* ( ) and select the cells in the range.
Access	Specifies whether the range is for input, output, or both. Select one of the following options: | Option	Description
Read	The external application can read data from the range.
Write	The external application can write data to the cells in the range.
Read/Write	The external application can read data in the range, as well as write data to the cells in the range.	

6.2.1.11.1　Setting up external interface connections

When you export a model that contains an *External Interface Connection* to HTML, Dashboards creates an HTML file and a SWF file. The HTML file includes Object and Embed tags needed to display the SWF file. You must add JavaScript code to the HTML file so that when the HTML file is opened in a web browser, the embedded Dashboards SWF file calls out to notify JavaScript when its external interface connection is ready to be used. For more information about adding the JavaScript code, see *Using Xcelsius External Interface Connection methods*.

> **i Note**
>
> After you export models that contain external interface connections, you can combine the HTML into a single file. However, each Object/Embed tag pair represents a different SWF file and must have a unique Name/ID. Each SWF model is identified by its Name/ID.

> **i Note**
>
> For additional HTML samples using the Dashboards external interface connection methods, see the sample files available in the *...Xcelsius\Connectivity\External Interface* folder in your Dashboards installation package.

⋆ Example

This example shows the HTML output for a SWF file containing an external interface connection:

```html
<HTML>
<HEAD>
<TITLE>eic2.swf</TITLE>
</HEAD>
<BODY>
<OBJECT classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://fpdownload.adobe.com/pub/shockwave/cabs/ flash/
swflash.cab#version=9,0,0,0"
WIDTH="849" HEIGHT="300" id="myMovieName">
<PARAM NAME="movie" VALUE="eic2.swf">
<PARAM NAME="quality" VALUE="high">
<PARAM NAME="bgcolor" VALUE="#FFFFFF">
<PARAM NAME="play" VALUE="true">
<PARAM NAME="loop" VALUE="true">
<PARAM NAME=bgcolor VALUE="#FFFFFF">
<EMBED src="eic2.swf" quality=high bgcolor=#FFFFFF WIDTH="849" HEIGHT="300"
NAME="myMovieName" ALIGN="" TYPE="application/ x-shockwave-flash"
play="true" loop="true"
PLUGINSPAGE="http://www.adobe.com/shockwave/download/index.cgi?
P1_Prod_Version=ShockwaveFlash">
</EMBED>
</OBJECT>
</BODY>
</HTML>
```

Note that `<PARAM NAME="movie" VALUE="eic2.swf">` specifies the SWF filename and, if you combine the HTML output for multiple SWFs into a single file, the `id="myMovieName"` and `NAME="myMovieName"` must be updated to a unique name/id for each SWF included in the file.

6.2.1.11.2   Using Dashboards External Interface Connection methods from JavaScript

After you export the SWF model to an HTML file, you must edit the HTML file so the embedded Dashboards SWF file can notify JavaScript by calling the `setSWFIsReady` method when its *External Interface Connection* is ready to be used.

➡ Tip

For additional HTML samples using these methods, see the sample files available in the *...Xcelsius\Connectivity\External Interface* folder in your Dashboards installation package.

setSWFIsReady Callback function

Add this function to the HTML output below the Object/Embed tags for the SWFs so the HTML file will respond to the SWF call.

Syntax

```
setSWFIsReady(movieName)
```

Parameters

movieName (String) is the OBJECT tag ID or EMBED tag name used to refer to the SWF in JavaScript.

Example

The following example shows the HTML for a single SWF model called "myMovieName" and the added function to respond to the SWF call.

```
<HTML>
<BODY>
<OBJECT
classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://fpdownload.adobe.com/pub/shockwave/cabs/flash/
swflash.cab#version=9,0,0,0"
WIDTH="270" HEIGHT="35" id="myMovieName">
<PARAM NAME="movie" VALUE="mymovie.swf">
<PARAM NAME="quality" VALUE="high">
<PARAM NAME="bgcolor" VALUE="#FFFFFF">
<PARAM NAME="play" VALUE="true">
<PARAM NAME="loop" VALUE="true">
<PARAM NAME=bgcolor VALUE="#FFFFFF">
 <EMBED src="mymovie.swf" quality=high bgcolor=#FFFFFF WIDTH="270" HEIGHT="35"
NAME="myMovieName" ALIGN="" TYPE="application/x-shockwave-flash"
play="true" loop="true"
 PLUGINSPAGE="http://www.adobe.com/shockwave/download/index.cgi?
P1_Prod_Version=ShockwaveFlash">
 </EMBED>
</OBJECT>
<SCRIPT language="JavaScript">

function setSWFIsReady(movieName)
{
// Add additional code using external Interface connection
// methods...
}
</SCRIPT>
</BODY>
</HTML>
```

describeDataSources method

Use this method to retrieve an XML representation of the ranges defined in the SWF *External Interface Connection*.

204

Dashboards and Presentation Design User Guide
Working with Data Sources

Syntax

```
describeDataSources ()
```

Returns

(String) XML representation of the ranges as a string.

getDataSource method

Use this method to retrieve the current value for the SWF *External Interface Connection* range.

Syntax

```
getDataSource (rangeName):Object
```

Parameters

<rangeName> (String) is the name of the range in the SWF *External Interface Connection*.

Returns

The current range value.

setDataSource method

Use this method to set the current value for the SWF *External Interface Connection* range.

Syntax

```
setDataSource (rangeName, value)
```

Parameters

- **`<rangeName>`** (String) is the name of the range in the SWF *External Interface Connection*.
- **`<value>`** (Object) is the new value.

addDataSourceChangeEventListener method

Use this method to register a JavaScript callback handler function that the SWF will call when a specific readable range value changes.

Syntax

```
addDataSourceChangeEventListener (rangeName, handlerFunctionName, sendCurrentValue)
```

Parameters

- **`<rangeName>`** (String) is the name of the range in the SWF *External Interface Connection*.
- **`<handlerFunctionName>`** (String) is the name of the JavaScript function the SWF calls when the range changes. For details, see `handlerFunctionName Callback method`.
- **`<sendCurrentValue>`** (Boolean) indicates if the current value should be passed to the SWF through the callback. It can be set to true (send the current value) or false (don't send the current value).

Example

The following example registers the callback function to listen for changes to the slider range and pass the current value and any changes to the range to myRangeHandler function:

```
addDataSourceChangeEventListener("slider", "myRangeHandler", true);
```

handlerFunctionName Callback method

Use this method to define the **`<handlerFunctionName>`** used in the `addDataSourceChangeEventListener` method.

206

Dashboards and Presentation Design User Guide
Working with Data Sources

Syntax

```
<functionName> (<movieName>, <rangeName>)
```

Parameters

- **<functionName>** (String) is the name of the function the addDataSourceChangeEventListener method registered as the callback function in the **<handlerFunctionName>** parameter.
- **<movieName>** (String) is the value specified in the OBJECT tag called *id* and the EMBED tag called *name* that identify the applicable SWF file.
- **<rangeName>** (String) is the name of the range in the SWF *External Interface Connection*.

Example

The following example defines the function "**<myRangeHandler>**" that is used in the addDataSourceChangeEventListener method. The OBJECT tag id and EMBED tag name value is "myMovieName" and the range name is "myRangeName":

```
function myRangeHandler (movieName, rangeName);
{
    // Get the movie object (browser dependant).
    var movie = null;
    if (navigator.appName.indexOf("Microsoft") != -1)
    {
            // Browser is Internet Explorer.
        movie = window[movieName];
    }
      else
      {
        // Browser is not Internet Explorer.
            movie = document[movieName];
    }

    // Get the movie current range value.
        var newValue = movie.getDataSource(rangeName);

    // Add additional code using external Interface connection methods...

}
```

removeDataSourceChangeEventListener method

Use this method to remove (unregister) a JavaScript callback handler function.

Syntax

```
removeDataSourceChangeEventListener (rangeName, handlerFunctionName)
```

Parameters

- **<rangeName>** (String) is the name of the range in the SWF *External Interface Connection*.
- **<handlerFunctionName>** (String) is the name of the function the `addDataSourceChangeEventListener` method was using as the callback function when the range changed.

Example

The following example sets the SWF to stop listening for changes to the Slider range:

```
removeDataSourceChangeEventListener ("slider", "myRangeHandler");
```

6.2.1.12 Working with Excel XML maps

In Microsoft Excel 2003 and 2007, you can embed XML documents and schemas into spreadsheets. When you add an XML schema to a spreadsheet, you create an XML map. The XML map allows you to map cells in the spreadsheet to elements in the XML schema. These maps can be used to translate XML content from a remote data source to the rows and columns of a spreadsheet. If you want to load XML data directly into your model, you can include XML maps in the embedded spreadsheet in Dashboards.

By default, XML functionality is not included in the embedded spreadsheet in Dashboards. You can define the XML maps in Excel. When you import the spreadsheet into Dashboards, it retains the link to the XML source and the cell mappings to the XML schema. If you want to work with the XML maps directly in Dashboards, you can enable the XML features in the embedded software.

> **i** Note
>
> Data that has been mapped using the Excel mapping functionality can be manually refreshed by using a *Connection Refresh button* component in your model.

> **i** Note
>
> If a Dynamic URL is used with the XML Map feature and an interval refresh, it must be bound to an Excel cell that contains a formula.

6.2.1.12.1   To add Excel XML maps to models

To add an XML map to the embedded spreadsheet, you can set up the map in the Excel spreadsheet and then import the spreadsheet into Dashboards or, if XML features are enabled in Dashboards, you can create the XML map in Dashboards. For more information about creating Excel XML maps, see the Microsoft Excel Online Help. Once the map is available in the spreadsheet, you can add components to the canvas and link them to the spreadsheet, and add an Excel XML map data connection to retrieve data from the XML source.

1. To add an Excel XML map to the embedded spreadsheet, do one of the following:

   ○ If you set up the XML map in your Excel spreadsheet, import the Excel spreadsheet. For more information about importing spreadsheets, see *To import Excel spreadsheets*.
   ○ To create the XML map directly in Dashboards:

      1. On the embedded spreadsheet, for Excel 2007, click the *Developer* tab and then click *Import XML Data*, or for Excel 2003, on the toolbar, click *Import XML Data*.
      2. In the *Import XML* dialog box, in the *File Name* list, type the URL for your XML data source and click *Open*.
      3. If the XML content doesn't incorporate a schema definition, Excel can create a schema based on the source data. Click *OK*.
      4. In the *Import Data* dialog box, select where you want to store the data and click *OK*.
      5. Embed the URL that will be used to fetch data in the underlying spreadsheet, for example, enter a static query (the same URL used to import the XML data) in a cell in the spreadsheet.
      6. Set up input cells in the spreadsheet and convert the static query to an interactive query using a formula that reads the values from the spreadsheet cells.

2. Add components to your model and link them to the cells in the spreadsheet.
3. Click ▌▶ *Data* ▶ *Connections* ◢.
4. In the *Data Manager* dialog box, click *Add* and select *Excel XML Maps*.
   An Excel XML Map connection is added to the list of connections.

   > **i** Note
   >
   > The Excel XML Map Connections option only adds a connection if an Excel XML Map is defined in the embedded spreadsheet.

5. On the *Definition* tab, set the following options:

Option	Description
Name	Type a name for the connection.
XML Data URL	Select the location where the model will retrieve updated data. You can map this to the URL where the XML data is stored or to the cell in the spreadsheet where you defined an interactive query formula.

6. To set the refresh and load status options, click the *Usage* tab. For more information about setting these options, see *To set refresh options* and *To create load and idle messages*.
7. Save your model and export it to a suitable format.

6.2.1.12.2 Enabling XML in the embedded spreadsheet

The embedded spreadsheet does not make all Excel features available by default. If you want to define XML maps in Dashboards, you need to enable XML features in the embedded spreadsheet. The procedure to enable XML features depends on which version of Excel you are using.

To enable XML features with Excel 2003

In Excel 2003, to use the XML features in Dashboards, you must add them to the Excel toolbar.

1. If Dashboards or Excel are running, close both applications.

   > **i Note**
   >
   > To make sure there are no instances of either program still running, view the process in the Windows Task Manager. There should be no processes named Dashboards or Excel in the list of running processes.

2. Launch Excel 2003.
   This should be the only instance of Excel running.

3. In Excel, click ▶ *Tools* ❭ *Customize* ◣.

4. In the *Customize* dialog box, click the *Toolbars* tab, click *New*, and enter a name for the toolbar, for example, XML Toolbar. Click *OK*.
   An empty, floating toolbar is created.

5. In the *Customize* dialog box, click the *Commands* tab.

6. In the *Categories* list, select *Data*, then in the *Commands* list, select *XML Source*, *Import XML Data*, and *XML Map Properties*. Drag these items to the floating toolbar you created in step 4.

7. Drag the floating toolbar and park it along the other Excel toolbars at the top of your application window, and then close Excel.

The next time you start Dashboards, the new XML toolbar should be available.

To enable XML features with Excel 2007

In Excel 2007, the XML features are available in the Developer ribbon tab.

1. If Dashboards or Excel are running, close both applications.

   > **i Note**
   >
   > To make sure there are no instances of either program still running, view the process in the Windows Task Manager. There should be no processes named Dashboards or Excel in the list of running processes.

2. Launch Excel 2007.
   This should be the only instance of Excel running.

3. In the main application menu of Excel 2007, click *Excel Options*.

210   Dashboards and Presentation Design User Guide
Working with Data Sources

4. Under Popular options, select *Show Developer Tab in the Ribbon* and click *OK.*

The next time you start Dashboards, the *Developer* tab should be available in the ribbon.

6.2.1.13 To configure Web Dynpro Flash Island connections

A Web Dynpro connection allows you to embed your model in a Web Dynpro application (WD application) as a Flash Island. When you configure a Web Dynpro connection, you specify a name, the connection type, and define the spreadsheet cells which are bound to the WD application.

The connection between your model's embedded spreadsheet and the WD application is bidirectional: changes made in the WD application context will update the corresponding spreadsheet cells, and similarly, changes in the spreadsheet will update the WD application.

> **i Note**
>
> While each model has a limit of one Web Dynpro connection, you can create unlimited bindings between the spreadsheet and the WD application. This is because a Flash Island (your dashboard) only has one data context. The ranges you add correspond to *Properties (Single Values)* and *Data Sources (Tables)* within that context.

1. Click ▶ *Data* ▶ *Connections* ▶.
2. In the *Data Manager* dialog box, click *Add* and then select *Web Dynpro Flash Island*.
3. Click *Add a Range* (⊞) to create a binding between the model's spreadsheet and the Web Dynpro application.
4. On the *Definition* tab, set the following options:

Option	Description
Type	Select one of the following connection types: ○ *Single Value (Property)* Properties must be bound to a single spreadsheet cell. ○ *Table (Data Source)* Data sources can be bound to a range of cells.
Name	Type a name for the connection.
Range	Type in a range, or click the *Cell Selector* ( ) and select a value from the spreadsheet.

More information about setting up a Web Dynpro application can be found by clicking this link: *http://help.sap.com/saphelp_nw70ehp2/helpdata/en/20/361af8255f4a33a2e8441956cc8f63/frameset.htm*.

6.2.1.14 To set refresh options

If you add a Web service query (Query as a Web Service), Web Service, XML Data, FS Command, Portal data, SAP NetWeaver BW, or Live Office connection to a model, you can specify how and when the data for that connection will be updated on the model.

> **i Note**
>
> If you add a *Connection Refresh Button* component to a model, you can also set refresh options for that component. When refresh options are set in the *Data Manager*, the data is refreshed when the model runs and before components are loaded. When refresh options are set for the *Connection Refresh Button*, the refresh options are applied when the component loads in the model. For more information about the *Connection Refresh Button*, see *Using the Web Connectivity components*.

1. Click ▌ *Data* ❯ *Connections* ◗.
   The *Data Manager* dialog box opens.
2. Click the connection you want to set refresh options for.
3. Click the *Usage* tab.
4. On the *Usage* tab, set the following options:

Option	Description
Refresh Before Components are Loaded	**i Note** For Web service query, Web Services, XML, SAP NetWeaver BW, and Live Office data connections only. Select this option if you want to refresh the data each time the model loads and use that data as the initial data for the model (if you use a Reset Button component, it will reset the data to the values from the last time the model was loaded).
Use Default Query Data	**i Note** For SAP NetWeaver BW connections when *Refresh Before Components are Loaded* option is selected. This option determines how data is refreshed when the model is run. When selected, the model uses the default values that were configured for the query and variable values. If not selected, the model refreshes the data using the input values set for each item. This option does not affect refresh options based on triggers or user interaction, such as button clicks. For example, If a model has a mandatory variable for country and the input value is linked to a cell in the embedded spreadsheet that contains the value for France. When the model runs, a Mandatory variable dialog box opens and the user enters Germany for the country value. The query is initialized for Germany but, if the *Use Default Query Data* option is not selected, the input values are then evaluated and the country is set to France so the model displays data for France. If this option is selected, the model ignores the input value from the spreadsheet and displays the data for the country specified in the dialog box.
Refresh Every	**i Note** For Web service query, Web Services, SAP NetWeaver BW, and XML Data connections only. Select this option to automatically refresh the data at specific intervals. In the first list, select the interval value and in the second list, select the time unit (minutes, seconds, and so on).
Refresh on Trigger	In the *Trigger Cell* box, type or select the cells that will trigger a refresh. Then select one of the following options:

Option	Description
	○ *When Value Changes* ○ *When Value Becomes* and enter the value that will trigger the refresh. **ⓘ Note** You can have only one trigger cell for each connection. If you have several variables to be set before the data is refreshed, you can either add a Refresh button to allow users to trigger the refresh once they have set all the options or create a global trigger cell that concatenates all the input values so the refresh is triggered when any of the input values change. To avoid multiple refreshes when loading models, use a Connection Refresh button rather than an automatic trigger for refresh or carefully design the trigger using Excel functions such as if-clauses.

Related Links

Managing data connections [page 185]

To configure Web service query (Query as a Web Service) connections [page 189]

To configure Web Service connections [page 191]

To add XML Data connections [page 193]

Integrating with SAP NetWeaver BW [page 251]

Using Live Office Data Connections [page 237]

6.2.1.15   To create load and idle messages

When you create a model that has a *Web service query (Query as a Web Service)*, *Web Service Connection*, *SAP NetWeaver BW Connection*, or *XML Data* connection, you can define messages that will be displayed while the model loads or is idle. You can also specify whether the cursor will be visible during loading and whether mouse input will be available.

1. Click ▌ *Data* ❯ *Connections* ▌.
   The *Data Manager* dialog box opens.
2. Click the connection you want to set messages for and click the *Usage* tab.
3. Under *Load Status*, set the following options:

Option	Description
Loading Message	Type or select a message to display while the model is loading.
Idle Message	Type or select a message to display while the model is idle.
Insert In	Type or use the Cell Selector (🔲) to select the cells in the embedded spreadsheet where the loading and idle messages will be inserted.
Enable Load Cursor	Select this option to display a different cursor while the model is loading.
Disable Mouse Input on Load	Select this option to prevent users from entering mouse input while the model is loading.

Related Links

Integrating with SAP NetWeaver BW [page 251]

6.2.2 Query overview

6.2.2.1 To browse queries

You can create new queries and view current queries using the *Query Browser*. The *Query Browser* is only available in SAP BusinessObjects Dashboards.

Click ▶ *View* ❯ *Query Browser* ❯ to make the *Query Browser* visible if it is not already available.

Icon	Action	Result	Note
Add Query	To add a query	The *Add Query* dialog box opens.	Available only to add the model's first query. This may require a login to the SAP BusinessObjects Business Intelligence platform.
	To add a query	The *Add Query* dialog box opens.	Opens the Add/Edit dialog box. This may require login to the BI platform.
	To edit a selected query	The *Edit Query* dialog box opens.	Opens the Add/Edit dialog box at the editing step required by the query's current state. This may require login to the BI platform.
✕	To delete a selected query	The query is deleted from your model.	
	To refresh a selected query or all queries	The selected query or all queries are refreshed.	You can specify new prompt values before you run your queries.
↑ ↓	To change the order of queries	The selected query is moved to a new position in the list of queries.	
⌄ ⌃	To expand or contract query information	The selected query's details are expanded or hidden. The details displayed are: ○ Result objects ○ Filters (if any) ○ Prompts (if any)	

Icon	Action	Result	Note
▼ **Result Objects**	To expand or contract query result objects information	A list of the result objects is displayed.	Queries will always have at least one result object, but may not have filters or prompts.
Insert in Spreadsheet ▼	To insert results object data into the embedded spreadsheet	The embedded spreadsheet icon appears.	You can choose a cell range where data from a result object can be inserted.
	To select a cell range in the embedded spreadsheet	The *Select a Range* dialog box for the embedded spreadsheet opens.	You are able to select a range with the mouse, or type the cell names in the text box.
▶ **Filters**	To expand or contract filter information	All filter objects are shown, if the query has any.	All filters are displayed, along with their AND/OR relationships if there is more than one filter.
▼ **Prompts**	To expand or contract prompt information	All prompts are shown, if the query has any.	There are three ways to enter prompt values: with a *Query Prompt Selector* component, by binding the prompt to the embedded spreadsheet, or by insertion by other components. Once the values have been entered by one way, the other two ways are disabled.
Connect to Spreadsheet ▼	To connect a prompt to the embedded spreadsheet	Two *Select a Range* dialog boxes for the embedded spreadsheet open.	Prompts can use values from the embedded spreadsheet and they can insert a list of values (LOV) into the spreadsheet.
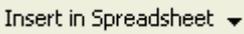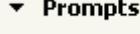
6.2.2.1.1 Universes and BEx queries

Universes and BEx (Business Explorer) queries are the data repositories you use to create queries. Using these data sources, you can create a query without seeing, or having to know anything about, the underlying data structures in the database. Universes and BEx queries organize objects into classes; a class is a container that groups objects that have a common purpose.

A universe is a file that organizes the data elements of a large database. Universes are designed and optimized by a universe designer, and can contain predefined filters that you can use in your queries. If you want to learn more about universes, see the *Universe design tool* documentation, available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ▶ *All Products* ◀.

A BEx query contains a group of predefined result objects that you can run reports on in Dashboards. While BEx queries are predefined, you can sort and filter the query's result objects while building your query. For more information about building queries, see *To build a query* [page 221].

Universes and BEx queries organize objects into the following categories:

Icon	Category Name
	Class or subclass (also called a folder)
	Analysis dimension
	Date analysis dimension
	Level-based hierarchy
	Value-based hierarchy
	Default level-based hierarchy
	Default value-based hierarchy
	Hierarchy level
	Dimension
	Attribute
	Measure
	Filter
	Hierarchy named set
	Calculated member
	Measure group folder

Related Links

6.2.2.1.2   Query Browser

The *Query Browser* is a compact window, similar to the *Object Browser*, that provides quick access to your queries, and the tools to modify them. Using the *Query Browser*, you can create a new query, or edit, refresh, or delete an existing query.

> **i   Note**

To open the *Query Browser*, on the main menu, click ▶ *View* ▶ *Query Browser* ▶.
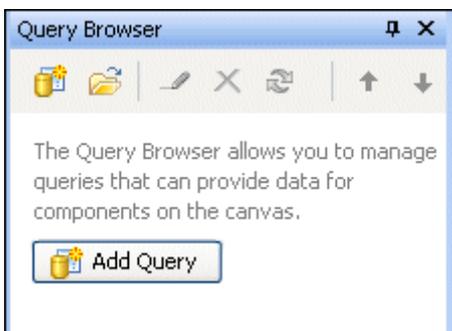
The *Query Browser* displays any current queries, as well as their properties. Each query has the following information listed:

Result Objects: the raw data returned by your query.

Filters: any filters you added while constructing your query.

Prompts: any prompts you added while constructing your query. You can drag any prompt (created in either the *Member Selector* dialog box or the *Filter* panel) onto the canvas to create a *Query Prompt Selector* component bound to that prompt.

Button	Description
	Create new query
	Open query
	Edit query
	Refresh query
	Move selected query up
	Move selected query down
	Expand query information
	Collapse query information

For more detailed information regarding each feature of the *Query Browser*, see the related topics.

Related Links

 217

6.2.2.1.3   To add or edit queries

A query is a body of data that is a subset of a universe. The data returned by a query is used to create your dashboard: the data can be linked to the embedded spreadsheet, or bound directly to certain components.

To start building a query, open the *Query Browser* and click *Add Query* ( ).

> **i Note**
>
> To open the *Query Browser*, on the main menu, click ▶ *View* ▶ *Query Browser* ◢.

> **i Note**
>
> To edit a query, select an existing query and click *Edit* ( ).

Clicking on the *Add Query* button displays the *Add Query* window, which is what you will use to build your query. The process of building a query is divided into five panes:

- *Select a Data Source*
  Use this pane to choose whether to use a universe or a BEx query as your main data source.
- *Select a Universe*/*Select a BEx Query*
  Use this pane to browse the Central Management Server you are connected to and select your data source.

  > **i Note**
  >
  > The name of this pane will reflect your data source selection in the first pane.

- *Build Query*
  Use this pane to build your query. This pane is divided into the following panels:

  - *Universe*
  - *Result Objects*
  - *Filters*
  - *Data Preview*

  You can drag data elements from the *Universe* panel into the *Result Objects* and *Filters* panels, and use the *Member Selector* on hierarchical data elements to further refine your query. You can also preview your current query in the *Data Preview* panel.

  > **i Note**
  >
  > The *Data Preview* panel is a smaller, but otherwise identical, version of the next pane in the *Add Query* process: *Preview Query Result*.

  > **i Note**
  >
  > This panel is renamed to *Result Set* once you preview data in it.

- *Preview Query Result*
  Use this pane to preview the results of your current query. If you are satisfied with your query and do not want to set custom refresh options or load messages, you can skip the next step.

- *Usage Options*
  Use this pane to set your query's refresh options, and its behavior while loading or idle.

Related Links

Member Selector [page 224]

To select a data source

Selecting a data source is the first step of building a query.

> **i Note**
>
> If the *Query Browser* is not already open, click ▌ *View* ❯ *Query Browser* ◣.

1. On the *Query Browser*, click *Add Query* (⬚).

   You can select a data source only for a new query; you cannot modify the choice you made here when editing a query.

   > **➡ Tip**
   >
   > There will be another *Add Query* button ( ) in the *Query Browser* if no queries exist.

   The *Add/Edit Query* dialog box will open to the *Select a Data Source* pane.

2. Click *Universe* or *BEx Query*.
   The *Next* button at the bottom of the pane will become active.

3. Click *Next*.

Related Links

To select a universe or BEx query [page 219]

To select a universe or BEx query

Selecting a universe file or BEx query is the second step of building a query. Note that if you wish to enable your model as an RRI sender, you must select a BEx query. For more information about RRI, see *Report-Report Interface* [page 220].

> **i Note**
>
> The title of this pane will be either *Select a Universe* or *Select a BEx Query*, depending on your selection in the first pane of the query building process (*Select a Data Source*).

1. Browse your Central Management Server's folders until you find the universe file or BEx query you want to use.

   Note that your particular BI platform connection may contain several universes or BEx queries in an expandable tree view of folders and files.

2. Click the universe filename or the BEx query filename.
   The *Next* button at the bottom of the pane will become active.
3. Click *Next*.

Related Links

To build a query [page 221]

Report-Report Interface

In SAP BusinessObjects Dashboards, you can enable your model to connect to the Insight to Action framework service provided by the SAP BusinessObjects Business Intelligence platform. The Insight to Action framework allows Dashboards models that use a BEx query as a data source to access the *Report-Report Interface* (RRI). RRI provides the ability to jump to targets (called "receivers") that have been attached to a BEx query. Targets can be reports, other queries, transactions, and web addresses. As part of the jump, RRI passes selection criteria to facilitate a context-sensitive jump.

To enable your model to be an RRI sender, the model must contain a query created using a BEx query as a data source, as well as a *Hierarchical Table* component on the canvas. For more information about enabling a *Hierarchical Table* to be an RRI sender, see *Using selector components* [page 79].

Targets are most commonly used to do the following:

- Navigate to various result object-specific targets, such as the following:
  - From the ID of a Cost Centre in a row in the hierarchical table, navigate to other reports containing data for that cost centre.
  - From the name of a sales representative in a row in the hierarchical table, navigate to that sales rep's employee fact sheet or order volume.
- Start workflows or activities, such as the following:
  - From a query showing cost per cost centre, create a "request for clarification", which is sent to an accountant.

> **i Note**
>
> In order for Dashboards to use RRI, the following three platform services must be selected when installing SAP BusinessObjects Business Intelligence platform:
>
> - *Web Application Container Server*
> - *RESTful Web Services*
> - *Insight to Action Service*

Related Links

Using selector components [page 79]
Selector general properties [page 87]
To build a query [page 221]

Value-based hierarchies and RRI

When using value-based hierarchies as result objects in your BEx query, they may be unable to pass context-sensitive information to the invoked RRI target. For example, if you include a value-based hierarchy in your query,

then bind that query to an RRI-sender enabled *Hierarchical Table* component, the invoked RRI actions may return incorrect data.

If you want to pass data from a value-based hierarchy to the RRI target, then add a dimension object (🚩) or level-based hierarchy (🗂 , 🗂) to your query's result objects instead of a value-based hierarchy.

Related Links

Report-Report Interface [page 220]
Universes and BEx queries [page 215]

To build a query

The *Build Query* pane is the third step of building a query, or the first editable pane when editing an existing query.

You can click *Show/Hide Filter Pane* (▽) or *Show/Hide Data Explorer Panel* (▦) to configure the pane.

1. Click and drag at least one object or folder of objects from the *Universe* panel to the *Result Objects* panel. Not all universe objects can be used as result objects.

   > **i Note**
   >
   > If you plan on using hierarchical result objects in your query, it is recommended that you use only one hierarchy object, along with one or more measures. For more information, see *Designing queries for hierarchical tables* [page 222].

   A green icon (📥) will indicate whether the object is valid in the panel where you have dragged it. A red icon
   (📥) will indicate that it is invalid.

2. Click and drag additional objects or folders of objects from the *Universe* panel to either the *Result Objects* panel or the *Filters* panel.

   > **i Note**
   >
   > Objects of type dimension and measure can be used as either result objects or filters.

   > **i Note**
   >
   > Objects of type precondition can be used only as filters.

   > **i Note**
   >
   > A specific dimension or measure object can be used as both a result and as a filter.

3. You can preview your current query by clicking *Refresh* (🔄) in the *Data Preview* panel. If your query contains prompts, you will be asked to input values for those prompts.

   > **i Note**

> The *Data Preview* panel is a smaller, but otherwise identical, version of the next pane in the *Add Query* process: *Preview Query Result*.

4. Once your query has at least one result object, the *Next* button at the bottom of the pane will become active. Click *Next*.

You can refine your result objects and filters using the *Member Selector*. The *Member Selector* allows you to select specific members, children, parents, or siblings of hierarchical data objects, or create a prompt on the data object which allows dynamic member selection at runtime. For more information, see *Member Selector* [page 224].

Related Links

To preview a query result [page 232]

Best practices for designing queries with hierarchical data

Additional considerations must be taken when designing a query with hierarchical result objects. This section explains the recommended format for queries with hierarchical data, and describes how data is displayed in the *Hierarchical Table* when the recommendations are not followed.

Designing queries for hierarchical tables

To ensure that your query data is displayed properly when bound to a hierarchical table, keep the following in mind when designing your query:

* Add only a single hierarchical result object and one or more measures. Do not include dimension result objects, or result objects from multiple hierarchies.
* Add the hierarchical result object as your first result object.
* Avoid choosing sub-entries in the member selector without including their parent entry.
  For example, if you want to show global sales data, as well as sales data for the UK and for Munich, you can create a query with a *Geography* hierarchy that includes the following entries: the top-level Global entry; UK and Germany from the Country level; and Munich from the City level. You can then add measure result objects for the appropriate sales data.

For hierarchical tables bound to queries that do not follow the recommended structure, the following formatting is applied:

* Hierarchical tables with dimension result objects:
  If you add a dimension to your query, duplicate rows may appear in the hierarchical table when it is bound to the query, and some rows of the hierarchy may be blank.
  For example, if you add a Country dimension to the query described in the previous example, rows for all of the countries in the database will appear in the hierarchical table directly beneath the Global entry. The hierarchy column will be blank for these rows. To avoid this issue, you must select all the appropriate countries from the hierarchy result object, instead of adding a new dimension.

> ℹ **Note**
>
> This formatting remains even if you do not bind the dimension result objects to the hierarchical table.

* Hierarchical tables with missing parent entries:
  If you include a sub-entry in a hierarchy without including its parent entry, the sub-entry may be moved to a higher level of the hierarchical table.
  For example, if you include only the Global entry, the UK entry, and the Munich entry from the Geography hierarchy, the hierarchical table displays UK and Munich on the same level beneath the Global entry. To ensure that all entries appear on the correct level, you can either include all of the relevant parent entries for each level, or exclude the entire level.

222

Dashboards and Presentation Design User Guide
Working with Data Sources

Related Links

Hierarchical data with variable dependencies

BEx queries can contain dependent variables. For instance, a hierarchy node can depend on a hierarchy variable; when you select a specific hierarchy variable, the nodes are updated accordingly. However, since these dependencies are only handled in SAP BusinessObjects Dashboards and not in exported models, it is recommended you avoid including the following dependencies:

- Hierarchy node variables that depend on a hierarchy variable.
- BW variables that depend on a Key date variable.
- Filters added in the *Build Query* pane that depend on a BW variable.
  For more information about filters, see *To set parameters on filter objects* [page 228].

To set query properties

You can set query properties such as the query name, limitations on numbers of rows returned, context (if available) and the order of prompts if the query has more than one.

The query properties are accessible from the *Build Query* pane of the *Add Query* process.

1. Click *Query Properties* (   ).
   The *Query Properties* dialog box opens.
2. On the *Query Properties* dialog box, do any of the following:

   o Type a name for your query
   o Set Data options as follows:

Option	Description
Retrieve duplicate rows	This retrieves duplicate row entries. This is useful to leave unchecked if you have recurring data, such as year or product category, which is not useful if repeated. For example, if your query returned sales per quarter per year, your query would appear as follows: Retrieving duplicates:

Retrieving duplicates:

2010	Q1	10,000
2010	Q2	12,500
2010	Q3	11,000
2010	Q4	9,750

Not retrieving duplicates:

2010	Q1	10,000
	Q2	12,500
	Q3	11,000

Option	Description		
		Q4	9,750
Retrieve empty rows (only supported in OLAP universes)	An empty row occurs typically in multidimensional queries when the data for the intersection of two or more dimensions does not exist. When this option is checked, the result set includes rows that can contain empty cells. When this option is unchecked, the result set contains only rows with non-empty cells.		
Max retrieval time	Limit the amount of time (in seconds) that the query can run. To specify no limit, set the value to -1.		
Max rows	Limit the number of rows of data the query can return and, in the corresponding list, select the number of rows. To specify no limit, set the value to -1. **i Note** This setting applies to the query at refresh time. At design time, you can also limit the number of rows returned for the preview.		
Sample rows	Retrieve a random data set within the current selection criteria as a sample and, in the corresponding list, select the number of rows to return. This option is only usable if the database supports sample data.		
Fixed	If *Sample rows* is selected, select this option to have the same rows returned each time a sample query is run.		

○ Choose a *Context* if one is available in the universe.

> **i Note**
>
> Only available if a relational data source is selected.

○ Choose the order of prompts if there is more than one prompt in the query.

3. Click *OK*.

Member Selector

The member selector lets you browse the members of a hierarchical result object, and to select which of those members to include or exclude in your query.

To open the member selector window, click the down arrow () that appears in a hierarchical data element in the *Result Objects* panel.
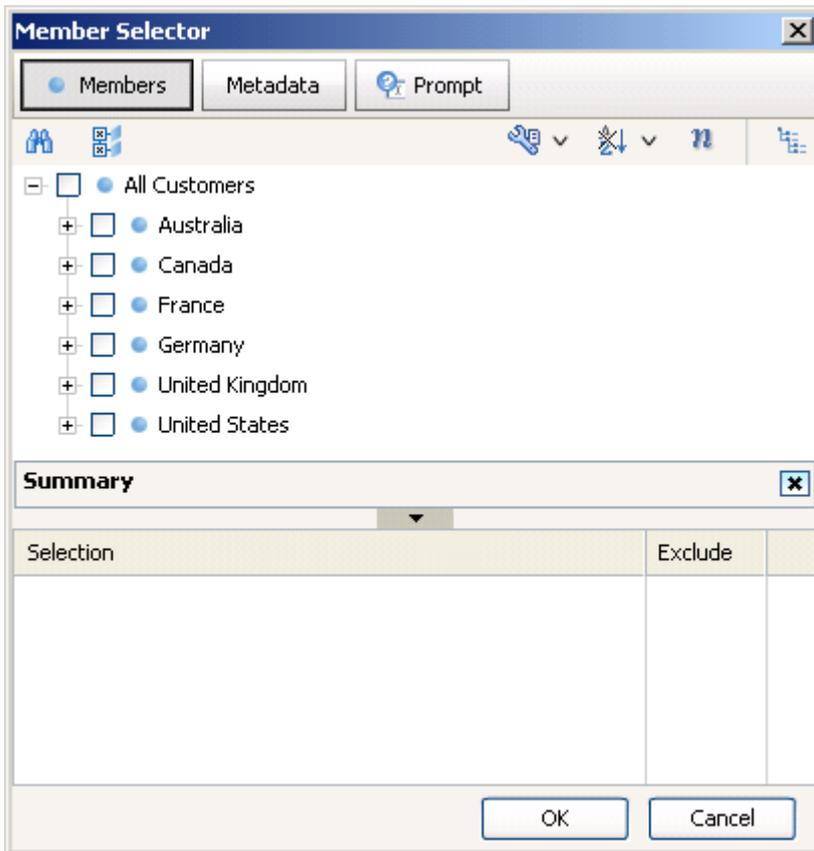
The member selector is divided into halves: the top half is where you browse for and select members, and the bottom half displays a *Summary* of all currently selected members.

The top half contains the three methods of selecting members:

- *Members*
  This tab represents the hierarchical result object as an expandable list. You can manually select any member or its relatives by expanding and browsing the hierarchy, or search for members using the *Search* option. This tab is best suited if you need to select relatives, such as descendants or siblings, of an element. For example, selecting all customers in Australia, displaying the States and Cities they're in.

- *Metadata*
  This tab lists the levels of the hierarchical result object. This tab is best suited for selecting a large number of members, or if you need to select one category of data.

  > **i Note**
  >
  > If the result object is a value-based hierarchy (as opposed to a level-based hierarchy), this tab may be blank.

  > **i Note**
  >
  > If your data source is a BEx query, this tab will be named *Levels* instead; however, the functionality remains the same.

- *Prompt*

This tab lets you create a prompt. A prompt is a way to select members, but has additional functionality: the prompt can be associated with a Query Prompt Selector component. The component can be dragged onto the canvas to provide dynamic member selection on the dashboard at runtime. This tab is best suited for very specific member selection, or if the dashboard user will need to change the member selection at runtime.

> ⚠ **Caution**
>
> Checking *Enable Parameter* in the prompt tab clears the *Summary* of any existing members selected using the Members or Metadata tabs.

> ℹ **Note**
>
> If your data source is a BEx query, this tab will not be available.

The bottom half of the member selector provides a summary of all the members you have selected. The summary can be used to exclude or delete a selected term.

To select members of a result object

When a result object has selectable members it will show an arrow in its top right corner.

1. Select the hierarchical result object and click the down arrow (⊙).
   The *Member Selector* opens, with the *Members* tab displaying the highest-level objects.

   > ➡ **Tip**
   >
   > You can also add members to a result object without opening the *Member Selector*: drag the individual members from the *Universe* panel to the result object in the *Result Objects* panel.

2. Explore the object tree by clicking on the expand/contract icons. Select an item and right-click it for options to add it or its relatives to the Summary.
   You can also right-click the object tree to make selections. A list of the objects you have selected will appear in the *Summary* panel in the lower part of the dialog box. (You may have to click open (▲) to see the list.)

   > ℹ **Note**
   >
   > In the *Summary* panel, you can also mark particular objects for exclusion by clicking in the *Exclude* check box, or delete objects from your members list by selecting the object and clicking *delete* (✖). You can discard all of your selection changes by clicking *Reset Member Selection* (✖) at the right of the *Summary* panel.

3. Click *OK*.
   Your selected members appear on the result object.

In the *Member Selector* you can also perform the following tasks:

Icon	Action	Result	Note
🔍	To search members	The Member Search dialog box opens.	You can search by member names or by member keys.

Icon	Action	Result	Note
	To select all members	All members at all levels are selected.	
	To change the member display mode	The mode list is displayed.	You can display members by caption, by unique name or by both.
	To estimate children count	An estimate of the number of a member's children displays beside the member.	
	To show all selected members	The selected members at all levels are shown.	

To select members of a result object by their metadata

When a result object has selectable levels (metadata) it will show an arrow in its top right corner.

> **i Note**
>
> If the result object is a value-based hierarchy (as opposed to a level-based hierarchy), the *Metadata* tab may be blank.

> **i Note**
>
> If your data source is a BEx query, this tab will be *Levels* instead; however, the functionality remains the same.

1. Select the hierarchical result object and click the down arrow ( ).
   The *Member Selector* opens, with the *Members* tab displayed. Click *Metadata* to switch to the *Metadata* tab.

   > **➡ Tip**
   >
   > You can also add levels to a result object without opening the *Member Selector*: drag the individual members or levels of members from the *Universe* panel to the result object in the *Result Objects* panel.

2. Expand the metadata tree until you find the level containing the objects that want to include or exclude from your query. Double-click a level to add it to the *Summary*.

   > **i Note**
   >
   > In the *Summary* panel, you can also mark particular objects for exclusion by clicking in the *Exclude* check box, or delete objects from your members list by selecting the object and clicking *delete* ( ). You can discard all of your selection changes by clicking *Reset Member Selection* ( ) at the right of the *Summary* panel.

3. Click *OK*
   Your edited list of metadata is shown in the result object. If you have excluded a level of members, this level is shown in strike-through text.

You can also search member metadata from a simple search box, and expand or collapse the metadata tree.

To set a prompt for a result object

The *Member Selector* dialog box is displayed. You want to allow your users to make run-time member selections (select member query parameters) of their own when they refresh your query.

> **i Note**
>
> If your data source is a BEx query, this tab will not be available.

1. Click the *Prompt* tab on the *Member Selector* dialog box.
2. Click the *Enable Parameter* checkbox.

   > **⚠ Caution**
   >
   > Checking this box will clear the summary of any terms that you added in the *Members* and *Metadata* tabs.

3. Accept or change the prompt message text.

   This is text that will describe the prompt's purpose to the dashboard user. If you associate this prompt with a *Query Prompt Selector* component, this is the text that will appear on the component at runtime.
4. Click the *Retain last values selected* checkbox to choose whether or not you want the user's parameter choices to be retained in the *Answer Prompts* dialog.
5. You can set pre-selected values for the prompt at runtime. To do this, click the *Set default values* checkbox. The *Edit* button becomes active. Click *Edit* and select default values in the *Select Parameter Values* dialog box. Click *OK*.
6. Click *OK*.

   The hierarchical result object is now associated with a prompt. You will be asked to enter values when creating your query, and the dashboard user can enter their own values at runtime if you place a *Query Prompt Selector* component bound to the prompt onto the canvas.

Related Links

To place a prompt on the canvas [page 234]

To set parameters on filter objects

You want to set filters on query results by specifying logical operators such as greater than, less than, equal to, etc., and by setting the values they will operate on.

1. Click the down arrow beside the logical operator box (shown below) and select one of the operators.
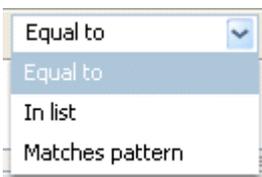
2. Click the drop-down arrow beside the value box (to the right of the logical operator box) and select one of the following:

   o Constant; type a constant directly into the box.
   o List of Values; the *Select Parameter Value* dialog box opens. For more information, see *To select filter parameters as a List of Values* [page 228].
   o Prompt; the Edit Prompt dialog box opens. For more information, see *To set options on filter prompts* [page 229].

To select filter parameters as a List of Values

The *Select Parameter Value* dialog box is open. You want to set an object to be the logical filter operator.

1. Explore the object tree or list and select an object, then click the *Move* arrow ($>$).
2. Click *OK*.
   The selected object is now the parameter for the filter.

To set options on filter prompts

The *Edit Prompt* dialog box is open. You want to set the filter as a prompt.

To set up a prompt for the selected filter object, do one of the following:

- Select *New Prompts* to create a prompt.

Option	Description
Prompt Text	Enter the text that will appear on the *Query Prompt Selector* component bound to this prompt.
Prompt with list of values	Check this to include a list of values in the drop-down list of the *Query Prompt Selector* component bound to this prompt. **i Note** If a multiple-level hierarchical filter object is used (for example,), then the following apply: ○ *Prompt with list of values* is checked and cannot be changed (since a multi-level hierarchy forces the prompt to use a list of values). ○ The drop-down list in the *Query Prompt Selector* component bound to this prompt becomes an expandable and collapsible hierarchical list. In this list, you can search for elements by name.
Select only from list	Check this to prevent the dashboard user from entering free values into the *Query Prompt Selector* component bound to this prompt. This makes selection only possible through the *Query Prompt Selector*'s list of values. **i Note** If a multiple-level hierarchical filter object is used (for example,), *Select only from list* is checked and cannot be changed (since a multi-level hierarchy forces the prompt to use a list of values). **i Note** If *Prompt with list of values* is unchecked, *Select only from list* is also unchecked and cannot be changed (since there would be no list to select from).
Retain last values	Check this to retain your last choices. Unchecking this will cause each new selection to overwrite the previous selection (unless you hold `Ctrl` while selecting) when using the *Query Prompt Selector* component at design time. **i Note** This option only affects the model at design time. It has no effect on an exported model at runtime.
Optional prompt	Check this to let the query run without needing an input for this filter prompt. **➡ Remember**

Option	Description
	Only filter prompts can be made optional; result object prompts are always mandatory.
Set default value(s)	Creates a pre-selected value for this filter prompt. For more information, see *To set default values for a filter prompt* [page 230].

- Click the radio button *Use Universe Prompts* to select from a table of predefined prompts (created by the universe designer). Note that the table may be empty if your universe has no predefined prompts.

Option	Description
Select from the table of prompts	The universe may contain several prompts. The table lists all universe prompts, along with the following information: ○ The prompt's name. ○ The prompt's question - this is the text that will appear on a *Query Prompt Selector* component bound to this prompt. ○ The type of data associated with the prompt, such as string or integer.
Use the selected universe prompt as optional	Check this to let the query run without needing an input for this filter prompt.

Related Links

To set default values for a filter prompt

When in the *Edit Prompt* dialog box, click browse ([...]) to open the *Select Parameter Value* dialog box.

You can set default values for your filter prompt so the dashboard user does not necessarily have to change or enter a new value at runtime.

1. Explore the object tree or list and select an object, then click the *Move* arrow ([>]).
2. Click *OK*.
   The selected object is now the default prompt value.

To set AND/OR relationships for filter objects

Filter objects can be related to each other with a logical AND relationship or a logical OR relationship. If you have more than one filter object in the Filters panel, a logical relationship between the objects is required. The default relationship is always AND.

To change the logical relationship between filters, do one of the following:

- Click the AND icon (And) to change the relationship to OR.

- Click the OR icon (Or) to change the relationship to AND.

To manage the sort orders of objects

> **i Note**

230   Dashboards and Presentation Design User Guide
Working with Data Sources

> The sort function is only available when using a relational data source. Sorting is not available for OLAP data sources.

On the *Build Query* pane of the *Add Query* dialog box you want to configure a sort order for your query's result objects.

1. To sort the objects, in the *Result Objects* pane, click *Sort* ( ).
   The *Sort* dialog box will appear.

2. Click *Insert sort object* ( ).
   The *Select an object* dialog box will appear with an expandable tree view of the objects in the universe.

3. Explore the tree view of the universe until you have found the object or objects you want to sort. Click *OK*.

   > ℹ **Note**
   >
   > You can use `Shift` + `Click` or `Ctrl` + `Click` to select multiple objects at once.

   > ℹ **Note**
   >
   > The *Select an Object* dialog box includes search, expand all, and contract all buttons to help browse the universe.

   > ℹ **Note**
   >
   > *OK* will be active only when you have chosen an object that can be sorted - some objects cannot be sorted, such as hierarchical or composite objects.

   All of the objects that you have selected for sorting will be listed in the *Sort* dialog box.

4. Select a sort object in the list and click the sort icon in the object's row to show a drop-down box

   ( ).

5. Select the object's sort order -- *Ascending* or *Descending*.

6. Click the up or down arrows ( ) to re-position the sort object within the list.

7. Click *OK*.

Each result object has both its own alphabetical ascending or descending sort order, and a position in the list of sort objects.

> ℹ **Note**
>
> The *Sort* dialog box also has *Delete all* ( ) and *Delete selection* ( ) buttons. These buttons remove sort objects from the list.

To remove objects

On the *Build Query* pane of the *Add/Edit Query* dialog box there will be at least one object in either the *Result Objects* panel or the *Filters* panel. To remove result objects or filters from your query, do one of the following:

- Drag the result object out of the pane. As you drag the object, the green icon ( ) will disappear. With this method, you can only remove one result object at a time.

- Select the result objects you want to remove and click *Delete Selected Objects* ( ).

> **i Note**
>
> You can use `Ctrl` + `Click` to select more than one object.

- To remove all of the result objects from the pane, click *Delete All* ( ).

Deleting result objects only removes them from your query. The objects will remain available in the Universe panel for re-selection.

To preview a query result

Previewing the result of your query is the fourth step of building a query, or the second editable pane when editing an existing query.

1. If your query contains prompts, you will be asked to enter values for the prompts. Once values have been chosen, the *OK* button becomes active.

   > **i Note**
   >
   > If your model has optional prompts or default values for all the prompts, the *OK* button will already be active.

2. If your query does not contain prompts, or once you have entered prompt values and clicked *OK*, review the data returned by your query, and do one of the following:

   - If you want to make any changes to your query's result objects or filters, click *Back* to return to the *Build Query* pane.
   - If you are satisfied with your query's result objects and filters, but want to customize the query's usage options, click *Next*. For more information about usage options, see *To set usage options* [page 233].
   - If you are satisfied with your query's result objects and filters and want to finish building your query, click *OK*.

     > **i Note**
     >
     > This will create the query with default usage options. You can always edit the query later to change the query's result objects, filters, and usage options.

Related Links

Setting usage options is the fifth step of building a query, or the third editable pane when editing an existing query. Setting usage options is optional; you can finish building your query in the previous pane (*Preview Query Result*).

- Choose *Refresh Options*:

  - *Refresh Before Components Are Loaded*
  - *Refresh Every...*
  - *Refresh on Trigger*

When Value Changes	Choose a changing value from the embedded spreadsheet
When Value Becomes	Choose a threshold value from the embedded spreadsheet

> **i** Note
>
> You must select a trigger cell in the embedded spreadsheet before the above options become active.

- Choose *Load Status*

  - Choose a *Loading Message*. Messages that are stored in cells in the embedded spreadsheet can be presented to the user during load or idle.
  - Choose an *Idle Message* from the embedded spreadsheet or write a message, then specify where the message should be inserted in the embedded spreadsheet. This would allow use of the message by a specific component that responds to that range.
  - Choose cursor and mouse behaviors.
  - Choose *Hide Error Message Window* and optionally choose a cell range in the embedded spreadsheet to receive the error message text.
- When you are finished configuring the query's usage options, click OK.
  The query is created and added to the *Query Browser*.

6.2.2.1.4   To delete queries

You want to delete an existing query.

1. On the *Query Browser*, select the query you wish to delete and click delete (✕).
2. If components are bound to the query, a warning dialog box listing the affected components will appear. Click *OK* to confirm the deletion.

> **i** Note
>
> The components bound to the query will lose their functionality when the query is deleted, but will not be removed from the canvas.

> **i** Note

Remember that components can be bound to a query either directly or indirectly (through the embedded spreadsheet). The warning dialog box may not list components bound indirectly to the query you are deleting.

Related Links

6.2.2.1.5   To refresh queries

You can refresh queries individually or all at once using the *Query Browser*. Refreshing a query causes it to refresh its connection to the selected data source and return the most up-to-date data available.

- Click the drop-down arrow beside *Refresh* ( ) and select one of the following:
    - *Refresh*
    - *Refresh with New Prompt Values*
    - *Refresh All*
    - *Refresh All with New Prompt Values*

    If you choose to use new prompt values, the *Answer Prompts* dialog box will open. Queries that are being refreshed that require prompt choices will be listed on the left panel and the range of possible choices will be shown on the right panel. Once all of the prompts have been answered, the *Run* button will become active.

Related Links

6.2.2.1.6   To place a prompt on the canvas

You have finished building your query and it includes one or more prompts. You want to bind the prompt to a *Query Prompt Selector* component so its result object or filter selection can be modified at runtime.

1. Click the query's expand arrow ( ).
2. Click the *Prompts* expand arrow ( ).
3. Click the name of the prompt, then drag the prompt onto the canvas.

    The cursor becomes a prompt icon ( ), and will create a *Query Prompt Selector* component already bound to that prompt.

6.2.2.1.7   To insert query result objects into the spreadsheet

You want to insert query results into the embedded spreadsheet so that other components can use or display the data returned by the query.

234

Dashboards and Presentation Design User Guide
Working with Data Sources

> **➡ Tip**
>
> To open the *Query Browser*, on the main menu, click ▶ *View* ▶ *Query Browser* ◢.

1. Click the query's expand arrow ( ).

2. Click the *Result Objects* expand button (▼ **Result Objects**).

3. Select the result object you want to insert into the spreadsheet.

4. Click the *Insert in Spreadsheet* expand button (Insert in Spreadsheet ▼).

5. Click *Select a Range* ( ) and use the mouse cursor to select a range of cells on the embedded spreadsheet.

6. Click *OK*.

6.2.2.1.8   To connect a prompt to the spreadsheet

You want values entered by a prompt to be inserted into the spreadsheet. This functionality is best used in conjunction with a *Query Prompt Selector* component bound to a prompt. This allows the *Query Prompt Selector*'s data, which can be entered at runtime, to be used by other components.

1. In the *Query Browser*, click the expand arrow ( ) of the query containing the prompt you want to connect.

2. Click the *Prompts* expand arrow (▼ **Prompts**).

3. Select the prompt object you want to connect to the spreadsheet.

4. Click the *Connect to Spreadsheet* expand arrow (Connect to Spreadsheet ▼).
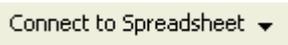
5. Click *Selected Values* ( ) and do one of the following:
   ○ Type the cell range directly into the input box.
   ○ Use the mouse cursor to select a range.

   The *Select a Range* dialog box will open.

6. Click *OK*.

7. Click *Insert List of Values* ( ) and do one of the following:
   ○ Type the cell range directly into the input box.
   ○ Use the mouse cursor to select a range.

8. Click *OK*.

Related Links

To place a prompt on the canvas [page 234]

7 Integrating with SAP Data Sources

SAP BusinessObjects Dashboards allows you to connect your model to external SAP data sources. These data sources include the following:

- The SAP BusinessObjects Business Intelligence platform
  SAP BusinessObjects Dashboards enables you to use data from managed SAP BusinessObjects Enterprise databases via a Live Office connections. You can also store Dashboards objects on the SAP BusinessObjects Business Intelligence platform where other users with appropriate permission can view them. For more information on creating a Life Office data connection, see *Using Live Office Data Connections* [page 237].
- SAP NetWeaver BW
  You can add one or more SAP NetWeaver BW connections to a model, configure them, link them to components, and publish your model on the SAP NetWeaver BW system as BW objects (TLOGO objects).

> ℹ **Note**
>
> When using a BEx query as a data source in your model, the SAP NetWeaver BW connection returns a snapshot of the BEx query pivot result set, as well as the keys for the characteristics and variables in one request. For more information on using a BEx query as a data source, see *To add or edit queries* [page 218].

7.1 Integrating with SAP BusinessObjects Business Intelligence platform

SAP BusinessObjects Dashboards enables you to create interactive presentations from managed BI platform data using SAP BusinessObjects Live Office connections. You can also store and manage Dashboards objects on the BI platform or Crystal Reports Server where any user with appropriate permissions can view these objects.

> ⚠ **Restriction**
>
> Dashboards supports managed Crystal reports only. If you have unmanaged files, you must move them to a managed environment to refresh Live Office connections in Dashboards. For more information about moving to a managed environment, see "Best Practices for Prompting" in the *SAP Crystal Reports 2011 User's Guide* available at *help.sap.com* under ▌ *SAP BusinessObjects* ❯ *All Products* ▐.

Using the Live Office connections in Dashboards and the Live Office Web Services in the BI platform or Crystal Reports Server, you can create and refresh models using managed BI platform data.

> ℹ **Note**
>
> Single Sign-On is not supported when connecting to the BI platform from Dashboards.

7.1.1 Using Live Office Data Connections

Live Office data connections allow you to create models using Excel spreadsheets constructed from managed Crystal Reports or BI platform data and refresh that data in the model from within the BI launch pad, or from outside the BI launch pad using a connection to the Crystal Reports Server or the BI platform.

Before you can add a Live Office data connection to your model, you must have a Live Office-enabled spreadsheet. A Live Office-enabled spreadsheet includes data that is actually a View created from either a managed Crystal Reports document or a managed BI platform document. You can create Live Office-enabled spreadsheets in Microsoft Excel outside of Dashboards and then import them into models to add Live Office data connections.

If you want to work with Live Office in Dashboards, you must have Live Office installed and, in Dashboards, Live Office compatibility mode must be enabled. For more information about enabling Live Office compatibility mode, see *To set preferences for the embedded spreadsheet*.

7.1.1.1 To create a Live Office-enabled spreadsheet in Dashboards

Creating a Live Office-enabled spreadsheet involves connecting to the repository where the data resides and inserting a Live Office object of the data in the embedded spreadsheet. Before you can insert the data into Dashboards, a Crystal report, Web Intelligence document, or universe query must be available and scheduled in the repository. For more information about creating and saving these documents to the repository, see the documentation for the product.

To create and work with Live Office-enabled spreadsheets in Dashboards, Live Office compatibility mode must be enabled in Dashboards. For information about enabling Live Office compatibility mode, see *To set preferences for the embedded spreadsheet* [page 174].

1. In the embedded spreadsheet toolbar, click *Live Office* and then click the type of data you want to insert (*Crystal Reports*, *Web Intelligence*, or *Universe Query*).

2. If the *Log on to SAP BusinessObjects Business Intelligence platform* dialog box opens, enter your logon credentials and click *OK*.

3. Follow the steps in the *Live Office Insert Wizard* to select and configure the data from the repository.
   For more information about using the *Live Office Insert Wizard*, see the *SAP BusinessObjects Live Office User Guide* available at *help.sap.com* under ▶ *SAP BusinessObjects* ▶ *All Products* ▶.

4. When you finish selecting and configuring the data, click *Finish*.

The report data is inserted into the embedded spreadsheet as a Live Office object.

You can now link the inserted data to components to display it in your model. To allow the data to refresh against the source data in the repository, you must also add a Live Office data connection to the model.

7.1.1.2 To add and configure Live Office data connections

To add Live Office data connections to models, SAP BusinessObjects Live Office must be installed on your machine and you must have a Live Office-enabled spreadsheet open in Dashboards.

Use a Live Office data connection when you want to use data from managed Crystal Reports or Web Intelligence documents in a model and make it refreshable against the source data.

> **i Note**
>
> The Live Office data connection option is not available until a Live Office-enabled spreadsheet is open.

1. Click ▶ *Data* ❯ *Connections* ◀.
2. In the *Data Manager* dialog box, click *Add* and select *Live Office Connections*.
3. On the Definition tab, set the following options:

Option	Description
Name	Displays the name of the Live Office connection.
Session URL	Verify that the default URL is correct for your web server.
	If your BI platform or Crystal Reports Server has a default Web Services installation, replace the web server in this default URL with the name of your Central Management Console (CMC).
Ranges	This option controls how many rows of data are returned. Set this range to include as many rows as can possibly be returned by this connection.
	> **i Note** > > It does not affect the number of columns.

4. Click the *Usage* tab and set the refresh and message options. For more information, see *To set refresh options* and *To create load and idle messages*.

The model is connected to a Live Office data source and, at runtime, the data will refresh as defined in the refresh options settings.

Related Links

Passing parameters to Live Office [page 238]

7.1.1.3 Passing parameters to Live Office

When a model has a Live Office data connection, you can set up the model to pass data values back to Live Office as prompt or parameter values. For example, you could add a selector component to the model so users can select prompt values at runtime. The selected value can then be sent back to the data source as the prompt or parameter value for retrieving data.

> **i Note**
>
> To insert and configure a Live Office object from within Dashboards, Live Office compatibility mode must be enabled. Alternatively, you can insert and configure the Live Office object in Microsoft Excel and then import the Live Office-enabled spreadsheet into Dashboards without enabling Live Office compatibility mode.

The overall process for setting up a model to pass parameters to Live Office:

1. Add a Live Office-enabled spreadsheet to Dashboards and enable Live Office compatibility mode.

   > ℹ **Note**
   >
   > The inserted Live Office object must be based on a report or document that is configured in the source location to accept and apply prompts or parameters.

2. Add a Live Office data connection.
3. Create the model based on the data in the Live Office object and add a selector component to allow users to select prompt values.
4. In the Live Office properties, link the prompts or parameters in the Live Office object to cells in the embedded spreadsheet.
5. Set the refresh options for the Live Office connection to update the data for the Live Office object when the trigger cell changes.

Related Links

To create a Live Office-enabled spreadsheet in Dashboards [page 237]
Using Live Office Data Connections [page 237]

7.1.1.3.1 To pass parameters to Live Office

To pass parameters to Live Office, the model must contain a Live Office connection.

When you adjust the Single Value component in the SWF file, you receive a prompt to log on to your BI platform system. When you have successfully logged on, the data is refreshed and the new value or values are reflected in your SWF file.

1. Place a Single Value component such as a dial or a slider on the canvas.
2. On the *General* view of the Single Values properties browser, click the Cell Selector button next to the *Data* field.
3. Click the cell that contains the bound parameter and then click *OK*.
4. Add and configure a Live Office connection. For more information, see *To add and configure Live Office data connections*.
5. On the *Usage* tab, click the Cell Selector button beside the *Trigger Cell* field.
6. Click the cell that contains the bound parameter and then click *OK*.
7. Ensure that *Trigger on Change* only is selected.

   Configuring the Trigger Behavior this way causes the SWF to refresh each time there is a change to the cell and in this case, the parameter value.
8. Click the *Preview* button to generate a SWF file.

7.1.1.4 Updating Live Office connection settings after system migration

When content is migrated from one BI platform system to another, SAP BusinessObjects Dashboards uses the Access URL property of SAP BusinessObjects Business Intelligence platform Web Services application to handle the changes made to the Live Office Web Services location. Check with your system administrator to ensure that the Access URL is configured correctly in the Central Management Console.

7.1.2 Embedding Flash objects and Dashboards objects in models

Models stored in the SAP BusinessObjects Business Intelligence platform can be configured to embed Flash objects and Dashboards objects that are also in the BI platform. To do this, you must add the DocumentDownload function to the URL that is used to call the file. The DocumentDownload function provides a standard way of opening Flash objects and Dashboards objects which have been added to the BI platform.

DocumentDownload opens the secondary Flash file within Dashboards, such as in a *Slide Show* component, which is useful when creating a "nested" or "parent-child" model within the BI platform.

> i Note
>
> Logon credentials are not passed from the parent Dashboards model file when calling a secondary Flash file. To send logon credentials from DocumentDownload, append CELogonToken to the URL.

> ➡ Tip
>
> To obtain the document ID, navigate to the document in BI launch pad, move your mouse over the document name hyperlink, and look for the ID number in the browser's status bar. You can also obtain the document ID from the Central Management console. For an example of the URL syntax, right-click a document and select *Document link*.

URL syntax for DocumentDownload in SAP BusinessObjects Enterprise XIR2

```
http://<<server>>:<<port>>/businessobjects/enterprise115/desktoplaunch/opendoc/
documentDownload?<parameter1>&<parameter2>&...&<parameterN>&CELogonToken=<<token>>
```

Replace the variables as indicated in the following table:

Variable	Replace with
`<<server>:<port>>`	The name and port of the machine that houses the BI platform repository where the document is stored.
`<<document ID>>`	When using the iDocID parameter, replace `<<document ID>>` with the document ID number of the document.

Variable	Replace with
`<<path>>`	When using the sPath parameter, replace **`<<path>>`** with the path to the document in the repository. Enclose each subfolder name in brackets, and separate subfolder names with commas. ⓘ Note Replace any spaces with plus signs.
`<<document name>>`	When using the sDocName parameter, replace **`<<document name>>`** with the name of the document in the repository. ⓘ Note Replace any spaces with plus signs.

ⓘ Note

Do not add `[Public+Folders]` to the path; start with the name of the first subfolder within `Public Folders`.

```
Public Folders
    folder 1
       folder 1.1
          folder 1.1.1
```

If your document is in folder 1.1.1, you would set sPath to: `[folder+1],[folder+1.1],[folder+1.1.1]`.

Example

DocumentDownload URL for SAP BusinessObjects Enterprise XIR2

```
http://cdi5boe:8080/ businessobjects/enterprise115/desktoplaunch/opendoc/
documentDownload?iDo
cID=10348&CELogonToken=SERVER%408757Jxnge3URUJ8L4
```

The following table identifies each of the parameters used in the sample URL:

Protocol:	http:// (can be https://)
Server name:	cdi5boe:
Port:	8080/
Function:	businessobjects/enterprise115/desktoplaunch/opendoc/documentDownload?
Document ID*:	iDocID=10348& ⓘ Note

	iDocID is the ID or the CUID of the SWF file.
LogonCredentials*:	CELogonToken=SERVER%408757JxmheURUJ8L4

> **i Note**
>
> Parameters marked with an asterisk (*) are required with the documentDownload function.

URL syntax for DocumentDownload in SAP BusinessObjects Enterprise XI3x and SAP BusinessObjects Business Intelligence platform 4.0

> **i Note**
>
> Additional parameters can be passed into the SWF file by adding FlashVar ranges to the URL, for example: "Range1=yyy&Range2=zzz&"

To pass the CELogonToken from the parent model to a child SWF, a Flash Variable connection must be added to the child model. Open the Data Manager, click Add, and select Flash Variables. In addition to the connection required for the CELogonToken, any additional parameters must have Flash Variable connections. Any additional parameters are converted to FlashVars by the Flash Player. For more information about configuring Flash Variable connections, see *To configure Flash variable connections*.

You can enter the URL directly into the URL field of a Slide Show component in the model or, to allow users to change the URL dynamically, configure the Slide Show component to retrieve the URL from a cell in the embedded spreadsheet. Then configure the Selector component using the Excel CONCATENATE function, to enter different URLs in that cell when the user makes a selection.

For XI3x, the URL syntax is as follows:

```
http://<<server>>:<<port>>/Xcelsius/opendoc/documentDownload?iDocID=<<document
ID>>&CELogonToken=<<token>>
```

For XIR4, the URL syntax is as follows:

```
http://<<server>>:<<port>>/BOE/Xcelsius/opendoc/documentDownload?iDocID=<<document
ID>>&CELogonToken=<<token>>
```

Replace the variables as indicated in the following table:

Variable	Replace with
`<<server:port>>`:	The name of the machine that houses the BI platform repository where the document is stored.
`<<document ID>>`	When using the iDocID parameter, replace `<<document ID>>` with the document ID number of the document.

Variable	Replace with
	ⅰ Note You can use the sDocName (document name) parameter as an alternative to the iDocID.
\<\<path\>\>	When using the sPath parameter, replace **\<\<path\>\>** with the path to the document in the repository. Enclose each subfolder name in brackets, and separate subfolder names with commas. **ⅰ Note** Replace any spaces with plus signs.
\<\<document name\>\>	When using the sDocName parameter, replace **\<\<document name\>\>** with the name of the document in the repository. **ⅰ Note** Replace any spaces with plus signs.
\<\<token\>\>	Log on credentials that are automatically sent from the Live Office BI platform as CELogonToken. **ⅰ Note** To use the logon credentials, add a Flash Variable connection called CELogonToken to the child model and link it to an empty cell in the spreadsheet. Then replace **\<\<token\>\>** with a reference to the table cell. When you export the SWF file to BI launch pad, the logon token is appended to the DocumentDownload URL.

ⅰ Note

Do not add `[Public+Folders]` to the path; start with the name of the first subfolder within `Public Folders`.

```
Public Folders
   folder 1
     folder 1.1
        folder 1.1.1
```

If your document is in folder 1.1.1, you would set sPath to: `[folder+1],[folder+1.1],[folder+1.1.1]`.

7.1.3   Embedding BI platform documents in models

Models can be configured to embed reports or other documents that are on the SAP BusinessObjects Business Intelligence platform. To do this, you must add the OpenDocument API (Application Programming Interface) to the URL that is used to call the document.

The OpenDocument API provides a standard method for opening documents and reports that have been added to the BI platform, using a simple URL. OpenDocument allows you to specify, not only the report, but also report parameters so the report returns only the desired data.

You can enter the URL directly into the URL field of a *URL Button* component in the model. To allow users to change the URL dynamically, configure the *URL Button* component to retrieve the URL from a cell in the embedded spreadsheet, then configure a selector component to enter different URLs in that cell when the user makes a selection.

> ➡ Tip
>
> You can also use OpenDocument with primary models that are not in the BI platform, if they have secondary files in the BI platform. The user will be prompted for BI platform credentials prior to launching the document.

For more information about creating OpenDocument URLs, see the *Viewing Documents Using OpenDocument* guide, available on the SAP Help Portal (*http://help.sap.com/bobip40/*).

URL syntax for OpenDocument in SAP BusinessObjects Business Intelligence platform 4.0

The URL syntax is as follows:

```
http://<<server>>:<<port>>/BOE/OpenDocument/opendoc/openDocument.jsp?
sIDType=CUID&iDocID=<<document ID>>&<<parameter1>>&<<parameter2>>&...&<<parameterN>>
```

Replace the variables as indicated in the following table:

Variable	Replace with
<<server>:<port>>	The name of the machine that houses the BI platform repository where the document is stored.
<<document ID>>	When using the iDocID parameter, replace **<<document ID>>** with the document CUID number of the document. ℹ Note When you view the document properties in BI launch pad, you can also see the document CUID. The CUID is unique so there is no need to use sDocumentName or sPath when using a CUID.

Variable	Replace with
	When you migrate your document from one system to another the CUID stays the same, which is why it is best to use the CUID.
`<<parameters>>`	For more information about parameters, see the *Viewing Documents Using OpenDocument* guide, available on the SAP Help Portal (*http://help.sap.com/ bobip40/*).

🍀 Example

Using iDocID to open a Crystal report by its document CUID

Given a Crystal report with a CUID of `Aa6GrrM79cRAmaOSMGoadKI`, the URL syntax is as follows:

```
http://mywebserver:8080/BOE/OpenDocument/opendoc/openDocument.jsp?
iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID
```

7.1.4   Storing models on the BI platform

When a model is saved on the SAP BusinessObjects Business Intelligence platform repository, it is saved as a Dashboards object that incorporates both the design-time and run-time models in one object. You can store and manage these objects on the BI platform or the SAP Crystal Reports Server.

ℹ Note

In earlier versions of the software, the Xcelsius object (design-time model) was stored separately from the Flash object (runtime model). Objects created in previous versions of the software may still be available in the repository and can be opened in Dashboards. However, to make use of the new features, you need to save these models as Dashboards objects. When you replace an Xcelsius object with a Dashboards object, you can choose to replace both the Xcelsius object and the corresponding Flash object so the Dashboards object uses the identification number from the original Flash object.

Once the objects are added to the BI platform repository, any user with appropriate permissions can access the files. You can manage objects saved to the BI platform through BI launch pad and the Central Management Console (CMC).

In BI launch pad, you can:

- View a list of Dashboards objects.
- Search for specific objects.
- Move or copy objects to different folders.

ℹ Note

Use the Lifecycle management console to ensure all BI platform objects are migrated with all dependencies intact. For information about migrating BI platform systems, see the *Lifecycle management console for SAP BusinessObjects Business Intelligence platform User Guide*.

If you have administrative rights on the CMC, you can also modify the edit and view rights for Dashboards objects within the CMC.

> **i Note**
>
> You cannot modify design-time models on the BI platform. To modify Dashboards objects that have been added to the BI platform, you must open the object within Dashboards. For more information, see *To open Dashboard Design objects from Enterprise folders*.

In Crystal Reports, you can also embed Dashboards objects in a Crystal report and have the objects use data from that report. To set up the Dashboards object to use source data from a Crystal report, add a *Crystal Reports Data Consumer* connection to the model.

When you save a model that contains a *Crystal Reports Data Consumer* connection as a Dashboards object, you can embed the object into the Crystal Reports source data file. For more information about working with Crystal Reports, see the *SAP Crystal Reports 2011 User's Guide* available on the SAP Help Portal (*help.sap.com*) under ▌▶ *SAP BusinessObjects* ▶ *All Products* ▌▶.

Related Links

To configure Crystal Reports Data Consumer connections [page 199]

7.1.4.1   To save models to BI platform folders

When you create a new model and save it to the platform, it is saved as a Dashboards object that includes both the design file (XLF) and the runtime file (SWF). However, if you open an existing Xcelsius object (XLF file) from the platform, when you save it, you can choose to save it to the new Dashboards object format and retain the existing runtime file (SWF) or to save it in the new format and delete the existing runtime file (SWF). Since the existing runtime file is not updated to reflect the changes made to the Xcelsius object (now Dashboards object), deleting the SWF file avoids confusion about which file is correct (the new Dashboards object or the SWF file).

1.   After creating or modifying a model, click *File* and select one of the following options:

Option	Description
▌▶ *Save To Platform* ▶ *Mobile Only* ▌▶	Use this option if you have created a new model, or have opened and modified an existing Dashboards model. Features unsupported on mobile devices are omitted. > **i Note** > For more information about unsupported features, see *Using the Mobile Compatibility panel* [page 267].
▌▶ *Save To Platform* ▶ *Desktop Only* ▌▶	Use this option if you have created a new model, or have opened and modified an existing Dashboards model.
▌▶ *Save To Platform* ▶ *Desktop and Mobile* ▌▶	Use this option if you have created a new model, or have opened and modified an existing Dashboards model. In the mobile file, features unsupported on mobile devices are omitted. > **i Note**

Option	Description
	For more information about unsupported features, see *Using the Mobile Compatibility panel* [page 267].
▶ *Save To Platform As* ▶ *Dashboards Object for Mobile Only* ◢	Use this option if you have opened and modified an existing Dashboards model, or have opened an Xcelsius object and want to save it as a new Dashboards object, but do not want to overwrite the existing object. Features unsupported on mobile devices are omitted. **ℹ Note** For more information about unsupported features, see *Using the Mobile Compatibility panel* [page 267].
▶ *Save To Platform As* ▶ *Dashboards Object for Desktop Only* ◢	Use this option if you have opened and modified an existing Dashboards model, or have opened an Xcelsius object and want to save it as a new Dashboards object, but do not want to overwrite the existing object.
▶ *Save To Platform As* ▶ *Dashboards Object for Desktop and Mobile* ◢	Use this option if you have opened and modified an existing Dashboards model, or have opened an Xcelsius object and want to save it as a new Dashboards object, but do not want to overwrite the existing object. In the mobile file, features unsupported on mobile devices are omitted. **ℹ Note** For more information about unsupported features, see *Using the Mobile Compatibility panel* [page 267].
▶ *Save To Platform As* ▶ *Dashboards Object to Replace Flash Object* ◢	Use this option if you opened an existing Xcelsius object and want to save it as a Dashboards object and delete the existing Flash object.

2. In the *Log On to SAP BusinessObjects Business Intelligence platform* dialog box, enter the following information:

Option	Description
System	Type the name of the Central Management console (CMC) where you want to save the file. The CMC stores information about the objects saved to the BI platform system. For more information about the CMC, refer to the BI platform documentation available on the SAP Help Portal (*help.sap.com*).
User name	Type your logon name.
Password	Type you password.
Authentication	Choose the appropriate authentication type.

3. Click *OK*.

4. If you are not replacing an existing file, select the folder where you want to save the Dashboards object, enter a file name, and click *Save*.

7.1.4.2 To open models from BI platform folders

Models may be stored on the platform as either Xcelsius objects or Dashboards objects. When you open a model from the platform, you can choose to open either object type.

1. Do one of the following:

   ○ To open an Xcelsius object, click ▶ *File* ❯ *Open from Platform* ❯ *Xcelsius object* ◢.

   ○ To open a Dashboards object, click ▶ *File* ❯ *Open from Platform* ❯ *Dashboards object* ◢

2. In the *Log On to SAP BusinessObjects Business Intelligence platform* dialog box, enter the following information:

Option	Description
System	Type the name of the Central Management Console (CMC) where you want to save the Dashboards object.
	The CMC stores information about the objects saved to the repository. For more information about the CMC, refer to the BI platform documentation available on the SAP Help Portal (*http://help.sap.com*).
User name	Type your logon name.
Password	Type you password.
Authentication	Click the appropriate authentication type.
	ⓘ Note
	If you are unsure of your logon information, contact your system administrator.

3. Click *OK*.
4. In the *Open* dialog box, select the folder that contains the object you want to open. Click *Open*.

The selected object opens in Dashboards.

7.1.5 Translating and globalizing models on the BI platform

If you want to make models available in multiple languages and formats, you can use the globalization features available through the BI platform. When models are saved to a BI platform repository, Dashboards offers two globalization features: regional formatting and text translation.

Regional formatting refers to how dates, times, numeric, and currency values are formatted. (For example, different regions may use different currency symbols or thousands separators). Text translation refers to converting the text displayed in the model to different languages.

To translate or globalize models, use the following overall process:

- In Dashboards:

  1. Set the Global Format preferences. See *To set default currency formats* [page 250].
  2. Create a model and set the text formats for each component element that will use regional formats. See *To format text elements* [page 43].
  3. Set the translation preferences for the model and specify any spreadsheet ranges that need to be translated. See *Setting up translation for models* [page 250].

4. Save the model to the BI platform repository. See *To save models to BI platform folders* [page 246].
* In the translation management tool:

> **i Note**
>
> For more information about using the translation management tool, see the *Translation management tool* documentation, available on the SAP Help Portal (*help.sap.com*) under ▶ *SAP BusinessObjects* ▶ *All Products* ▶.

1. Import the file from the Central Management Console (CMC).
2. Open the *Language Management* view and add languages to the *Selected Languages* list.
3. Open the Translation Editor or the *Text Editor* view and translate the text.
4. Use the Translation Editor to set the format options for numbers and date/time values.
5. Save the file.

7.1.5.1   Setting up models for regional formatting

When a model is saved to a BI platform repository, elements that use default format options automatically use the regional data formats associated with the preferred viewing locale specified for the user. However, if an element format is set to use a specific value, that setting will not be changed as part of regional formatting.

The following Excel-formatted values can be changed to suit regional formats based on the user's preferred viewing locale:

> **i Note**
>
> For currency values, regional formatting does not affect the currency symbol. Since regional formats do not apply currency exchange rates the values stay in the currency specified at design-time. Only the numeric format is changed to reflect regional preferences, for example, whether the symbol is before or after the number, the punctuation used, and so on.

Element	Example
Long Month	January, February, March
Short Month	Jan, Feb, Mar
Long Day	Sunday, Monday, Tuesday
Short Day	Sun, Mon, Tue
Date Separator	"/" as in 01/01/1900
Time Separator	":" as in 12:00:00
Decimal Separator	"." as in 1.5
Thousands Separator	"," as in 1,000,000

When you configure component properties in a model, you can set the text format for each element. To allow regional formats to apply, set the text format options to default values. To format the text correctly for the preferred language on your system, set the global text formats for the model.

7.1.5.1.1 To set default currency formats

To have a specific format used for currency when components are added to models, you can set up the default currency format. When you set the component properties, you can change these settings for the selected component or allow the component to use regional settings.

> ℹ **Note**
>
> Regional settings do not affect the currency symbol. Since regional formats do not apply currency exchange rates the values stay in the currency specified at design-time.

1. Click ▶ *File* ❯ *Preferences.* ❯
2. In the *Preferences* dialog box, click *Language*.
3. Under *Currency*, set the following options:

Option	Description
Negative Values	Specify the format to use for negative currency values.
Decimal Places	Select the number of decimal places to display for currency values.
Prefix	If desired, type the symbol to appear before the numbers in currency values.
Suffix	If desired, type the symbol to follow the numbers in currency values.

4. Click *OK*.

7.1.5.2 Setting up translation for models

Most text entered in the component properties can be translated, including titles, subtitles, series names, category names, and labels. To translate strings in the embedded spreadsheet, you must create a group for the cells and include them in the translation settings.

> ℹ **Note**
>
> Data added to the spreadsheet through an external data connection is not translated. To be translated, the data must be in the model at the time the model is saved to the repository.

> ℹ **Note**
>
> Text strings included in formulas are not translated.

> ➜ **Tip**
>
> When text is translated, the string length may change, which could affect the layout and design of the model. For example, a model designed in English might contain a label for "Sales", which is five characters. When translated into German, the label would be "Verkaufe", which is eight characters. To reduce the impact of changing word lengths on the model layout, set the translation settings to limit for the number of characters in each string. Select ▶ *SAP* ❯ *Translation Settings* ❯ to access translation settings.

7.1.5.2.1   To configure translation settings

1. Click ▎ *File* ▸ *Translation Settings* ▎.
2. In the *Translation Settings* dialog box, select *Enable translation of component text properties*.
3. To specify ranges in the spreadsheet to be translated, do the following:

    a) Below the *Groups* list, click the *Add* button (⊞).
    b) In the *Name* box, type a name for the selection.
    c) In the *Source Text* box, type the range of cells to be translated or click the *Cell Selector* (⬛) and select the cells in the embedded spreadsheet.
4. Click *OK*.

7.2   Integrating with SAP NetWeaver BW

In SAP BusinessObjects Dashboards, you can add one or more data connections to SAP NetWeaver BW, configure the connection and link it to components, and then publish your model on the SAP NetWeaver BW system as BW Objects (TLOGO objects). Published Flash models can be assigned to NetWeaver Enterprise Portal roles and accessed through an iView. These models support lifecycle through Data Warehousing transport of BW Objects.

The SAP NetWeaver BW connection returns a snapshot of the BEx query pivot result set, as well as the keys and value helps for the characteristics and variables in one request.

> ℹ **Note**
>
> Business Data Warehouse level authorization applies.

In addition, the connection provides metadata, such as the query and InfoProvider names, query filters, key date, refresh date, hierarchies, and the current message log.

> ℹ **Note**
>
> If you want to use a BEx query as a data source, you must use the *Query Browser* instead of the *Data Manager*. For more information, see *To add or edit queries* [page 218].

Related Links
To preview Dashboards models with SAP NetWeaver BW data [page 257]

7.2.1   Before connecting to SAP NetWeaver BW systems

To use the SAP BusinessObjects Dashboards BW connector, you must perform the following procedures:

*   Install the SAP GUI 7.10 Support Package 12 (or higher) with the BI 7.0 Add-On on the same machine as Dashboards.

- Ensure that NetWeaver Java and ABAP 7.0 ehp1 SP5 are installed.
- Ensure that Single Sign-On user mapping is established between the NetWeaver Java Application Server and the NetWeaver ABAP system.

For more information, see the SAP NetWeaver BW documentation at *help.sap.com*

7.2.2 To connect to SAP NetWeaver BW systems

When you enter a command that requires a connection to the SAP NetWeaver BW system, SAP BusinessObjects Dashboards attempts to connect to the system automatically. If authorization is required, you are prompted to log onto the SAP NetWeaver BW system. The following actions require a connection to the SAP NetWeaver BW system:

- Click ▶ *SAP* ❯ *Open* ◢
- Click ▶ *SAP* ❯ *Publish* ◢
- Click ▶ *SAP* ❯ *Publish As* ◢
- Click ▶ *Data* ❯ *Connections* ◢ and add and configure an *SAP NetWeaver BW connection*

> ℹ **Note**
>
> If Single Sign-On has been set up between SAP NetWeaver Java AS and SAP NetWeaver ABAP, you are prompted for authorization only once by Dashboards

7.2.3 To disconnect from the SAP NetWeaver BW system

If you are connected to an SAP NetWeaver BW system, you can connect and publish to that system. If you want to connect or publish to a different SAP NetWeaver BW system, you must first disconnect from the current SAP system.

1. Click ▶ *SAP* ❯ *Disconnect* ◢.
2. Click *OK*.

The connection with the SAP NetWeaver BW system ends. You can now connect to a different SAP system or change the user authentication.

7.2.4 To configure SAP NetWeaver BW connections

Use the SAP NetWeaver BW connection to create a connection to the Business Warehouse data and retrieve query data, information about the current characteristic selection, and the available values for the characteristics in the query. Each data connection supports one query. If you need the data from more than one query in a model, you can create multiple connections.

> ℹ **Note**

In addition to the items you configure and preview when you set up an SAP NetWeaver BW connection, the returned result set includes at least two rows of data that provide more details about the results. For example, these rows contain the descriptions, as well as information about the key figure. The result set also includes a row representing the Overall Result. If you don't want the result set to include the Overall Result row, in the BW query add the option *Result Rows* to the option *Always Suppress* for each characteristic.

1. Click ▶ *Data* ❯ *Connections* ◀.
    The *Data Manager* dialog box opens.
2. In the *Data Manager* dialog box, click *Add* and select *SAP NetWeaver BW Connection*.
3. On the *Definition* tab, set the following options:

Option	Description
Name	A name for the connection.
System	The name of the SAP system you are currently connected to. ℹ **Note** You can connect different components in a model to different queries or query views, but all queries and query views must be on the same SAP system.
Query	The name of the query or query view that is currently selected. Click *Browse* to select a different query or query view on the connected SAP system. ℹ **Note** If you replace an existing query, all links between the query data and ranges in the embedded spreadsheet are deleted.
Return Data As	From the list, select one of the following options to set the format for retrieved data: ○ *Raw* — data is not formatted and formats are applied in Dashboards. This option is best suited for use with Excel calculations or charts. ○ *Formatted* — data uses the BEx format and includes information such as sealing factor, number of decimals, currency, and sign change. This option is best suited for use with text-based components, such as the *Label*, *Spreadsheet*, *Table*, and *Scorecard* components.
Input Values	Lists the input-enabled values and selectable characteristics that are available to customize the data returned from the query or query view. ○ Filters — Lists the characteristics and key figures included in the query and that are available to filter the data returned from the query or query view. Use filters to allow model users to navigate into slices of the overall dataset. ○ Variables — Set the values for mandatory and optional query variables. ℹ **Note** Mandatory variables are initialized before the model loads and are displayed through standard BEx dialog boxes. Variables must have proper default values and exit variables must return a valid value (in internal format) for all cases.

Option	Description
	  To set a value for each input, select the input object from this list and then in the *Read From* box, type a value or click the *Cell Selector* (▦) to select a cell from the embedded spreadsheet where the selected object will retrieve its value.

Output Values	Lists the values returned by the query or query view, as well as any filters that can be applied to the data after it is returned.
	
	○ Cross-tab data — Returns a snapshot of the query or query view pivot table. The Header rows contain information about unit and currency, as well as the scaling factor. Basic, restricted, and calculated key figures are available cross-tab data. Cross-tab data reflects the Characteristics Value Display settings, such as Result Rows or Sort Characteristics.
	○ Characteristics — Returns characteristics and structures the value help text and keys for the filters applied in the *Input Values* box, and the hierarchy information.
	Each characteristic includes the following items:
	
	    ○ Description – Represents the description of the characteristic.
	    ○ Current Filter Value (Text) – Represents the technical name for the characteristic.
	    ○ Current Filter Value (Key) – Shows the current filter selection for the characteristic.
	    ○ Value Help – The options available under this category depend on the InfoObject. They represent the options for the characteristic's descriptions and all the available display attributes. You can use this option to create a list box to allow users to select values.
	○ Variables — Returns the variable help texts and keys, as well as the applied filters.
	○ Static Filters — Returns any restrictions defined in the BEx query designer. These restrictions cannot be changed within Dashboards.
	○ Information — Returns general query text elements, such as name, last refresh and data update, and key date value.
	
	ℹ Note
	
	Key date value cannot be set through the Dashboards BEx query connection.
	
	○ Message — Returns error text messages from the SAP NetWeaver BW system.
	
	To set a location to store each output value, select the output object from this list and then, in the *Insert In* box, click the *Cell Selector* (▦) to select a cell from the embedded spreadsheet to store the values retrieved from the query or query view.
	
	ℹ Note

Option	Description
	By default, Dashboards allows up to 512 rows to be retrieved and inserted into the embedded spreadsheet. To increase the number of rows allowed, see *To set preferences for the embedded spreadsheet*.

4. Click the *Data Preview* tab.

5. Click *Refresh Data Preview*.

    The sample data is generated and displayed to show the data structure.

> **i Note**
>
> The data preview does not retrieve data from the BEx query, and filters and parameters set on the *Definition* tab are not applied.

6. View and adjust the data structure as desired using the following options:

Option	Description
Tree view in left panel	Expand the tree to show the Columns, Rows, and Free Characteristics. By default, they are set as they were defined in the BEx query designer.
Up and Down buttons	To change the order of the data, select an item in the tree and click the *Up* or *Down* buttons to move the item up or down one position within the list. After changing the order of items, click the *Refresh Data Preview* button to see the changes.
Move Item menu	To move items to different structure element, drag the item to the new location or select the item and click *Move Item* ( ), then one of the following options: ○ *Move to Columns* -- moves the data to a column structure. ○ *Move to Rows* -- move the data to a row structure. ○ *Move to Free Characteristics* -- moves the data to a free characteristic structure. ○ *Reset Query Defaults* -- sets the structure back to how it was defined in the query. After changing the order of items, click the *Refresh Data Preview* button to see the changes.

7. Click the *Usage* tab to set the refresh and message options.

> **i Note**
>
> The *Refresh Before Components are Loaded* option executes the query for the first time before any input value filters can be specified by the user. To avoid retrieving large amounts of data, check the *Use Default Query Data* option to use the BEx query default values.

For more information about setting refresh and message options, see *To set refresh options* and *To create load and idle messages*.

Once your model is configured and connected to the SAP NetWeaver BW system, you can publish it to the SAP system.

7.2.4.1 Working with queries modified in SAP Business Explorer

When you create a model with an *SAP NetWeaver BW Connection*, the query data is read at the time the data connection is defined. If the query is subsequently changed in SAP BEx Query Designer, Dashboards does not notice the changes but, when the model runs, it will use the most current query. If the changes do not affect the query structure and adjust only settings that do not directly affect the model logic or structure, you do not need to update the model and the query changes will automatically take effect when the model runs. However, if the query definition or structure has changed, the definitions in the model may no longer match the query.

If the query definition or structure changes, check the definition in the model and, if necessary, re-import the query information by replacing the query in the data connection.

> **i Note**
>
> When you change the query in the data connection, all links to the embedded spreadsheet are removed and you will need to re-map the data to cells and cells ranges.

The following query changes do not affect the query structure and it is not necessary to update the model after making these changes:

- Add/modify a static filter
- Add/modify default values
- Modify settings for sorting characteristics
- Change the definitions of restricted key figures

The following query changes might affect the query structure. Check the model design to determine if you will need to update the query:

- Add/remove characteristic subtotals
- Add/remove free characteristics
- Change variable values

The following query changes will affect the model structure. After making these changes, re-import the query into the data connection:

- Modify characteristics display options (no display, key, text, key and text, text and key)
- Add/remove rows or columns

7.2.5 To publish models to SAP NetWeaver BW systems

Use this option to generate a SWF and publish the Dashboards model to the SAP NetWeaver BW system. Publishing a model allows it to be transported and translated like other BEx Objects. When a model is published, Dashboards publishes three files: the XLF, an SWF, and a TXT file. The TXT file is required for translation on the SAP NetWeaver BW system.

> **i Note**
>
> You must publish the model on the same SAP system that the SAP NetWeaver BW query connections connect to.

1. Do one of the following:

   ○ To save a new model, click ▶ *SAP* ❯ *Publish* ◤.

   ○ To save an existing model with a different name or location, click ▶ *SAP* ❯ *Publish As* ◤.

2. If you are not connected to the SAP system, the *SAP Logon* dialog box opens. Select the appropriate SAP system and enter your credentials. Click *OK.*

> **i Note**
>
> If you are connected to a different SAP system than you want to publish to, you must disconnect from the SAP network before continuing these steps.

   The *Open* dialog box opens.

3. Navigate to the location you want to save the file and enter a filename. Click *OK.*

The SWF file is published to the SAP NetWeaver ABAP system.

7.2.6   To preview Dashboards models with SAP NetWeaver BW data

After an SAP BusinessObjects Dashboards model with an SAP NetWeaver BW connection is published to an SAP system, you can use this procedure to preview it with data from the connected SAP system.

> **i Note**
>
> You can use the *Preview* button in Dashboards to view the model configuration and to test basic workflow, but the data from the SAP NetWeaver BW queries and query view result sets are not available.

1. Click ▶ *SAP* ❯ *Launch* ◤.
   A web browser session opens with the SAP NetWeaver Portal Logon screen.
2. Enter your portal credentials and click *Login*.

Once you are logged in, the model loads inside a Dashboards BW template, which contains a single model. You can use this URL in an iView template to add the model to an SAP NetWeaver Portal role.

7.2.7   To open XLF files from SAP NetWeaver BW servers

1. Click ▶ *SAP* ❯ *Open* ◤.
2. The *SAP Logon* dialog box opens. Select the appropriate SAP system and enter your credentials. Click *OK.*
   The *Open* dialog box opens.
3. Select the Dashboards file that you want to open and click *OK*.

7.2.8 Translating and globalizing Dashboards models on SAP NetWeaver BW systems

When you save a model to an SAP system, you can enable translation settings so text in the model can be translated as a BW object. When text is translated, the string length may change, which could affect the layout and design of the model. For example, a model designed in English might contain a label for "Sales", which is five characters. When translated into German, the label would be "Verkaufe", which is eight characters. These additional three characters might affect the layout of the label and fields around it. You can limit this impact when you set up translation by setting a limit for the number of characters in each string.

Dashboards also displays numeric, currency, date, and time values in a format specific to the location set based on the SAP NetWeaver BW user locale. The following Excel-formatted elements can also be changed dynamically based on your SAP profile locale when running the SWF file:

Element	Example
Long Month	January, February, March
Short Month	Jan, Feb, Mar
Long Day	Sunday, Monday, Tuesday
Short Day	Sun, Mon, Tue
Date Separator	"/" as in 01/01/1900
Time Separator	":" as in 12:00:00
Decimal Separator	"." as in 1.5
Thousands Separator	"," as in 1,000,000
Currency Symbol	£100, 50¢, 10F50

The following restrictions apply to Dashboards globalization settings:

- The order that value elements appear are not changed at runtime, even if the location setting where the model runs uses a different element order. For example, if the model is created in an operating system where the location settings use the date format MM/DD/YYYY. When the model is run in an environment that uses the date format, DD/MM/YYYY, the order of the month and day are not updated for the new location. This restriction also applies to whether currency symbols are at the front or the end of the value.

- Only values that are in the embedded spreadsheet are globalized. If values are manually entered into the *Properties* panel or into the SWF at runtime, those values are not changed according to the location setting when the model runs.

- When designing and editing models, changing location settings does not affect existing, bound components. If you change the location setting and want the change to apply to existing components, you must rebind the components to the embedded spreadsheet.

- If a region-specific format is applied to cells in the embedded spreadsheet, the format for those cells does not change when the model is run in a different location. For example, in Microsoft Excel, when you apply the Currency format, you can choose None, a plain symbol, or symbols associated with a specific region. If you select the plain symbol, for example €, the symbol changes depending on the location setting when the model runs. But, if you choose a symbol for a specific region, for example *€ French (France)*, the symbol does not change when the model is run in a different location.

7.2.8.1   To set up translation for models on SAP NetWeaver BW systems

Before you set up translation for a model, it must have an SAP NetWeaver BW connection, be open on the canvas, and be meant to be published to the SAP NetWeaver BW system.

You can set up static text for translation so translated text displays when the Flash model is viewed. To allow translation, you must designate the cells in the spreadsheet that can be translated. When the model is published, the content in the specified cells is available to SAP NetWeaver BW. Once translation is set up, translated text, based on the SAP NetWeaver BW user locale, is passed to the model at run-time. For more information about the translation options, see the *SAP NetWeaver BW* documentation, available on the SAP Help Portal (*help.sap.com*).

> **i Note**
>
> Text translation is not available from Dashboards.

1. Click ▶ *SAP* ▶ *Translation Settings* ▶ .
2. If you are not connected to the SAP system, the *SAP Logon* dialog box opens. Select the appropriate SAP system and enter your credentials. Click *OK.*

   > **i Note**
   >
   > If you are connected to a different SAP system than you want to publish to, you must disconnect from the SAP network before continuing these steps.

3. In the *Translation Settings* dialog box, select the *Enable Support for Translation* option.
4. Below the *Ranges* list, click *Add.*
5. In the *Name* box, type a name for the range.
6. Beside the *Range* box, click the *Cell Selector* (🔳) and select a cell.
7. To translate multiple texts, Repeat steps 4 to 6.
8. To minimize the impact of translated words on the model design and spacing, do one of the following:

   > **i Note**
   >
   > This setting is not supported in SAP NetWeaver 7.0 Enhancement package 01 SP 05. However, it is a good practice to set this option for models, so they will work correctly once the option is supported by SAP NetWeaver and you won't need to republish them.

   ○ To enter specific values, beside the *Maximum Number of Characters* box, click *Manual Edit* (🖉) and, on the *Number of Characters Settings* dialog box, specify the maximum number of characters allowed for each string.
   ○ To select values from the embedded spreadsheet, beside the *Maximum Number of Characters* box, click the *Cell Selector* (🔳) and select the cells that will store the values for the maximum number of characters.
9. Click *OK*.

8   Exporting and Deploying Models

You can export your model once you have finished configuring your model's components, bindings, and data sources. For an exhaustive list of supported file extensions, see *Exporting models* [page 260].

When deploying a model as a SWF file, Adobe Flash Player security restrictions may apply. This section covers Adobe Flash Player restrictions, as well as restrictions if your model uses FS Command connections, to let the dashboard user view the exported Dashboards file on their local drive.

8.1   Exporting models

This table lists the supported export formats:

Export format	SAP Crystal Presentation Design	SAP Crystal Dashboards, personal and departmental editions	SAP BusinessObjects Dashboards
Macromedia Flash (SWF)		☑	☑
Adobe AIR		☑	☑
Adobe Acrobat 6 (PDF)	☑	☑	☑
Adobe Acrobat 9 (PDF)	☑	☑	☑
HTML			☑
Microsoft PowerPoint			☑
Microsoft Outlook			☑
Microsoft Word			☑
BusinessObjects Enterprise			☑

For all formats, the model is saved as a Flash SWF file and then inserted into the selected program. As a SWF file, the model does not require Dashboards to be on the local drive to run. If the model contains universe or BEx query data, you must also save the model to SAP BusinessObjects Business Intelligence platform. For more information about using universe queries, see *To browse queries* [page 214].

> ℹ **Note**
>
> The SWF file created by Dashboards includes an item in the SWF context menu that appears when users right-click the open SWF file. The item identifies Dashboards and links to an SAP website. The information that appears in this context menu is stored in the following text file: `....\Xcelsius\assets\RightClick.txt`. To change the information, modify the `RightClick.txt` file using the following format

260

Dashboards and Presentation Design User Guide
Exporting and Deploying Models

`<<menuItemName>><<URL> >`where `<<menuItemName>>` is the name of the item in the context menu (for example, *About Dashboards*) and `<<URL>>` is the address for the website that opens when users click the menu item. Re-export the SWF to apply the changes.

8.1.1   To export models

If you are exporting to Adobe AIR, the Adobe Flex 3.0 SDK is installed on your system. For more information, see *To setup your system for Adobe AIR*.

Not all export options are available in all versions of the program. For a detailed table of supported export formats, see *Exporting models* [page 260].

1. Click ▸ *File* ▸ *Export* ◂ and then select one of the following options:

Option	Description
Flash (SWF)	Generates an Adobe Flash (SWF) file that can be executed locally with the standalone Flash player, opened in a browser, used as part of a web site, or imported into a number of different programs. ⚠ **Caution** Any external files are saved in a folder in the folder where the model is saved. For the model to work properly, the external file folder must remain in the same folder as the model.
AIR	Exports the model to a browserless Runtime for Rich Internet Applications (RIAs) format that can be run on desktops across multiple operating systems. Before you can export to Adobe AIR, you must configure your system. For more information, see *To setup your system for Adobe AIR*. ℹ **Note** To run AIR files, you must install the *AIR player from the Adobe website*.
HTML	Embeds the Adobe Flash SWF model in an HTML file to ensure users can view the model by opening one file even if their computer is not configured to open Flash files in the browser. ⚠ **Caution** When the model is exported, the software creates an HTML file and an SWF file with the same name and saves them in the selected directory. For the model to work properly, these files must be in the same folder.
PDF	Generates a PDF file that can be viewed locally, opened in a browser, or deployed on a web site. The model is embedded in the PDF and is dynamic and interactive. ℹ **Note** You can save your PDF files to be compatible with either Acrobat 6 (and later versions) or Acrobat 9. Acrobat 9 requires users to have the latest version of Acrobat Reader installed to view the model. In the *Save As* dialog box, in the *Save as Type* list, select the Acrobat version. ⚠ **Caution**

Option	Description
	Any external files are saved in a folder in the folder where the model is saved. For the model to work properly, the external file folder must remain in the same folder as the model.
PowerPoint Slide	Generates a PowerPoint (PPT) file with the same name. A SWF will be embedded in the PPT file. The model is on the first slide of the PowerPoint presentation. When the presentation runs, the model is dynamic and interactive. ⚠ **Caution** Any external files are saved in a folder in the folder where the model is saved. For the model to work properly, the external file folder must remain in the same folder as the model.
Outlook	Generates an SWF file and attaches it to a Microsoft Outlook message, which you can then customize and send.
Word	Generates a Microsoft Word document and embeds an SWF of your model in the document. The model is dynamic and interactive. ℹ **Note** If, in Microsoft Word, the model is static, on the Microsoft Word Control toolbar, click *Exit Design Mode* (⊠).

2. Do one of the following:

   ○ If you are exporting to Flash (SWF), HTML, PDF, PowerPoint Slide, or Word: Enter a name for the exported file and click *Save*.
   ○ If you are exporting to Outlook: Enter a message and click *Send* to send the message.
   ○ If you are exporting to AIR: Set the following options and then click *Export*:

Option	Description
Name	Enter a name for the application. The name must be unique for each installation on the same system. This name will display on the installation screen. The default value is the current document name.
ID	Enter a unique ID to identify the application. The ID must be unique for each installation on the same system. The default value is "com.air.flash.**<DOCUMENT_NAME>**".
Version	Enter a version number for the application. The default value is "1.0".
Copyright	Enter the copyright information, which will display on the installation screen when users install the application on a Macintosh system.
Window Style	Select a window style from the following options: ○ *System Chrome* - the window adopts the style of windows in the environment where it is running.

Option	Description
	○ *Custom Chrome (Opaque)* - the window has its own unique look specific to AIR and has a solid background. ○ *Custom Chrome (Transparent)* - If the canvas for the model is set to Transparent, when you use this setting the window has a transparent background.
Window Size: Use Custom Size	Select this option if you want to set a custom size for the window. The set the size values in the *Width* and *Height* boxes. **i Note** When a custom size is used, users cannot change the size of the window at run-time.
Application Icons (PNG only)	Select the image files you want to use for the application icons. **i Note** Adobe AIR only supports PNG files for application icon images.
Destination	Enter the location where Adobe will save the AIR Application Installer. **i Note** If you have trouble exporting your model, select *Export with Temporary Certificate*.

8.1.2 To setup your system for Adobe AIR

If you want to export model content as an Adobe Integrated Runtime (AIR) desktop application, you must install the Adobe Flex 3.0 SDK.

1. Download the *Adobe Flex 3.0 SDK from the Adobe website*.
2. Extract the files in the zipped Adobe Flex SDK file to the following path: `C:\Program Files\Adobe\Flex Builder 3 Plug-in\sdks\3.0`

   i Note

   If you want to change the location for the SDK file, after you install Dashboards, change the folder location in file `<<Dashboards' Installation Path>>\Xcelsius\assets\air\sdkPath.dat`

3. Download and install the Java 2 Runtime Environment (J2RE 1.4 or higher).

To run the exported AIR file, you will need to download and install the *AIR player from the Adobe website*.

8.2 Adobe Flash Player Security Restrictions

Models generated by Dashboards are based on Adobe Flash technology and are subject to its security protection rules.

The Adobe Flash Player 9 and later includes security restrictions that affect SWF files that call content files stored on a local drive. If you want to run a Dashboards SWF file locally, you must add the file to the trusted locations in Adobe Flash Player Settings Manager.

Adobe Flash Player also restricts access to external data sources. If your SWF is hosted on a web server, any web server that your SWF file connects to for live data must have a cross domain policy file in the web server root.

For more information about Adobe Flash Player security, see the following documents on the Adobe website.

> **i Note**
>
> Flash Player documentation is version specific and does not address previous versions. Users with Adobe Flash Player 9 will need to read both documents.

- *Security changes in Flash Player 8*
- *Policy file changes in Flash Player 9 and Flash Player 10*

8.2.1 Running models locally

Flash (SWF) files attempting to call content files (images, XML data, and so on), using a URL file reference, can deliver the content only when the local drive is listed as a trusted location in the Adobe Flash Player security settings. If you create a snapshot or export to a local format, such as PPT, HTML PDF, Word, or SWF, you may need to add the SWF file or the host application (PowerPoint, Microsoft Word, and so on) to the trusted locations. Users need to add the SWF or host application file to the Flash Player trusted locations if they want to open an Dashboards or Presentation Design model in the following applications:

- A local HTML web page
- The standalone Flash Player
- A PDF (Acrobat 9 or later)
- A PowerPoint presentation
- A Microsoft Word document

8.2.1.1 To make a file trusted in Adobe Flash Player Settings Manager

If you want to run a model locally, you must add the file location to the trusted locations in the Adobe Flash Player Settings Manager.

1. To open the Adobe Flash Player Settings Manager, go to *http://www.macromedia.com/support/ documentation/en/flashplayer/help/settings_manager04.html*.

2. On the Adobe Flash Player Settings Manager page, in the *Table of Contents*, click *Global Security Panel Settings*.

3. Select the *Always Allow* option.

4. In the *Always trust files in these locations* list, select ▌▶ *Edit locations* ▶ *Add location* ▌.

5. Browse to the location of the SWF file (or container file such as PowerPoint, HTML, and so on) and click *Open*.

6. Click *Confirm*.

8.2.2   Running models from a web server

Because models contain a Flash (SWF) file, if a model is hosted on a web server, any other web server that the model connects to for live data must have a cross-domain policy file in the web server root (differs for each web server). Without a cross domain policy file, the model may not be able to connect to the web server to retrieve data.

The Adobe Flash Player security restrictions do not allow a SWF file to access data that resides outside the web domain from which the SWF originated. If a SWF file attempts to connect to an external data source, Flash Player may display a variety of messages. The most common message is `Unable to Load URL: http://boe-server:8080/dswsbobje/services/session`.

A cross-domain policy file is a simple XML file that gives the Adobe Flash Player permission to access data in a given domain without displaying a security dialog box. The cross domain policy file on your web server controls which SWF files, running on which domains, can access your web server. When placed in a root folder of a server, it tells the Flash Player to allow direct access to data on that server, without prompting the user to grant access.

8.2.2.1   To create a cross-domain policy file

Adobe Flash Player security restrictions prevent SWF files from accessing data across domains unless a cross-domain policy file gives permission. If you plan to deploy your model on a web server and want it to access external data sources, you must create a cross-domain policy file and place it in the root folder of all the servers the file needs to access.

For more information about cross-domain policy files, see the *Adobe website*.

1. Download or create a `crossdomain.xml` file.
   For example, the following cross-domain policy file allows any SWF, running on any domain, to access the web server.

   ```
   <?xml version="1.0"?>
   <!DOCTYPE cross-domain-policy SYSTEM
           "http://www.macromedia.com/xml/dtds/cross-domain-policy.dtd ">
   <cross-domain-policy>
   <allow-http-request-headers-from domain="*" headers="*" secure="false" />
   <allow-access-from domain="*" secure="false" />
   </cross-domain-policy>
   ```

2. Place the cross-domain policy file in the root folder of all the servers the model need to access.
   For example, to allow access between different BI platform servers, you would need to place the cross-domain policy file in the associated Tomcat folders on each server, that is, Location: `C:\Program Files \Business Objects\Tomcat\webapps\ROOT`

8.2.3   FS Command restrictions

From Flash Player 9 onwards Adobe recommends using External Interface to communicate with JavaScript (Dashboards requires Flash Player 9 or later). It is therefore recommended that you use External Interface instead of FS Command. FS Command is provided for backwards compatibility only.

The FS Command function lets a SWF file communicate with a script in a web page. However, the allowScriptAccess setting in the browser controls access to scripts for the web page. If the allowScriptAccess setting is set to "always", a SWF file can always access web page scripts. If it is set to "sameDomain", the SWF file can only access scripts that are in the same domain as the web page. In Flash Player 9 the attribute is set to "sameDomain" by default.

Some uses of Adobe's FS Command action are not supported in the Adobe Flash Player 9. Although FS Command cannot be used to pass commands to the Flash Player directly, it can be used communicate with the program hosting the player or pass messages to Macromedia Director, Visual Basic, Visual C++, and other programs that can host ActiveX controls.

The six predefined FS Command commands that are used to manipulate SWF files within the Flash Player do not currently function. This is a known issue to be resolved by Adobe. Dashboards supports the use of the FS Command action to pass commands and parameters to a scripting language, such as JavaScript, in a Web browser.

FS Command can also be used to send messages to Macromedia Director or Visual Basic events to programs that can host ActiveX controls. These uses may be functional in Dashboards, however they are not officially supported.

For more information on this topic, please see the Adobe article at: *http://www.adobe.com/support/flash/ action_scripts/actionscript_dictionary/actionscript_dictionary372.html*.

8.2.4   Flash Player Error Numbers and Messages

Once you export your model, it is saved as a SWF file and is subject to Adobe Flash Player policies and restrictions. If the Flash Player encounters a problem when running the model, it may display an error message or number. Flash Player generates these messages.

For more information about specific Adobe Flash Player error numbers and messages, see the *Adobe web site*.

9 Mobile support

SAP BusinessObjects Dashboards allows dashboard consumers to view models on mobile devices such as the iPad. Certain components and dashboard features are unavailable or behave differently on mobile devices. You can use the *Mobile Compatibility* panel to ensure that your model will run correctly on mobile devices.

Add-on components created using the Dashboards Component SDK can be made mobile-compatible. For more information, refer to the Dashboards Component SDK documentation, available on the SAP Help Portal (*http:// help.sap.com/bodash41*).

> **i Note**
>
> To view models on mobile devices, an SAP BusinessObjects Business Intelligence platform installation is required (4.0 Support Package 5 or higher).

> **i Note**
>
> Models created in versions of Dashboards older than 4.1 cannot be viewed on Android devices.

9.1 Using the Mobile Compatibility panel

The *Mobile Compatibility* panel helps ensure that your model will run correctly on mobile devices. The panel displays a warning icon for each component or feature in your model that is unsupported on mobile devices. Selecting a component in the *Mobile Compatibility* panel selects it on the canvas and opens its property sheet.

To view all incompatible components or features in the model, in the *Mobile Compatibility* panel, click *Refresh* (). Incompatible items are categorized as follows:

- *General* - Miscellaneous warnings and suggestions, such as an oversized canvas or an unsupported theme.
- *Component* - Unsupported components. Component warnings are split into the following subcategories:

    o Unsupported components. These components are not displayed in the model when the model is viewed on mobile. Components in this category are denoted by a red warning icon.

    o Supported components with unsupported properties. These components are displayed when the model is viewed on mobile but may behave differently or have certain properties disabled. Components in this category are denoted by a yellow warning icon.

    o Visual variant components. These components are replaced by their fallback component when the model is viewed on mobile with all settings preserved. For example, *Gauge2* or *Gauge3* would be replaced by *Gauge* on mobile. Components in this category are denoted by a yellow warning icon.

    > **i Note**
    >
    > *Background2*, *Background4*, and *Background5* are supported.

- *Connectivity* - Unsupported external connections. This includes all connections added through the *Data Manager*.

9.2 Supported components in mobile dashboards

The following is a summary of all components supported on mobile devices, along with any of their unsupported properties or features.

> **i Note**
>
> All components have *Entry Effect* and *Insert On*: *Mouse Over* disabled on mobile.

Chart components

Component	Unsupported properties
Bar Chart *Column Chart*	• All *Enable Run-Time Tools*, *Plot Area*, *Title Area*, and *Range Label* options. • *Allow Zoom Out Only*, *Logarithmic*, and *Fixed Label Size* scale options. • *Show Fill*, *Show Border*, *Border Thickness*, and *Check Box* legend options. • All underline font styles. • *Alerts*.
Bubble Plot	• All *Enable Run-Time Tools* options. • *Allow Zoom Out Only*, *Logarithmic*, and *Fixed Label Size* scale options. • *Show Fill*, *Show Border*, *Border Thickness*, and *Check Box* legend options. • All underline font styles. • *Alerts*.
Combination Chart *Line Chart*	• All *Enable Run-Time Tools*, *Plot Area*, *Title Area*, *Legend Area*, and *Range Label* options. • *Allow Zoom Out Only*, *Logarithmic*, and *Fixed Label Size* scale options. • *Show Fill*, *Show Border*, *Border Thickness*, and *Check Box* legend options. • All underline font styles. • *Alerts*.
Pie Chart	• All *Plot Area*, *Title Area*, and *Legend Area* options. • All underline font styles.
Stacked Bar Chart *Stacked Column Chart*	• All *Enable Run-Time Tools*, *Plot Area*, *Title Area*, *Legend Area*, and *Range Label* options. • *Allow Zoom Out Only*, *Logarithmic*, and *Fixed Label Size* scale options. • *Show Fill*, *Show Border*, *Border Thickness*, and *Check Box* legend options.

Component	Unsupported properties
	• The *Marker Overlap* appearance option. • All underline font styles. • *Alerts*.
Waterfall Chart	• All *Enable Run-Time Tools* options. • *Allow Zoom Out Only*, *Logarithmic*, *Fixed Label Size*, and *Minor Divisions* scale options. • *Show Fill*, *Show Border*, *Border Thickness*, and *Check Box* legend options. • All underline font styles.

Container components

Component	Unsupported properties
Canvas Container *Tab Set*	All *Scroll Bar* color options.

Selector components

Component	Unsupported Properties
Check Box *Icon* *Push Button* *Radio Button* *Toggle Button*	All properties are supported.
Combo Box	• All *Scroll Bar* color options. • *Drop-Down Animation*. • *Alerts*.
Label Based Menu	• All *Interaction* options except *Always Expanded* and *Expand On Mouse Click*. • All *Title* options. • *Alerts*.
List Box	• All *Scroll Bar* color options. • *Alerts*.
List View	• All *Scroll Bar* color options.

Component	Unsupported Properties
Scorecard	• Change *Column Width* at runtime and *Column Width* by percentage.

Single Value components

Component	Unsupported properties
Gauge	• *Object Elasticity*, *Adjustable* and *Open Limits*, *Mouse Sensitivity*, and *Vertical Mouse Tracking* behavior options. • *Play Button* and *Limit Background Color* appearance options.
Horizontal Progress Bar *Vertical Progress Bar* *Horizontal Slider* *Vertical Slider*	• *Adjustable* and *Open Limits* behavior options. • *Play Button* and *Limit Background Color* appearance options.

Text components

Component	Unsupported properties
Input Text	All properties are supported.
Input Text Area	*Enable HTML Formatting*.
Label	• All *Scroll Bar* color options. • *Enable HTML Formatting*.

Art and background components

Component	Unsupported properties
Background *Background2* *Background4* *Background5* *Ellipse*	All properties are supported.

Component	Unsupported properties
Horizontal Line Rectangle Vertical Line	
Image	• _Import_ SWF files. • _Hide SWF Background Color_.

Web Connectivity

Component	Unsupported properties
URL Button	All properties are supported. **i Note** Linking a _URL button_ to unsupported data sources will not work. For example, an OpenDocument URL will not pass FlashVars to the model since FlashVars are not supported. For more information about unsupported data sources, see _Unsupported features in mobile dashboards_ [page 273]. **i Note** On an iPad, external URLs (such as http://help.sap.com) are opened in Safari.

Universe Connectivity

Component	Unsupported properties
Query Refresh Button	All properties are supported.

9.2.1 Using components in mobile dashboards

Mobile dashboard users interact with components using the device's touch screen. The user can interact with components and items by tapping, touching, or dragging them. The user can also pinch to zoom chart component scales or models with a canvas size larger than the mobile device's resolution.

Component interaction that involves clicking or dragging is unchanged on mobile. Component interaction that involves mousing over is unsupported, with the following exceptions:

Column Chart and Line Chart

When viewing a model on a computer, the user can mouse over a column on a *Column Chart* or a point on a *Line Chart* to show detailed values. On mobile devices, the user must briefly touch the column or point. The values are displayed on the column or point, as shown below:
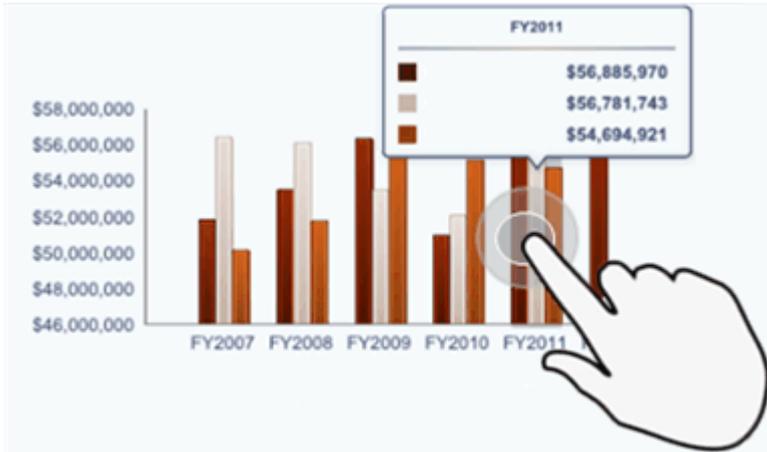

List View and Scorecard

If the text in a cell is larger than the cell's width, the user can tap and hold the cell to show the full text. The text is displayed in a tooltip above the cell.
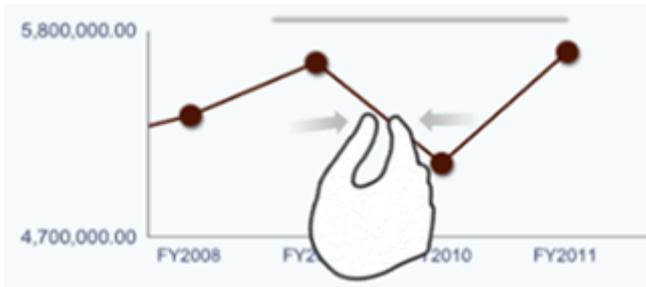
Pie Chart

When viewing a model on a computer, the user can mouse over a slice of a *Pie Chart* to show detailed values. On mobile devices, the user must tap the slice. The values are displayed in the center of the chart, as shown below:
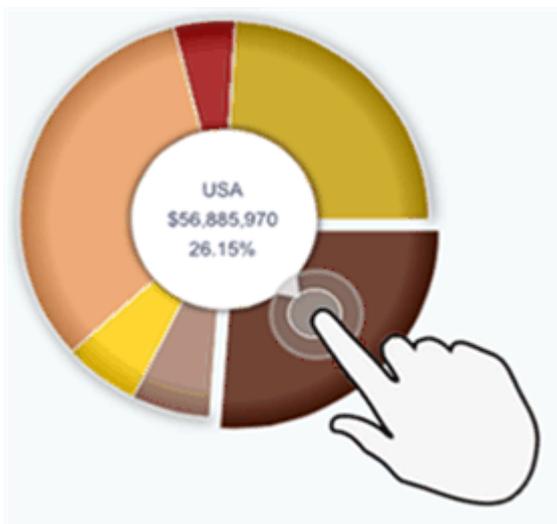
Pinch to zoom

The user must pinch to zoom chart component scales. Pinching to zoom a chart's scale is shown below:

Waterfall Chart

The user can tap and hold a column to show detailed values. The values are displayed in a tooltip above the selected column.

9.3   Unsupported features in mobile dashboards

The following is a summary of unsupported or partially supported features when viewing a model on mobile devices.

Fonts

For a list of fonts supported on mobile, see *iosfonts.com*. All listed fonts that are installed on your local machine will be added to Dashboards' font list with *(iOS 5+)* appended.

> **i Note**
>
> If a font is not supported, it defaults to Verdana.

Themes

Only the *Nova* theme is supported.

> **i Note**

If another theme is used, it defaults to *Nova*. The color scheme remains the same.

Canvas size

If the canvas is larger than the mobile device's native resolution, scroll bars will be used if the dashboard is viewed in *Original Size Mode*. The native resolution of the iPad is 1024x768, so it is recommended you set the canvas to this size when designing your model.

Model connectivity

If the model is running in offline mode, or if the mobile device has no network connection, all connections in the model will be disabled and will use cached data.

> **i Note**
>
> All connections added from the *Query Browser* are supported. Connections added through the *Data Manager* are supported if denoted by an asterisk.

www.sap.com/contactsap

SAP