Actuate One™

One Design
One Server
One User Experience

Using BIRT Studio - Java Component Edition

Contains information proprietary to:
Actuate Corporation, 951 Mariners Island Boulevard, San Mateo, CA 94404

www.actuate.com
www.birt-exchange.com

Document No. 120201-2-761301 October 19, 2012

Contents

Chapter 5
Filtering data 81

Chapter 6
Working with summary tables 117

About *Using BIRT Studio - Java Component Edition*

Using BIRT Studio - Java Component Edition provides information about using and configuring the BIRT Studio report design tool. This manual explains how to design a report and how to configure and customize BIRT Studio.

- *About Using BIRT Studio - Java Component Edition.* This chapter provides an overview of this guide.
- *Part 1. Designing reports using BIRT Studio.* This part describes the tasks that users complete to design reports.
- *Chapter 1. Getting started.* This chapter introduces BIRT Studio and describes how to create, save, and run a report. The chapter describes how to select data from a data source, and how to lay out report data.
- *Chapter 2. Editing and formatting report content.* This chapter describes the formatting options in BIRT Studio, including applying a theme, changing font properties, changing the display of data, formatting data based on conditions, and modifying the page layout. The chapter also describes how you can reorder columns, hide columns, display hidden columns, merge several columns into a single column, and work with data in a merged column.
- *Chapter 3. Inserting calculated data.* This chapter describes how to write expressions to calculate data.
- *Chapter 4. Organizing data in a report.* This chapter describes how to sort, group, and aggregate data.
- *Chapter 5. Filtering data.* This chapter describes how to create filters to display only the information that your report needs. The chapter also explains how to prompt for filter values at run time by creating regular and dynamic filter parameters.

- *Chapter 6. Working with summary tables.* This chapter describes how to create and modify summary tables, as well as how to change table types.
- *Chapter 7. Presenting data in a cross tab.* This chapter describes how to create a cross tab, obtain data for a cross tab, and how to format and analyze data in a cross tab.
- *Chapter 8. Presenting data in a chart.* This chapter describes the types of charts you can create and discusses the procedures for displaying data in regular charts, Flash charts, and HTML5 charts.
- *Chapter 9. Adding interactive viewing features.* This chapter introduces the interactive features you can add to a BIRT report. The chapter describes how to use bookmarks and hyperlinks in a report element, and explains the procedures to create, modify, and delete bookmarks and hyperlinks.
- *Chapter 10. Functions and operators.* This chapter is a reference to all the supported functions and operators you can use to create calculated data. The chapter describes the functions you can use to create aggregate calculations, and the operators you can use in expressions, conditional formatting, as well as in filter conditions.
- *Part 2. Customizing and integrating BIRT Studio.* This part is a guide to customizing and integrating BIRT Studio.
- *Chapter 11. Introduction to BIRT Studio customization.* This chapter introduces BIRT Studio, provides an overview of the types of customizations, and lists the Actuate software required to perform the customization tasks.
- *Chapter 12. Creating and publishing report templates.* This chapter provides guidelines for designing effective report templates for use with BIRT Studio. The chapter also describes the procedures for publishing the templates and their associated resources.
- *Chapter 13. Providing data.* This chapter describes the different ways to provide BIRT Studio users with data for their reports. The chapter provides guidelines for developing custom data sources.
- *Chapter 14. Configuring BIRT Studio.* This chapter discusses how to modify the appearance of BIRT Studio pages, how to control the functionality available to users, how to customize BIRT Studio for different user groups, and how to configure the BIRT Studio application.
- *Chapter 15. Actuate BIRT Studio URIs.* This chapter discusses how to access BIRT Studio using a URI and how to use the BIRT Studio servlet.

Part One

Designing reports using BIRT Studio

Chapter

1

Getting started

This chapter contains the following topics:

- About BIRT Studio
- Starting BIRT Studio
- Creating a report
- Saving a report
- Running and viewing a report

About BIRT Studio

BIRT Studio is a web-based report design tool for users who want to create professional-looking reports quickly and easily without detailed understanding of database architecture or report design techniques. You use predefined data sources and templates that provide the data and basic layout for your reports. Then, using BIRT Studio's intuitive graphical interface, you can add a wide variety of charts; rearrange and reformat content; sort, group, and filter data. As you edit and format a report, a preview feature provides an instant view of your changes with sample data.

BIRT Studio is a highly customizable application. A system or application administrator creates and integrates custom data sources and templates that you use to create reports. An administrator can also customize the user interface and functionality in BIRT Studio to tailor the tool to your organization's report design requirements.

This guide provides instructions for using the features included in the default installation of BIRT Studio. The functionality available to you can differ depending on how your system administrator configures and customizes BIRT Studio.

Starting BIRT Studio

The configuration of your organization's application system determines how you access BIRT Studio. Your system administrator should provide a URL and, in most cases, login information. In a typical installation, you start BIRT Studio after you log into another application. Figure 1-1 shows an example of an application that includes a link you choose to launch BIRT Studio.

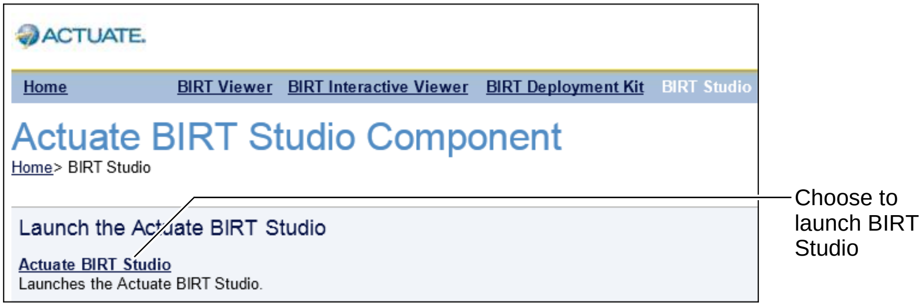


Figure 1-1 Launching BIRT Studio from an application

BIRT Studio opens in your web browser, as shown in Figure 1-2.

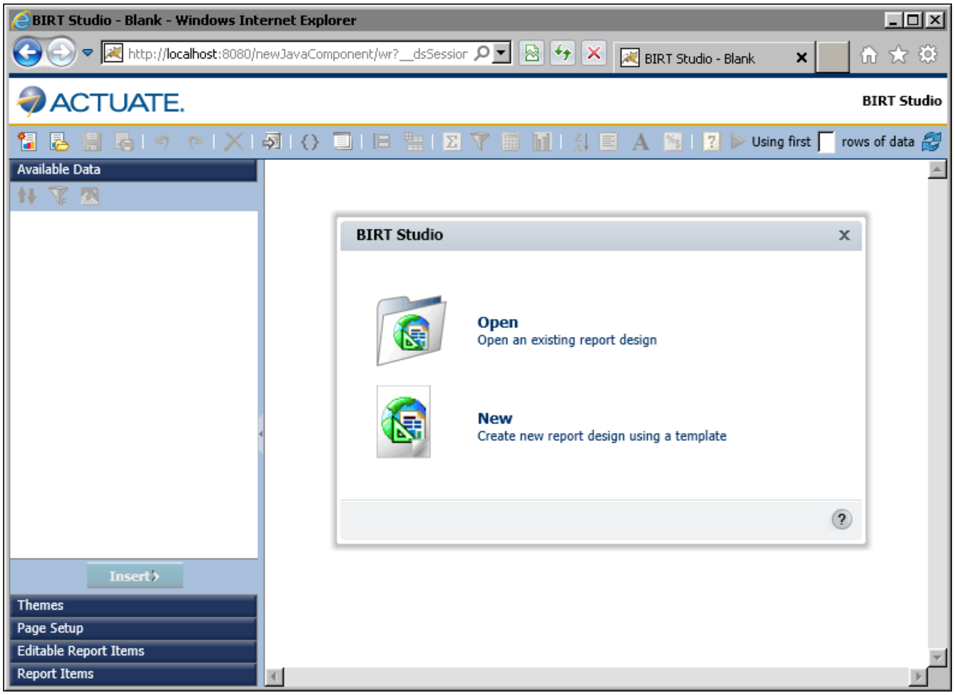


Figure 1-2 Opening an existing report or creating a new one in BIRT Studio

Creating a report

After you start BIRT Studio, you can either create a new report or open an existing report. When you choose to create a report, BIRT Studio displays a series of dialog boxes to guide you through the following required steps:

- Select a template on which to base your report.
- Select a data source that provides the data for your report.

After you select a template and data source, you perform the following tasks:

- Select and insert the data in the report.
- Edit and format the report.
- Save, run, and view the report.

Selecting a template

Every report you create in BIRT Studio is based on a template. A template determines the basic structure for a report and typically contains the following elements:

- Report items, such as a report title
- A report table to display data
- Page footer items, such as a date and time stamp, page number, or author name

Figure 1-3 shows two templates included in the default installation of BIRT Studio.

Your version of BIRT Studio most likely provides custom templates designed for the types of reports you need to create. BIRT Studio organizes templates into categories. A category is simply a convenient way to group similar templates. Templates can be organized by report types or by departments, or they can be stored in a single category, depending on the requirements at your site and the configuration your system administrator implements. Your access privileges determine which categories and templates you can view and use.

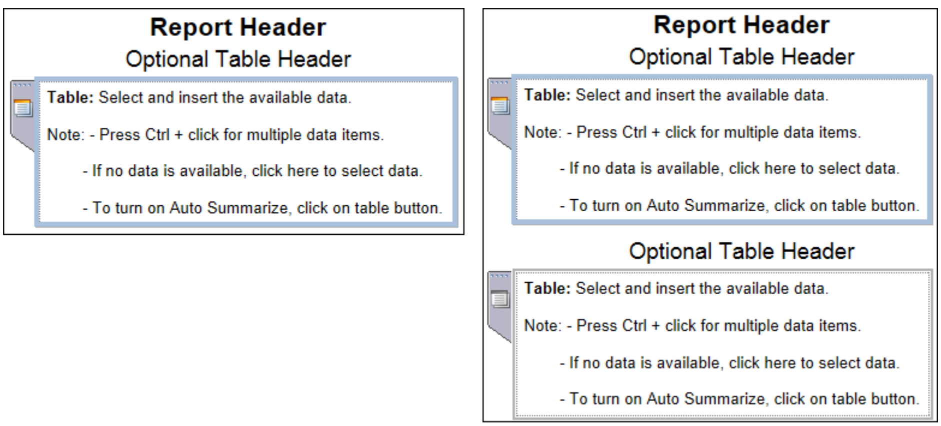


Figure 1-3 Two templates included in the default installation of BIRT Studio

Figure 1-4 identifies the items a template typically includes. Notice that the table contains instructions for inserting data. Templates often provide information about using items. After you select a template and data source, a typical next step is to insert data in a table.

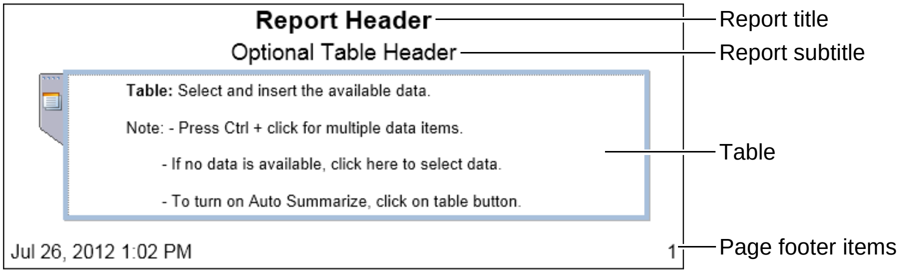


Figure 1-4 Items in a typical template

How to select a template for your report

1 When you create a new report, BIRT Studio displays Report Template. On Report Template, select a template category from the drop-down list.

Report Template displays thumbnail images of the templates in the selected category. Figure 1-5 shows sample templates in the Standard category included in the default installation of BIRT Studio.

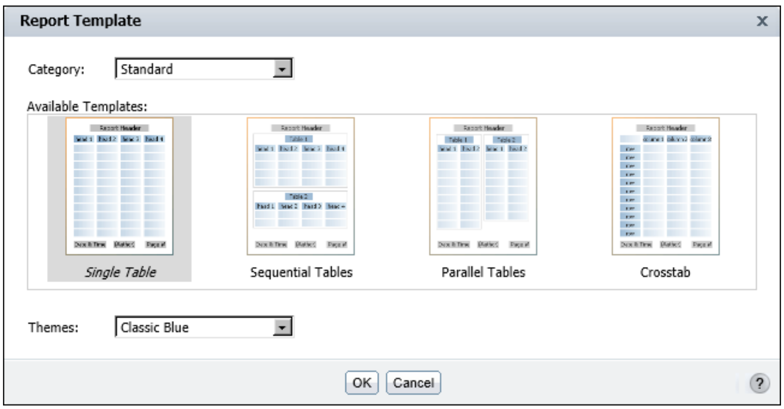


Figure 1-5 Selecting a report template

2 To view a description of the template, move your cursor over the thumbnail image of the template.

3 Select the template to use, then choose OK.

Selecting a data source

After you select a template, you select a data source from a list of available sources. Depending on your system configuration, the data source can be one of the following:

- A data set included in the selected template
- A custom data source, such as an ODA data source

Data sets classify the data fields retrieved from a data source, in dimensions, measures, and attributes. These categories are specified by the template or data source developer, at the time of designing the template or data source, and cannot be modified using BIRT Studio. It is useful to understand these categories to use them appropriately in a report design.

Dimensions are data fields that group other data fields. Each dimension field can contain multiple attributes. Attributes are typically associated with a dimension. Table 1-1 lists the possible attributes for each dimension field.

Table 1-1 Dimensions with associated attribute fields

Dimension	Attribute fields
Product Line	Product name, product code
Region	Country, state, city

Measures are data fields that can be aggregated, or computed such as revenue, profit, price, and so on.

Data sources contain all the information to connect to a data repository, such as a database, and retrieve a specific set of data. Both types of data sources provide easy access to the data fields that contain the data for a report. For example, a data set named CustomerData contains customer information fields, such as customer name, address, phone number, and so on. You choose this data source to create a report that lists customers and their contact information. The data in the data set is often organized into categories called dimensions, measures, and attributes. This classification is set by the developer at the time of designing the data set, and cannot be modified in BIRT Studio.

You can limit the data available to a report design by applying filters to a data set in a template. Using filters to limit the data available to a report design is useful when you need to create a report that displays specific data quickly. For example, if the data set displays sales totals for the years 2003, 2004, and 2005, and you need to view the sales totals for the most recent of those years, you can specify a filter condition such that BIRT Studio retrieves and displays data only for the year 2005 in the report design.

BIRT Studio also provides a convenient Table Builder wizard that enables you select data fields from a data set in a template, and insert them into the report table. Alternately, if you want to preview the data as you select and insert the data fields into the report design, you can manually select and insert fields in a report design. If you use a custom data source, you must first select the data fields in the ODA editor, and then manually select and insert them into a report design. The following section describes how to create a report design.

How to create a report design using a data set in a template

1 In Report Template, select a template that includes one or more data sets. Choose OK. Data appears, as shown in Figure 1-6.

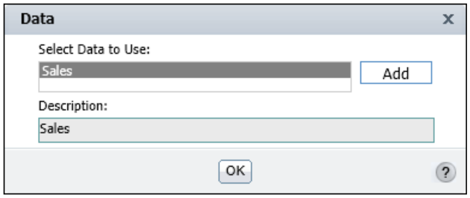


Figure 1-6 Selecting a data set

Data displays the data sets included in the template.

2 Select a data set, then choose OK.

Table Builder appears listing the data fields in the selected data set. Do one of the following:

- Use Table Builder to select and insert the data fields in the report design. You can also limit the data retrieved from the data set using Table Builder.
- To preview the data as you manually select and insert data fields in the report design, choose Cancel on Table Builder, without making any selections. All the data fields from the selected data set appear in Available Data in BIRT Studio.

3 On Table Builder, as shown in Figure 1-7, complete the following steps:

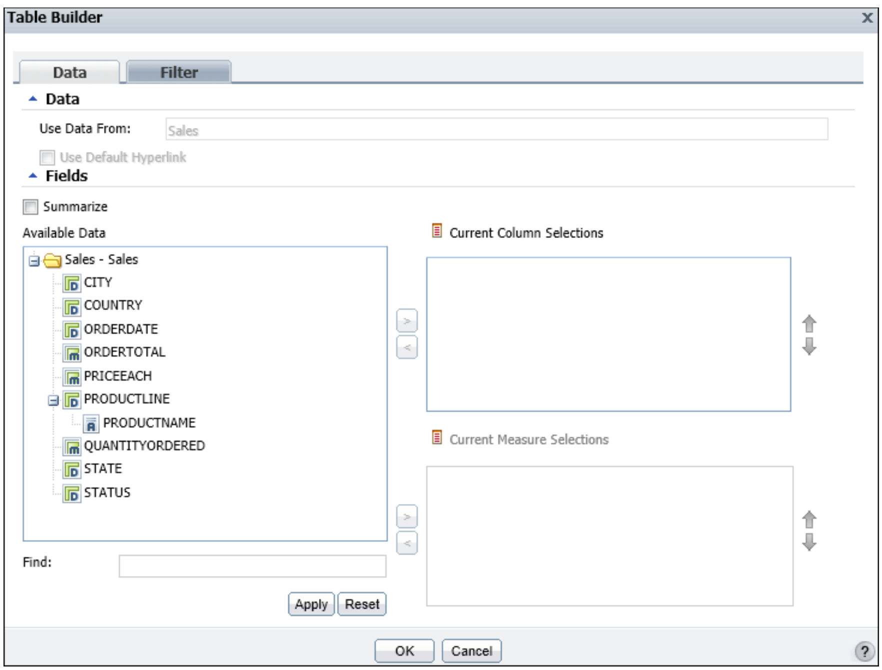


Figure 1-7 Table Builder displaying fields in the selected data set

1 In Data, in Use Data From, select a different data set, if necessary.

2 In Fields, do one of the following:

- To create a summary table, select Summarize.
- To create a detail report, deselect Summarize.

3 In Available Data, do one of the following:

- Press Ctrl, and select each data field to use from the list of available data fields. Then choose the right arrow. The selected data fields appear in Current Column Selections, as shown in Figure 1-8.

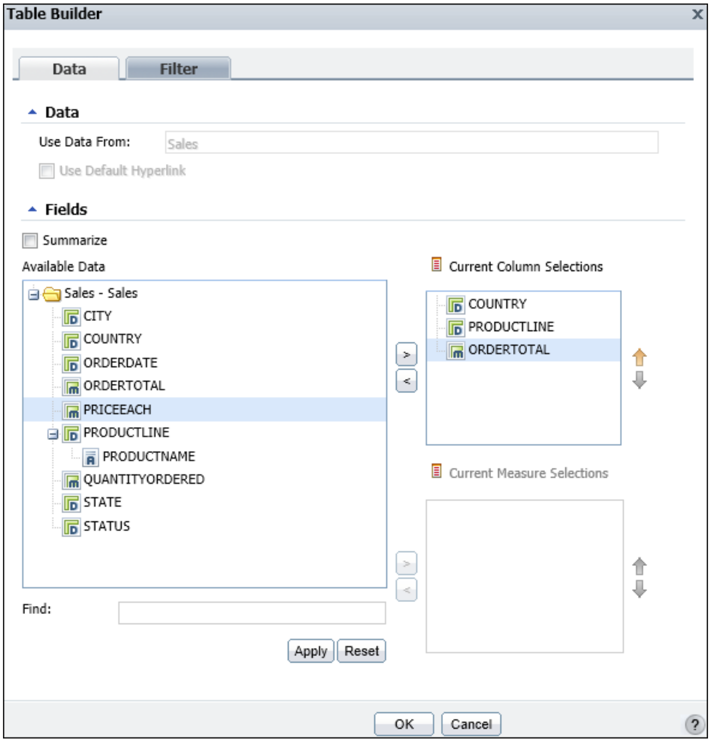


Figure 1-8 Current Column Selections displaying the selected data fields

- If the data source contains a large number of data fields, to search for a data field by name, do the following:
  - Type the name of the field in Find, then choose Apply.
  - Type the name of the field in Find, then press Enter.

  If the data field is found, Table Builder displays the data field, and the category containing the data field in Available Data.

- Select the data field, then choose the right arrow.

  The data field appears in Current Column Selections.

- Repeat this step to retrieve each additional data field for which you can supply a name.

Use the up and down arrows to rearrange the order in which the data fields are displayed in the report design.

You can now specify a filter condition if you want to limit the data retrieved from the data source and displayed in the report. Using filters is useful when working with data sources that contain large amounts of data.

How to limit the data displayed in a report

You can use Table Builder—Filter to specify a filter at the data set level.

1. In Table Builder, select Filter.
2. In Table Builder—Filter, choose Add Filter, as shown in Figure 1-9.

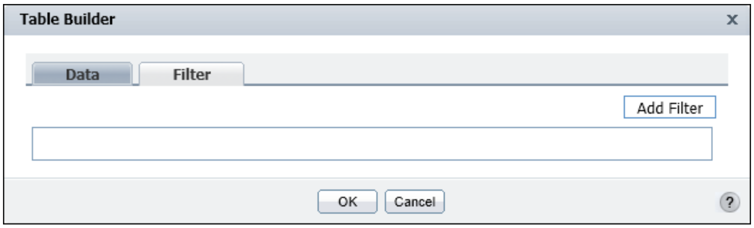


Figure 1-9 Adding a filter condition at the data set level

3. Filter appears. In Filter, complete the following steps:
   1. In Filter By, select the column to evaluate.
   2. In Condition, select a comparison operator from the drop-down list. Based on your selection, either one or two fields appear in Value.
   3. In Value, do one of the following:
      - Type a value.
      - Choose Select Values, then select a value from the list of values that appears.
      - To specify a null value, select No value.
   4. Choose Add Condition. Filters displays the condition, as shown in Figure 1-10. Choose Validate. On the message confirming that the syntax of the filter condition is valid, choose OK. In Table Builder—Filter, choose OK. The data fields appear in the report design, as shown in Figure 1-11.

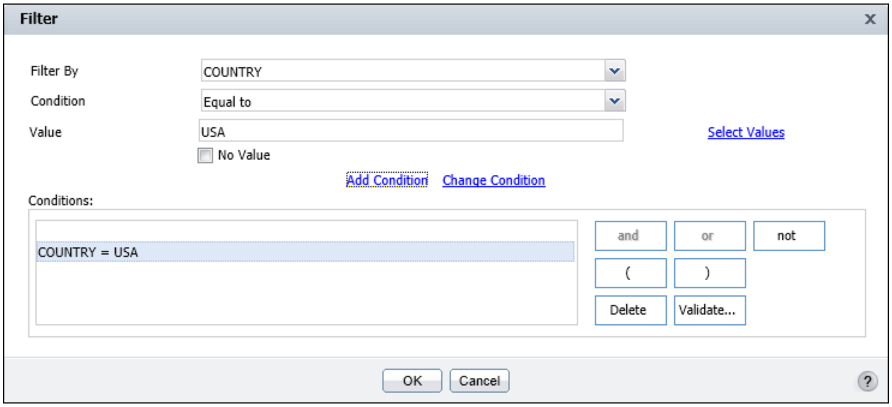


Figure 1-10 Specifying a data set filter condition

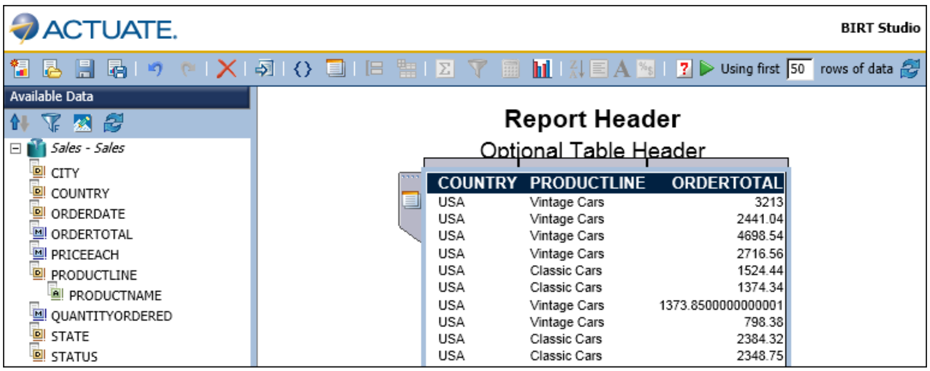


Figure 1-11 BIRT Studio displaying the selected data

How to create a report design using a custom data source

If you use a custom data source, you must manually select and insert data fields in a report table.

1. On Report Template, select a template that does not contain a data set. Choose OK.
2. Data Source appears, displaying available sample and custom data sources, as shown in Figure 1-12. Select the data source to use. Choose OK.

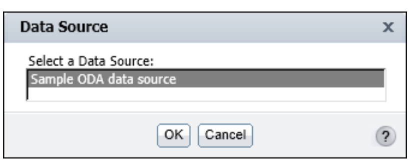


Figure 1-12 Selecting a data source

3 Sample ODA Editor appears, as shown in Figure 1-13. Select the data fields to use in the report design.

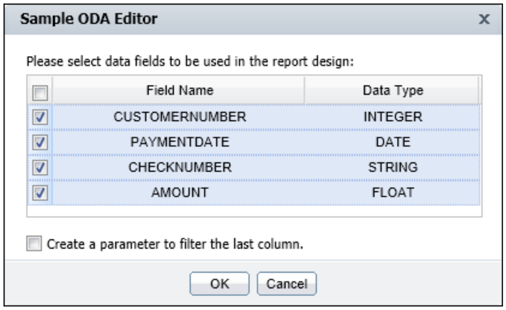


Figure 1-13 Selecting data fields

4 Optionally select Create a parameter to filter the last column.

- Select this option to activate any existing parameters set for the last data field, Amount, as shown in Figure 1-13. After you insert the data field in the report design, choose Parameters and make your selection to specify viewing parameters for data in that column.
- Deselect this option to view all data rows in the data field.

Choose OK. The selected data fields appear in Available Data in BIRT Studio.

The next step is to manually select and insert the data fields to use in the report table.

How to manually select and insert data fields in a report table

To manually select and insert data fields in a report table, in BIRT Studio, press Ctrl, and in Available Data, select each data field to insert in the table. Then choose Insert. Alternatively, drag each data field and drop it in the table. The order in which you select the data fields is important because they appear in the report in the order of selection.

For each data field you insert, a column appears in the table, displaying up to 50 values. Notice that the data field names, COUNTRY, PRODUCTLINE, and ORDERTOTAL appear at the top of each column. These items are called column headers, and BIRT Studio creates them when you insert data fields in a table.

Figure 1-14 shows an example of three data fields, Country, Product Line, and Order Total, inserted in the report table.

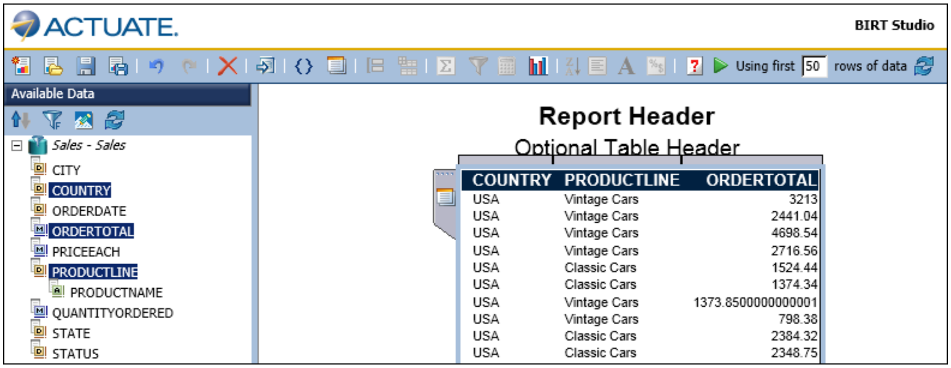


Figure 1-14 BIRT Studio displaying the selected data

You are now ready to format the report design.

Editing and formatting a report

So far, you have learned how to create a simple listing report in three steps. The next step is to enhance the report design. At the very least, the example report. as shown in Figure 1-14, needs a real title, and the names used in the column headers need editing. If you do not like the fonts and colors in the template, you can change them. Editing and formatting tasks are described later in this document.

As you edit and format a report, the report design provides a preview of your changes. The report design, however, does not show all the rows of data. By default, the report design shows the first 50 rows. To see more rows of data, change the number in Using first 50 rows of data, then press Enter. You can set a number up to 200.

Saving a report

Depending on the platform on which you are working, you save a report to a folder in a file system. Your access privileges determine the folders in which you can save a report.

The first time you save a report, BIRT Studio prompts you to select the folder and provide a name and description for the report. The next time you save the report, the report is saved to the same folder and you can choose whether to replace the original report or create a new version.

How to save a new report

1 Choose Save. The first time you save a report, Save Report Design appears, as shown in Figure 1-15. This dialog box typically displays your home folder.

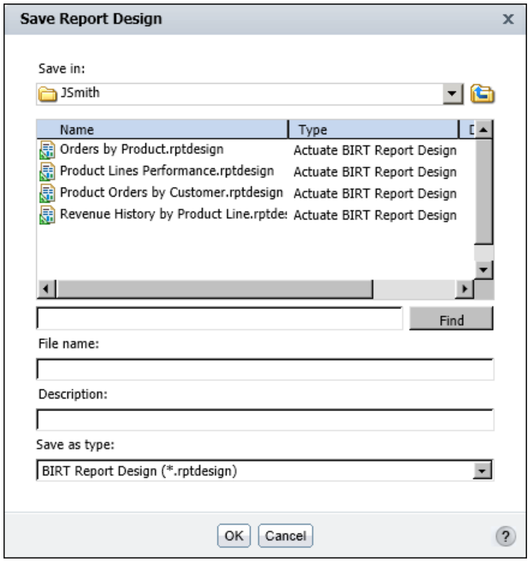


Figure 1-15 Save Report Design

2 If necessary, navigate through the folder structure to select a different folder.

3 If you know the name of the file you want to overwrite, type the name, or relevant keywords in Find. Then, press Enter, or choose Find. Save Report Design displays the reports that match the string you provided.

4 In File Name, type a name for the report. The file name can contain alphanumeric characters and spaces, but it cannot contain punctuation.

5 Optionally, in Description, type a brief description of the report. A description is helpful if you are sharing this report with other users.

   Save as type shows that the report is saved as a BIRT Report Design file with a .rptdesign file-name extension. Choose OK.

If you modify an existing report, BIRT Studio prompts you to confirm that you want to replace the existing version with the modified version. To create a new version of the report design, change the file name, or save the report design in a different folder.

How to save a report and change its name or location

Sometimes you need to create a new version of a report while retaining the original report or you want to save the report in a different folder. To accomplish either task, choose Save As. In Save Report Design, select a folder and type a file name in the same way as when using Save.

Running and viewing a report

As you design a report, the data you see is just a preview. To view all the data, save the report, then run the report. You run a report by choosing Save and View. The generated report appears in Actuate BIRT Viewer in your web browser. Figure 1-16 shows the viewer displaying the finished version of the report design, as shown in Figure 1-14.

In the viewer, you can page through the report, print the report, export the report to another format, and even modify the content, organization, and formatting of the report. For more information about all the report operations you can perform in the viewer, see *Working with Actuate BIRT Viewers.*

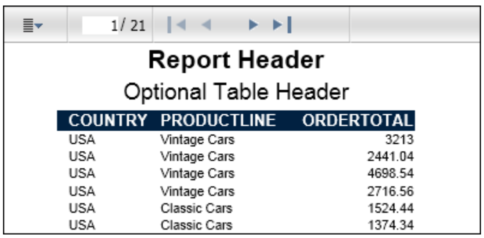


Figure 1-16 Actuate BIRT Viewer displaying a report

Chapter

2

Editing and formatting report content

This chapter contains the following topics:

- Editing and formatting options
- Removing items from a report
- Editing labels
- Applying a theme
- Selecting an item for formatting
- Changing font properties
- Changing the alignment of text
- Applying a number format
- Applying a date-and-time format
- Applying a string format
- Working with predefined data formats
- Formatting data based on conditions
- Managing columns
- Merging data into one column
- Adding a new column header row
- Changing the page layout to fit data

Editing and formatting options

The template you choose determines the basic layout of your report, and the items you can use in the report design. Report items can include common elements, such as a report title, page numbers, report-creation date, and a table in which to display report data. You can choose which report items to include in your report design.

In a typical template, some report items are editable, and others are not. Examples of items you can edit include report titles or tables in which you insert data. Examples of non-editable items include standard copyright or confidentiality statements.

A typical template also defines the style of a report—the color schemes, fonts, spacing and alignment, page size, and other presentation aspects. A template also can provide multiple sets of styles called themes, which you use to change the appearance of a report with one click. Templates save time and effort that you would otherwise spend formatting your report to make it visually appealing. Sometimes, however, you need to make formatting changes to refine the appearance of a report. A template serves as a good base for reports, but the particular contents in individual reports can require you to change, for example, the size of the page, the width of columns, or the size of the font.

This chapter describes the editing and formatting tasks you can perform using the default installation of BIRT Studio. Some editing or formatting functionality may not be available to you if your system administrator configured BIRT Studio to omit that functionality.

Removing items from a report

When you create a new report from a template, by default, all the items in the template appear in the report. If there is an item, such as a report-generation date or a label, that you do not want to display in your report, you can exclude the item. If you change your mind, you can select the item for display again.

How to select the items to remove or display

1. In the side menu, select Report Items. A list of every item in the template appears. Figure 2-1 shows an example of such a list.
2. Select the items to display in the report. If you are not sure which item in the list corresponds to which item in the report design, select or deselect one item at a time, choose OK, and see what item appears or disappears from the design.

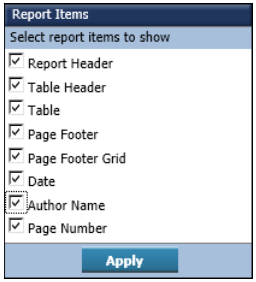


Figure 2-1 Report items included in a template

Editing labels

Labels contain static text. In a typical template, some labels are editable, and others are not. Examples of labels you can edit include report titles or author names. Examples of non-editable labels include standard copyright or confidentiality statements.

When you insert data fields in a table, BIRT Studio displays the values in rows and columns and creates labels, or column headers, that show the names of the data fields. These names are appropriate in a database, but usually are not suitable for a report. You typically edit the text in these column headers.

How to edit a label

1. Double-click the label. The label becomes an editable field that contains a blinking cursor.
2. Delete the existing text, and type the text to display.
3. Press Enter. The edited text appears in the label.

Applying a theme

A theme in a template is similar to a theme in Microsoft Windows. A Windows theme helps you personalize, with one click, your computer display to show a particular background and custom color and fonts settings for various items, such as title bars, buttons, and icons. Similarly, a template theme can determine the colors used in the report, such as the background color for column headers or data rows. A theme can also determine the font used in specific report items, the spacing between rows, the styles of borders, the alignment of text, and much more.

Themes are optional. A template can, but does not have to, include themes. A template includes multiple themes if the template designer decides to provide users with different style options to apply to a report. On the other hand, a template does not include themes if, for example, a particular type of report must adhere to a specific layout and style.

How to apply a theme

1 In the side menu, choose Themes. A list of available themes appears, if the template includes themes. Figure 2-2 shows an example of a list of themes.

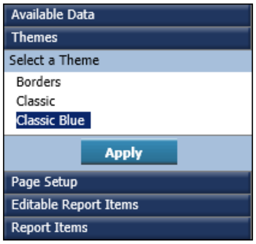


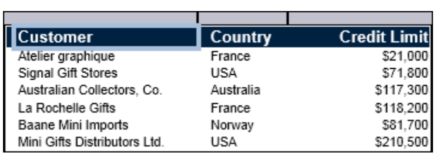
Figure 2-2 Themes included in a template

2 Select a theme and choose Apply. The report design reflects the styles defined in the theme.

Selecting an item for formatting

You can format only items that the template sets as editable. Typically, these items are titles and tables. In a table, you can format the column headers and the data in the columns. To format an item, first, select the item, then choose the formatting action from the toolbar or from a context menu that appears when you right-click the selected item. The toolbar provides access to the most common actions. The context menu provides access to all the actions you can perform on a selected item.

Selecting a label or a column header is straightforward. You simply click the item. To select data for formatting, select the entire column by clicking the column area or the column handle situated above the column header. A box appears around the selected item. Figure 2-3 shows an example of a selected column header. Figure 2-4 shows an example of a selected column.



Customer	Country	Credit Limit
Atelier graphique	France	$21,000
Signal Gift Stores	USA	$71,800
Australian Collectors, Co.	Australia	$117,300
La Rochelle Gifts	France	$118,200
Baane Mini Imports	Norway	$81,700
Mini Gifts Distributors Ltd.	USA	$210,500

Figure 2-3 A selected column header in a table

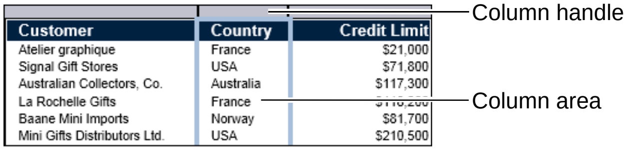


Figure 2-4 A selected column in a table

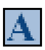
You cannot select a single row of data in a table, nor can you select a single data value. If you want to highlight a particular row or value, use conditional formatting. This topic appears later in this chapter.

Changing font properties

You can change the font properties of editable labels, column headers, and data in a column. The properties you can set include the font, font size and color, and the background color, and you can make the text bold, italic, or underlined.

How to change font properties

1 Select the label, column header, or column. A box appears around the selected element.



2 Choose Font. Font, as shown in Figure 2-5, displays the current font property values. A value of Auto means the property uses a default value, which is specified by a theme, the template, or the software.

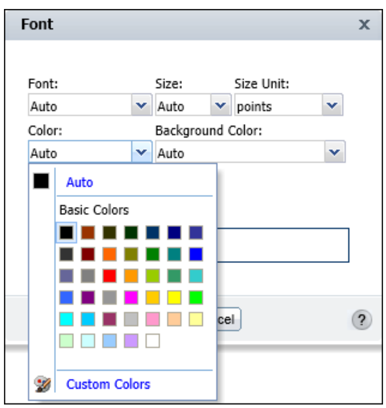


Figure 2-5 Selecting a font

3 Use the drop-down lists to specify the font, size, and size unit for the selected element.

4 In Color, and Background Color, do one of the following:

- Select a color in Basic Colors.

■ Select Custom Colors, and use the spectrum bar to specify a color group. Then, select a shade in the gradient square, and choose Pick Color, as shown in Figure 2-6.

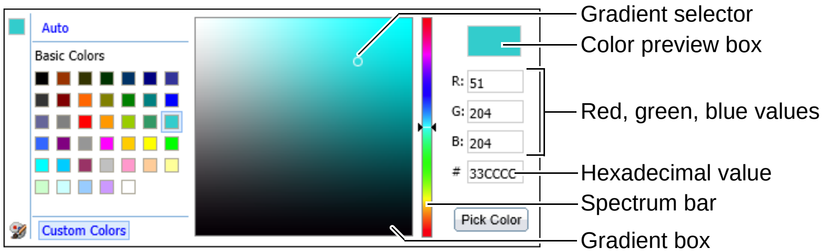


Figure 2-6 Specifying a custom color using color picker

The selected options appear in the Preview field. The RGB value of the selected color appears in Color. Choose OK.

Changing the alignment of text

You can change the alignment of text in editable labels, column headers, and columns.

How to change the alignment of text

1 Select the label, column header, or column. A box appears around the selected element.

2 Choose one of the alignment options, Align Left, Align Center, or Align Right, as shown in Figure 2-7.

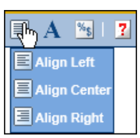


Figure 2-7 Selecting alignment options

Applying a number format

The appearance of numeric data depends on the following factors:

- How the data is stored
- The locale to which the system is set
- The format, if any, set by the template or theme

BIRT Studio provides common formats you can use to change how numbers appear. You can, for example, display numbers with or without decimal values, in scientific notation, as a percentage, or with a currency symbol.

Even when a column displays numbers, the data in the column can be of string type. Postal codes, for example, are frequently stored as string data. Numeric formats do not apply to numbers of string type. When you select a column for formatting, the title of the formatting dialog box tells you what type of data the column contains.

How to apply a number format

1 Select a column that contains numeric data.

2 Choose Format Data.

3 On Number Column Format, in Format Number as, select a format, as shown in Figure 2-8. You can select a standard format or define a custom format. The standard and custom formats are described in the next sections.

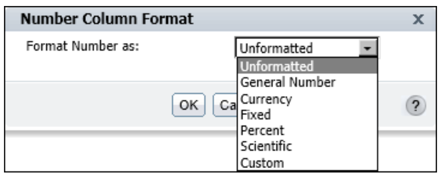


Figure 2-8 Formats for numeric data

Selecting a standard number format

Table 2-1 describes the standard number formats that BIRT Studio supports. The examples in the table reflect the English (United States) locale. If you work in a different locale, the data appears different. For example, a number that appears as 1352.45 in the English (United States) locale appears as 1352,45 in the French (France) locale. When you select a number format, you can set additional formatting options, such as the number of decimal places, the inclusion of a thousands separator, a currency symbol, and so on.

Table 2-1 Standard number formats

Format	Example of data display
General Number	6066.45 or 6066.5 or 6067, depending on the original value. This format displays up to two decimal places. Whole numbers and numbers with one or two decimal places appear in their original format.
Currency	$6,067.45 or ¥6067 or 6067€, depending on the symbol, symbol position, decimal place, and thousands separator values you set.

(continues)

Table 2-1 Standard number formats (continued)

Format	Example of data display
Fixed	6067 or 6067.5 or 6,067.45, depending on the decimal place and thousands separator values you set.
Percent	45% or 45.8% or %45, depending on the symbol position and decimal place values that you set. This format multiplies the original value by 100 and adds the percent (%) symbol.
Scientific	2E04 or 2.67E04, depending on the decimal place value you set. The number after the E represents the exponent of 10. For example, 2.67E04 means 2.67 multiplied by 10 raised to the fourth power.

Defining a custom number format

You can define a custom number format using special symbols to construct a format pattern. A format pattern shows where to place currency symbol, as well as which characters to use for the thousands separator, and the decimal separator. The thousands separator and the decimal separator are locale-dependent. In the US (English) locale, for example, the thousands separator is the comma, and the decimal separator is the period. In most European locales, the thousands separator is a space, and the decimal separator is the comma. Table 2-2 shows examples of custom format patterns and their effects on numeric data.

Table 2-2 Results of custom number formats

Format pattern	Data in original format	Result of formatting
0000.00	12.5 124.5 1240.553	0012.50 0124.50 1240.55
#.000	100 100.25 100.2567	100.000 100.250 100.257
$#,###	2000.00 20000.00	$2,000 $20,000
ID #	15	ID 15

Applying a date-and-time format

As with numeric data, the appearance of date-and-time data depends on how the data is stored, your system's locale, and the date-and-time format, if any, defined in the template or theme.

BIRT Studio provides common formats you can use to change how dates and times appear. You can, for example, display dates as January 19, 2008 11:00:00 AM PST or Jan 19, 2008 or 1/19/08, and so on.

How to apply a date-and-time format

1 Select a column that contains date-and-time data.

2 Choose Format Data.

3 On Date/Time Column Format, select a format from the drop-down list. You can select a standard format or define a custom format. The standard and custom formats are described in the next sections.

Selecting a standard date-and-time format

The formats that appear in the drop-down list differ depending on the locale in which you are working. Figure 2-9 and Figure 2-10 show the sample formats that appear in the English (United States) and the Japanese locales, respectively.

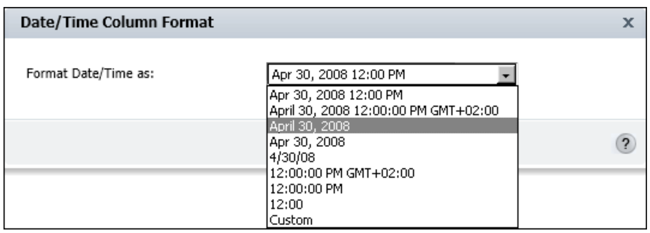


Figure 2-9 Date-and-time formats displayed in English (United States) locale

Figure 2-10 Date-and-time formats displayed in Japanese locale

Defining a custom date-and-time format

BIRT Studio supports the ability to define custom date-and-time formats. You can define a custom date-and-time format using special symbols, shown in Table 2-3, to construct a format pattern. Use custom date-and-time formatting only for reports viewed in a single locale. Custom formats always display dates in the format you set. This format can be inappropriate for other locales.

For example, if you use the format MM-dd-yy, the date January 10, 2006, always appears as 01-10-06, regardless of the locale in which the report is viewed. For locales in which dates customarily appear in date-month-year format, the date 01-10-06 is interpreted as October 1, 2006.

Table 2-3 Symbols for defining custom date-and-time formats

Symbol	Description	Example
yy	Short year	08
yyyy	Long year	2008
MM	Month as a number	07
MMM	Short month name	Jul
MMMM	Full month name	July
d	Day in month	10
W	Week in month	2
w	Week in year	28
DD	Day in year	192
E	Short day of week	Thu
EEEE	Long day of week	Thursday
H	Hour in day (0 - 23)	0
k	Hour in day (1 - 24)	24
K	Hour in AM/PM (0 - 11)	0
h	Hour in AM/PM (1 - 12)	12
mm	Minutes	30
ss	Seconds	55
a	AM or PM	8:00:00AM

Table 2-4 shows examples of custom formats and their effects on a date that is stored as April 15, 2006 12:15:30 PM.

Table 2-4 Results of custom date-and-time formats

Format pattern	Result of formatting
MM-dd-yy	04-15-06
E, M/d/yyyy	Fri, 4/15/2006
MMM d	Apr 15
MMMM	April
yyyy	2006

Table 2-4 Results of custom date-and-time formats

Format pattern	Result of formatting
W	3 (the week in the month)
w	15 (the week in the year)
DD	105 (the day in the year)
h:mm:ss	12:15:30

Applying a string format

A column that displays string data displays the data as it is stored in the data source. You can format string data to modify text that contains inconsistent capitalization, or to include characters, such as a space or a punctuation mark, at a specific place in the string. For example, you can display telephone numbers in one of the following formats:

```
(415) 555-2121
415.555.2121
415-555-2121
```

How to apply a string format

1 Select a column that contains string data.

2 Choose Format Data.

3 On String Column Format, select a format from the drop-down list. You can select a standard format or define a custom format. Standard and custom formats are described in the following sections.

Selecting a standard string format

Table 2-5 describes the string formats you can choose and provides examples of how the formatted data appears.

Table 2-5 Standard string formats

Format	Description
Lowercase	The string displays in all lowercase, for example: `john smith`
Uppercase	The string displays in all uppercase, for example: `JOHN SMITH`

Defining a custom string format

You can define a custom string format using special symbols to construct a format pattern. Table 2-6 describes these symbols.

Table 2-6 Symbols for defining custom string formats

Symbol	Description
@	Character placeholder. Each @ character displays a character in the string. If the string has fewer characters than the number of @ symbols that appear in the format pattern, spaces appear. Placeholders are filled from right to left, unless you specify an exclamation point (!) at the beginning of the format pattern. See Table 2-7 for examples.
&	Same as @, except if the string has fewer characters, spaces do not appear. See Table 2-7 for examples.
!	Specifies that placeholders are to be filled from left to right. See Table 2-7 for examples.
>	Converts string characters to uppercase.
<	Converts string characters to lowercase.

Table 2-7 shows examples of custom string format patterns and their effects on string data.

Table 2-7 Results of custom string formats

Format pattern	Data in original format	Results of formatting
(@@@) @@@-@@@@	6175551007 5551007	(617) 555-1007 ( ) 555-1007
(&&&) &&&-&&&&	6175551007 5551007	(617) 555-1007 () 555-1007
!(@@@) @@@-@@@@	6175551007 5551007	(617) 555-1007 (555) 100-7
!(&&&) &&&-&&&&	6175551007 5551007	(617) 555-1007 (555) 100-7
!(@@@) @@@-@@@@ + ext 9	5551007	(555) 100-7 + ext 9
!(&&&) &&&-&&&& + ext 9	5551007	(555) 100-7 + ext 9
>&&&-&&&&&-&&	D1234567xy	D12-34567-XY
<&&&-&&&&&-&&	D1234567xy	d12-34567-xy

Working with predefined data formats

When you use an information object data source to create a BIRT report, BIRT Studio translates any predefined Actuate data formats as discussed in the following section.

Table 2-8 shows examples of how BIRT Studio translates predefined numeric data formats.

Table 2-8 Working with predefined numeric formats

Predefined Actuate format	BIRT Studio format
general number	General Number
currency	##,##0.00
fixed	Fixed
standard	Standard
percent	0.00%
scientific	Scientific

Table 2-9 shows examples of how predefined date-and-time formats in the English locale, are translated by BIRT Studio.

Table 2-9 Working with predefined date-and-time formats in the English locale

Predefined Actuate format	Data in original format	BIRT Studio format
General date	04/15/2006	General date
Long date	Thursday, April 15, 2006	Long time
Medium date	15-Apr-06	Medium date
Short date	04/15/06	Short date
Long time	12:00:00 AM	Long time
Medium time	12:00 AM	Medium time
Short time	00:00	Short time
Short week	W16 2006	"W"w yyyy
Short month	4/2006	M/yyyy
Long year	2006	yyyy

Table 2-10 shows examples of custom date-and-time format patterns and their effects on a date that is stored as 04/15/2006 in the data source.

Table 2-10 Working with predefined custom date-and-time format patterns

Predefined Actuate format	Data in original format	BIRT Studio format
mm-dd-yy	04-15-06	MM-dd-yy
ddd, m/d	Thu, 4/15	E, M/d
dddd, m/dd/yy	Thursday, 4/15/06	EEEE, M/dd/yy
mmm d	Apr 15	MMM d
mmmm	April	MMMM
"Quarter" Q	Quarter 2	'Quarter' q
"Day" y	Day 106	'Day' D

If an entry is not included in the tables, BIRT Studio translates that format in the default format. Predefined formats for other types of data, such as string data and Boolean data, do not get converted by BIRT Studio and the resulting data is displayed in the default format.

Formatting data based on conditions

When you format data in a selected column, the format applies to all the values. Often, it is useful to change the format of data when a certain condition is true. For example, you can display sales numbers in red if the value is a negative number and in black if the value is a positive number.

You also can change the format of data in a column according to the values in another column. For example, in a report that shows customer names and the number of days each customer's invoice is past due, you can highlight in blue any customer name that has an invoice past-due value that is between 60 and 90 days. Then, you can highlight in red and bold any customer name that has an invoice past-due value that is greater than 90 days.

To apply conditional formatting, create a rule to define when and how to change the appearance of data. You can apply conditional formats only to data in columns. The following example shows an example of a rule specified for a selected column, Profit.

How to set conditional formats

1 Select the column containing the data you want to format. If the column is not selected, you do not see the correct context menu.

2 Choose Format➛Conditional Formatting from the context menu. Conditional Formatting appears, as shown in Figure 2-11.

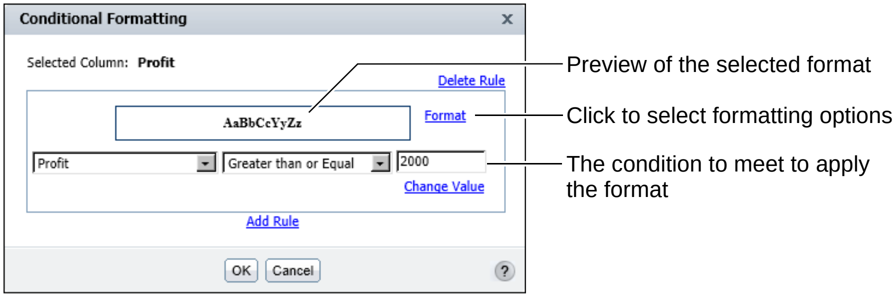
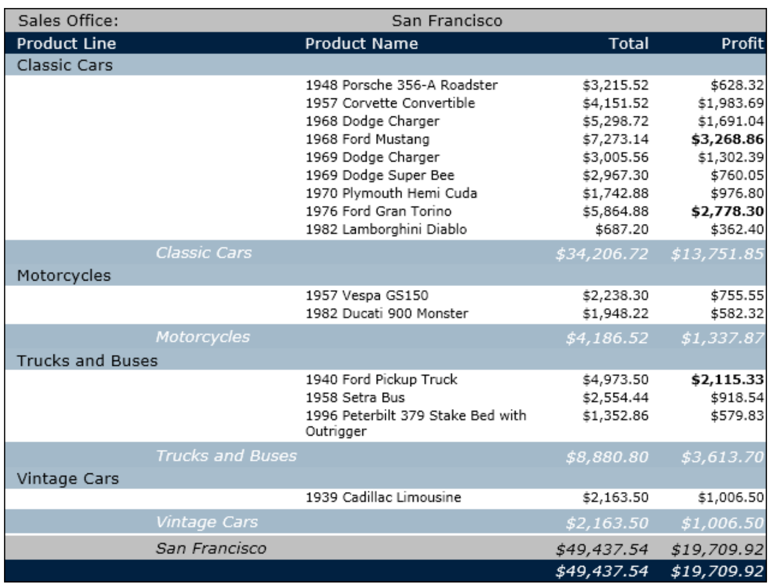


Figure 2-11 Conditional Formatting displaying a rule

3 On Conditional Formatting, create a rule to specify:

- The format to apply. Choose Format to select formatting options.
- The condition that must be true to apply the format, for example, Profit Greater than or Equal to 2000. For more information about specifying a condition, see the next section.

4 Choose OK. Figure 2-12 shows the conditional formatting applied to the report. In the Profit column, numbers greater than 2000 are displayed in bold. If the column contains aggregate data, aggregate values do not reflect the applied conditional formatting properties.



Sales Office:	San Francisco		
Product Line	**Product Name**	**Total**	**Profit**
Classic Cars			
	1948 Porsche 356-A Roadster	$3,215.52	$628.32
	1957 Corvette Convertible	$4,151.52	$1,983.69
	1968 Dodge Charger	$5,298.72	$1,691.04
	1968 Ford Mustang	$7,273.14	**$3,268.86**
	1969 Dodge Charger	$3,005.56	$1,302.39
	1969 Dodge Super Bee	$2,967.30	$760.05
	1970 Plymouth Hemi Cuda	$1,742.88	$976.80
	1976 Ford Gran Torino	$5,864.88	**$2,778.30**
	1982 Lamborghini Diablo	$687.20	$362.40
Classic Cars		*$34,206.72*	*$13,751.85*
Motorcycles			
	1957 Vespa GS150	$2,238.30	$755.55
	1982 Ducati 900 Monster	$1,948.22	$582.32
Motorcycles		*$4,186.52*	*$1,337.87*
Trucks and Buses			
	1940 Ford Pickup Truck	$4,973.50	**$2,115.33**
	1958 Setra Bus	$2,554.44	$918.54
	1996 Peterbilt 379 Stake Bed with Outrigger	$1,352.86	$579.83
Trucks and Buses		*$8,880.80*	*$3,613.70*
Vintage Cars			
	1939 Cadillac Limousine	$2,163.50	$1,006.50
Vintage Cars		*$2,163.50*	*$1,006.50*
San Francisco		*$49,437.54*	*$19,709.92*
		$49,437.54	*$19,709.92*

Figure 2-12 A report design displaying conditional formatting

Specifying a condition

The condition part of a conditional formatting rule is an If expression that must evaluate to true. For example:

```
If the order total is less than 1000
If the customer credit limit is between 10000 and 20000
If the sales office is Tokyo
If the order date is 7/21/2008
```

The Conditional Formatting dialog box helps you construct the If expression by breaking it down to its logical parts. In Figure 2-11, the expression consists of three parts. In Figure 2-13, the expression has four parts.

In the first field, select a column. This column contains the value that determines when conditional formatting takes effect. The selected column does not have to be the same as the column you selected for formatting in the report design. For example, if Product Name is the column selected for formatting, you can select Profit in this field to indicate that for a certain profit amount, conditional formatting applies to the product name.

In the second field, select the comparison test, or operator, to apply to the selected column. You can select Equal to, Less than, Less than or Equal to, and so on. If you select Is Null, Is Not Null, Is True, or Is False, the If expression does not require additional information. For more information about the operators, see Chapter 10, "Functions and operators."

If the selected operator requires a comparison to one or more values, one or more additional fields appear. For example, if you select Less than or Equal to, a third field appears. In this field, you type the comparison value. If you select Between or Not Between, a third and fourth field appear. In these fields, type the lower and upper values, as shown in Figure 2-13.

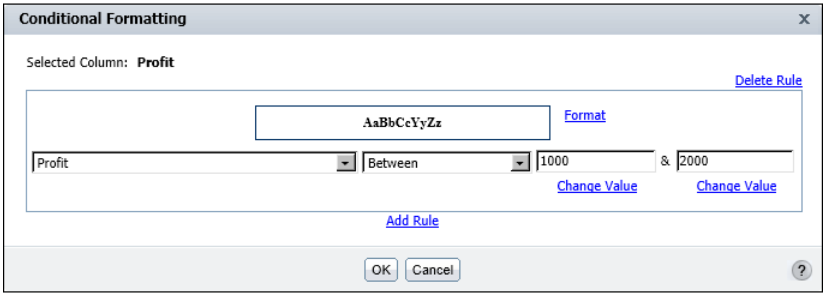


Figure 2-13 Defining a condition that uses the Between operator

Comparing to a literal value

The conditional expression, as shown in Figure 2-13, evaluates the Profit column and compares each value to determine if it is between 1000 and 2000. The 1000 and 2000 values are literal values that you type. Alternatively, you can select from a list of values from the Profit column. Selecting from a list of values is useful if the comparison value is a product name, and you do not know the exact product names, or if the comparison value is a date, and you do not know the date format to type. If the comparison value is a date, BIRT Studio also provides a calendar tool, which you can use to select a date.

How to select a comparison value from a list of values

1. On Conditional Formatting, below the field that takes a comparison value, choose Change Value.
2. On Value, select Specify literal value, and choose Select Values. The values in the selected column appear. Figure 2-14 shows an example of the values in a product name column.

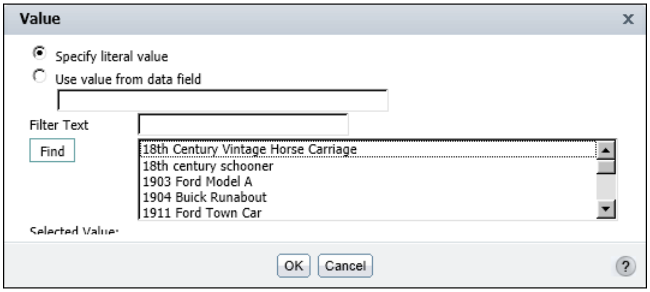


Figure 2-14 Selecting a comparison value

3. If there are too many values in the list, you can type a string in Filter Text to search for values that begin with that string. For example, you can type 18 to view all product names that begin with 18. You cannot search for strings that appear in the middle of a name.
4. Select a value from the list, then choose OK. The value appears in the comparison value field on Conditional Formatting.

Comparing to a value in another column

In a conditional expression, you can compare the values of one column with the values of another column. For example, in a report that displays products, sales prices, and MSRP (Manufacturer Suggested Retail Price), you can create a conditional formatting rule that compares the sale price and MSRP of each product, and highlight the names of the products whose sales price is greater than MSRP.

How to compare to a value in another column

1. On Conditional Formatting, below the field that takes a comparison value, choose Change Value.
2. On Value, select Use value from data field. A list of columns used in the report appears.

3 Select a column from the list, then choose OK. The column name appears in the comparison value field on Conditional Formatting.

Figure 2-15 shows a condition comparing the sale price value with the MSRP value. If the sale price value is greater, the product name appears in bold.

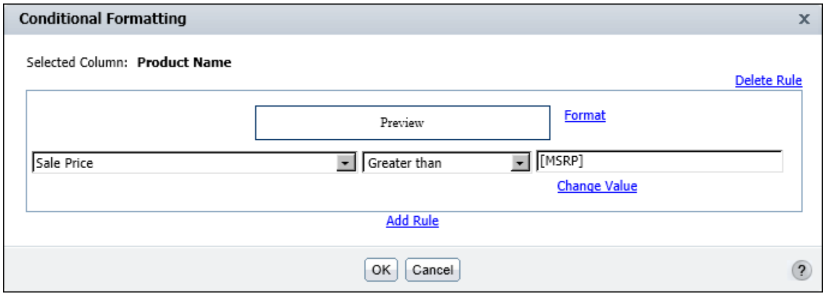


Figure 2-15 Creating a rule that compares values in two columns

Figure 2-16 shows the report design with conditional formatting applied.

Sales Office:	San Francisco		
Product Line	**Product Name**	**Sales Price**	**MSRP**
Classic Cars			
	1948 Porsche 356-A Roadster	$66.99	$69.30
	1956 Porsche 356A Coupe	$139.03	$126.39
	1957 Corvette Convertible	$133.92	$133.92
	1961 Chevrolet Impala	$76.80	$72.76
	1968 Dodge Charger	$110.39	$105.70
	1968 Ford Mustang	$173.17	$175.11
	1969 Dodge Charger	$103.64	$103.64
	1969 Dodge Super Bee	$65.94	$72.37
	1970 Plymouth Hemi Cuda	$72.62	$71.82
	1971 Alpine Renault 1600s	$52.05	$55.11
	1976 Ford Gran Torino	$139.64	$132.29
	1982 Lamborghini Diablo	$34.36	$33.98
Motorcycles			
	1957 Vespa GS150	$49.74	$55.95
	1974 Ducati 350 Mk3 Desmo	$91.85	$91.84
	1982 Ducati 900 Monster	$67.18	$62.33
	1982 Ducati 996 R	$38.62	$36.21

Figure 2-16 A report design with conditional formatting

Specifying multiple conditional formatting rules

You can create up to three conditional formatting rules for a single column. You can, for example, create three rules to set the values of a profit column to one of three colors, depending on the dollar amount. Figure 2-17 shows an example.

For each row of data in the report, BIRT Studio evaluates the rules in the order in which they appear in the list of rules. As it evaluates each rule, BIRT Studio applies the specified format properties if the condition is met.

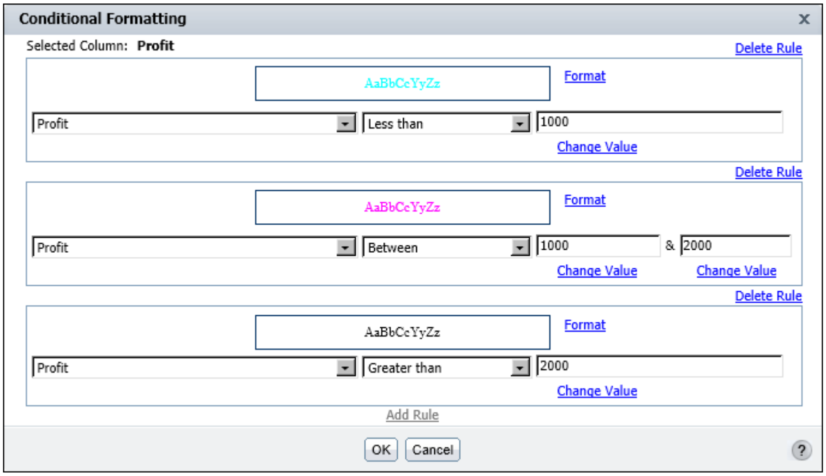


Figure 2-17 Defining multiple conditional formatting rules

When creating multiple rules for a column, be careful that the conditions do not cover overlapping values. Consider the following scenario:

- The first rule sets a profit value to blue if the value exceeds 5000.
- The second rule sets the profit value to green if the value exceeds 1000.

If the profit value is 6000, the value appears in green, not blue as you expect, because the condition in both rules is true (6000 exceeds 5000 and 1000), and the second rule supersedes the first rule. For the rules to make sense, the second rule should set the profit value to green if the value is between 1000 and 5000.

The example, as shown in Figure 2-17, displays three conditions. The first rule specifies a condition for profit values less than 1000, the second rule specifies a condition for profit values between 1000 and 2000, and the third rule specifies a condition for profit values greater than 2000.

Reverting to default formats

When you change the font properties or text alignment for a label or a column of data, your changes override the formats in the template or theme. To revert an element to its default font formats, select the element, choose Font, and set the properties to Auto. To reset text alignment, select the element, and choose an alignment option. You cannot reapply a theme to revert to all the default formats in one step.

If you applied a number, date-and-time, or string format to a column of data, you also can restore these values to the original format that the data source specifies. To do so, select the column, choose Format Data, and select the Unformatted option.

Managing columns

You can change the order of the columns in a table at any time. You can also hide columns, and display hidden columns in a report design.

How to change the order of columns

Use one of the following methods to change the order of columns:

- Right-click any selected column and choose Column➛Reorder Columns. Re-order Columns appears, as shown in Figure 2-18.

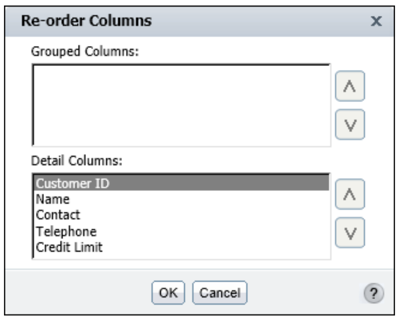
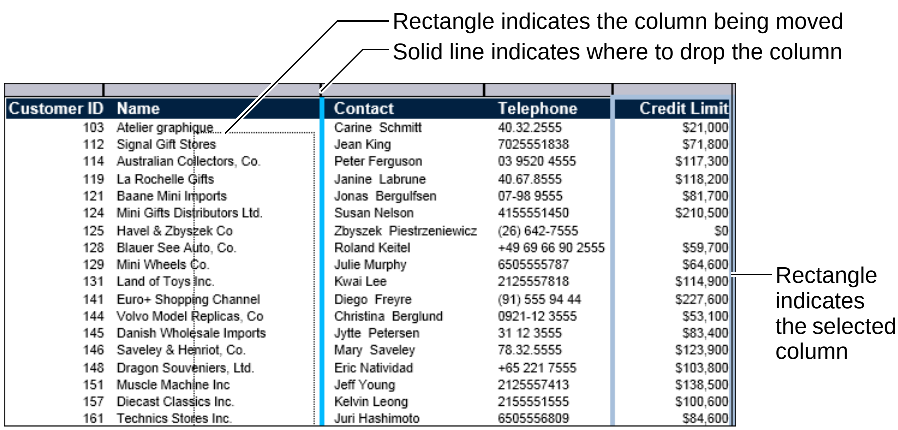


Figure 2-18 Changing the order of columns

On Re-order Columns, as shown in Figure 2-18, use the up or down arrow to move a selected column into the new position. Moving a column up the list moves the column to the left in the table.

- Select the column, and drag it to the new location. As you drag the column, a solid vertical line indicates where you can drop the column, as shown in Figure 2-19.



Customer ID	Name	Contact	Telephone	Credit Limit
103	Atelier graphique	Carine Schmitt	40.32.2555	$21,000
112	Signal Gift Stores	Jean King	7025551838	$71,800
114	Australian Collectors, Co.	Peter Ferguson	03 9520 4555	$117,300
119	La Rochelle Gifts	Janine Labrune	40.67.8555	$118,200
121	Baane Mini Imports	Jonas Bergulfsen	07-98 9555	$81,700
124	Mini Gifts Distributors Ltd.	Susan Nelson	4155551450	$210,500
125	Havel & Zbyszek Co	Zbyszek Piestrzeniewicz	(26) 642-7555	$0
128	Blauer See Auto, Co.	Roland Keitel	+49 69 66 90 2555	$59,700
129	Mini Wheels Co.	Julie Murphy	6505555787	$64,600
131	Land of Toys Inc.	Kwai Lee	2125557818	$114,900
141	Euro+ Shopping Channel	Diego Freyre	(91) 555 94 44	$227,600
144	Volvo Model Replicas, Co	Christina Berglund	0921-12 3555	$53,100
145	Danish Wholesale Imports	Jytte Petersen	31 12 3555	$83,400
146	Saveley & Henriot, Co.	Mary Saveley	78.32.5555	$123,900
148	Dragon Souveniers, Ltd.	Eric Natividad	+65 221 7555	$103,800
151	Muscle Machine Inc	Jeff Young	2125557413	$138,500
157	Diecast Classics Inc.	Kelvin Leong	2155551555	$100,600
161	Technics Stores Inc.	Juri Hashimoto	6505556809	$84,600

Figure 2-19 Moving a column

How to hide a column

Select a column. From the context menu choose Column➛Hide Column. BIRT Studio displays the report without the selected column.

How to display a hidden column

Select a column. From the context menu, choose Column➛Show Column. Show Column appears listing the hidden columns, as shown in Figure 2-20. In Pick Hidden Column, select the column to display. Choose OK. The column appears in the report.

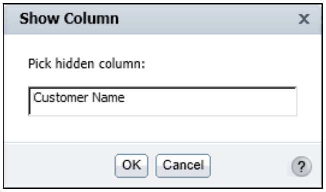


Figure 2-20 Selecting a hidden column to display

Merging data into one column

You can merge the data from two or more columns into one column. When you merge data, the data appears on multiple lines. This format is ideal for some types of data, such as addresses. This format also enables you to include more columns without exceeding the width of a page.

Compare the report designs in Figure 2-21 and Figure 2-22. In Figure 2-21, the report design displays each piece of the address information in a separate column. In Figure 2-22, the data in the AddressLine1, City, State, and Zip columns are merged.

Customer	Phone	AddressLine1	City	State	Zip
American Souvenirs Inc	(203) 555-7845	149 Spinnaker Dr.	New Haven	CT	97823
Cambridge Collectables Co.	(617) 555-5555	4658 Baden Av.	Cambridge	MA	51247
Diecast Classics Inc.	(215) 555-1555	7586 Pompton St.	Allentown	PA	70267
Land of Toys Inc.	(212) 555-7818	897 Long Airport Avenue	NYC	NY	10022
Mini Gifts Distributors Ltd.	(415) 555-1450	5677 Strong St.	San Rafael	CA	97562
Mini Wheels Co.	(650) 555-5787	5557 North Pendale Street	San Francisco	CA	94217
Muscle Machine Inc	(212) 555-7413	4092 Furth Circle	NYC	NY	10022
Signal Gift Stores	(702) 555-1838	8489 Strong St.	Las Vegas	NV	83030
Technics Stores Inc.	(650) 555-6809	9408 Furth Circle	Burlingame	CA	94217

Figure 2-21 A report design displaying address information in multiple columns

Customer	Phone	AddressLine1 City State Zip
American Souvenirs Inc	(203) 555-7845	149 Spinnaker Dr. New Haven CT 97823
Cambridge Collectables Co.	(617) 555-5555	4658 Baden Av. Cambridge MA 51247
Diecast Classics Inc.	(215) 555-1555	7586 Pompton St. Allentown PA 70267
Land of Toys Inc.	(212) 555-7818	897 Long Airport Avenue NYC NY 10022

Figure 2-22 A report design displaying address information in a merged column

When you merge data from multiple columns, the column headers also appear on multiple rows, as shown in Figure 2-22. You can improve the format by merging the column headers to remove the City, State, Zip column headers, and editing the remaining column header. Figure 2-23 shows the improved format.

Customer	Phone	Address
American Souvenirs Inc	(203) 555-7845	149 Spinnaker Dr. New Haven CT 97823
Cambridge Collectables Co.	(617) 555-5555	4658 Baden Av. Cambridge MA 51247
Diecast Classics Inc.	(215) 555-1555	7586 Pompton St. Allentown PA 70267
Land of Toys Inc.	(212) 555-7818	897 Long Airport Avenue NYC NY 10022

Figure 2-23 A report design displaying merged column headers in a single row

Creating a merged column

The following section describes how to create a merged column, and how to merge column headers in a single row.

How to merge data into one column

1. Select all the columns that contain the data to merge. Press Ctrl as you select each column.
2. Right-click one of the selected columns, then choose Column➛Merge Columns, as shown in Figure 2-24.

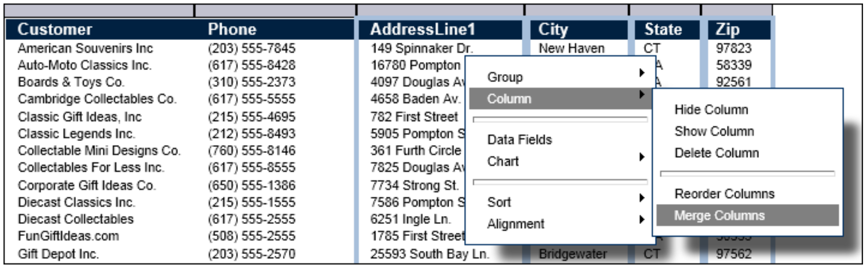


Figure 2-24 Merging columns

How to merge column headers

1 Select the column header in the bottom row, as shown in Figure 2-25. Right-click the selected column header, then choose Cell➛Merge Up.

2 Repeat step 1 until only one column header remains.

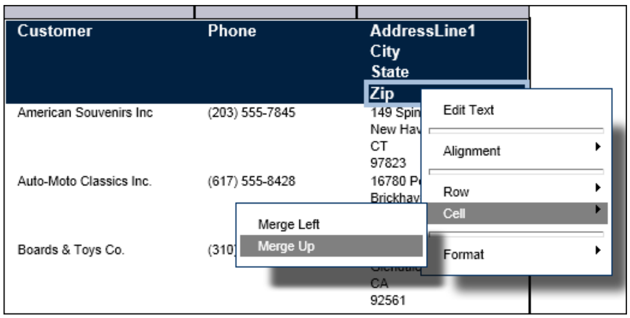


Figure 2-25 Merging column headers

Working with data in a merged column

When working with a merged column in BIRT Studio, you can perform actions such as modifying font, specifying conditional formatting rules, formatting data strings, creating filters, and so on, just as you would for a regular column.

To work with a merged column, select the column, then choose an option from the context menu. Select Data Item appears, as shown in Figure 2-26. The drop-down list contains a list of columns in the merged column.

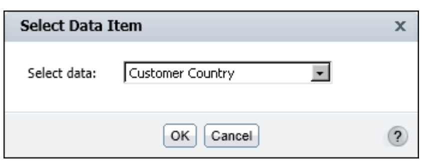


Figure 2-26 Selecting a column from the list of merged columns

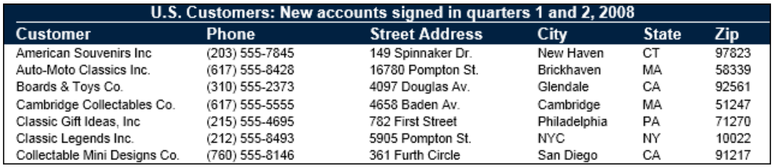
In Select data, select a column for which you want to perform an action such as formatting, or filtering. Choose OK.

Repeat this task for every column in the merged column for which you want to perform an action.

Adding a new column header row

If you need to add more text or more space in the header area, you can add multiple column header rows. Figure 2-27 shows a row with text added above the default column header row.

When you insert a row, the new row contains the same number of columns as the other rows in the table. To display text that spans multiple columns, as shown in Figure 2-27, you must first merge the columns in that row. As Figure 2-27 shows, you also can add borders to a row.



U.S. Customers: New accounts signed in quarters 1 and 2, 2008					
Customer	Phone	Street Address	City	State	Zip
American Souvenirs Inc	(203) 555-7845	149 Spinnaker Dr.	New Haven	CT	97823
Auto-Moto Classics Inc.	(617) 555-8428	16780 Pompton St.	Brickhaven	MA	58339
Boards & Toys Co.	(310) 555-2373	4097 Douglas Av.	Glendale	CA	92561
Cambridge Collectables Co.	(617) 555-5555	4658 Baden Av.	Cambridge	MA	51247
Classic Gift Ideas, Inc	(215) 555-4695	782 First Street	Philadelphia	PA	71270
Classic Legends Inc.	(212) 555-8493	5905 Pompton St.	NYC	NY	10022
Collectable Mini Designs Co.	(760) 555-8146	361 Furth Circle	San Diego	CA	91217

Figure 2-27 A report design displaying text in a row above the column header row

How to add a new row

1. Select a column header.
2. Right-click the column header, then choose Row➛Insert Row Above to add a row above the selected column header, as shown in Figure 2-28. Alternatively, choose Row➛Insert Row Below to add a row below the selected column header.

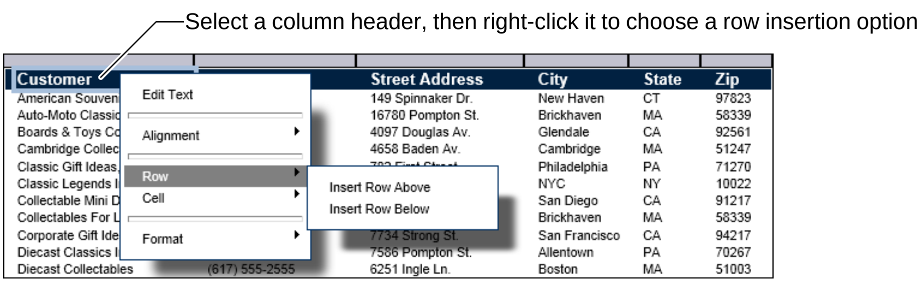
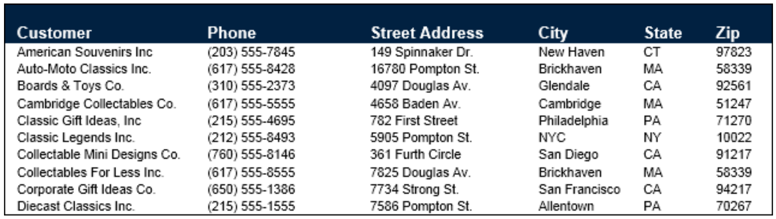


Figure 2-28 Inserting a new row above or below the column header row

A new row appears. Figure 2-29 shows a new row inserted above the default column header row. The row has the same formatting as the row that follows.



Customer	Phone	Street Address	City	State	Zip
American Souvenirs Inc	(203) 555-7845	149 Spinnaker Dr.	New Haven	CT	97823
Auto-Moto Classics Inc.	(617) 555-8428	16780 Pompton St.	Brickhaven	MA	58339
Boards & Toys Co.	(310) 555-2373	4097 Douglas Av.	Glendale	CA	92561
Cambridge Collectables Co.	(617) 555-5555	4658 Baden Av.	Cambridge	MA	51247
Classic Gift Ideas, Inc	(215) 555-4695	782 First Street	Philadelphia	PA	71270
Classic Legends Inc.	(212) 555-8493	5905 Pompton St.	NYC	NY	10022
Collectable Mini Designs Co.	(760) 555-8146	361 Furth Circle	San Diego	CA	91217
Collectables For Less Inc.	(617) 555-8555	7825 Douglas Av.	Brickhaven	MA	58339
Corporate Gift Ideas Co.	(650) 555-1386	7734 Strong St.	San Francisco	CA	94217
Diecast Classics Inc.	(215) 555-1555	7586 Pompton St.	Allentown	PA	70267

Figure 2-29 An empty row above the column header row in a report design

How to merge column headers to type text that spans multiple columns

1 Select the first column header in the new row. Right-click the column header, then choose Cell➛Merge Right, as shown in Figure 2-30. The first cell merges with the second cell to become a single, wider cell.

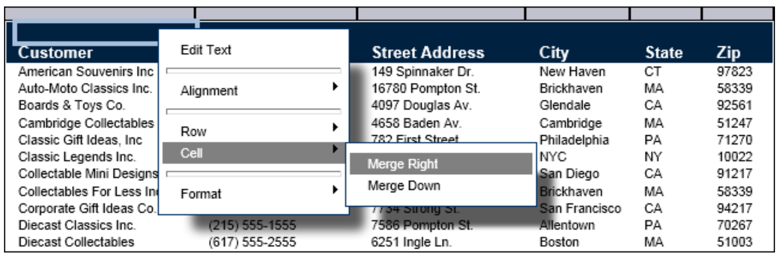
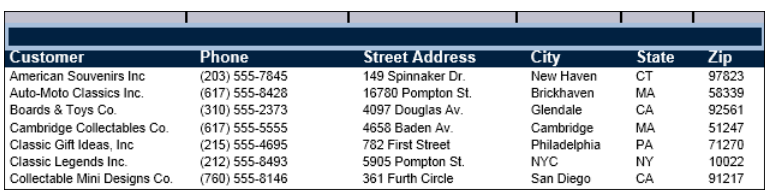


Figure 2-30 Merging column headers

2 Repeat step 1 to merge all cells in the column. Figure 2-31 shows the selection of a single cell created by merging six cells.



Customer	Phone	Street Address	City	State	Zip
American Souvenirs Inc	(203) 555-7845	149 Spinnaker Dr.	New Haven	CT	97823
Auto-Moto Classics Inc.	(617) 555-8428	16780 Pompton St.	Brickhaven	MA	58339
Boards & Toys Co.	(310) 555-2373	4097 Douglas Av.	Glendale	CA	92561
Cambridge Collectables Co.	(617) 555-5555	4658 Baden Av.	Cambridge	MA	51247
Classic Gift Ideas, Inc	(215) 555-4695	782 First Street	Philadelphia	PA	71270
Classic Legends Inc.	(212) 555-8493	5905 Pompton St.	NYC	NY	10022
Collectable Mini Designs Co.	(760) 555-8146	361 Furth Circle	San Diego	CA	91217

Figure 2-31 Results of merging all the column headers

3 To display text in this cell, double-click in the cell, type the text, then press Enter.

4 To add borders:

1. Select the cell. Right-click it, and choose Format➛Border, as shown in Figure 2-32.

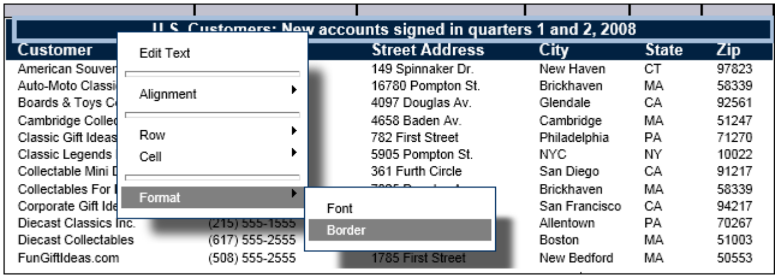


Figure 2-32 Adding borders around the merged column header

2 On Border, specify the type of border or borders to add. You can set the border style, color, and width. Figure 2-33 shows the bottom border set to create a solid white border, one pixel wide. Choose OK.

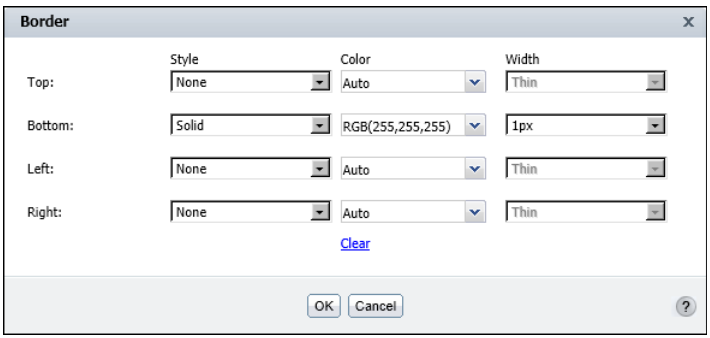


Figure 2-33 Specifying the border type

Changing the page layout to fit data

When you insert data fields in a table, by default, the table expands to accommodate the data. The width of the columns depend on several factors, including the column sizes defined in the template and the data field sizes defined in the original data source.

Figure 2-34 shows an example of a table that uses the default layout. The columns are approximately the same width, even though the data in the state and zip code columns, for example, occupy less space than the data in the Customer and Street Address columns. If you insert many data fields in a table, you probably need to adjust the width of columns to improve the spacing between the columns.

Customer	Customer ID	Credit Limit	Contact	Phone	Street Address	City	State	Zip Code
Signal Gift Stores	112	71,800	Jean King	(702) 555-1838	8489 Strong St.	Las Vegas	NV	83030
Mini Gifts Distributors Ltd.	124	210,500	Susan Nelson	(415) 555-1450	5677 Strong St.	San Rafael	CA	97562
Mini Wheels Co.	129	64,600	Julie Murphy	(650) 555-5787	5557 North Pendale Street	San Francisco	CA	94217
Land of Toys Inc.	131	114,900	Kwai Lee	(212) 555-7818	897 Long Airport Avenue	NYC	NY	10022
Muscle Machine Inc	151	138,500	Jeff Young	(212) 555-7413	4092 Furth Circle	NYC	NY	10022
Diecast Classics Inc.	157	100,600	Kelvin Leong	(215) 555-1555	7586 Pompton St.	Allentown	PA	70267
Technics Stores Inc.	161	84,600	Juri Hashimoto	(650) 555-6809	9408 Furth Circle	Burlingame	CA	94217

Figure 2-34 A report design displaying data in the default table layout

If you intend to print a report or export the report to a page-based format such as PDF or Word, check the report output in those formats to ensure that all the data fits on the page. Figure 2-35 shows a portion of the example report in PDF format. The Zip Code column does not fit on the page.

Customer Directory

Customer	Customer ID	Credit Limit	Contact	Phone	Street Address	City	State
Signal Gift Stores	112	71,800	Jean King	(702) 555-1838	8489 Strong St.	Las Vegas	NV
Mini Gifts Distributors Ltd.	124	210,500	Susan Nelson	(415) 555-1450	5677 Strong St.	San Rafael	CA
Mini Wheels Co.	129	64,600	Julie Murphy	(650) 555-5787	5557 North Pendale Street	San Francisco	CA
Land of Toys Inc.	131	114,900	Kwai Lee	(212) 555-7818	897 Long Airport Avenue	NYC	NY
Muscle Machine Inc	151	138,500	Jeff Young	(212) 555-7413	4092 Furth Circle	NYC	NY
Diecast Classics Inc.	157	100,600	Kelvin Leong	(215) 555-1555	7586 Pompton St.	Allentown	PA

Figure 2-35 Default PDF document layout does not accommodate all the data

You can change the page layout to better fit the data in your report by using the following techniques:

- Adjusting the widths of the columns
- Changing the layout type, page size, and orientation

Adjusting column widths

If you want to control precisely the widths of columns in a table, first set the Layout Preference property in Page Setup to Fixed Width. The option to set a column’s width is not available if the layout type is Auto Expand Width.

How to set a column’s width

1. Select the column, right-click it, then choose Column➛Column Width.
2. On Column Properties, as shown in Figure 2-36, type the width value. The default unit is inches, but you can select cm, mm, picas, or points.

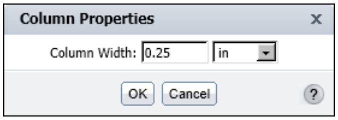


Figure 2-36 Setting a column's width using Column Properties

Choose OK.

Changing the layout type, page size, and orientation

As described previously, a table's width expands to display all the columns of data. This feature is typical for viewing a report online. If, however, you want to design a report that shows the same output, whether it is viewed online or in a page-based format such as PDF, you can change the report's layout type from Auto Expand Width to Fixed Width. If you select Fixed Width, you can specify a page width, and this value is reflected in the web report, as well as in PDF and printed reports. If you use the default Auto Expand Width setting, the page size values can differ for the web report, and the PDF or printed reports. Additionally, you can modify column width if you use Fixed Width layout type, but this option is not available if you use Auto Expand Width layout type.

How to set the layout type, page size, and orientation

1 In the side menu, choose Page Setup. Page Setup appears, as shown in Figure 2-37.

2 Set the following options:

   1 In Layout Preference:

      - Select Auto Expand Width to set the column widths based on the other page options you select and on the number of columns in the table.
      - Select Fixed Width to set the page width and precise column widths yourself.

   2 In Orientation, accept the default selection of Auto, or select Portrait or Landscape mode. If you choose Auto, BIRT Studio sets the page orientation depending on whether the report is longer or wider. If your table is too wide to print in a standard 8.5-inch page width, select Landscape.

   3 In Paper Size, select one of the standard sizes, A4, US Letter, or US Legal. Alternatively, select Custom to set a custom page width and height.

   4 If you choose Custom, specify the page size values in Width and Height.

   5 In Top Margin, Bottom Margin, Left Margin, and Right Margin, specify the margin sizes to use for each page.

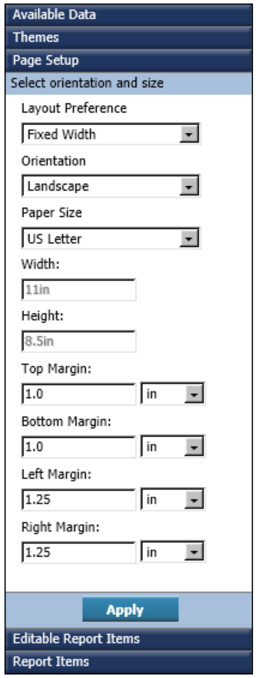


Figure 2-37 Page setup options

Chapter

3

Inserting calculated data

This chapter contains the following topics:

- About calculated data
- Creating a computed column
- Adding data fields for an expression
- Writing expressions

About calculated data

Most business reports require calculations to track sales, finances, inventory, and other critical business activities. You can keep a count of items in a warehouse, or provide more complex financial data, such as tracking stock portfolio performance over time. Some of this calculated data can be included in the data set if the person who created the data set anticipated that you and other users need to work with this type of data.

Sometimes, however, a data set does not provide all the data you want to display in your report. In this case, you can create your own calculations. To do so, you create a new field called a computed column. Figure 3-1 shows a report design that uses a computed column, Total, to display the total of each order line item. In this example, the Total values need to be calculated because the data set does not include this data.

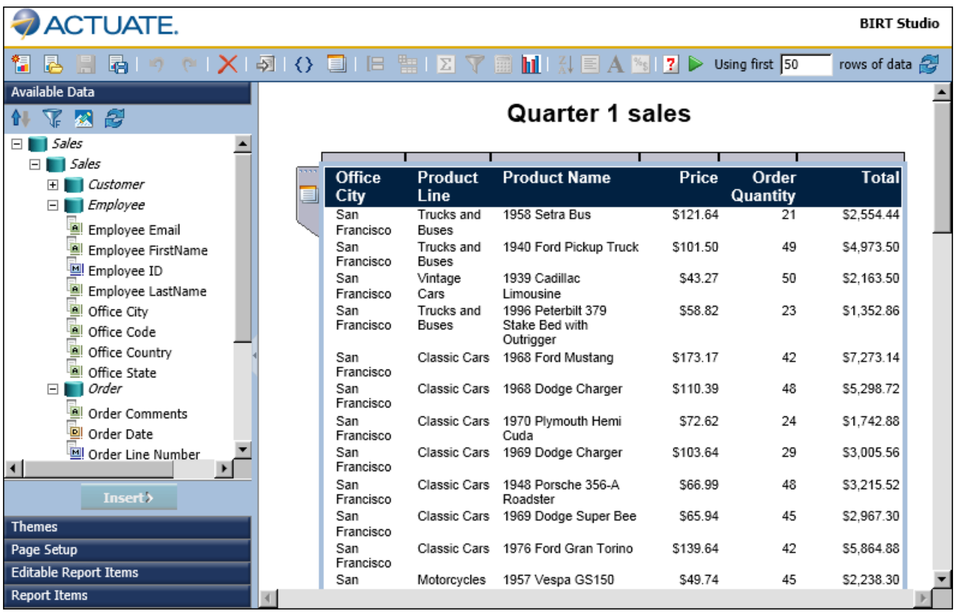


Figure 3-1 A report design displaying a computed column, Total

Creating a computed column

When you create a computed column, you write an expression, which is a statement that indicates how to calculate the data. In the report, as shown in Figure 3-1, each value in the Total column is calculated by multiplying the value in the QUANTITYORDERED field with the value in the PRICEEACH field.

Figure 3-2 shows the following expression specified for the computed column, Total:

```
[QUANTITYORDERED] * [PRICEEACH]
```

When you refer to a data field in an expression, you must enclose the field name within brackets ([ ]). The field names are case-sensitive.

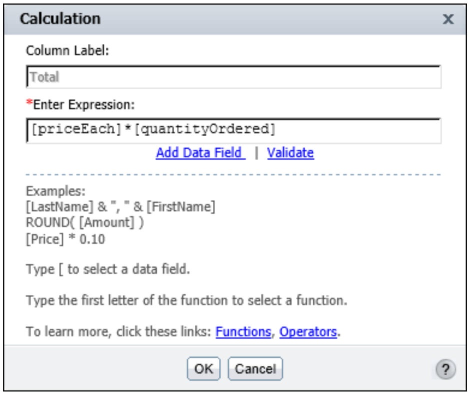


Figure 3-2 Calculation displaying an expression for the computed column

About expressions

BIRT Studio supports typical mathematical operations, such as addition, subtraction, multiplication, and division. Computed columns, however, are not limited to mathematical calculations. BIRT Studio also supports many functions for manipulating date-and-time and string data.

A function is a set of instructions that do something and return a result. For example, if a customer name field contains values with leading or trailing blank characters, you can remove the blank characters by using the TRIM( ) function as follows:

```
TRIM([CustomerName])
```

In this example, [CustomerName] is the input value, or argument, you supply to the TRIM( ) function to tell the function to trim the values in the CustomerName field.

More examples of expressions used in computed columns appear later. For descriptions of the supported functions, see Chapter 10, “Functions and operators.”

How to create a computed column

1. Select the table column to the left of the new computed column you want to insert.
2. On the toolbar, choose New Computed Column, as shown in Figure 3-3.

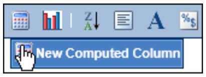


Figure 3-3 Choosing New Computed Column

Calculation appears.

3 In Column Label, type a name for the computed column. The name you specify appears in the column header.

4 In Enter Expression, type the expression to calculate the values you want to display:

- To use a data field in the expression, type the left bracket ([), then select a field from the list that appears, as shown in Figure 3-4.

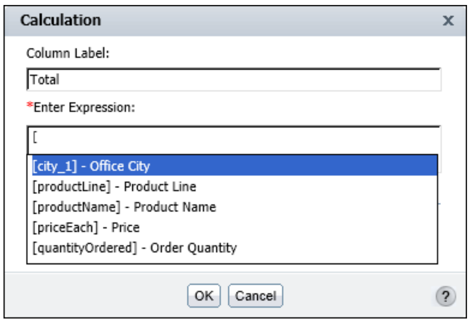


Figure 3-4 Selecting a data field to use in an expression

The list shows only the fields in use in the report design. It does not show all the fields in the data set. To use a field that is not in the list, you must add the field to the report design. This task is described later in this chapter.

- To use a function, type the first letter of the function, then select a function from the list that appears, as shown in Figure 3-5. The functions indicate the arguments, if any, you need to supply.

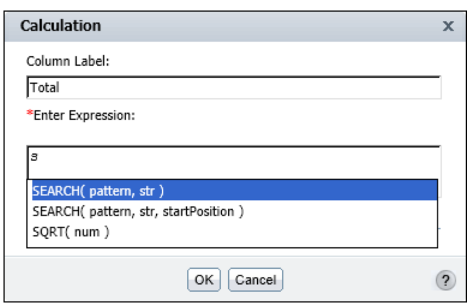


Figure 3-5 Selecting a function to use in an expression

5 When you finish writing the expression, choose Validate. If the expression is syntactically correct, the dialog box displays the message, This Expression is valid. If the expression contains an error, the dialog box displays a message that describes the error.

6 After you validate the expression, Choose OK. The computed column appears in the report design.

Creating aggregate data in a computed column

BIRT Studio enables you to create a new computed column, as well as create aggregate data for the values in this column, for certain types of calculated columns. For more information about the types of aggregate calculations you can create, see “Aggregating data” in Chapter 4, “Organizing data in a report.”

Adding data fields for an expression

It is common to use a data field in an expression for a computed column. As mentioned earlier, a computed column only has access to data fields that are used in the report design. Sometimes, however, you need to write an expression that uses a field provided by the data set, but you do not want to display the field values in the report.

For example, if you insert a CustomerName field and a Phone field in a table, then you insert a computed column, the computed column only has access to the CustomerName and Phone fields. You cannot create an expression that combines, for example, values from the AddressLine1, City, State, and PostalCode fields. To create such a computed column, you must first add the data fields to the table, but without actually inserting the fields in the table.

How to add a data field

1 Right-click the table handle, as shown in Figure 3-6.

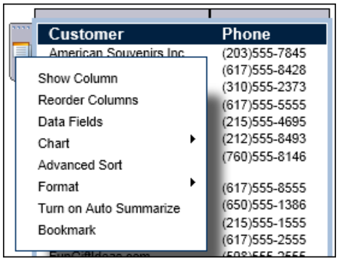


Figure 3-6 Adding a data field to a table

2 Choose Data Fields. The Data Fields dialog box appears. Figure 3-7 shows an example of Data Fields displaying two fields, customerName and phone, currently used in the table.

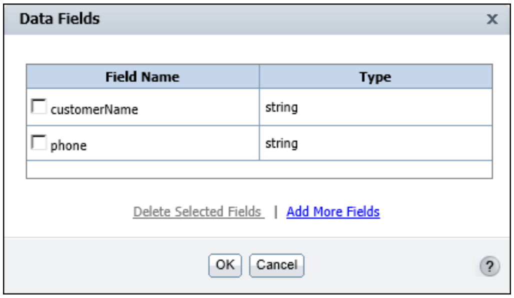


Figure 3-7 Data Fields displaying the fields currently used in the table

3 Choose Add More Fields. Data Set displays all the fields in the data set, as shown in Figure 3-8.

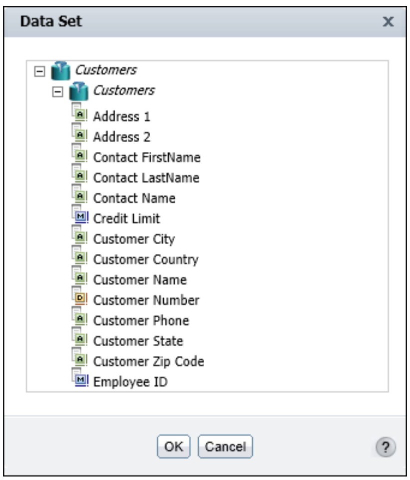


Figure 3-8 Data Set displaying all the available fields

4 Select the field to add. To add multiple fields, press Ctrl as you select each field. Choose OK. The selected fields are available to use in any expression.

To define an expression for a computed column, you can choose the Add Data Field link, as shown in Figure 3-9, then follow steps 2 to 4 to add fields.

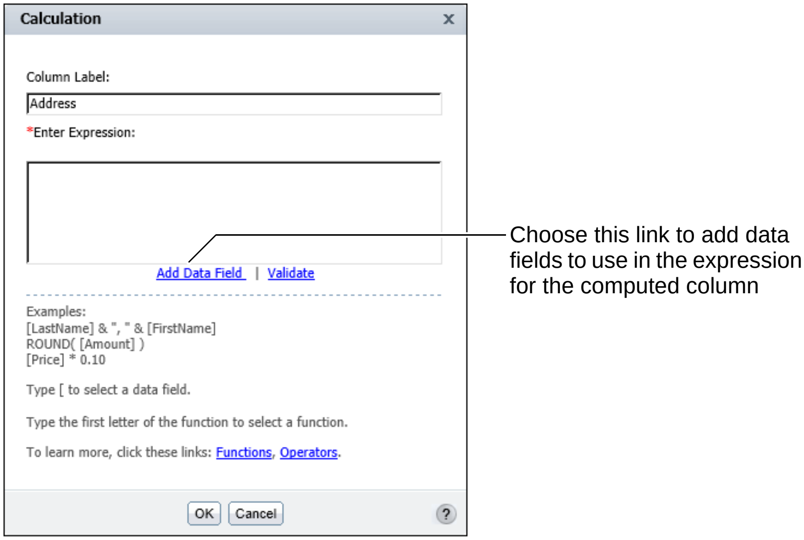


Figure 3-9 Choose Add Data Field to add fields

Writing expressions

An expression is a statement that produces a value. An expression can be a literal value, such as:

```
1.23
"Hello, World!"
```

An expression can contain any combination of literal values, operators, functions, and references to data fields, as shown in the following examples.

The following expression displays a customer's first and last names stored in two fields. The & operator concatenates string values.

```
[FirstName] & [LastName]
```

The following expression displays a full address by concatenating values from four data fields and adding commas where appropriate:

```
[Address1] & ", " & [City] & ", " & [State] & " " & [Zipcode]
```

The following expression calculates a gain or loss in percent. The expression uses the subtraction, division, and multiplication operators, -, /and *.

```
([SalePrice] - [UnitPrice])/[UnitPrice] * 100
```

The following expression uses the DIFF_DAY( ) function to calculate the number of days it took to process an order for shipping. The function takes two arguments, a start date and an end date. In the example, two data fields are supplied as the arguments. BIRT Studio runs this expression against each value in the fields and calculates a value for each data row.

```
DIFF_DAY([OrderDate], [ShippedDate])
```

The following expression uses the ADD_DAY( ) function to calculate a payment due date when the payment term is net 30. The function takes two arguments, a start date and the number of days to add.

```
ADD_DAY([InvoiceDate], 30)
```

The following expression uses the IF( ) function to evaluate if the value in the country column is UK. If the condition is true, the function replaces the value with United Kingdom. If the condition is false, the country values appear as stored.

```
IF(([Country]="UK"),"United Kingdom",[Country])
```

Using numbers and dates in an expression

When you create an expression that contains a literal number, you must type the number according to the conventions of the US English locale. In other words, use a period (.), not a comma (,) as the decimal separator, even if you are working in, for example, the French locale. For example:

```
Correct: ([Quantity] * [Price]) * 1.5
Incorrect: ([Quantity] * [Price]) * 1,5
```

Similarly, when you create an expression that contains a literal date, type the date according to the conventions of the US English locale. For example, if you are working in the French locale, type 03/12/2007 to represent March 12, 2007. Do not type 12/03/2007, which is the convention for the French locale. You must enclose literal date values in double quotation marks(" "), as shown in the following expression that calculates the number of days from the order date to Christmas:

```
DIFF_DAY([OrderDate], "12/25/08")
```

About reserved characters in an expression

Some characters are reserved for internal use and have a special meaning. For example, as you saw in the expression examples in the previous sections, BIRT Studio uses brackets to denote a data field. The following characters are reserved in BIRT Studio:

```
[
]
?
' (single quotation mark)
```

If the name of a data field contains reserved characters, BIRT Studio removes the reserved characters when you select the data field to use in an expression. For example, if the name of a data field is OBSOLETE?, BIRT Studio changes it to [OBSOLETE'?'] in the expression. If you type [OBSOLETE?] in the expression, the dialog box displays an error message. To avoid syntax errors, always select the field and let the software construct the correct expression. Figure 3-10 shows an example of the Calculation dialog box.

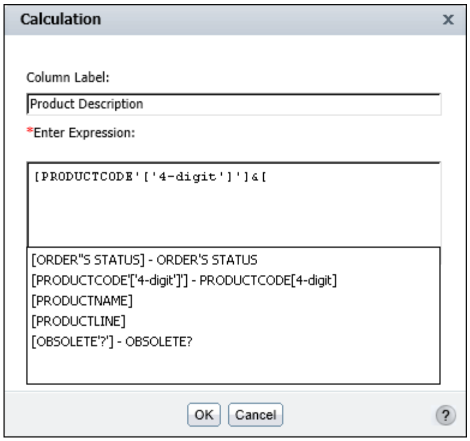


Figure 3-10 Calculation displaying a list of available data fields

The example, as shown in Figure 3-10, displays a list of data fields, three with reserved characters in their names. The third, fourth, and fifth fields show both versions of their names, the changed name and the original name with reserved characters, as follows:

```
[ORDER''S STATUS] - ORDER'S STATUS
[PRODUCTCODE'['4digit']'] - PRODUCTCODE[4-digit]
[OBSOLETE'?'] - OBSOLETE?
```

Chapter

4

Organizing data in a report

This chapter contains the following topics:

- Sorting data
- Organizing data in groups
- Organizing data in sections
- Aggregating data
- Hiding details
- Suppressing duplicate values
- Starting each group or section on a new page

Sorting data

When you insert data in a report design, the data set determines the default sort order for the data rows. If the data set sorts a field in ascending order, the column values appear in ascending order in the design. Typically, however, data appears randomly. A field is likely to display customer names, for example, in the order in which customers were added to the database, rather than in alphabetical order. Sorting data, therefore, is an important task in creating a useful report.

Compare the reports in Figure 4-1.

Country	Customer	Credit Limit
Australia	Australian Collectors, Co.	117300
Denmark	Danish Wholesale Imports	83400
France	Atelier graphique	21000
France	La Rochelle Gifts	118200
France	Saveley & Henriot, Co.	123900
Germany	Blauer See Auto, Co.	59700
Norway	Baane Mini Imports	81700
Poland	Havel & Zbyszek Co	0
Singapore	Dragon Souveniers, Ltd.	103800
Spain	Euro+ Shopping Channel	227600
Sweden	Volvo Model Replicas, Co	53100
USA	Land of Toys Inc.	114900
USA	Mini Gifts Distributors Ltd.	210500
USA	Mini Wheels Co.	64600
USA	Signal Gift Stores	71800

Country	Customer	Credit Limit
Australia	Australian Collectors, Co.	117300
Denmark	Danish Wholesale Imports	83400
France	Atelier graphique	21000
France	La Rochelle Gifts	118200
France	Saveley & Henriot, Co.	123900
Germany	Blauer See Auto, Co.	59700
Norway	Baane Mini Imports	81700
Poland	Havel & Zbyszek Co	0
Singapore	Dragon Souveniers, Ltd.	103800
Spain	Euro+ Shopping Channel	227600
Sweden	Volvo Model Replicas, Co	53100
USA	Signal Gift Stores	71800
USA	Mini Gifts Distributors Ltd.	210500
USA	Mini Wheels Co.	64600
USA	Land of Toys Inc.	114900

Figure 4-1 Report designs displaying unsorted and sorted data

The report on the left displays the data rows in the order the data set returns them. The report on the right displays the same data, except that the rows are sorted by country in alphabetical order.

You can sort data in ascending or descending order, and you can sort data by multiple columns. For example, Figure 4-2 shows the results of sorting by country, then by customer name. For rows where the country names repeat, France and USA, the customer names appear in alphabetical order.

Country	Customer	Credit Limit
Australia	Australian Collectors, Co.	117300
Denmark	Danish Wholesale Imports	83400
France	Atelier graphique	21000
France	La Rochelle Gifts	118200
France	Saveley & Henriot, Co.	123900
Germany	Blauer See Auto, Co.	59700
Norway	Baane Mini Imports	81700
Poland	Havel & Zbyszek Co	0
Singapore	Dragon Souveniers, Ltd.	103800
Spain	Euro+ Shopping Channel	227600
Sweden	Volvo Model Replicas, Co	53100
USA	Land of Toys Inc.	114900
USA	Mini Gifts Distributors Ltd.	210500
USA	Mini Wheels Co.	64600
USA	Signal Gift Stores	71800

Figure 4-2 A report design displaying data sorted by country and customer

Sorting on a single column

To sort data on a single column, first select the column. Then choose Sort Ascending or Sort Descending, as shown in Figure 4-3.

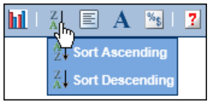


Figure 4-3 Selecting a sort option

Sorting on multiple columns

You can sort data by up to three columns. When you sort by multiple columns, you do not use the Sort Ascending or Sort Descending buttons. Instead, you use the Advanced Sort dialog box, as shown in Figure 4-4.

When you sort on multiple columns, it is important to understand the order of precedence for the sort. On Advanced Sort, select the columns in the order in which to sort data. For example, to sort data by city first, then by customer name, you must select the columns in that order on Advanced Sort.

How to sort data on multiple columns

1. Select a column. You can select any column in the table.
2. Right-click the selected column, then choose Sort➛Advanced Sort from the context menu.
3. On Advanced Sort, select a column from the first drop-down list, and choose either Ascending or Descending order.
4. Select another column from the second drop-down list, and choose the sort order.
5. Optionally, select a third column on which to sort. Figure 4-4 shows the sort criteria for the report design in Figure 4-2.

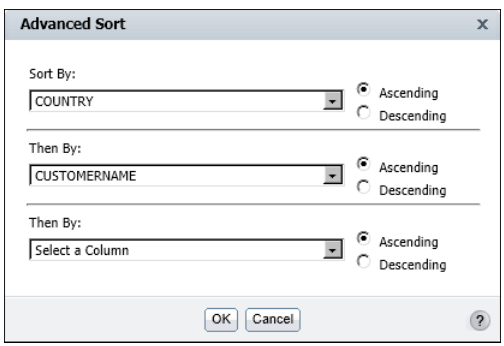


Figure 4-4 Specifying advanced sorting

Restoring data to its original order

To restore data to its original, unsorted order, complete the following steps.

1 Select a column. You can select any column in the table. Right-click the column, then choose Sort➛Advanced Sort.

2 In Advanced Sort, for each column specified for a sort, select Select a Column from the drop-down list, as shown in Figure 4-5. Choose OK.

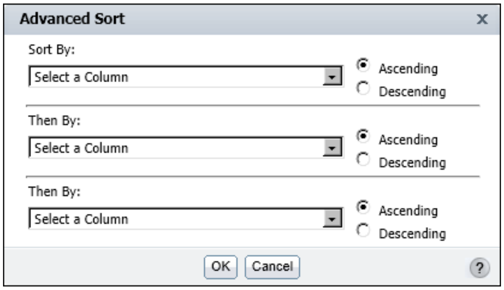


Figure 4-5 Removing sorting criteria

Organizing data in groups

As the previous section describes, sorting data makes a report more readable and useful. Sorting, however, is only one of the ways to organize data in a report. It is common for reports to present data that is organized into meaningful groups, especially reports that contain large amounts of data.

Consider the task of listing every item a corporation owns, along with information such as the category, purchase price, purchase date, inventory code, location, and supplier. If a report presents all these items in an unorganized list or even in a sorted list, there is no way to determine how much the corporation has spent, for example, on office furniture or computer hardware, because this information cannot be calculated. The report cannot help you see which year had the most purchases, nor is it easy to tell which items are located in a field office and which items are in the warehouse.

To organize this information into a useful inventory report, you create data groups. Data groups contain related data rows. For example, you can create a report that lists items by category—all office furniture in one group, all computer hardware in another group, all computer software in a third group, and so on. For each group, you can calculate aggregate data, such as the total purchase price or the count of items in a group. Organizing data in groups makes it easier to compare and analyze information.

Compare the reports in Figure 4-6 and Figure 4-7. The report design in Figure 4-6 displays sales information in a simple list. The data rows are sorted by sales office, then by product line. Notice the repeated sales office name and product lines.

Sales Office	Product Line	Product Name	Total	Profit
San Francisco	Classic Cars	1952 Alpine Renault 1300	$10,286.40	$5,554.56
San Francisco	Classic Cars	1992 Porsche Cayenne Turbo Silver	$3,782.61	$1,061.19
San Francisco	Classic Cars	1970 Dodge Coronet	$1,441.59	$502.86
San Francisco	Classic Cars	1970 Triumph Spitfire	$4,096.03	$1,246.51
San Francisco	Classic Cars	1998 Chrysler Plymouth Prowler	$3,389.28	$1,054.55
San Francisco	Classic Cars	1969 Ford Falcon	$4,069.44	$2,076.24
San Francisco	Classic Cars	1969 Corvair Monza	$3,834.54	$1,427.76
San Francisco	Classic Cars	2001 Ferrari Enzo	$7,241.83	$3,322.64
San Francisco	Classic Cars	1952 Alpine Renault 1300	$6,557.76	$3,008.88
San Francisco	Classic Cars	1992 Porsche Cayenne Turbo Silver	$2,420.00	$884.84
San Francisco	Classic Cars	1962 LanciaA Delta 16V	$5,138.56	$1,312.02
San Francisco	Classic Cars	1958 Chevy Corvette Limited Edition	$1,088.32	$483.74
San Francisco	Classic Cars	1970 Dodge Coronet	$2,236.95	$780.30
San Francisco	Classic Cars	1992 Ferrari 360 Spider red	$3,386.80	$1,828.80
San Francisco	Classic Cars	1998 Chrysler Plymouth Prowler	$7,368.00	$2,292.50

Figure 4-6 A report design displaying data in a simple list

Sales Office	Product Line		Product Name	Total	Profit
San Francisco					
	Classic Cars				
			1948 Porsche 356-A Roadster	$3,215.52	$628.32
			1956 Porsche 356A Coupe	$6,534.41	$1,914.31
			1957 Corvette Convertible	$4,151.52	$1,983.69
			1961 Chevrolet Impala	$1,766.40	$1,022.81
			1968 Dodge Charger	$5,298.72	$1,691.04
			1968 Ford Mustang	$7,273.14	$3,268.86
			1969 Corvair Monza	$4,636.50	$1,694.88
			1969 Dodge Charger	$3,005.56	$1,302.39
			1969 Dodge Super Bee	$2,967.30	$760.05
			1969 Ford Falcon	$7,720.18	$3,899.88
			1970 Plymouth Hemi Cuda	$1,742.88	$976.80
			1970 Triumph Spitfire	$2,929.92	$723.84
			1971 Alpine Renault 1600s	$1,405.35	$363.69
			1976 Ford Gran Torino	$5,864.88	$2,778.30
			1982 Lamborghini Diablo	$687.20	$362.40
			2001 Ferrari Enzo	$5,485.92	$2,331.45
	Sum	*Classic Cars*		*$64,685.40*	*$25,702.71*
	Motorcycles				
			1957 Vespa GS150	$2,238.30	$755.55
			1974 Ducati 350 Mk3 Desmo	$2,755.50	$1,071.60
			1982 Ducati 900 Monster	$1,948.22	$582.32
			1982 Ducati 996 R	$1,274.46	$477.84
			2002 Yamaha YZR M1	$3,472.48	$1,969.00
	Sum	*Motorcycles*		*$11,688.96*	*$4,856.31*
	Ships				
			18th century schooner	$3,771.57	$1,054.35
	Sum	*Ships*		*$3,771.57*	*$1,054.35*
	Trains				
			Collectable Wooden Train	$2,107.60	$621.28
	Sum	*Trains*		*$2,107.60*	*$621.28*
	Trucks and Buses				
			1940 Ford Pickup Truck	$4,973.50	$2,115.33
			1958 Setra Bus	$2,554.44	$918.54
			1996 Peterbilt 379 Stake Bed with Outrigger	$1,352.86	$579.83
	Sum	*Trucks and Buses*		*$8,880.80*	*$3,613.70*
	Vintage Cars				
			1903 Ford Model A	$6,098.72	$2,888.62
			1904 Buick Runabout	$1,916.88	$653.04
			1912 Ford Model T Delivery Wagon	$3,643.15	$1,344.56
			1939 Cadillac Limousine	$2,163.50	$1,006.50
	Sum	*Vintage Cars*		*$13,822.25*	*$5,892.72*
Sum San Francisco				*$104,956.58*	*$41,741.07*
Sum				*$104,956.58*	*$41,741.07*

Figure 4-7 A report design displaying data grouped by sales office and product line

The report design in Figure 4-7 shows the same data. Unlike the report design in Figure 4-6, this design groups the rows by sales office then by product line. Data appears in a hierarchy. The report design does not contain repeated values for office names and product lines. In addition, totals, displayed in italics, appear at the end of each product line and sales office group.

As the example reports Figure 4-6 and Figure 4-7 show, a report design that groups data provides a more effective way to view data. When you group data, you can:

- Add subtotals, counts, averages, or other aggregate information at the beginning or end of each group.
- Insert a page break before or after each group to start each group of data on a new page.
- Hide the details of each group to view a summary report.

In addition, BIRT Studio completes the following actions when you group data:

- Removes duplicate values.
- Sorts the values of each group. For example, a product line group displays the product lines in alphabetical order (Classic Cars to Vintage Cars), as shown in Figure 4-7.
- Generates a table of contents when you view the finished report in the viewer. The table of contents, which displays the group values, supports navigating to specific locations in the report. This feature is particularly useful when a report contains many pages.

How to group data

1. Select the column that contains the data to group.
2. Choose Add Group, as shown in Figure 4-8.

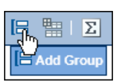


Figure 4-8 Adding a group

If the column you select contains string or numeric data, BIRT Studio groups the data by each unique value in the column. If the column you select contains date-and-time data, you have additional grouping options described in the following section.

Grouping data on a date-and-time column

When you group data on a date-and-time column, you have two options. You can show every individual date or time value, or you can group the data by a specific time interval. The second option is more typical. A shipping report, for example, can organize shipment data by month or by quarter, rather than by date.

Figure 4-9 shows three report designs that contain the same data. The first report design displays data that has not been grouped, the second design groups shipment dates using individual date values, and the third design groups shipment dates by month.

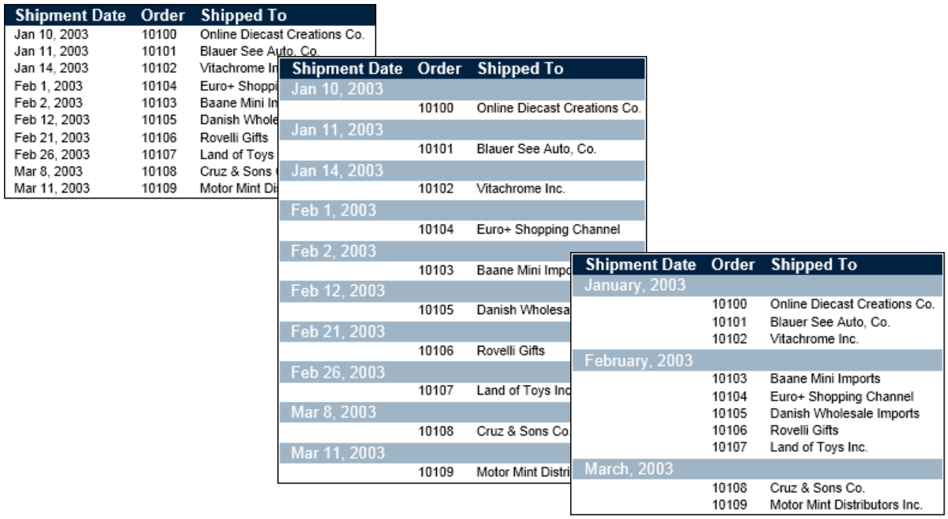


Figure 4-9 Comparing different date grouping options

How to group date-and-time data

1 Select the column that contains the date-and-time values by which to group.

2 Choose Add Group.

3 On Group Details, select one of the group options. If you opt to group by interval:

- Select an interval type, such as weeks, months, or quarters, from the drop-down list.
- Type a number by which to group the selected interval type. For example, if you selected Weeks, type 2 to group data in two-week periods.

Figure 4-10 shows grouping the SHIPPEDDATE column by month.

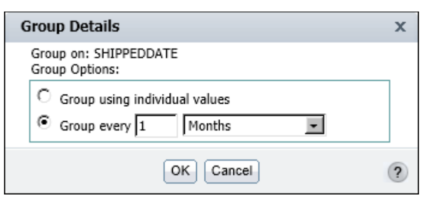


Figure 4-10 Grouping dates by month

Grouping on multiple columns

Just as you do when you sort on multiple columns, think about the order of precedence before you create groups. If you want to group customer data by state, then by city, create the groups in that order. The report design, as shown in Figure 4-7, groups sales data first by sales office, then by product line. You can identify the order of the data groups by the order of the columns in the table. The first group appears as the first column in the table, the second group appears next to it, and so on.

Changing the grouping order

After you create groups, you can change their order. Consider the effect of changing the grouping order. For example, changing the order of the state and city groups to city and state produces an illogically organized report. On the other hand, changing the order of the sales office and product line groups to product line and sales office provides a different perspective on the sales data. The focus shifts from the sales office performance to the product line performance.

How to change the grouping order

1 Select any column.

2 Right-click the selected column, then choose Column➛Reorder Columns. Re-order Columns, as shown in Figure 4-11, displays the grouped columns in the order in which the report design currently groups the data.

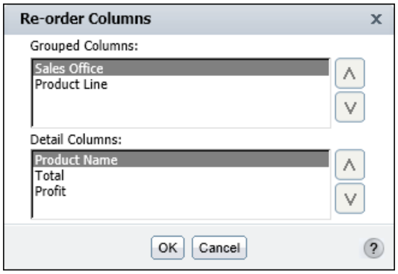


Figure 4-11 Changing the grouping order

3 Under Grouped Columns, select a column and use the up or down buttons to move the column up or down the list. Choose OK.

Deleting a group

To re-display the values in the detail row, delete a group. Select the column whose values you want to ungroup, then choose Delete Group, as shown in Figure 4-12.

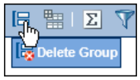


Figure 4-12 Choosing Delete Group

Deleting a group deletes all information associated with the group. For example, if the report design displays aggregate data at the group level, this data is removed.

Organizing data in sections

A section is functionally equivalent to a group. When you create a section, you are also grouping data. Like groups, you can create multiple sections, calculate aggregate data for each section, start each section on a new page, and hide the details of each section.

One of the obvious differences between a section and a group is how the information is arranged. Compare the report designs in Figure 4-13 and Figure 4-14. The report in Figure 4-13 organizes data in two groups: sales office and product line. The report in Figure 4-14 organizes the same data in two sections: sales office and product line.

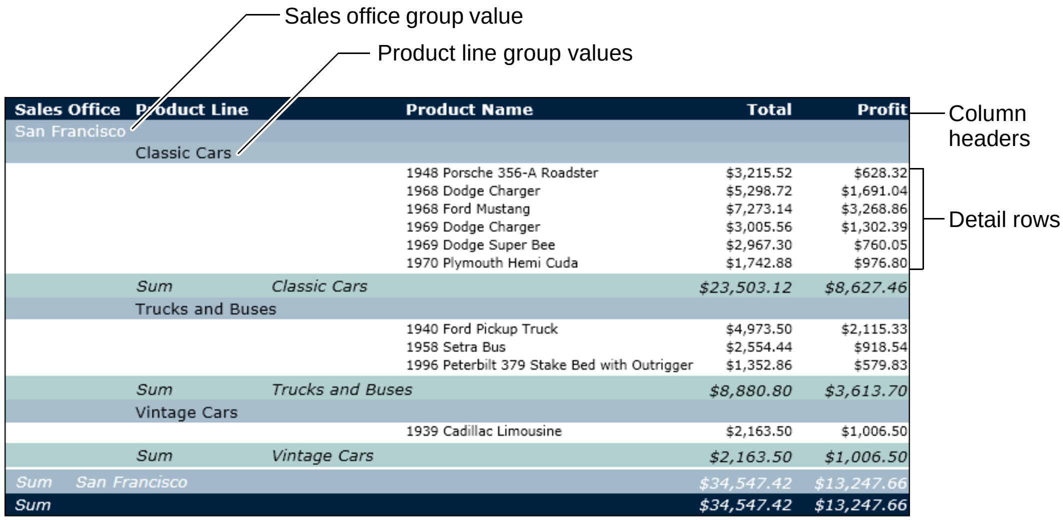
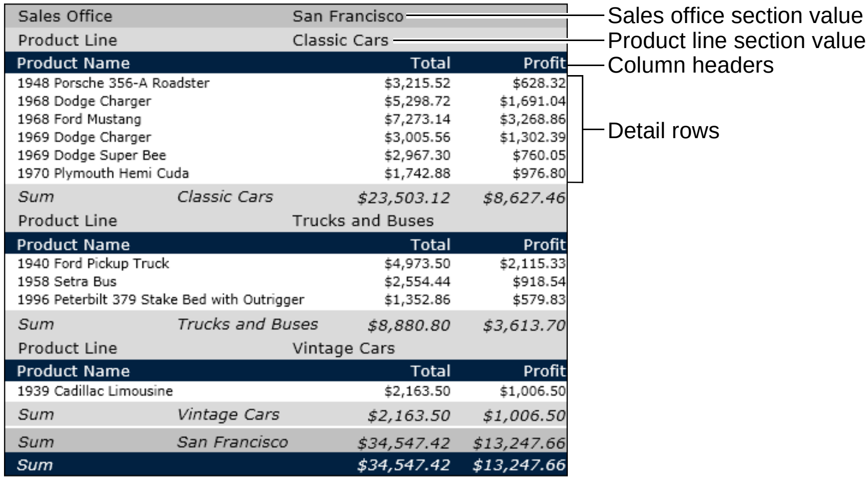


Figure 4-13 Using groups to organize data



Sales Office		San Francisco	
Product Line		Classic Cars	
Product Name		Total	Profit
1948 Porsche 356-A Roadster		$3,215.52	$628.32
1968 Dodge Charger		$5,298.72	$1,691.04
1968 Ford Mustang		$7,273.14	$3,268.86
1969 Dodge Charger		$3,005.56	$1,302.39
1969 Dodge Super Bee		$2,967.30	$760.05
1970 Plymouth Hemi Cuda		$1,742.88	$976.80
Sum	Classic Cars	$23,503.12	$8,627.46
Product Line		Trucks and Buses	
Product Name		Total	Profit
1940 Ford Pickup Truck		$4,973.50	$2,115.33
1958 Setra Bus		$2,554.44	$918.54
1996 Peterbilt 379 Stake Bed with Outrigger		$1,352.86	$579.83
Sum	Trucks and Buses	$8,880.80	$3,613.70
Product Line		Vintage Cars	
Product Name		Total	Profit
1939 Cadillac Limousine		$2,163.50	$1,006.50
Sum	Vintage Cars	$2,163.50	$1,006.50
Sum	San Francisco	$34,547.42	$13,247.66
Sum		$34,547.42	$13,247.66

Figure 4-14 Using sections to organize data

In Figure 4-13, the report design with groups displays the data from five fields in five columns. The sales office and product line values appear in the initial columns, below the column headers.

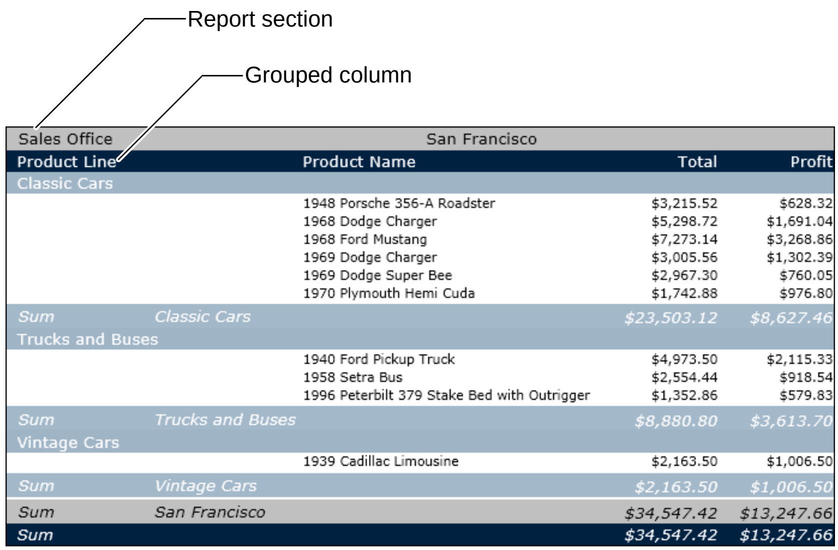
In Figure 4-14, the report design with sections, the sales office and product line values appear above the column headers.

As you can see from the examples, organizing data in sections reduces the overall width of the table. If your report design displays many columns, and space is tight, this difference can be reason enough to create sections instead of groups.

Aside from the differences in the way information is arranged, organizing data in sections provides the following benefits over organizing data in groups:

- You can insert multiple charts, one in each section, to reflect the aggregated data in each section. In a table that contains data groups, you can display only one chart, which presents the aggregated data for the overall table.
- You can display additional data fields in a section heading. By default, a section heading displays only a label and the section value, as shown in the report design in Figure 4-14.

You can use both groups and sections in a single report design. Figure 4-15 shows a report design that contains the same data as the previous examples, except the sales office column is a section, and the product line column is grouped.



Sales Office		San Francisco		
Product Line		Product Name	Total	Profit
Classic Cars				
		1948 Porsche 356-A Roadster	$3,215.52	$628.32
		1968 Dodge Charger	$5,298.72	$1,691.04
		1968 Ford Mustang	$7,273.14	$3,268.86
		1969 Dodge Charger	$3,005.56	$1,302.39
		1969 Dodge Super Bee	$2,967.30	$760.05
		1970 Plymouth Hemi Cuda	$1,742.88	$976.80
Sum	Classic Cars		$23,503.12	$8,627.46
Trucks and Buses				
		1940 Ford Pickup Truck	$4,973.50	$2,115.33
		1958 Setra Bus	$2,554.44	$918.54
		1996 Peterbilt 379 Stake Bed with Outrigger	$1,352.86	$579.83
Sum	Trucks and Buses		$8,880.80	$3,613.70
Vintage Cars				
		1939 Cadillac Limousine	$2,163.50	$1,006.50
Sum	Vintage Cars		$2,163.50	$1,006.50
Sum	San Francisco		$34,547.42	$13,247.66
Sum			$34,547.42	$13,247.66

Figure 4-15 Using a group and a section to organize data

How to create a section

1 Select the column that contains the data you want to organize in a section.

2 Choose Add Section, as shown in Figure 4-16.

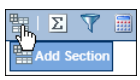


Figure 4-16 Adding a section

Just as with groups, if the column you selected contains string or numeric data, BIRT Studio creates a section for each unique value in the column. If the column you selected contains date-and-time data, you can create sections based on individual date values, or you can create sections that group dates by interval.

Displaying additional information in a section heading

As described previously, when you create a section, BIRT Studio creates for each section, a section heading that contains a label and the section value. The label displays the column name. Figure 4-17 shows an example of a section heading.

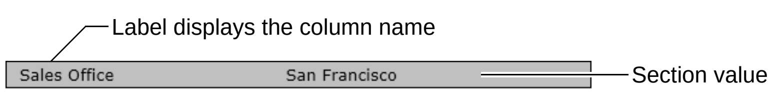


Figure 4-17 A section heading

You can add the contents of other data fields in a section heading. In the sales office section heading, for example, you can add the address of the sales office or the e-mail of the sales representative. Figure 4-18 shows an example.

Figure 4-18 A section heading with an additional field

The data fields that you can add to a section heading are the data fields that are currently in use in the table in the report design. If a data field is not in the table, but is included in the data set, you can add the data field to the table.

How to add content to a section heading

1 Select the section heading.

2 Right-click the section heading, then choose Section Heading. A dialog box displays a list of the data fields currently used in the report design.

3 Select the data field to add to the section heading, as shown in Figure 4-19. Choose OK. The data field appears in the section heading.

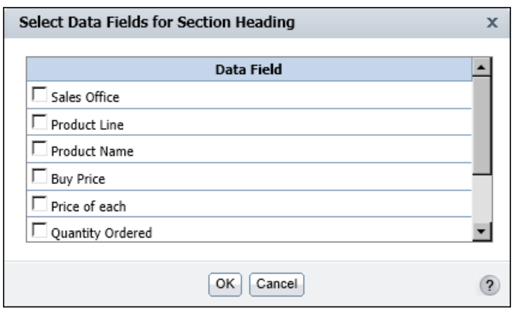


Figure 4-19 Select Data Fields for Section Heading displaying fields that you can add to a section heading

Deleting a section

You can delete a section to reorganize data. When you delete a section, the values in the section do not reappear in the detail rows. Instead, the values appear in a grouped column. When you delete a section in a report design, any aggregate data for the section is deleted. In addition, any chart you inserted in the section is also deleted.

To delete a section, select the section heading, then choose Delete Section, as shown in Figure 4-20.

Figure 4-21 shows the report design after the product line section is deleted from the design in Figure 4-20.

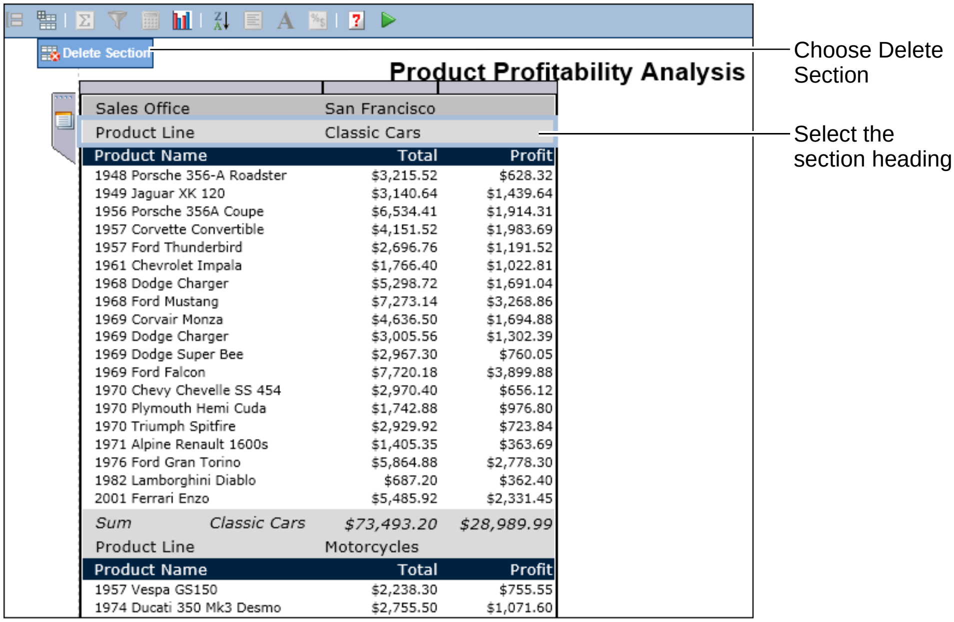


Figure 4-20 Deleting a section

Sales Office		San Francisco		
Product Line	**Product Name**		**Total**	**Profit**
Classic Cars				
	1948 Porsche 356-A Roadster		$3,215.52	$628.32
	1968 Dodge Charger		$5,298.72	$1,691.04
	1968 Ford Mustang		$7,273.14	$3,268.86
	1969 Dodge Charger		$3,005.56	$1,302.39
	1969 Dodge Super Bee		$2,967.30	$760.05
	1970 Plymouth Hemi Cuda		$1,742.88	$976.80
Trucks and Buses				
	1940 Ford Pickup Truck		$4,973.50	$2,115.33
	1958 Setra Bus		$2,554.44	$918.54
	1996 Peterbilt 379 Stake Bed with Outrigger		$1,352.86	$579.83
Vintage Cars				
	1939 Cadillac Limousine		$2,163.50	$1,006.50
	Sum	*San Francisco*	*$34,547.42*	*$13,247.66*
	Sum		*$34,547.42*	*$13,247.66*

Figure 4-21 A report design displaying data after a section is deleted

The product line section becomes a grouped column, displayed below the sales office section.

Aggregating data

One of the key features of any report is the ability to display summary, or aggregate, data. Many of the example report designs shown in this document contain aggregate data. The product profitability report design, for example, shows sales and profit subtotals for each product line by sales office, a sales and profit total for each sales office, and grand totals for the company.

Aggregating data involves performing a calculation on a set of values. For a simple listing report that does not organize data in groups or sections, aggregate calculations are performed on values in a selected column, over all the data rows in a table. The listing report design in Figure 4-22 displays aggregate data for the Total and the Profit columns at the end, or footer, of the table.

Product Name	Total	Profit
1992 Porsche Cayenne Turbo Silver	$2,557.14	$952.20
1970 Dodge Coronet	$1,706.40	$541.08
1952 Alpine Renault 1300	$10,286.40	$5,554.56
1961 Chevrolet Impala	$2,121.28	$1,086.72
1992 Porsche Cayenne Turbo Silver	$4,471.20	$1,331.10
1992 Porsche Cayenne Turbo Silver	$3,782.61	$1,061.19
1970 Dodge Coronet	$1,441.59	$502.86
1970 Triumph Spitfire	$4,096.03	$1,246.51
1998 Chrysler Plymouth Prowler	$3,389.28	$1,054.55
1968 Ford Mustang	$8,317.80	$4,027.50
	Max $10,286.40	*Max $5,554.56*
	Avr $4,216.97	*Avr $1,735.83*
	Sum $42,169.73	*Sum $17,358.27*

Aggregate data

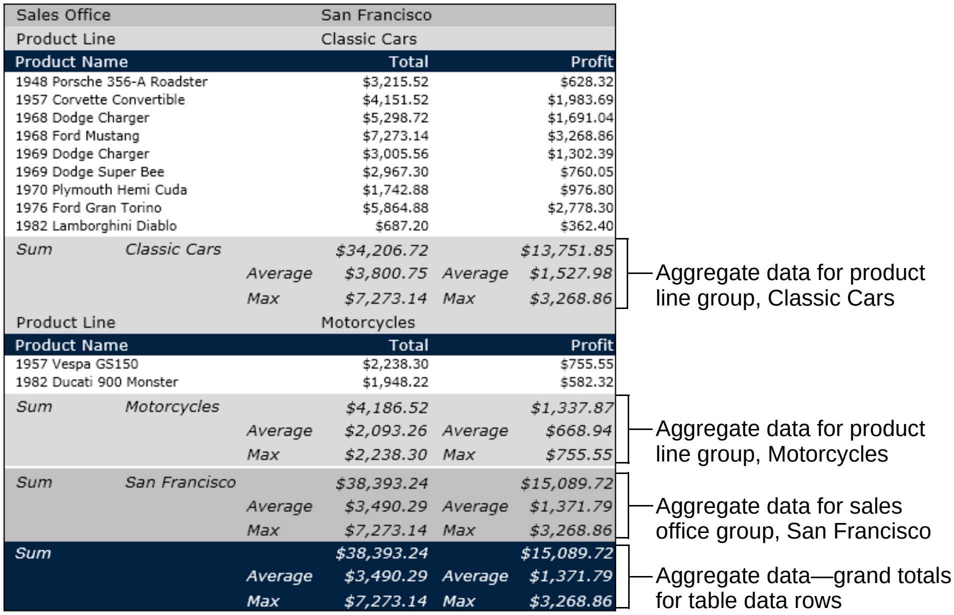
Figure 4-22 Displaying aggregate data in a simple listing report

For a report design that organizes data in groups or sections, as shown in Figure 4-23, you can display aggregates for a selected column or columns, for each group of data rows, and for all the data rows in the table. In this report design, the aggregate data appears in the footer of each group and in the footer of the table.

When you aggregate data in a selected column, you specify the following information:

- The type of aggregate calculation. For both the Total and Profit columns, the report design in Figure 4-23 displays the sum of the values, the highest (max) value, and the average value. For each column, you can display a maximum of three aggregations. Each aggregation appears on a separate row.
- Whether to perform the aggregate calculation over all the data rows in the table, or over the data rows in each group, or both. Aggregating data for groups applies only to report designs that organize data in groups or sections.
- Whether to display the aggregate data in the footer or header of the table or the groups. The previous examples show aggregate data in the footers, which is typical. Select header to display the aggregate data at the beginning of the table or group.

The aggregate calculations you can perform on a column depend on the column's data type. You can perform aggregate functions that involve a mathematical calculation, such as Sum and Average, only on numeric data.



Sales Office			San Francisco		
Product Line			Classic Cars		
Product Name			Total		Profit
1948 Porsche 356-A Roadster			$3,215.52		$628.32
1957 Corvette Convertible			$4,151.52		$1,983.69
1968 Dodge Charger			$5,298.72		$1,691.04
1968 Ford Mustang			$7,273.14		$3,268.86
1969 Dodge Charger			$3,005.56		$1,302.39
1969 Dodge Super Bee			$2,967.30		$760.05
1970 Plymouth Hemi Cuda			$1,742.88		$976.80
1976 Ford Gran Torino			$5,864.88		$2,778.30
1982 Lamborghini Diablo			$687.20		$362.40
Sum	*Classic Cars*		*$34,206.72*		*$13,751.85*
		Average	*$3,800.75*	*Average*	*$1,527.98*
		Max	*$7,273.14*	*Max*	*$3,268.86*
Product Line			Motorcycles		
Product Name			Total		Profit
1957 Vespa GS150			$2,238.30		$755.55
1982 Ducati 900 Monster			$1,948.22		$582.32
Sum	*Motorcycles*		*$4,186.52*		*$1,337.87*
		Average	*$2,093.26*	*Average*	*$668.94*
		Max	*$2,238.30*	*Max*	*$755.55*
Sum	*San Francisco*		*$38,393.24*		*$15,089.72*
		Average	*$3,490.29*	*Average*	*$1,371.79*
		Max	*$7,273.14*	*Max*	*$3,268.86*
Sum			*$38,393.24*		*$15,089.72*
		Average	*$3,490.29*	*Average*	*$1,371.79*
		Max	*$7,273.14*	*Max*	*$3,268.86*

Figure 4-23 Displaying aggregate data for groups and sections

The most common aggregate functions you perform on string data are Count and Count Value. Count returns the number of values in a column, including duplicate values. Count Value returns the number of distinct values in the column.You can use Count Value on a customer name column to get the number of customers. For date-and-time data columns, you can use the Min and Max functions to get the earliest and the latest date, respectively, from a column displaying order dates. For descriptions of the supported aggregate functions, see Chapter 10, "Functions and operators."

How to aggregate data

1 Select the column that contains the data to aggregate.

2 Choose Aggregation. Aggregation appears, as shown in Figure 4-24.

3 In Select Function, select the aggregate function to use.

4 In Aggregate on:

- Select table to aggregate data over all the data rows in the table. If you select table, select either header or footer as the location in which to display the aggregate data.
- Select group to aggregate data at the group level. If you select group, select either header or footer as the location in which to display the aggregate data.

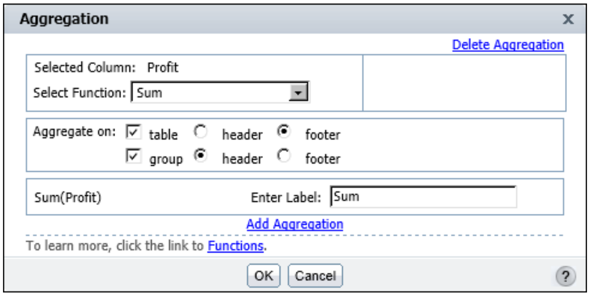


Figure 4-24 Aggregating data

5 In Enter Label, type the text to display next to the aggregate value. By default, BIRT Studio uses the name of the function you selected. You can, for example, replace Max with the following text:

```
Highest value
```

6 If you want to calculate and display a second aggregation, choose Add Aggregation. Follow steps 3 to 5 to define the next aggregation.

Choose OK. Figure 4-25 shows two aggregations in addition to the one, as shown in Figure 4-24, defined for the report design, as shown in Figure 4-23.

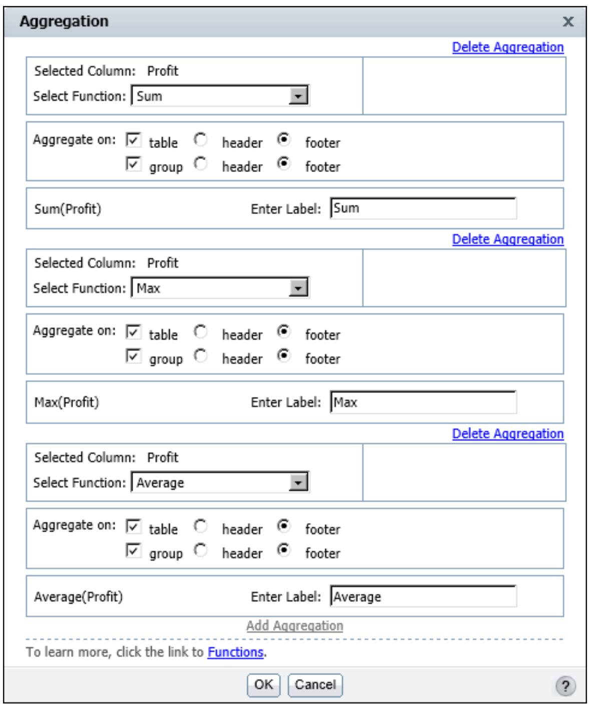


Figure 4-25 Defining three aggregations

Formatting aggregate values

Using BIRT Studio, you can format aggregate values in the following ways:

- Specify alignment properties.
- Modify font type, size, color, and style.
- Apply a format to the aggregate values.

To format an aggregate value, select the value, and choose a formatting option from the context menu.

Filtering aggregate values

You can use filters to display aggregate values according to certain conditions. Because BIRT Studio displays only a preview of up to 200 rows of the actual data, when you perform actions such as aggregating data, or filtering aggregate data values, the resulting report design displays different results in BIRT Studio and in Actuate BIRT Viewer. This disparity occurs because BIRT Studio applies the specified aggregation function to the previewed data, and applies the filter condition to this subset of data.

When you view a report created in BIRT Studio, in the BIRT Viewers, all data rows are retrieved and displayed in the report design. If you now create an aggregate calculation in Interactive Viewer, the aggregate function is applied across all rows of data in the report design. If you create a filter condition for the aggregate data, the filter is also applied across all rows of data, resulting in more accurate results. Always view the report design in Actuate BIRT Viewer to verify that the results are what you expect.

When you create a filter condition that excludes certain values, such as viewing transactions that closed over a month ago or listing products that are not the top 5 selling products, BIRT Studio recalculates the aggregate data according to the data BIRT Studio retrieves. In some cases, it is also useful to view aggregate values for the unfiltered data, for example, to perform percentage calculations on this data.

To address both situations in detail tables containing aggregate data, BIRT Studio provides an option on the Filter, Advanced Filter, and Top/Bottom N filter dialog boxes that enables a user to specify whether to recalculate aggregate data based on the filter condition created, or not.

To create a filter on aggregate values, complete the following steps:

1. In BIRT Studio, select the aggregate value, then choose Filter➛Filter, or Filter➛Filter Top/Bottom N from the context menu, as shown in Figure 4-26.
2. Define a filter condition.

BIRT Studio executes the filter condition, and displays the resulting data in the report design. Run the report in Actuate BIRT Viewer to verify that the results are what you expect.

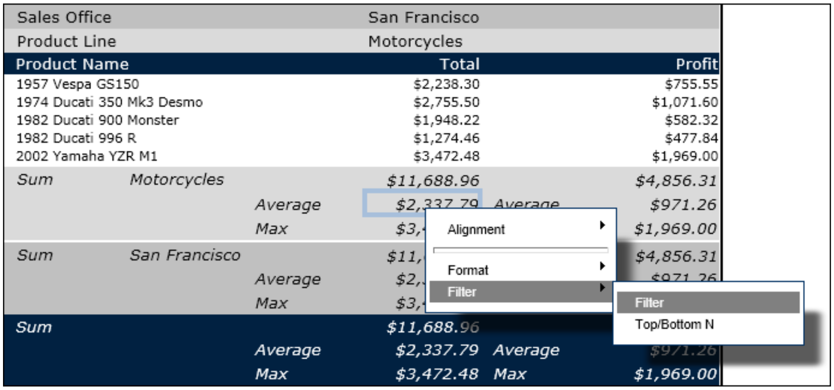


Figure 4-26 Selecting the filter option for an aggregate value

Hiding details

When you create a report that organizes data in groups or sections, you can change the appearance of the report to a summary report by hiding the details of a group or section. Hiding details, particularly for a report that runs into hundreds of pages, shows key information at a glance.

For reports containing large amounts of aggregate data, you can also choose to create a summary report. A summary report aggregates data at the data source level, providing a concise view of report data, while improving response time, and also reducing the load on the server. For more information, see Chapter 6, “Working with summary tables.”

Figure 4-27 shows two report designs that contain the same data. The first design shows all details. The next design hides the details within each product line group.

How to hide the details of a group

Right-click the grouped column for which to hide details, then choose Group➛Hide Detail, as shown in Figure 4-28.

In the example shown in Figure 4-27, in the report on the right, because you hid details at the group level, all the detail rows for every product line group are hidden.

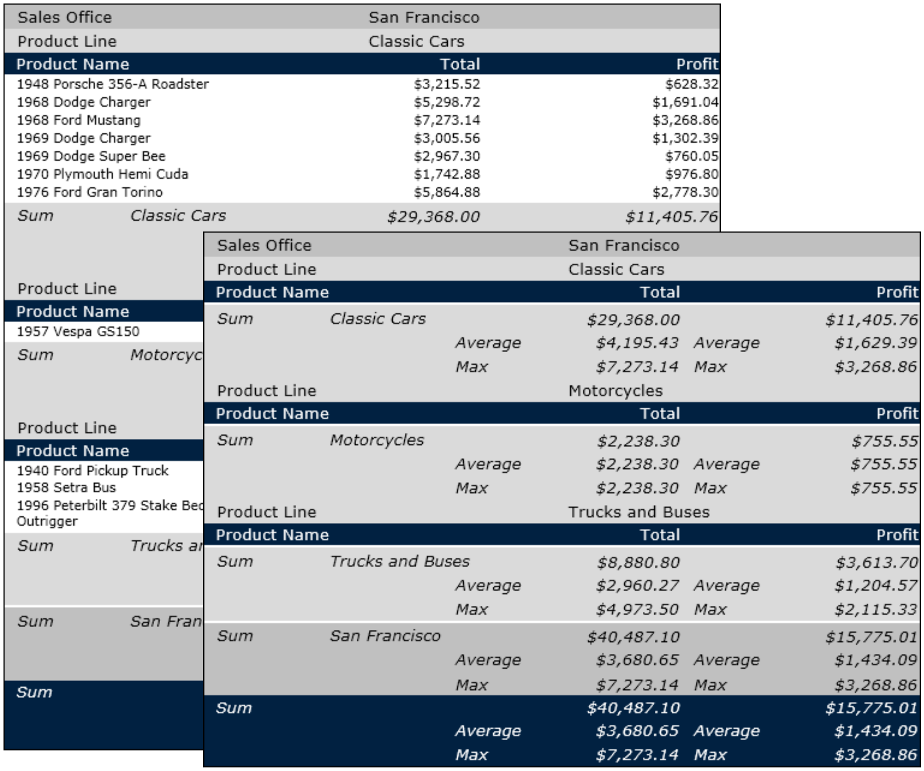


Figure 4-27 Showing and hiding detail rows

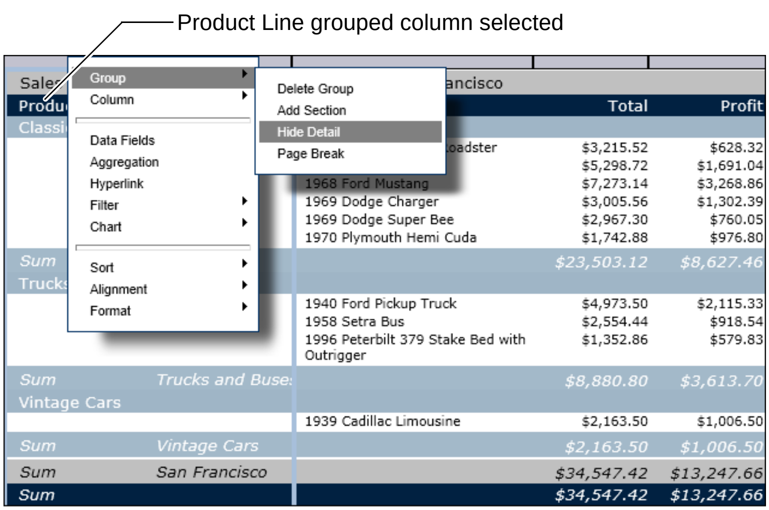


Figure 4-28 Hiding the details of a selected grouped column

How to hide the details of a section

1. Select the section heading of the section whose details to hide.
2. Right-click the section heading, then choose Section➛Hide Detail, as shown in Figure 4-29.

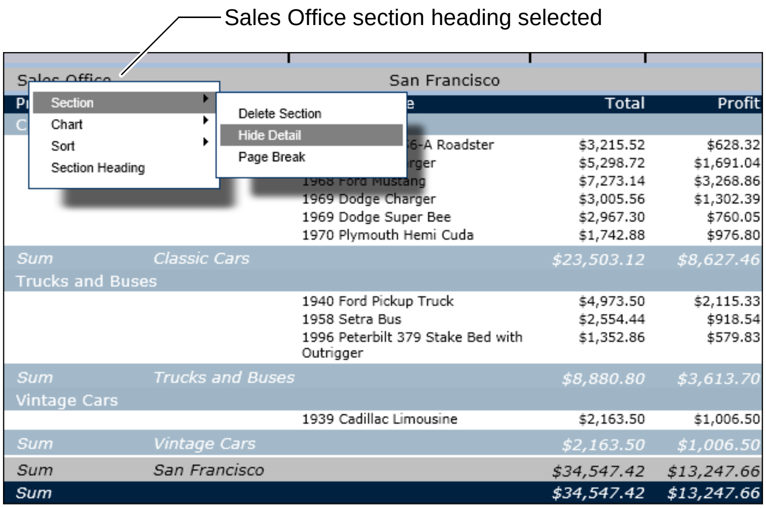


Figure 4-29 Hiding the details of a selected section

How to re-display the details of a group or section

Select the grouped column or section heading, right-click the object, then choose Group➛Show Detail or Section➛Show Detail.

Suppressing duplicate values

Frequently, you find that you do not need to display all the data present in a report. For example, sometimes a report column displays duplicate values in consecutive data rows. When the duplication is unnecessary or makes the report difficult to read, you can suppress consecutive duplicate values.

You can use BIRT Studio to suppress duplicate values that appear in a grouped column. For example, a report that lists customer orders, grouped by customer, shows some data values such as the address, contact information, customer number, and so on, multiple times for every order placed by the customer. You can prevent the duplicate occurrence of these values by moving data rows from the column to the group header.

Avoiding repeated values in a column

In the report shown in Figure 4-30, the Location column shows the city name each time the name occurs.

customerName	city	state	salesRepEmployeeNumber
Diecast Classics Inc.	Allentown	PA	1216
Diecast Collectables	Boston	MA	1188
Gifts4AllAges.com	Boston	MA	1216
Collectables For Less Inc.	Brickhaven	MA	1188
Online Mini Collectables	Brickhaven	MA	1188
Auto-Moto Classics Inc.	Brickhaven	MA	1216
Gift Depot Inc.	Bridgewater	CT	1323
Signal Collectibles Ltd.	Brisbane	CA	1165
West Coast Collectables Co.	Burbank	CA	1166
Technics Stores Inc.	Burlingame	CA	1165
Marta's Replicas Co.	Cambridge	MA	1216

Figure 4-30 Viewing duplicate values in columns

As shows Figure 4-31, the report appears neater and more organized when duplicate values are not repeated.

customerName	city	state	salesRepEmployeeNumber
Diecast Classics Inc.	Allentown	PA	1216
Diecast Collectables	Boston	MA	1188
Gifts4AllAges.com			1216
Collectables For Less Inc.	Brickhaven		1188
Online Mini Collectables			
Auto-Moto Classics Inc.			1216
Gift Depot Inc.	Bridgewater	CT	1323
Signal Collectibles Ltd.	Brisbane	CA	1165
West Coast Collectables Co.	Burbank		1166
Technics Stores Inc.	Burlingame		1165
Marta's Replicas Co.	Cambridge	MA	1216

Figure 4-31 Viewing a report with suppressed duplicate values

Using BIRT Studio, you can suppress consecutive occurrences of duplicate values. In the Location column in Figure 4-31, the Boston value is suppressed in the second, third, fourth, and fifth rows. If Boston occurs again after the listing for NYC, that occurrence of Boston is visible and subsequent consecutive occurrences are suppressed. The values must be exact duplicates to be suppressed.

If a column extends across multiple pages, the first row on each page displays a value, even if duplicate values are suppressed for that column.

How to suppress duplicate values in a column

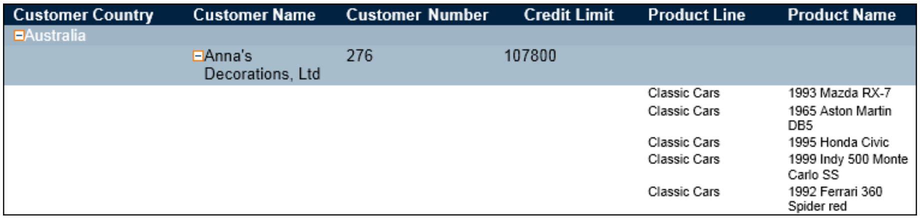
Select the column containing duplicate values. From the context menu, choose Column➛Do Not Repeat Values.

How to show hidden duplicate values in a column

Select a column that does not repeat duplicate values. From the context menu, choose Column➛Repeat Values.

Avoiding repeated values in a group

The example report in Figure 4-32 displays the values for E-mail, and Credit Limit multiple times for the same customer. By moving the data row values to the group header, you limit the appearance of these fields of data so that they only appear once.



Customer Country	Customer Name	Customer Number	Credit Limit	Product Line	Product Name
Australia					
	Anna's Decorations, Ltd	276	107800		
				Classic Cars	1993 Mazda RX-7
				Classic Cars	1965 Aston Martin DB5
				Classic Cars	1995 Honda Civic
				Classic Cars	1999 Indy 500 Monte Carlo SS
				Classic Cars	1992 Ferrari 360 Spider red

Figure 4-32 Viewing duplicate values in a group

How to suppress duplicate values in a grouped column

1 Select the column with the duplicate values. From the context menu, choose Column➛Move to Group Header.

2 On Move to Group Header, as shown in Figure 4-33, select the group to which you want to move the data values, if the report contains more than one group.

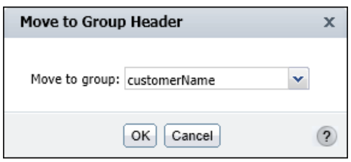
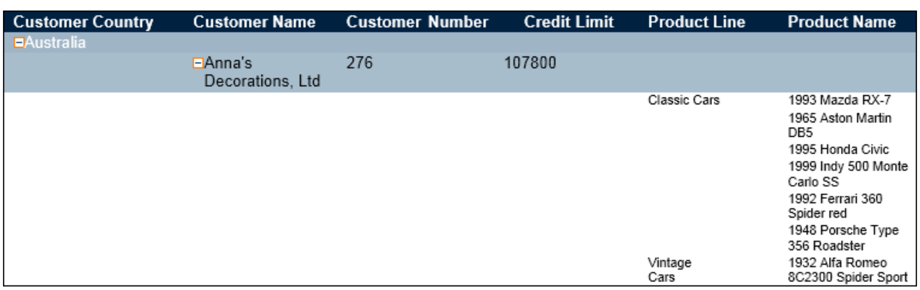


Figure 4-33 Selecting a group header

Choose OK.

Repeat steps 1 and 2 for each report column for which you want to move data rows to the group header. The value from the first data row in each group appears in each group header. The Credit Limit and E-mail columns display a single data row for each Customer Number group header, as shown in Figure 4-34.



Customer Country	Customer Name	Customer Number	Credit Limit	Product Line	Product Name
Australia					
	Anna's Decorations, Ltd	276	107800		
				Classic Cars	1993 Mazda RX-7
					1965 Aston Martin DB5
					1995 Honda Civic
					1999 Indy 500 Monte Carlo SS
					1992 Ferrari 360 Spider red
					1948 Porsche Type 356 Roadster
				Vintage Cars	1932 Alfa Romeo 8C2300 Spider Sport

Figure 4-34 Viewing a report that suppresses duplicate values in a group

Starting each group or section on a new page

Lengthy reports or reports that consist of distinct sections typically look more organized if each section appears on a separate page. The product profitability report design example appears compact, because it is previewed with a small number of rows. For example, the report design in Figure 4-29 is set to display only 20 rows of data. When the report is run and appears in the viewer, it contains 60 pages of data.

As you design a report, make sure you periodically view the generated report to see if you need to set page breaks to make the report more usable. You can set a page break for each group and section you create. In the report design in Figure 4-29, you can start each sales office section and each product line group within the sales office section on a new page.

How to set a page break

1 Select a grouped column or section heading. Right-click the object, then choose Group➛Page Break or Section➛Page Break. Page Break appears, as shown in Figure 4-35.

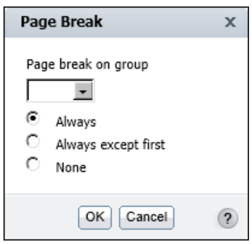


Figure 4-35 Setting a page break

2 On Page Break, select After or Before from the drop-down list to insert a page break after or before a group or section.

- If you select After, choose one of the following options:
  - Choose Always to always insert a page break after each group or section.
  - Choose Always except last to always insert a page break after each group or section, but not after the last one. This is the typical option to avoid a blank page at the end of the report.
- If you select Before, choose one of the following options:
  - Choose Always to always insert a page break before each group or section.

- Choose Always except first to always insert a page break before each group or section, but not before the first one. This is the typical option to avoid a blank page at the beginning of the report.

- Choose None to remove an existing page break.

3 Choose OK. Run the report to view the modified design.

Chapter

5

Filtering data

This chapter contains the following topics:

- About data filtering
- About filtering options
- Creating a filter
- Prompting for filter values at run time

About data filtering

A data set often provides more information than your report needs. You can select specific data rows to use in a report by using a filter. For example, rather than list all customer sales, you can create a filter to select only the sales data for a particular week or only the sales data for a particular region.

Filtering data helps you work effectively with large amounts of data. It enables you to find the necessary pieces of information to answer specific business questions, such as which sales representatives generated the top ten sales accounts, which products generated the highest profit in the last quarter, which customers have not made a purchase in the past 90 days, and so on.

Filtering data can also have a positive effect on processing speed. Limiting the number of data rows used in the report can reduce measurably the load on the server, because it does not need to return all the rows every time the report is run.

BIRT Studio supports the use of static filters and dynamic filters. You can use a static filter to define a specific filter condition during report design. The data displayed in the report depends on the filter condition specified in the report design. You use a dynamic filter to prompt a user to specify values for which BIRT Studio displays data, when they run the report. The data that appears in the report depends on the values the user specifies when running the report.

When you use filters in a report design, BIRT Studio typically evaluates the filter condition against the rows of data previewed in the report design, 50 rows by default. Increase this number to 200 rows, when creating filters in a report design. If you use the default value of 50 data rows to test the filter condition, BIRT Studio might frequently not be able to return any data that matches the filter you define. Also run the report in BIRT Viewer to view the actual data, and verify that the result is what you expect.

About filtering options

You can filter data at the data set level and at the table level. Filtering at the data set level narrows the scope of data available to a report design and can improve design-time performance if the data set returns a particularly large amount of data. Filtering at the table level narrows the scope of data displayed in a table and is the typical filtering option.

If you filter data at both the data set and table levels, BIRT Studio executes the filter at the data set level first, then at the table level. Design the filters accordingly. For example, if a data set returns all customer sales for ten years, you can create a filter at the data set level to limit the sales data to the current year. Then, when you design your report, you can create a filter at the table level to display sales data for a particular quarter in the current year. You cannot display sales data for the previous year, because the filter at the data set level excludes this data.

Creating a filter

When you create a filter, you define a condition that specifies which data rows to include in the report. A filter condition is an If expression that must evaluate to true in order to include a data row. For example:

```
If the order total is greater than 10000
If the sales office is San Francisco
If the order date is between 4/1/2008 and 6/30/2008
```

Figure 5-1 shows an example of a condition defined in Filter. As the illustration shows, Filter helps you define the condition by breaking it down to the following parts:

- The column to evaluate, such as Total.
- The comparison operator that specifies the type of comparison test, such as Greater Than. For information about all the operators, see Chapter 10, “Functions and operators.”
- The value to which all values in the column are compared, such as 10000.

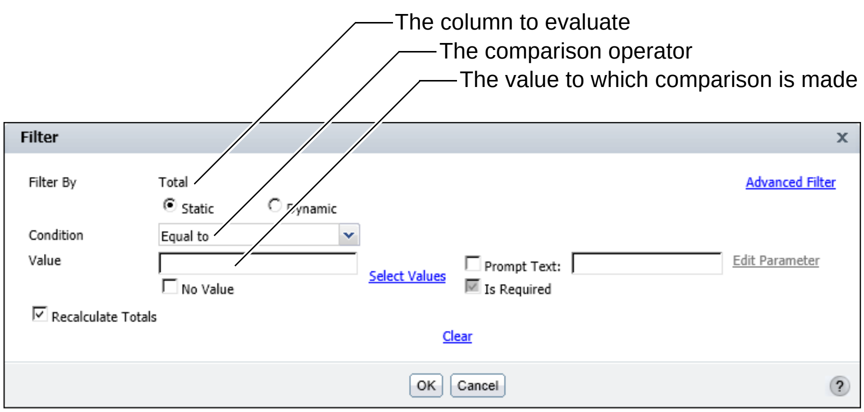


Figure 5-1 Filter displaying a filter condition

Creating a filter at the report table level

This section describes the procedure to create a filter at the table level. You can create a simple filter as described in the previous section. In addition, you can also choose multiple values for a filter condition by using the In operator, choose to exclude specific data, create a filter on empty values in the data fields, display the top or bottom N values, or specify a date as the comparison value. You can create a filter condition that compares values in a column containing string data to a string pattern instead of to a specific value. You can also create a filter condition that compares values in one column to values in another column.

When BIRT Studio evaluates a filter condition, it displays data that meets the specified condition, but users can specify whether or not the aggregate data in the report is recalculated to meet the filter condition. This feature is useful when comparing the filtered data values with the unfiltered totals, for example, when performing a percentage calculation of the unfiltered aggregate totals. You can specify whether to recalculate totals when creating a filter condition at the report table level.

How to create a filter at the table level

1 Select the column in the report design, containing the value that determines when the filter takes effect. For example, to create a filter that retrieves data rows where the sales office is Boston, select the sales office column.

2 Choose Filter. Filter appears, as shown in Figure 5-2.

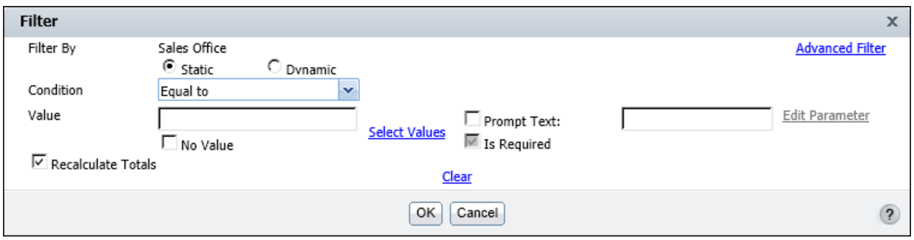


Figure 5-2 Filter displaying the name of a selected column

Filter By displays the column on which to apply the filter.

3 Select Static, as the type of filter, if necessary.

4 In Condition, select the comparison test, or operator, to apply to the selected column. Depending on the operator you select, Filter displays one or two additional fields, or a completed filter condition.

5 If you select an operator that requires a comparison value, you can specify the value in one of the following ways:

- In Value, type the value.
- To select from a list of values, choose Select Values. A list of values appears. Select a value from the list. Figure 5-3 shows the selection of Boston from the list of available sales office values.
- If the list of values is long, type a value in Filter Text and choose Find. If found, BIRT Studio highlights this value in the list. When you select the value, it appears in Value.
- To specify a null value, select No value.

- If you are filtering a detail table containing aggregate data, or a summary table, to recalculate the aggregate data values for the retrieved data rows, accept the default selection of Recalculate Totals. To calculate totals for all the data rows in the report table deselect Recalculate Totals.

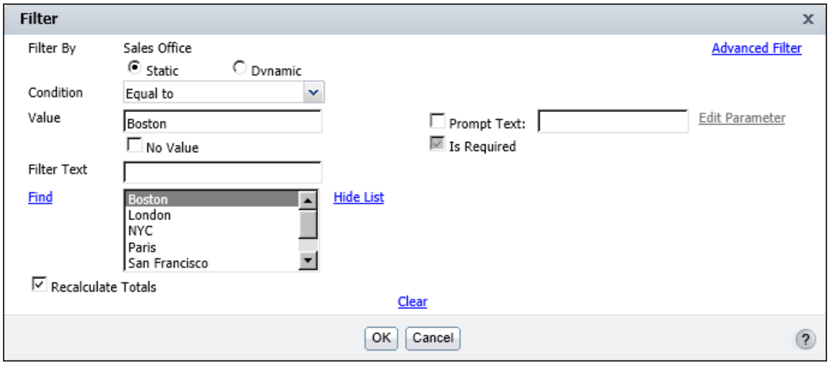


Figure 5-3 Specifying values

6 Choose OK. The filter takes effect.

In cases where the sample data in the report design does not meet the filter condition, the report design does not display any data. Always test a filter by running and viewing the generated report in Actuate BIRT Viewer.

BIRT Studio displays the message, as shown in Figure 5-4.

Figure 5-4 The message displayed when the sample data in the report design does not meet the filter condition

Selecting multiple values for a filter condition

The preceding filter examples specify one comparison value. Sometimes you need to view more data, for example, sales details for several sales offices, not for only one office. o select more than one comparison value, select the In operator, choose Select Values, then select each value. Figure 5-5 shows the selection of London and Paris from a list of sales office values.

To select multiple values, press Ctrl as you select each value. To select contiguous values, select the first value, press Shift, and select the last value. This action selects the first and last values and all values between them.

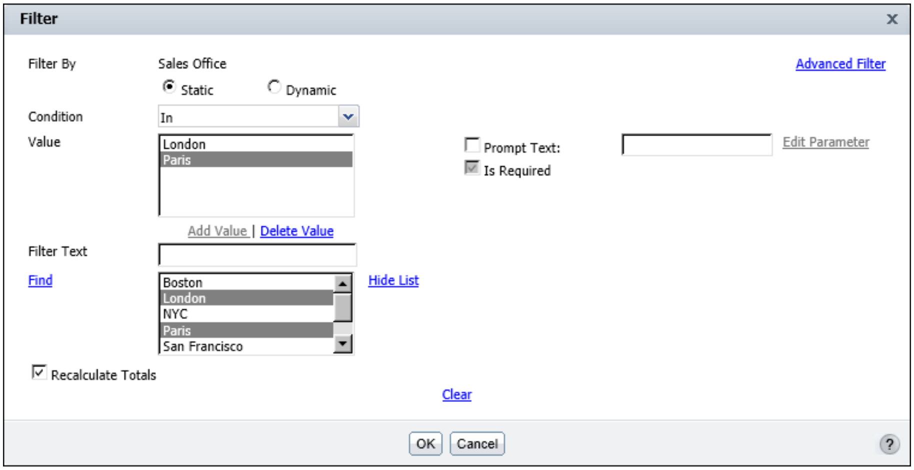


Figure 5-5 Selecting multiple comparison values

Excluding data

You use comparison operators, such as Equal to, Greater Than, or Less Than, to evaluate the filter condition to determine which data to include. Sometimes it is more efficient to specify a condition that excludes a small set of data. For example, you need sales data for all countries except the USA. Instead of selecting all the available countries and listing them in the filter condition, simply use the Not Equal To operator. Similarly, use Not Between to exclude data in a specific range, and Not Like to exclude data that matches a string pattern.

For example, the following filter condition excludes orders with amounts between 1000 and 5000:

```
OrderAmount Not Between 1000 and 5000
```

The filter condition in the next example excludes products with codes that start with MS:

```
ProductCode Not Like MS%
```

Filtering empty or blank values

Sometimes, rows display nothing for a particular column. For example, suppose a customer database table contains an e-mail field. Some customers, however, do not supply an e-mail address. In this case, the e-mail field might contain an empty value or a blank value. An empty value, also called a null value, means no value is supplied. Null values apply to all data types. You can create a filter to exclude data rows where a particular column has null values.

When filtering to exclude null values, use the Is Not Null operator. To view only rows that have null values in a particular column, use Is Null. For example, the following filter condition excludes customer data where the e-mail column contains null values:

```
e-mail Is Not Null
```

The following filter condition displays only rows where the e-mail column contains null values:

```
e-mail Is Null
```

Displaying top or bottom values

For a report that presents a large amount of numeric data, you find it useful to view, for example, the top 50 order totals or the counties whose median home prices are in the bottom 10 percent. To display top or bottom values, choose Filter➛Filter Top/Bottom N, then create a condition using the Top N, Top Percent, Bottom N, or Bottom Percent operators.

For example, the following filter condition displays the top 100 orders:

```
OrderAmount Top N 50
```

The filter condition in the next example displays the median prices in the bottom 10 percent. If there are 1000 rows of data, 100 rows are displayed.

```
MedianPrice Bottom Percent 10
```

These operators cannot be used to create a filter at the data set level. You can display top or bottom values for the report table level, or at the group level in the report design. For example, in a report design that lists profit and revenue for each product line and sales office, you can define a filter condition that displays profit values in the bottom 10 percent for each sales office city.

In Top/Bottom N, in Filter, select an operator, then specify a value in the next field, as shown in Figure 5-6. If you are filtering a detail table containing aggregate data, or a summary table, to recalculate the aggregate data values for the retrieved data rows, accept the default selection of Recalculate Totals. To display aggregate data values for the unfiltered data in the report table, deselect Recalculate Totals.

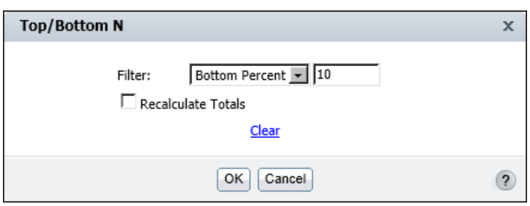


Figure 5-6 Specifying a bottom percent filter condition

Choose OK. The report design displays the data rows that match the filter condition you defined, as shown in Figure 5-7.

To modify the filter condition you just created, because the filter is created at the group level, use the Top/Bottom N dialog box to change the condition. To modify a top or bottom filter created at the report table level, you can either use the Top/Bottom N dialog box or the Advanced Filter dialog box. For information about using the Advanced Filter dialog box to perform this task, see “Changing a condition,” later in this chapter.

Product Line	Sales Office	Product Name	Total	Profit
Classic Cars				
	San Francisco			
		1971 Alpine Renault 1600s	$1,405.35	$363.69
		1982 Lamborghini Diablo	$687.20	$362.40
	Sum San Francisco		*$2,092.55*	*$726.09*
Sum Classic Cars			*$2,092.55*	*$726.09*
Motorcycles				
	San Francisco			
		1982 Ducati 996 R	$1,274.46	$477.84
	Sum San Francisco		*$1,274.46*	*$477.84*
Sum Motorcycles			*$1,274.46*	*$477.84*
Ships				
	San Francisco			
		Pont Yacht	$1,831.44	$532.74
	Sum San Francisco		*$1,831.44*	*$532.74*
Sum Ships			*$1,831.44*	*$532.74*
Vintage Cars				
	San Francisco			
		1939 Chevrolet Deluxe Coupe	$1,242.54	$362.31
	Sum San Francisco		*$1,242.54*	*$362.31*
Sum Vintage Cars			*$1,242.54*	*$362.31*
Sum			*$6,440.99*	*$2,098.98*

Figure 5-7 Displaying profits in the bottom 10 percent for each city

Specifying a date as a comparison value

When you create a filter condition that compares the date-and-time values in a column to a specific date, the date value you supply must be in one of the following formats, regardless of your locale:

```
3/26/2008
3/26/2008 2:30:00 PM
```

For your convenience, BIRT Studio provides a calendar you can use to select a date. Figure 5-8 shows the calendar and how to access it.

Choose the calendar icon to open the calendar, as shown in Figure 5-8. The calendar appears displaying the current date, month, and year. The current date is highlighted by default.

- To accept the current date, choose Today.
- To choose another date, select it.

The selected value appears in Value, in Filter. Choose OK for the filter condition to take effect.

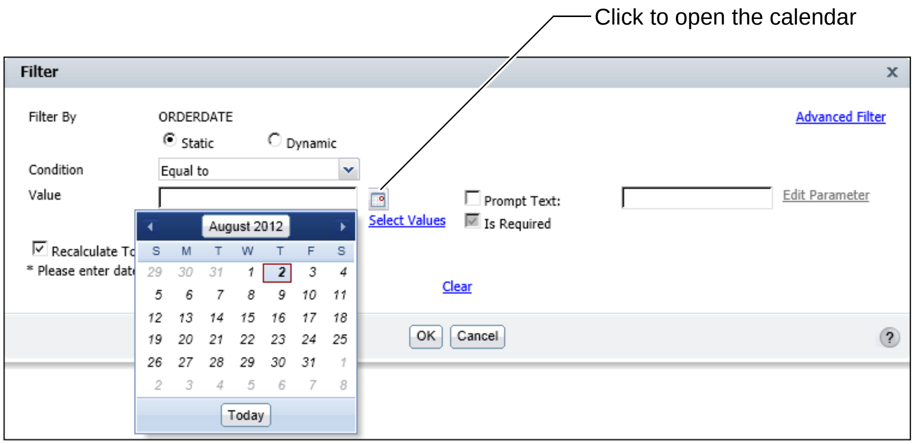


Figure 5-8 Selecting a date using the calendar

Comparing to a string pattern

For a column that contains string data, you can create a filter condition that compares each value to a string pattern instead of to a specific value. For example, to display only customers whose names start with M, use the Like operator and specify the string pattern, M%, as shown in the following filter condition:

```
Customer Like M%
```

You can use the following special characters in a string pattern:

- % matches zero or more characters. For example, %ace% matches any value that contains the string ace, such as Ace Corporation, Facebook, Kennedy Space Center, and MySpace.
- _ matches exactly one character. For example, t_n matches tan, ten, tin, and ton. It does not match teen or tn.

To match the percent sign (%) or the underscore character (_) in a string, precede those characters with two backslash characters (\\). For example, to match S_10, use the following string pattern:

```
S\\_10
```

To match 50%, use the following string pattern:

```
50\\%
```

Comparing to a value in another column

Use a filter condition to compare the values in one column with the values of another column. For example, in a report that displays products, sale prices (Price), and MSRP (Manufacturer Suggested Retail Price), you can create a filter condition to compare the Price and MSRP of each product and display only rows where the Price is greater than MSRP.

How to compare to a value in another column

1 Select the column that contains the values to compare, then choose Filter.

2 On Filter, choose Advanced Filter. Advanced Filter appears, as shown in Figure 5-9.

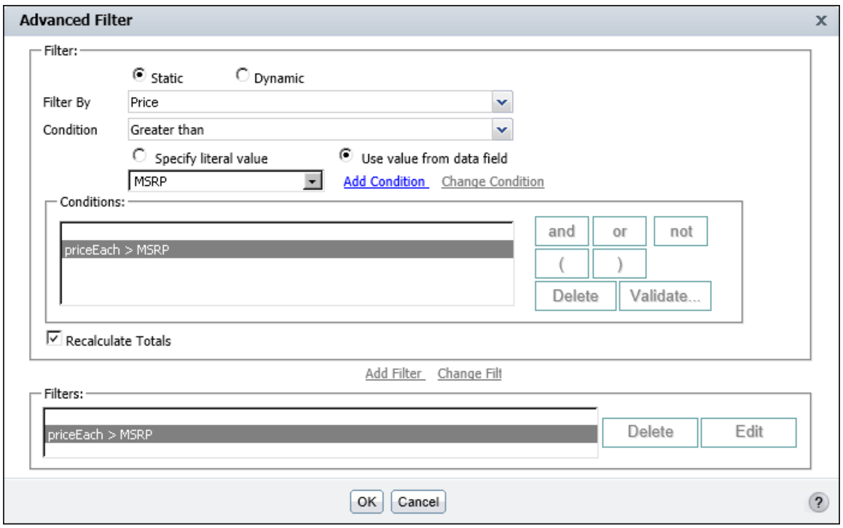


Figure 5-9 Building a comparison filter using Advanced Filter

3 In Condition, select a comparison operator.

4 Choose Use value from data field.

5 From the drop-down list that appears, choose the column that contains the values to compare to the first column.

Figure 5-9 shows an example of a filter condition that compares the values in the Price column with the values in the MSRP column.

6 Choose Add Condition.

7 Choose Add Filter, then choose OK.

Defining multiple filter conditions

When you create a filter, you define at least one filter condition, but there is no limit on the number of filter conditions that you can add. Each condition you add narrows the scope of data further. For example, you can create a filter that returns rows where the customer's credit rank is either A or B and whose open orders total between $250,000 and $500,000. Each condition you add also adds complexity to the filter. Design and test filters with multiple conditions carefully.

To define multiple filter conditions, you can use the Filter dialog box to select a column and define a filter, then select the next column and define a filter, and so on, or you can use Advanced Filter.

Adding a condition

Figure 5-10 shows two filters, the first on the SALES OFFICE column and the second on the PRODUCTLINE column. When you use this method, BIRT Studio constructs the following filter expression:

```
SALES OFFICE = San Francisco
and PRODUCTLINE = Vintage Cars
```

This filter returns only data rows that meet both conditions. To create a filter that returns data rows when either condition is true, or to create a more complex filter, use Advanced Filter, as shown in Figure 5-11.

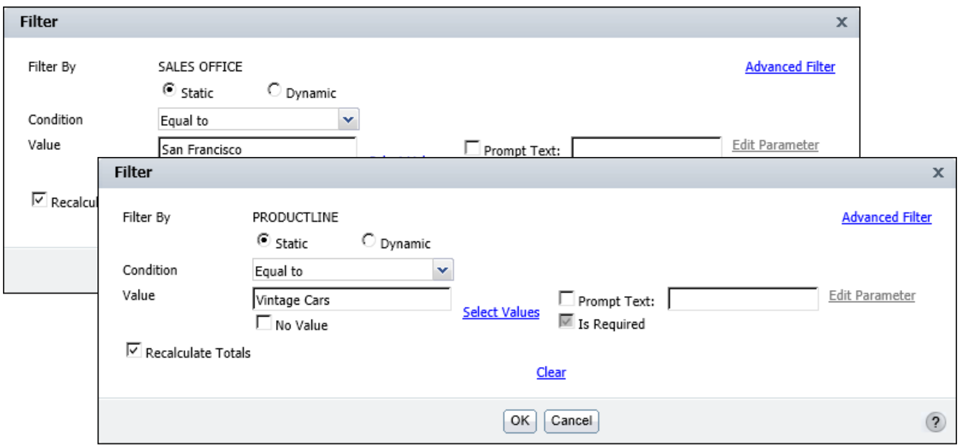


Figure 5-10 Creating multiple filter conditions

As Figure 5-11 shows, this dialog box provides more options for defining a filter with multiple conditions. It also shows all the filter conditions for the table.

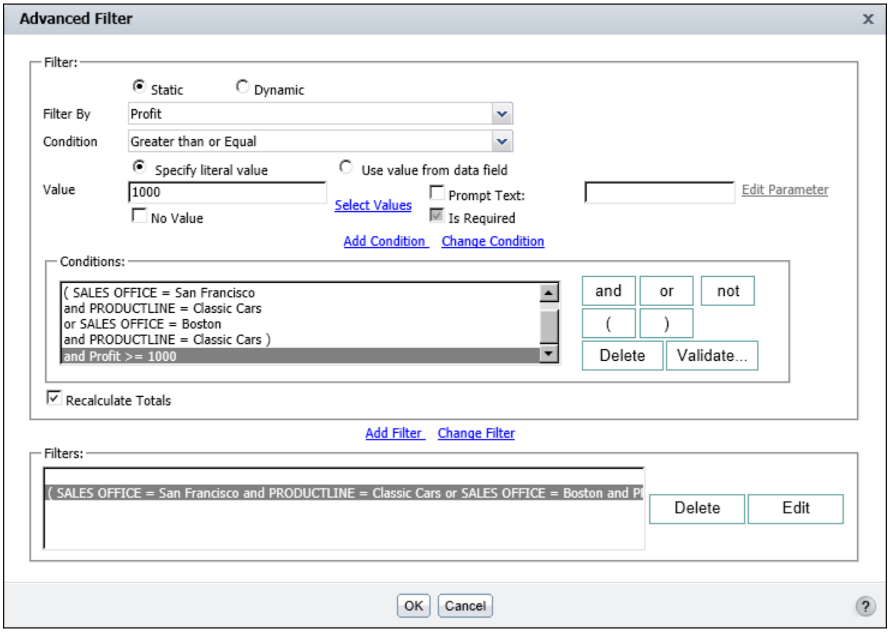


Figure 5-11 Displaying multiple conditions in a complex filter

How to define multiple filter conditions

1 In BIRT Studio, select the column that contains the values to filter by, then choose Filter.

2 On Filter, choose Advanced Filter.

3 On Advanced Filter, select Static, if necessary.

Advanced Filter displays the selected column in Filter By. You can change this column by selecting a different column from the drop-down list.

4 Define a filter condition as follows:

1. In Condition, select a comparison operator.
2. Specify a value.
    - To type or choose a specific value, choose Specify literal value.
        - In Value, type the value, then proceed to step 3.
        - To select from a list of values, choose Select Values. A list of values appears. Select a value from the list, then proceed to step 3.
        - If the list of values is long, type a value in Filter Text, then choose Find. If found, BIRT Studio highlights this value in the list. Proceed to step 3.

- To compare values in the selected column, to values in another column, choose Use value from data field. Select a column from the drop-down list.
- To specify a null value, select No value.

3 Choose Add Condition.

The filter condition appears in the Conditions area.

5 Define the next filter condition on Advanced Filter as follows:

1 In Filter By, select another column.

2 In Condition, select a comparison operator.

3 Specify a value.

- To type or choose a specific value, choose Specify literal value.
  - In Value, type the value. Proceed to step 4.
  - To select from a list of values, choose Select Values. A list of values appears. Select a value from the list. Proceed to step 4.
  - If the list of values is long, type a value in Filter Text, then choose Find. If found, BIRT Studio highlights this value in the list. Proceed to step 4.
- If you choose Use value from data field, select a column from the drop-down list.
- To specify a null value, select No value.

4 Choose Add Condition.

In the Conditions area, the second filter condition appears after the first condition, as shown in Figure 5-12. By default, the second condition is preceded by the logical operator, And.

6 Select a different logical operator, if necessary.

7 If the report contains aggregate data, do one of the following:

- To recalculate aggregate data values across the filtered data rows, select Recalculate Totals if necessary.
- To calculate aggregate data for the unfiltered data, deselect Recalculate Totals.

8 To add additional filter conditions, repeat step 4.

9 If you create more than two filter conditions and you use different logical operators, you can use the parentheses buttons to group conditions to determine the order in which conditions are evaluated.

10 Choose Validate to verify the syntax of the filter conditions, then choose Add Filter.

The defined filter conditions appear in the Filters area, as shown in Figure 5-12. Choose OK.

11 Run the report to verify that it displays the results you expect.

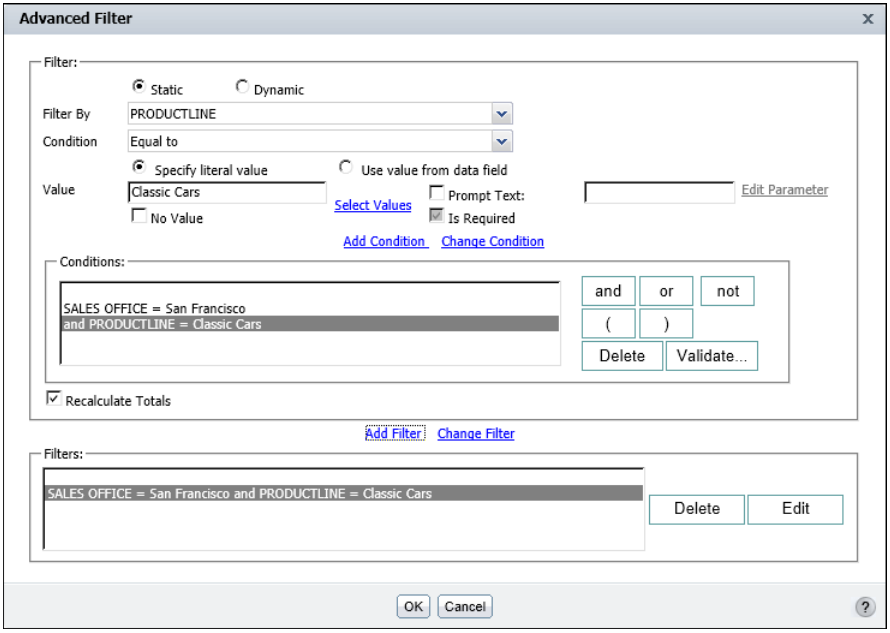


Figure 5-12 Advanced Filter displaying two conditions

Selecting a logical operator

As you add each filter condition, the logical operator And is inserted between each filter condition. You can change the operator to Or. The And operator means both filter conditions must be true to include a data row in the report. The Or operator means only one condition has to be true to include a data row. You also can add the Not operator to either the And or Or operators to exclude a small set of data.

For example, the following filter conditions return only sales data for classic car items for the San Francisco office:

```
Sales Office = San Francisco
And Product Line = Classic Cars
```

The following filter conditions return all sales data for the San Francisco and Boston offices:

```
Sales Office = San Francisco
Or Sales Office = Boston
```

The following filter conditions return sales data for all product lines, except classic cars, sold by the San Francisco office:

```
Sales Office = San Francisco
And Not (Product Line = Classic Cars)
```

Specifying the evaluation order

BIRT Studio evaluates filter conditions in the order in which they appear. If you define more than two conditions, you can use parentheses to group conditions to get the results you expect. For example, A And B Or C is evaluated in that order, so A and B must be true or C must be true to include a data row. In A And (B Or C), B Or C is evaluated first, so A must be true and B Or C must be true to include a data row. To illustrate the difference a pair of parentheses can make, compare the following examples.

The following filter contains four conditions and none of the conditions are grouped:

```
Sales Office = San Francisco
and ProductLine = Classic Cars
and Total >= 8000
or Profit >= 4500
```

Figure 5-13 shows the generated report. Although the filter specifies the San Francisco office and the Classic Cars product line, the report displays data for other sales offices and product lines. Without any grouped conditions, the filter includes rows that meet either conditions 1, 2, and 3, or just condition 4.

Sales Office	Product Line	Product Name	Total	Profit
San Francisco				
	Classic Cars			
		1956 Porsche 356A Coupe	$6,534.41	$1,914.31
		1968 Dodge Charger	$5,298.72	$1,691.04
		1968 Ford Mustang	$7,273.14	$3,268.86
		1969 Ford Falcon	$7,720.18	$3,899.88
		1976 Ford Gran Torino	$5,864.88	$2,778.30
		2001 Ferrari Enzo	$5,485.92	$2,331.45
	Vintage Cars			
		1903 Ford Model A	$6,098.72	$2,888.62
		1917 Grand Touring Sedan	$7,031.20	$3,216.40

Figure 5-13 A report displaying the results of a complex filter that does not use parentheses to group conditions

The following filter contains the same four conditions, but this time the third and fourth conditions are grouped:

```
Sales Office = San Francisco
and ProductLine = Classic Cars
and (Total >= 8000
or Profit >= 4500)
```

Figure 5-14 shows the generated report. This time, only two rows meet the conditions. The Sales Office = San Francisco and ProductLine = Classic Cars conditions must be true, and either the Total >= 8000 condition or the Profit >= 4500 condition is true.

Sales Office	Product Line	Product Name	Total	Profit
San Francisco				
	Classic Cars			
		1956 Porsche 356A Coupe	$6,534.41	$1,914.31
		1968 Dodge Charger	$5,298.72	$1,691.04
		1968 Ford Mustang	$7,273.14	$3,268.86
		1969 Ford Falcon	$7,720.18	$3,899.88
		1976 Ford Gran Torino	$5,864.88	$2,778.30
		2001 Ferrari Enzo	$5,485.92	$2,331.45

Figure 5-14 A report displaying the results of a complex filter that uses parentheses to group conditions

Changing a condition

To change a filter condition, in the Filters area of Advanced Filter, select the condition, then choose Edit. Modify the condition by changing the values in Filter By, Condition, or Value. Select Change Condition, then select Change Filter, and choose OK, as shown in Figure 5-15.

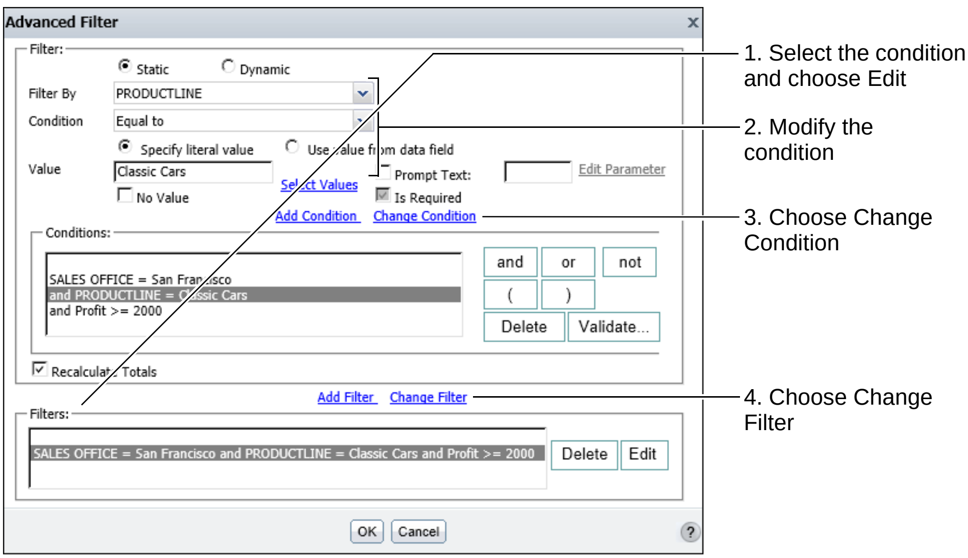


Figure 5-15 Changing a filter condition in Advanced Filter

Deleting a condition

To delete a filter condition, in the Filters area of Advanced Filter, select the Filter, and choose Delete. Then, select the Condition and choose Delete. Choose OK. Verify that any remaining filter conditions display the expected results.

Defining a filter at the data set level

You create a filter at the data set level using one of the following methods:

- Use Advanced Filter.
- Use the BIRT Studio Table Builder—Filter wizard.

How to create a filter at the data set level

1 In Available Data, choose Filter, as shown in Figure 5-16.

Filter Pane appears below the report design. It is empty, as shown in Figure 5-17, if you have not yet defined any filters at the data set level.

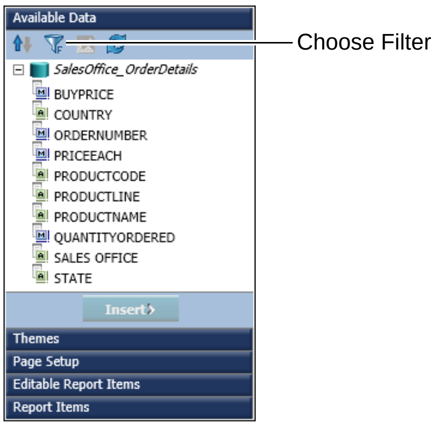


Figure 5-16 Choosing Filter

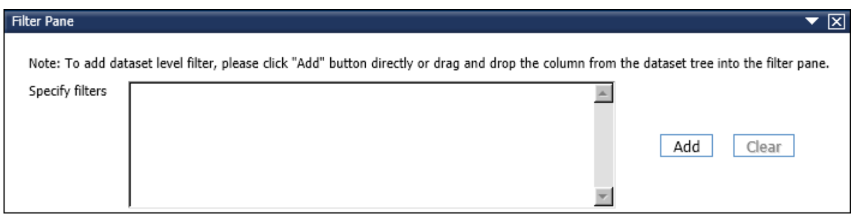


Figure 5-17 Filter Pane

2 Choose Add to create a new filter. Advanced Filter appears.

3 Define the filter condition.

How to create a data set filter using Table Builder

1 Create a new report using a data set in a template, or a custom data source.

2 On Table Builder—Data, select the data fields to use in the report design.

3 Choose Filter. On Table Builder—Filter, choose Add Filter.

4 Filter appears, as shown in Figure 5-18. To create a filter condition, complete the following steps:

1. In Filter By, select a column from the drop-down list.
2. In Condition, select an operator. Depending on the operator you choose, one or two value fields appear. The example in Figure 5-18 shows the selection of the operator Between, and two corresponding value fields.
3. In Date From and Date To, do one of the following:
    - Type a value.
    - Choose Select Values, then select a value from the list of values that appears.
    - Use the calendar tool to select a date.
    - Select No value to choose a Null value.
4. Choose Add Condition. Filters displays the new condition, as shown in Figure 5-18.

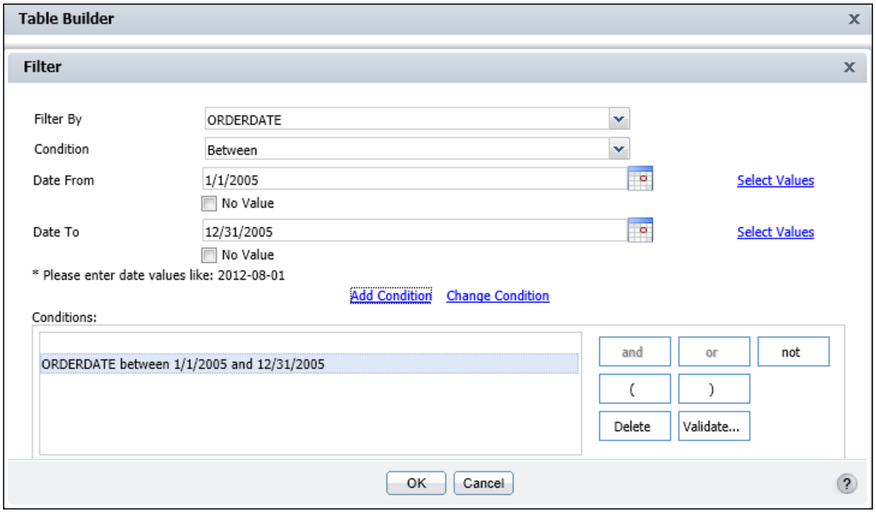


Figure 5-18 Specifying a data set filter in Table Builder

5. Choose Validate to verify the syntax of the new condition.

Choose OK. BIRT Studio displays data fields that meet the filter condition in the report design.

How to modify a data set filter

To modify an existing filter condition, complete the following steps on Table Builder—Filter:

1 Select Edit, as shown in Figure 5-19.

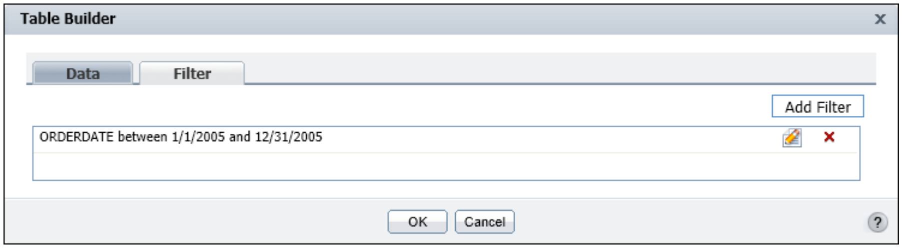


Figure 5-19 Modifying a filter condition

2. Filter appears. In Filter, Conditions displays the existing filter conditions. Select the condition to modify.
3. In Filter By, select a new column from the drop-down list.
4. In Condition, select an operator. Depending on the operator you choose, one or two value fields appear.
5. In Value, do one of the following:
   - Type a value.
   - Choose Select Values, then select a value from the list of values that appears.
   - Use the calendar tool to select a date.
   - Select No value to choose a Null value.
6. Choose Change Condition. Filters displays the modified condition.
7. Choose Validate to verify the syntax of the modified condition.

   Choose OK. BIRT Studio displays data fields that meet the filter condition in the report design.

Prompting for filter values at run time

You can create a prompt that accepts a value or multiple values for a filter when a report runs. A prompt, also called a parameter, enables users reading your report to control the content of the report without having to edit the report. For example, in a report that displays sales data by sales office, instead of creating a filter that returns data for a specific office, you can create a prompt that asks the user to select the office for which they want to view data in the report. Even if you are the only one reading the report, a prompt makes it easy for you to view the report with different content. Without a prompt, you must edit the filter condition to supply a different value.

About report parameters

Report parameters provide a mechanism for collecting values from a report user or a program. You typically use report parameters in filters to collect information that determines the data to display in a report. In BIRT Studio, you can create a static filter parameter or a dynamic filter parameter.

To construct a filter parameter, complete the following tasks:

- Choose a filter parameter type.
- Define a condition.
- Specify whether the parameter is required or optional. Dynamic filter parameters are always optional.
- Select a display type from the list of available display options described in Table 5-1.
- Specify a list of values from which a user can select values.

Choosing a parameter type

A static filter parameter is a filter condition that the report developer defines, that prompts the user to specify one or more values when running a report. The filter condition typically consists of an operator, specified by the report developer, and corresponding values, to be specified by the report user when running a report.

A dynamic filter parameter differs from a static filter parameter in one important aspect. Using a dynamic parameter, the report developer can provide report users with a list of operators to use to construct a filter condition. The user selects an operator, then specifies one or more corresponding values when running the report. The report displays data for the filter parameter condition the user specifies.

Making a filter parameter optional

When you create a prompt, you can require the user to specify a value, or you can make the value optional. It is typically good practice to make the value optional, so the user can view the report with all the data. For example, if Parameters contains a list of sales offices for the user to choose from, if the prompt is not required, the user can leave the field blank to view sales data for all the sales offices.

You can also require that the user supply a value if displaying all the data results in a very long report. A report that runs into hundreds of pages is not only difficult to read, but takes longer to generate. In BIRT Studio, a dynamic filter parameter is always optional.

How to make a prompt value optional

To make a prompt value optional, in Filter, select Prompt Text. Prompt Text and Is Required appear highlighted. Deselect Is Required, as shown in Figure 5-20.

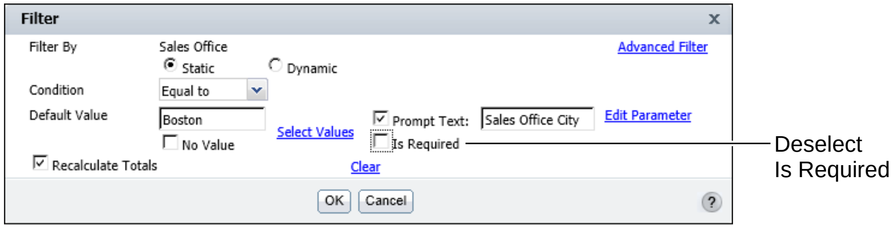


Figure 5-20 Specifying an optional prompt

No Value appears in Filter. Select No Value to display data rows that have null values.

Accepting multiple values

Users often want to select any number of values for a filter condition. In an inventory report, for example, the user might need to view data for several vendors. To support the selection of multiple values, create a filter parameter as follows:

- Select List Box as the display type.
- Select the In operator as one of the operators to provide to the user.
- Create a list of values.

Specifying a display type

When specifying properties for a static filter parameter, you can select the display types shown in Table 5-1 for the list of values. When specifying properties for a dynamic filter parameter, you can specify List Box or Text Box as the display type.

Table 5-1 Choosing a display type for the list of values

Option	Description	Use
Combo Box	Provides a combination of a drop-down list and a text box, where the user can either select a value from the list or type a value	Enables the user to select one or more values, or type a specific value for which BIRT Studio displays data.

(continues)

Table 5-1 Choosing a display type for the list of values (continued)

Option	Description	Use
List Box	Displays available values in a drop-down list	Enables users to select one or more values. If you use the In operator, the list box is the only available display type for the list of values.
Radio Button	Displays the list of values as radio buttons	Enables the user to select one value at a time. Additionally, this option enables a report developer to provide the user with a limited number of values from which to choose.
Text Box	Requires that the user type a value for which to display data	Enables the user to type a value for which BIRT Studio displays data.
Text Box-Auto Suggest	Provides the user with the available values that match a certain number of typed characters	Activate auto suggest after any number of typed characters. For example, if you activate auto suggest after one character, and if the user types S, all the values beginning with that character appear in a list from which the user selects a value. In case there is no match, BIRT Studio displays the message No Suggestions.

Providing the user with a list of values

To create a helpful prompt, you can provide the user with a list of values from which to choose. Do not assume that the report user knows what values to supply. For example, a user probably does not know that an order status field takes one of three values, Open, Closed, and In Process. In some cases, providing a list of values, such as customer names or invoice numbers, is necessary.

You create a list of values using one of the following techniques:

- Create a static list of values. In a static list, the values you select to display to the report user are fixed in the report design. You can either select from a list of values, or type values to populate the list of values.
- Create a dynamic list of values. In a dynamic list, the software generates the list of values when the report runs, using the current values in the data source. You can create a dynamic list of values at the data set level, or at the report column level.

Create a dynamic list for values that are frequently updated in the data source. For example, fields such as new customer names or product names are periodically updated in data sources. If you create a static list of these values, you need to update the list manually to match the values in the data source. Create a static list if you want to control the list of values displayed to the report user, for example, if you want to display only some of the values.

Creating a static filter parameter

For a static filter parameter, the report developer specifies the filter condition, consisting of an operator and display type to specify the value for which BIRT Studio displays data. The report developer also creates a list of values. A static filter parameter can be required or optional.

How to create a static filter parameter at the report table level

1 In BIRT Studio, select the column that contains the values for which you want to create a static prompt, then choose Filter.

2 Filter appears. In Filter, select Static, if necessary.

3 In Condition, select an operator.

4 Select Prompt Text, then type the text to display.

5 If the report contains aggregate data, do one of the following:

   - To recalculate aggregate data values across the filtered data rows, select Recalculate Totals, if necessary.
   - To calculate aggregate data for the unfiltered data, deselect Recalculate Totals.

6 Choose Edit Parameter, as shown in Figure 5-21.

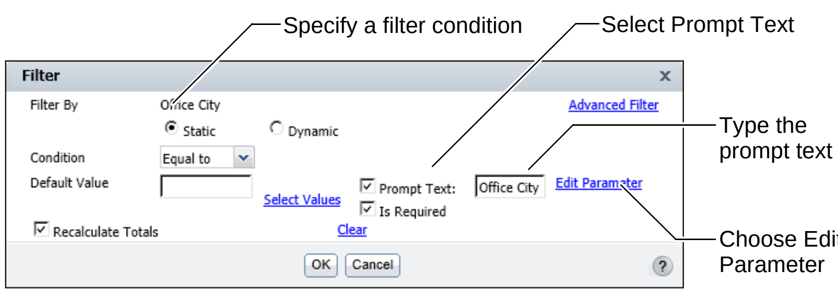


Figure 5-21 Creating a filter that prompts for a value when the report runs

7 Edit Parameter appears. On Edit Parameter, complete the following steps:

   1 In Prompt Text, modify the existing text, if necessary.

   2 In Help Text, optionally type a tooltip to assist the user in selecting the values for which BIRT Studio displays data.

3 Accept the default selection of Is Required. To make the parameter optional, deselect Is Required.

4 In Display Type, select an option from the list. The example in Figure 5-22 shows the display type set to Text Box.

5 Select the values to make available to the report user when the report runs. Choose Select Values. Select a value from the list of values that appears in Find, then press Ctrl and select the each additional value you want to display.

6 If you selected Text Box as the display type, select a value to set as the default. If you do not specify a default value, the first value you selected is set as the default value. The example in Figure 5-22 displays Boston as the default value.

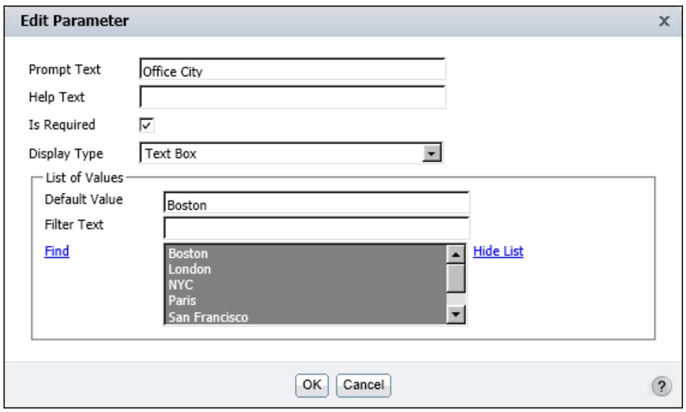


Figure 5-22 Creating a static parameter

Choose OK. The specified default value appears in Default Value, on Filter, as shown in Figure 5-23. Choose OK, the filter condition takes effect immediately.

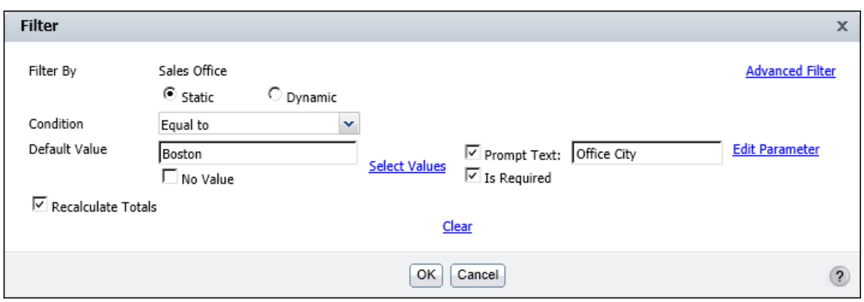


Figure 5-23 Filter displaying the specified parameter properties

How to create a static filter parameter with a list of values

When you create a static filter parameter, and select an available display type other than Text Box, you can create a static or dynamic list of data values to provide to the user. To create a static filter parameter with a list of values, on Edit Parameter, complete the following steps:

1. In Prompt Text, if necessary, type the display text for the prompt.
2. In Help Text, optionally type a tooltip to assist the user in selecting values for which BIRT Studio displays data.
3. Accept the default selection of Is Required. To make the parameter optional, deselect Is Required.
4. In Display Type, select an option other than Text Box from the list of options. The example in Figure 5-24 shows the selection of List Box.
5. In List of Values:
   - Select Static to create a static list of values, that the report developer specifies. If you select Static, proceed to step 6.
   - Select Dynamic to create a list of values that is retrieved from the data source when the report runs. If you select Dynamic, proceed to step 8.

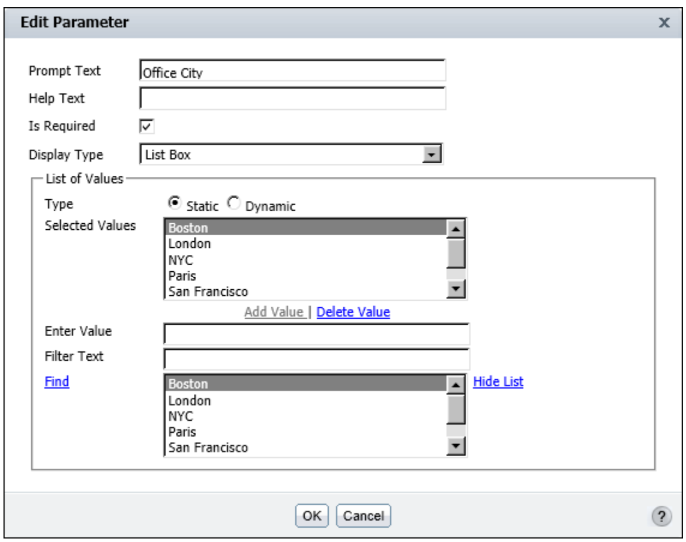


Figure 5-24 Creating a static list of values

6. To select the values to make available to the report user when the report runs, choose Select Values.

   Select a value from the list of values that appears in Find, then press Ctrl and select any additional values you want to display. You can also type each value in Enter value, then choose Add value. The added values appear in Selected Values.

7 In Selected Values, select a value to set as the default value, as shown in Figure 5-24.

- If the parameter is optional, you do not need specify a default display value.
- If the parameter is optional, No Value appears on Edit Parameter. Select No Value to display data rows that have null values.

Choose OK. Proceed to step 10.

8 In Data Set, from the drop-down list, select Current column (No Data Set), or select an available data set. If you selected the current column complete the following steps, as shown in Figure 5-25:

- Choose Select Values. Select a value from the list of values that appears in Find, then press Ctrl and select each additional value to display. The first value you select is set as the default value.

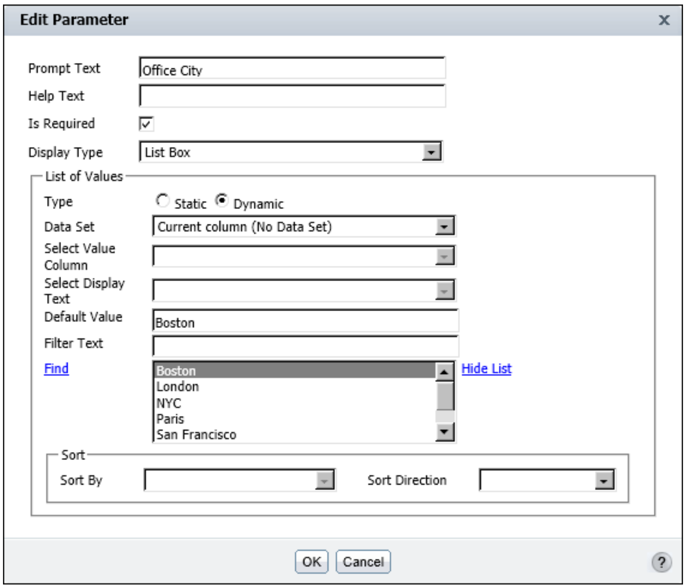


Figure 5-25 Creating a dynamic list of values

- In Sort, in Sort Direction, optionally select Ascending or Descending. If you do not specify a sort direction, BIRT Studio displays the default value first, and arranges the remaining values in ascending order below the default value.

Choose OK. Proceed to step 10.

9 In Data Set, if you selected an available data set, complete the following steps:

- In Select Value Column, select the column whose values are filtered according to the parameter condition.
- In Select Display Text, select a column from the drop-down list, containing the values displayed to the user. For example, you can choose to display the office code values for the user to choose from, instead of the names of sales office cities.
- Choose Select Values. Select a value from the list of values that appears in Find, then press Ctrl and select the remaining values you want to display. The first value you select is set as the default.
- In Sort, specify a field to sort the list of values by.
- In Sort Direction, select Ascending or Descending from the drop-down list.

  If you do not specify a sort direction, BIRT Studio displays the default value first, and arranges the remaining values in ascending order below the default value.

Choose OK.

10 On Filter, choose OK.

11 Test the prompt in BIRT Studio by choosing Parameter. Parameters displays the prompt, as shown in Figure 5-26. In this example, the default parameter value is Boston.

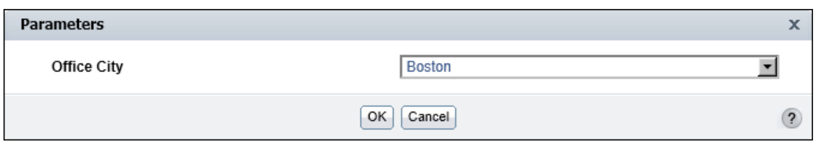


Figure 5-26 Parameters displaying the specified prompt

How to create a static filter parameter at the data set level

1 Create a new report by selecting a template containing a data set.

2 Select the data fields to insert in the report, and lay out the report.

3 In BIRT Studio, in Available Data, choose Switch Filter Pane.

4 Filter Pane appears, as shown in Figure 5-27. Choose Add to define a filter condition.

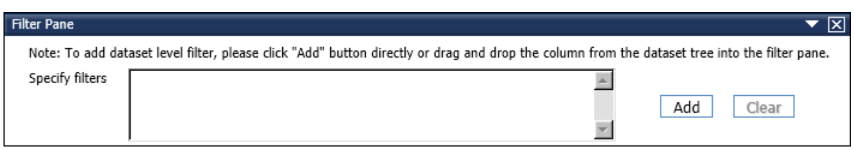


Figure 5-27 Filter Pane

5 Advanced Filter appears, as shown in Figure 5-28. On Advanced Filter, in Filter By, select a column from the drop-down list.

6 In Condition, select an operator from the drop-down list.

7 Select Prompt Text, then type the text to display. Edit Parameter is highlighted, and Is Required appears selected, as shown in Figure 5-28.

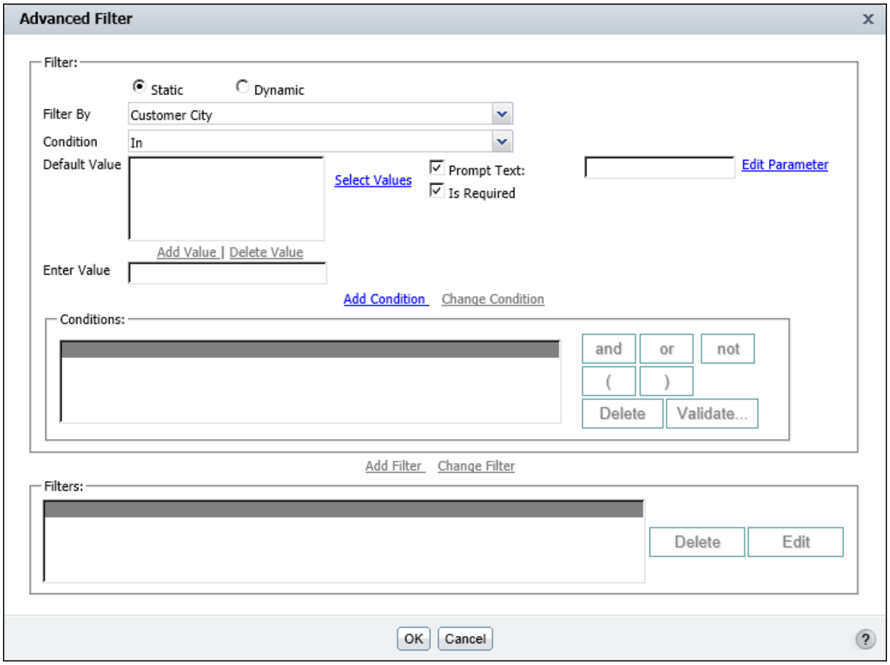


Figure 5-28 Defining a static filter parameter in Advanced Filter

8 Choose Edit Parameter. In Edit Parameter, create the parameter condition as follows:

- To create a static filter parameter using a list of values complete steps 1 to 9 in "How to create a static filter parameter with a list of values," earlier in this chapter.
- To create a static filter parameter displaying a text box where the user types the value, complete step 7 in "How to create a static filter parameter at the report table level," earlier in this chapter.

Choose OK. Figure 5-29 displays the completed filter parameter condition.

The selected default value appears in Default Value, as shown in Figure 5-29.

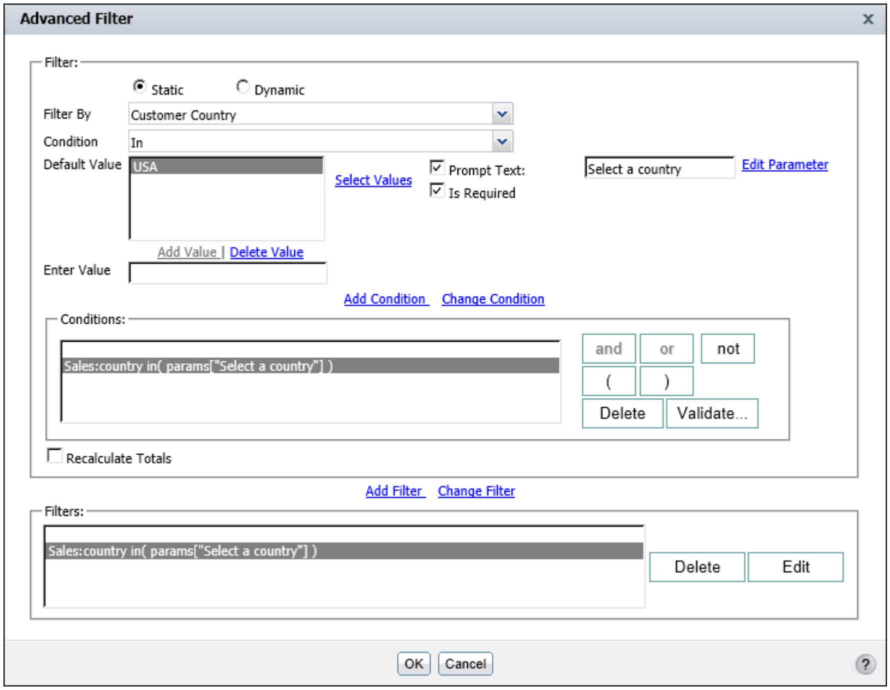
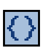


Figure 5-29 Advanced Filter displaying the static filter parameter condition

9 Choose Add Condition. The condition appears in Conditions, as shown in Figure 5-29.

10 Choose Add Filter. The static filter parameter condition appears in Filters.

Choose OK. Filter Pane displays the specified condition.



11 Test the prompt in BIRT Studio by choosing Parameter. Parameters displays the prompt, as shown in Figure 5-30. The default value specified in the example condition is USA.

Figure 5-30 Parameters displaying the static parameter condition

Creating a dynamic filter parameter

In BIRT Studio, dynamic filter parameters modify the underlying query, so filtering occurs at the data source level. This method of querying the data source for the most recent information also provides users with more accurate information. Only data rows that meet the filter criteria are retrieved from the data source. Retrieving a limited number of rows improves performance and reduces the time it takes for the data to appear in the report.

Dynamic filter parameters provide users with more control over the data they view in a report. Instead of specifying only the value on which to filter, the report user can specify conditions, such as Profit Less than 1000, Profit Between 1000 and 2000, or Profit Greater than 2000. The user can also choose to view all totals, omitting any filter conditions. A dynamic filter parameter is always optional.

The basic properties you specify when you create a dynamic filter parameter, are similar to the properties for a static filter parameter, with one exception. On Edit Parameter, you specify the operators to provide to users, and set a default operator. Like a static filter parameter, a dynamic filter parameter can also provide the user with a list of values. However, to provide the user with a static or dynamic list, values must be presented in a list box.

Figure 5-31 shows an example of a dynamic filter parameter that includes a list of values. The display type is set to List Box.

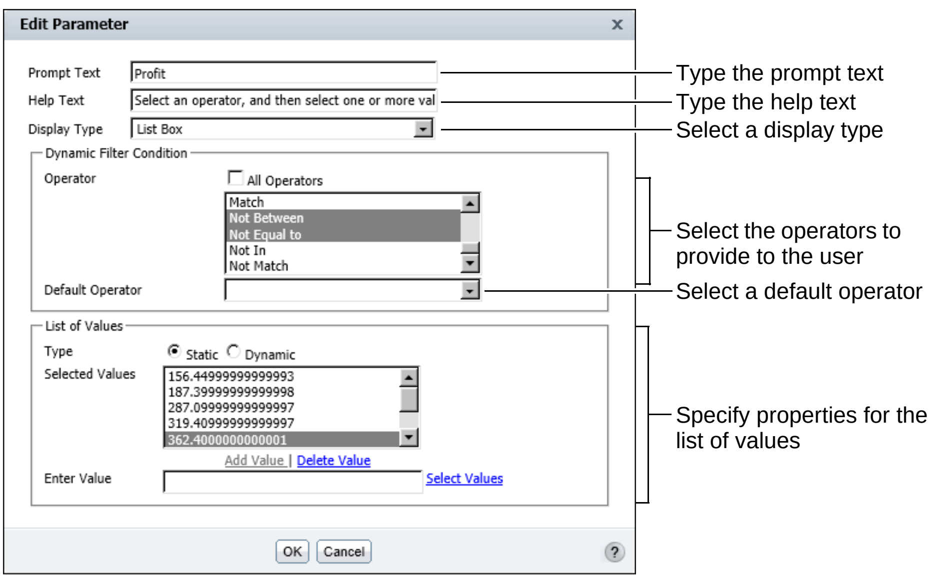


Figure 5-31 Specifying properties for a dynamic filter parameter

How to create a dynamic filter parameter at the report table level

1 Select the column for which you want to create a prompt, and choose Filter. Filter appears. The selected column appears in Filter By.

2 Select Dynamic, as shown in Figure 5-32.

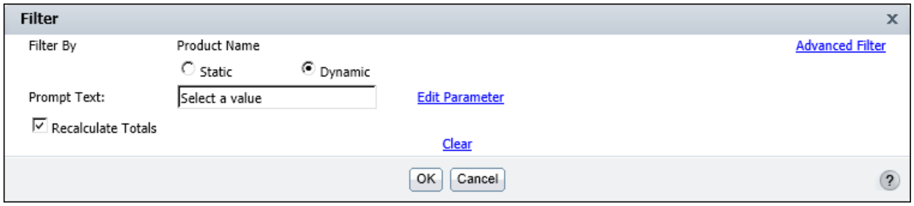


Figure 5-32 Specifying filter parameter properties

3 In Prompt Text, type a description for the column you selected in step 1. Then, choose Edit Parameter.

4 If the report contains aggregate data, do one of the following:

- To recalculate aggregate data values across the filtered data rows, select Recalculate Totals, if necessary.
- To calculate aggregate data for the unfiltered data, deselect Recalculate Totals.

5 Choose Edit Parameter. In Edit Parameter, complete the following steps:

1. In Prompt Text, if required, modify the text you typed earlier.
2. In Help Text, optionally type a tooltip that assists the user in selecting parameter values, for which BIRT Studio displays data.
3. In Display Type, choose List Box.
4. In Dynamic Filter Condition, in Operator, select one or more operators from the drop-down list to provide to the user. By default, all operators are selected. To select operators from the list, deselect All Operators. Then, press Ctrl and select each operator from the list of available operators.
5. Select a default operator from the drop-down list.
6. In List of Values:
   - Select Static to create a static list of values. A static list contains values specified by the report developer. If you selected Static, proceed to step 7.
   - Select Dynamic to create a dynamic list. A dynamic list contains values that are generated at run time from the data source. If you selected Dynamic, proceed to step 9.

7 Choose Select Values. In Find, select the values from the list of values. You can also type each value in Enter value, then choose Add value. The specified values appear in Selected Values.

8 In Selected Values, select a value to set as the default.

Choose OK. Proceed to step 6.

9 In Data Set, select Current Column (No data set), or the available data set. If you selected the current column, complete the following steps:

- Choose Select Values. Select a value from the list of values that appears in Find to set as the default value.
- In Sort, in Sort Direction, select Ascending or Descending. If you do not specify a sort direction, BIRT Studio displays the default value first, and arranges the other values in ascending order below the default value.

  Choose OK. Proceed to step 6.

10 In Data Set, if you selected an available data set, complete the following steps:

- In Select Value Column, select the column whose values are filtered according to the parameter condition. The example, as shown in Figure 5-33, uses the Product Name column.

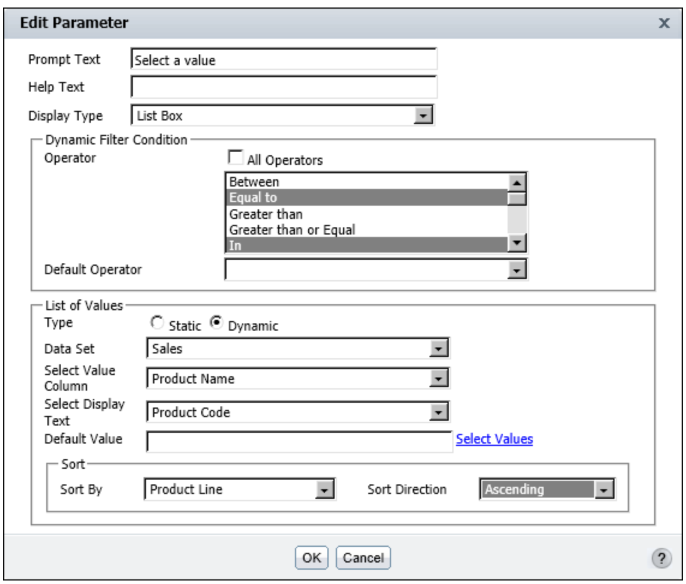


Figure 5-33 Specifying a dynamic list of values

- In Select Display Text, optionally select a column from the drop-down list, containing the values displayed to the user. In the example, as shown in Figure 5-33, displayed values are selected from the Product Code column.

- In Enter Value, optionally specify a default value.
- In Sort By, select a sort field by which the data in the list of values is sorted. In the example, as shown in Figure 5-33, displayed values are sorted by Product Line.
- In Sort Direction, select Ascending or Descending. If you do not specify a sort direction, BIRT Studio displays the default value first, and arranges the remaining values in ascending order below the default value.

Choose OK.

6 On Filter, choose OK.

7 To test the prompt in BIRT Studio choose Parameter. Parameters displays the prompt, as shown in Figure 5-34.

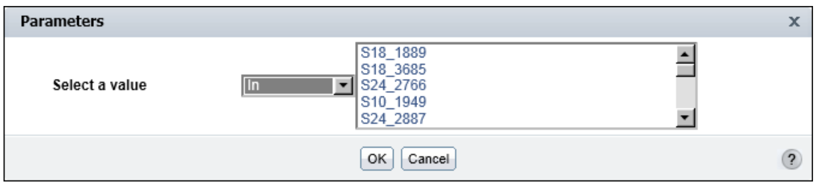


Figure 5-34 Parameters displaying a dynamic prompt

How to create a dynamic filter parameter at the data set level

1 Create a new report by selecting a template containing a data set, or a custom data source.

2 Select the data fields to insert in the report.

3 In BIRT Studio, in Available Data, choose Switch Filter Pane.

4 Filter Pane appears. Choose Add to define a filter condition. Advanced Filter appears.

5 On Advanced Filter, select Dynamic.

6 In Filter By, select a column from the drop-down list.

7 Choose Edit Parameter. In Edit Parameter, create the parameter condition by completing step 5 in "How to create a dynamic filter parameter at the report table level," earlier in this chapter.

In Advanced Filter, Prompt Text displays the text you entered earlier.

Choose Add Filter. The dynamic filter parameter condition appears in Filters, as shown in Figure 5-35. Choose OK. Filter Pane displays the condition.

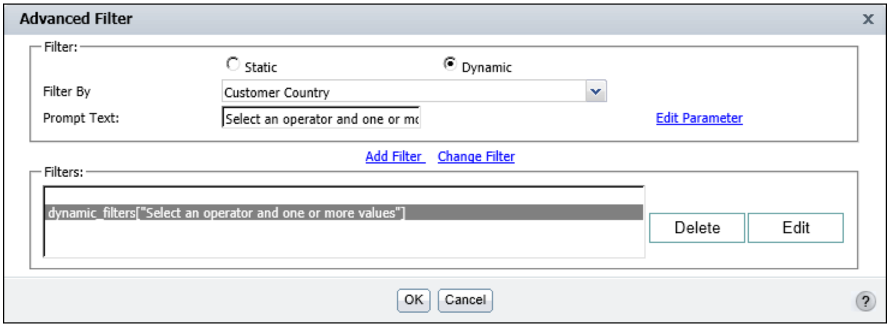


Figure 5-35 Advanced Filter displaying the dynamic filter parameter

8 To test the prompt in BIRT Studio, choose Parameter. Parameters displays the prompt, as shown in Figure 5-36. The default operator specified in the example condition is In, and the default value is USA.

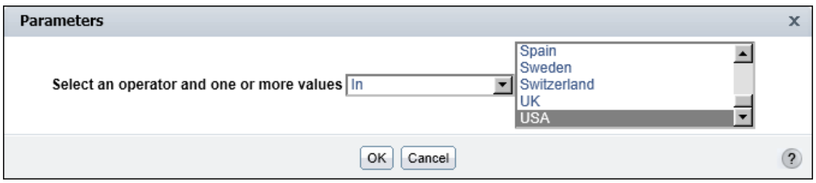


Figure 5-36 Parameters displaying the dynamic prompt

When working with a dynamic prompt set on a column containing numeric data that uses the Match or Not Match operators, if you use the value 1 as the string value to match you must enclose the value in double quotation marks (" "). Type the value as "1" in the text box or combo box, so that BIRT Studio evaluates it accurately as a literal string. If you use other values such as 2, 3, and so on, you do not need to enclose them in double quotation marks.

Creating multiple filter parameters

A report developer can also define multiple conditions, for which a user specifies values, further narrowing the scope of data displayed in a report. For example, you can define the first condition such that the user specifies a Sales Office value, the next condition where the user specifies one or more Product Lines, and a third condition where the user specifies a Profit value or range. The user can specify that the Sales Office value is equal to Boston, the Product Line is In Classic Cars, and Motorcycles, and that the Profit is less than or equal to 1001.10, as shown in the example. BIRT Studio displays the data rows that fulfill all three conditions.

You can specify multiple static filter parameters, dynamic filter parameters, or a combination of both types. You create multiple filter parameters on Advanced Filter. The following section describes how to define multiple filter parameters in a report design.

How to create multiple filter parameters

1. Select the first column for which you want to create a prompt, and choose Filter. Filter appears. The selected column appears in Filter By.
2. Select Static or Dynamic.
   - To create a static filter parameter, complete steps 1 to 9 in "How to create a static filter parameter with a list of values," earlier in this chapter.
   - To create a dynamic filter parameter, complete steps 1 to 9 in "How to create a dynamic filter parameter at the report table level," earlier in this chapter.
3. On Filter, choose Advanced Filter.
4. On Advanced Filter, choose Add Condition. The filter parameter condition you previously specified appears in the Filters area in Advanced Filter.
5. Choose Add Filter.
6. To define the next condition, in Advanced Filter, complete the following steps:
   1. In Filter, select Static or Dynamic.
   2. In Filter By, select a column from the drop-down list.
   3. In Prompt Text, type the display text for the prompt.
   4. If the report contains aggregate data, do one of the following:
      - To recalculate aggregate data values across the filtered data rows, select Recalculate Totals if necessary.
      - To calculate aggregate data for the unfiltered data, deselect Recalculate Totals.
   5. Choose Edit Parameter.
      - If you selected Static in step 1, complete steps 1 to 9 in "How to create a static filter parameter with a list of values," earlier in this chapter.
      - If you selected Dynamic in step 1, complete steps 1 to 9 in "How to create a dynamic filter parameter at the report table level," earlier in this chapter.
   6. In Advanced Filter, choose Add Filter.
7. To define additional filter parameters, in Advanced Filter, repeat step 6. Choose OK. The example in Figure 5-37 shows Advanced Filter displaying three filter parameter conditions.

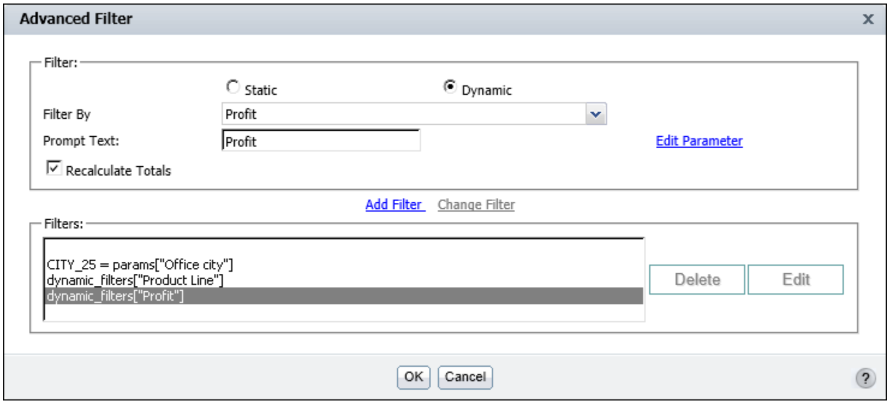


Figure 5-37 Defining multiple filter parameters in Advanced Filter

8 To test the prompt in BIRT Studio choose Parameter. Parameters displays the prompt with all defined conditions, as shown in Figure 5-38. The first condition is a static filter parameter, and the remaining two conditions are dynamic filter parameters.

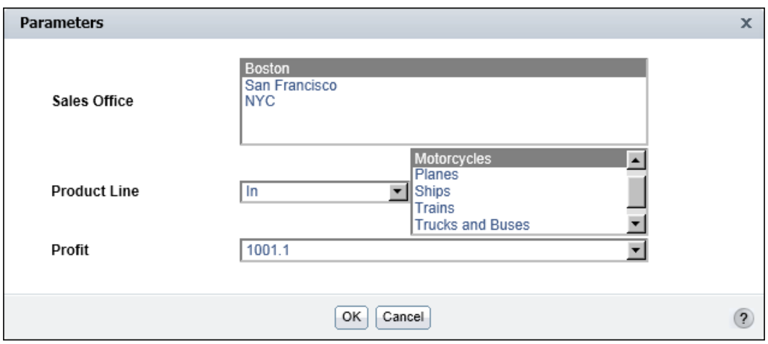


Figure 5-38 Specifying multiple filter parameter values

Chapter

6

Working with summary tables

This chapter contains the following topics:

- About summary tables
- Creating a summary table
- Modifying a summary table
- Inserting a chart in a summary table
- Changing table types

About summary tables

A summary table presents aggregate data information in a report, providing users with a concise view of the data. A summary table helps improve response time, reduce network traffic, and enables customers to view the data they need in a format that is easily understood. The summary table feature in BIRT Studio enables report users to display only aggregate data from data sets that contain a large volume of data. The aggregate calculation occurs at the data source level, which enables users to quickly view summary information at a glance.

When a report design presents all detail rows of a data set, with or without aggregate data, the table is called a detail table. By hiding the details for a group in a detail table, you can provide the user with a simplified view of the aggregate data. Using a summary table to achieve the same result improves performance, and reduces the load on the server, when dealing with large volumes of data.

For example, from a data source that contains two fields Order Status and Amount, using a summary table, the user can view the Total Amount for each Status. In an ordinary report design, BIRT Studio retrieves the sales for each status and calculates the sum of the Amount. When you use a summary table, BIRT Studio calculates the sum of the Amount field at the data source level, and presents the corresponding values in the report design. This section describes how you can use BIRT Studio to create and modify a summary table.

About columns in a summary table

The columns in a summary table are dimension columns, attribute columns, and measure columns. Dimension columns group data in other columns. Measure columns contain the aggregated values that are evaluated. Some examples of dimension columns include order date, country, state, or product line. Dimension columns containing date-and-time data can be grouped in intervals, in a summary table. Depending on the type of data in a measure column, BIRT Studio makes available specific aggregate functions to use on the column. Some examples of measure columns include quantity ordered, profit, and revenue.

In a summary table, columns that provide additional information about a dimension field, are called attribute columns. For example a product name, or product code data field can be attributes of the Product Line dimension. Each dimension can contain several attribute fields.

Designing a summary table

When you design a summary table, first evaluate which dimension, attribute, and measure columns to select, based on the aggregate data you want to display. In the example report, as shown in Figure 6-1, data in the measure column is grouped by the dimension, and displays subtotals in the group footer. Duplicate values in a grouped dimension column do not appear in the summary table.

Although there are two dimension columns in the example in Figure 6-1, the outermost column is used to group data in the other columns. The attribute column, Product Name, is an attribute of the Product Line dimension. You can perform multiple aggregate calculations for data in a single measure column. A grand total is displayed in the report table footer.

Dimension columns
Attribute column
Measure columns
Subtotal
Subtotal
Grand total

COUNTRY	PRODUCTLINE	PRODUCTNAME	TOTAL QUANTITY ORDERED	TOTAL ORDERS
France				
	Classic Cars	1965 Aston Martin DB5	156	$14,571.44
	Motorcycles	1996 Moto Guzzi 1100i	71	$6,066.78
	Vintage Cars	1938 Cadillac V-16 Presidential Limousine	43	$1,676.14
Sub Total (France)			*270*	*$22,314.36*
USA				
	Classic Cars	1957 Ford Thunderbird	194	$15,739.55
	Vintage Cars	1917 Grand Touring Sedan	254	$16,902.43
Sub Total (USA)			*448*	*$32,641.98*
Grand Total			*718*	*$54,956.34*

Figure 6-1 Example of a summary table

Although each attribute column must be associated with a dimension column for it to be useful in a summary table, you do not need a dimension column to create a meaningful summary table. The example, as shown in Figure 6-2, displays order totals, the measure column, for each state, the attribute column, and a grand total for all states.

STATE	ORDER TOTALS
NV	$80,180.98
Victoria	$42,768.03
Grand Total	*$122,949.01*

Figure 6-2 Summary table without groups displaying the grand total

The example, as shown in Figure 6-3, displays the grand total for two peer attribute columns and a measure column.

PRODUCTNAME	PRODUCTCODE	SUM (QUANTITYORDERED)
1965 Aston Martin DB5	S18_1589	767
1996 Moto Guzzi 1100i	S10_2016	283
P-51-D Mustang	S18_2581	192
1938 Cadillac V-16 Presidential Limousine	S24_2022	439
Grand Total		*1681*

Figure 6-3 Grand totals displayed for attribute columns

Because these are peer attribute columns, and the summary table does not contain the dimension field they are associated with, data is not grouped, and subtotals are not displayed. The classification of data fields into dimensions, attributes, and measures is defined in the data set.

Creating a summary table

To create a summary table, complete the following steps:

1 Select a template.

2 Select a data source.

3 Select the dimension, attribute, and measure fields to use in the summary table.

As you did when creating a detail report, you do one of the following:

- Use Table Builder to select the data fields for the summary table. Using Table Builder, you can also specify the following information:
  - Group date-and-time dimension columns by an interval.
  - Specify one or more aggregate functions to use for the selected measure columns.
  - Optionally create a filter condition at the data set level for a dimension or attribute column to limit the data displayed in the summary table.
- Manually select and insert data fields to create a summary table.

How to create a summary table using Table Builder

This section describes the process to create a summary table using a data set in a template.

1 Launch BIRT Studio, select a template, then select a data set.

2 Table Builder appears. In Table Builder, complete the following steps:

   1 In Table Builder—Data, in Fields, select Summarize to create a summary table. Current Measure Selections is highlighted.

   2 In Fields, in Available Data:

      1 Press Ctrl and select each dimension and attribute data field to insert in the report design. Then choose the right arrow.

      2 Press Ctrl and select the measure fields to insert in the report design. Then choose the right arrow.

   The selected data fields, Order Date and Product Name, appear in Current Column Selections. The selected measure fields, Order Total and Quantity Ordered, appear in Current Measure Selections, as shown in Figure 6-4. Because Product Name is an attribute of the Product Line dimension, the data field is listed as a subcategory of the Product Line dimension in Current Column Selections.

3 Use the up and down arrows to modify the order in which you want the selected fields to appear in the report design.

4 If the data source contains a large number of data fields, type the name of the field in Find, then choose Apply. If the data field is found, Table Builder displays the data field in Available Data. Select the data field, then choose the right arrow. The data field appears in Current Column Selections. Repeat this step to retrieve each additional data field for which you can supply a name.

5 For each selected dimension field containing date-and-time data, you can group the data in intervals. Select an available interval from the Date Value drop-down list. Figure 6-4 shows the Order Date field grouped in quarters.

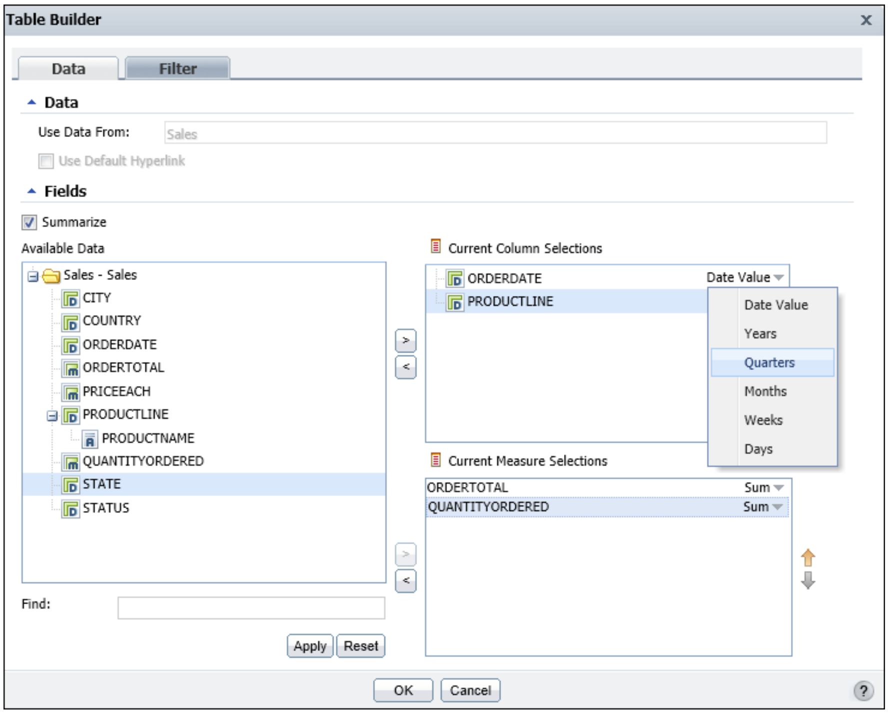


Figure 6-4 Grouping date dimension column in quarterly intervals

6 For every measure field containing numeric or string data, select an aggregate function to apply to the data from the drop-down list. For columns containing numeric data, the default aggregation function selected is Sum. To aggregate a measure field by an additional function, do the following:

1 Select the field once again in Available Data. Then select the right arrow so that the measure field appears once again in Current Measure Selections.

2 Use the menu to choose a new aggregate function to apply to the field. Figure 6-5 shows the average function applied to the Quantity Ordered field.

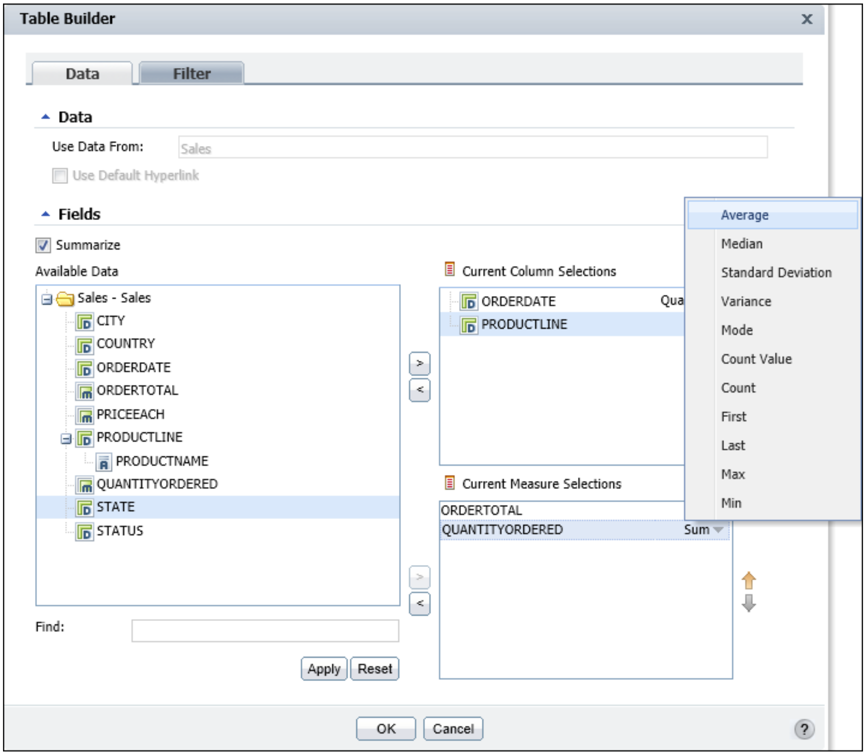
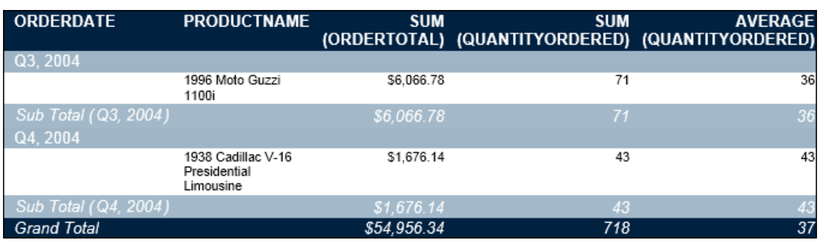


Figure 6-5 Aggregating measure columns

7 To limit the data displayed in the report, create a data set filter for dimension and attribute fields, in the same way you did for a detail report.

When done, in Table Builder, choose OK. BIRT Studio displays the selected data fields in a summary table, as shown in Figure 6-6. You are now ready to modify the summary table.



ORDERDATE	PRODUCTNAME	SUM (ORDERTOTAL)	SUM (QUANTITYORDERED)	AVERAGE (QUANTITYORDERED)
Q3, 2004				
	1996 Moto Guzzi 1100i	$6,066.78	71	36
Sub Total (Q3, 2004)		*$6,066.78*	*71*	*36*
Q4, 2004				
	1938 Cadillac V-16 Presidential Limousine	$1,676.14	43	43
Sub Total (Q4, 2004)		*$1,676.14*	*43*	*43*
Grand Total		*$54,956.34*	*718*	*37*

Figure 6-6 Summary table report design

How to create a summary table using an ODA data source

The following section describes the process to create a summary table using an ODA data source.

1 In BIRT Studio, create a new report by selecting a template and selecting the data source to use.

2 Choose Turn on Auto Summarize.

3 In Available Data, select the dimension, attribute, and measure columns to use, then choose Insert.

A summary table displaying aggregate data for the selected columns appears in the report design. You are now ready to modify the summary table.

Modifying a summary table

Using BIRT Studio, you can work with summary tables to format data, organize data in groups, sort data, create filters, and insert charts as you would for a detail table.

This section describes the actions you can perform on dimension columns, attribute columns, measure columns, and each column header. Table 6-1 lists the actions you can perform on dimension columns, measure columns, attribute columns, column headers, and the entire report table.

Table 6-1 Modifying data in a summary table

Action	Dimension column	Attribute column	Measure column	Column header	Report table
Advanced filtering					✓
Advanced sorting	✓		✓		✓
Create a new computed column			✓		
Create aggregate data			✓		
Create borders				✓	✓
Create data set filters					✓
Create filters	✓	✓	✓		✓
Create groups	✓				
Create hyperlinks	✓	✓	✓		

(continues)

Table 6-1 Modifying data in a summary table (continued)

Action	Dimension column	Attribute column	Measure column	Column header	Report table
Create page breaks before or after a group	✓				
Create sections					
Delete a group	✓				
Delete column	✓	✓	✓		
Edit the column header text				✓	
Filter Top/Bottom N			✓		
Format data	✓	✓	✓		
Group date-and-time data in intervals	✓				
Hide column					
Hide details for a group	✓				
Insert a chart		✓			✓
Reorder columns	✓	✓	✓		
Sort column data	✓	✓	✓		✓
Specify alignment properties	✓	✓	✓	✓	✓
Specify column width	✓	✓	✓		
Specify conditional formatting rules			✓		
Specify font properties	✓	✓	✓	✓	✓

You can perform the following actions at the report table level in a summary table:

- Insert a chart at the report table level.
- Hide a table, and display the corresponding chart.
- Sort data.
- Specify font properties.
- Specify alignment properties for column data.
- Format data types.
- Add a border for the table.

- Add data fields from the data set.
- Reorder columns.
- Create a bookmark to the summary table.

To add a table border, add data fields, insert a chart, reorder columns, or create a bookmark to the summary table, choose an option from the context menu of the report table, as shown in Figure 6-7.

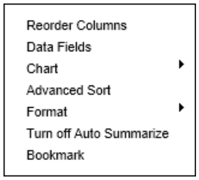


Figure 6-7 Modifying the report table

Modifying labels

You can perform the following actions on a column header, or an aggregate label, in a summary table:

- Edit the header or label text.
- Modify the alignment of label text.
- Format font properties.
- Create a border for the label.

To modify a column header, or an aggregation label select the column header, and choose an option from the context menu. The context menu appears, as shown in Figure 6-8.

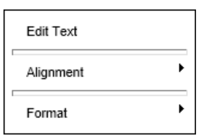


Figure 6-8 Modifying a column header

Working with a dimension column

If a report has one or more dimension columns, the order in which you select and insert the columns determines the group level. For example, if you select and insert Product Line before Country, Product Line is a higher level, or outer group. Country is the lower level, or inner group.

A grouped dimension column displays subtotals for the data in each group. When you group a dimension column, BIRT Studio eliminates duplicate data values when calculating aggregate data. Figure 6-9 displays a summary table containing three dimension columns. The Order Date column is grouped in quarterly intervals, and is located in the innermost position.

COUNTRY	PRODUCTLINE	ORDERDATE	SUM (QUANTITYORDERED)	SUM (ORDERTOTAL)
Australia				
	Motorcycles			
		Q2, 2003	212	$21,814.69
	Sub Total (Motorcycles)		*212*	*$21,814.69*
	Planes			
		Q2, 2003	192	$16,926.89
	Sub Total (Planes)		*192*	*$16,926.89*
	Vintage Cars			
		Q2, 2003	78	$4,026.45
	Sub Total (Vintage Cars)		*78*	*$4,026.45*
Sub Total (Australia)			*482*	*$42,768.03*

Innermost grouped date dimension

Figure 6-9 Grouping the innermost date-and-time dimension column

When a summary table contains more than one dimension column, data in the outer dimension columns can be grouped, as shown in Figure 6-9, and subtotals are displayed for each value in the grouped column.

When working with a dimension column that contains date-and-time data, you can group the data by intervals such as day, week, month, quarter, or year. When placed in the innermost position, dimension columns that contain other types of data, cannot be grouped, as shown in Figure 6-10. If a dimension column contains date-and-time data, you can always group data in the column by intervals.

COUNTRY	ORDERDATE	PRODUCTLINE	SUM (QUANTITYORDERED)	SUM (ORDERTOTAL)
Australia				
	Q2, 2003			
		Motorcycles	212	$21,814.69
		Planes	192	$16,926.89
		Vintage Cars	78	$4,026.45
	Sub Total (Q2, 2003)		*482*	*$42,768.03*
Sub Total (Australia)			*482*	*$42,768.03*

Innermost ungrouped dimension

Figure 6-10 Grouping dimension columns

The innermost dimension column does not display subtotals, irrespective of the type of data in the column, as shown in Figure 6-9 and Figure 6-10.

For a summary table containing a single dimension column, and one or more associated attribute columns, you cannot group the data in the dimension column. No subtotals appear. The summary table displays a grand total value for all product lines, as shown in Figure 6-11.

PRODUCTLINE	PRODUCTNAME	SUM (ORDERTOTAL)
Classic Cars	1965 Aston Martin DB5	$73,290.33
Motorcycles	1996 Moto Guzzi 1100i	$27,881.47
Planes	P-51-D Mustang	$16,926.89
Vintage Cars	1938 Cadillac V-16 Presidential Limousine	$27,164.68
Grand Total		*$145,263.37*

Figure 6-11 A summary table displaying a grand total

Modifying a dimension column

You can perform the following actions on a dimension column, as shown in Figure 6-12:

- Delete a group.
- Hide detail rows for a grouped column.
- Specify page breaks before or after a group.

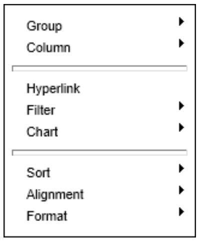


Figure 6-12 Modifying a dimension column

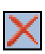
To modify a dimension column, select the column, and choose an option from the context menu. An example of the context menu for a dimension column, as shown in Figure 6-12, displays the available options.

Moving a dimension column

When you move a dimension column in a summary table, you can move it only among other dimension columns. Dimension columns and associated attribute columns always appear preceding the measure columns in a summary table.

Dimension columns can also contain associated data fields, called attributes. You cannot move a dimension column independently of its associated attribute columns.

Deleting a dimension column



To delete a dimension column, select the column, and choose Delete. When you delete a dimension column containing associated attributes from a summary table, the attribute fields are not deleted.

Working with an attribute column

Attributes represent subcategories of a broader category, or dimension. For example, Product Code and Product Name can be attributes of a dimension, Product Line. For an attribute to be useful in a summary table, it must be associated with a dimension field in the data set. The developer typically sets this property at the time of designing the data set. Multiple attribute columns associated with a single dimension field are called peer attribute columns. In a summary table that contains one or more dimension columns, and associated attributes, data in the attribute columns cannot be grouped and does not display subtotals.

In a summary table that contains one or more peer attribute columns, and no dimension columns, data in the attribute columns cannot be grouped and does not display subtotals. In a summary table that contains one or more non-peer attribute columns and no dimension columns, data in the outermost attribute column is grouped and displays subtotal aggregate values.

Inserting an attribute column

Attribute columns appear following the associated dimension column in a summary table. When you insert an attribute column in a summary table containing the associated dimension column, the attribute appears after the dimension column.

The example in Figure 6-13 shows a dimension column, Country, an associated attribute column, State, and a measure column, Quantity Ordered. There are no grouped columns in the summary tables and BIRT Studio displays aggregate values for the entire table.

COUNTRY	STATE	SUM (QUANTITYORDERED)
Australia	Victoria	482
USA	NV	929
Grand Total		*1411*

Figure 6-13 Dimension, associated attribute, and measure columns

When you insert an attribute column in a summary table that does not contain an associated dimension column, or other attribute columns, the attribute column is inserted to the left in the report table, but cannot be grouped if it is the only attribute column in the summary table. When you insert one or more non-peer attribute columns in a summary table with no dimension columns, you can group data in the outermost attribute column, and subtotal values are displayed for this grouped column.

The example in Figure 6-14 shows two non-peer attribute columns, State and Product Name, and one measure column. Data in the State column is grouped, and aggregate data values are displayed for each group.

STATE	PRODUCTNAME	SUM (QUANTITYORDERED)
NV		
	1957 Ford Thunderbird	611
	1917 Grand Touring Sedan	318
Sub Total (NV)		*929*
Victoria		
	1996 Moto Guzzi 1100i	212
	P-51-D Mustang	192
	1928 Ford Phaeton Deluxe	78
Sub Total (Victoria)		*482*
Grand Total		*1411*

Figure 6-14 Attribute and measure columns

When you insert an attribute column in a summary table containing a dimension column, and the inserted attribute column is not associated with the dimension column or the other attribute columns in the summary table, the new column appears in the position you select, as shown in Figure 6-15.

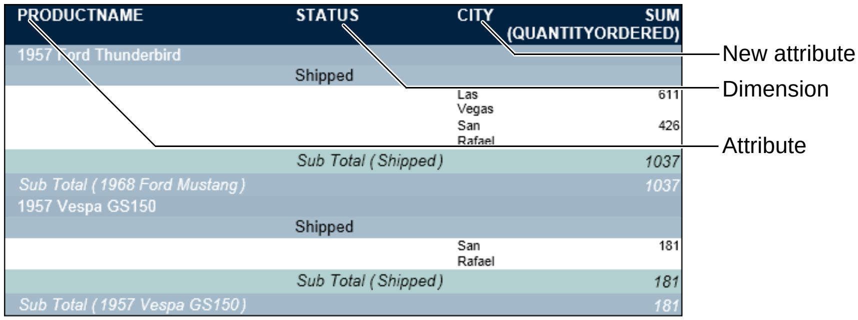


Figure 6-15 Inserting a non-associated attribute field in a summary table

Modifying an attribute column

In an attribute column, you can change column width, reorder columns, create hyperlinks, create filters, sort data, modify alignment, format data, and change font properties. To modify an attribute column, select the column, and choose an option from the context menu, as shown in Figure 6-16.

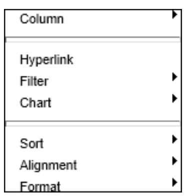


Figure 6-16 Modifying an attribute column

Moving an attribute column

In a summary table containing multiple dimensions and associated attribute columns, you can reorder attribute columns only among peer attribute columns. To move an attribute column, select the column, and choose Reorder Columns from the context menu. When you move a dimension column with its associated attribute columns in a summary table, the columns are placed in their new position as a single unit. You cannot move the dimension column without moving the associated attribute columns.

Deleting an attribute column

To delete an attribute column, select the column, and choose Delete. When a dimension column containing associated attributes is deleted from a summary table, the attributes remain in the summary table. To delete each attribute column select each column, and choose Delete.

Working with a measure column

Every summary table must contain at least one measure column to display aggregate data. The developer typically sets default aggregation functions for each measure column in the data set. You can modify the default aggregate function and add additional aggregate functions to data in measure columns.

Inserting a measure column

A measure column is typically inserted to the right of dimension and attribute columns in a summary table. When you insert a measure column in a summary table, aggregate data is displayed based on the default aggregation function set by developer at the time of creating the data set. You can modify the aggregate function and add additional aggregation functions using BIRT Studio.

Modifying a measure column

You can perform the following actions on a measure column:

- Insert a new computed column.
- Specify conditional formatting rules.
- Use the following numeric and non-numeric subtotal functions to modify aggregate data:
  - First
  - Last
  - Count value
  - Max
  - Min
  - Sum
  - Average
  - Median
  - Standard deviation
  - Variance
  - Mode

To modify a measure column, select the column, and choose an option from the context menu, as shown in Figure 6-17.

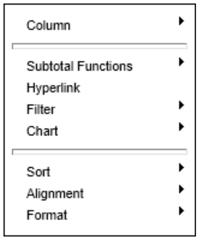


Figure 6-17 Modifying a measure column

Adding an aggregate calculation

You can apply multiple aggregate functions to a single measure column in a summary table. To add an aggregate function, complete the following steps:

1. In Available Data, select the measure column, and choose Insert.
2. In Select Subtotal Function, as shown in Figure 6-18, select an aggregate function from the drop-down list.

   Choose OK. The measure column you selected appears in the report design displaying aggregate values for the new aggregate function you defined.

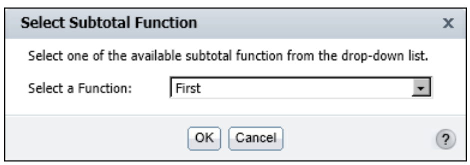
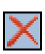


Figure 6-18 Adding an aggregate function

Moving a measure column

When you move a measure column in a summary table, you can move it among other measure columns. Measure columns always appear to the right of dimension and attribute columns in a report design.

Deleting a measure column



To delete a measure column, select the column, and choose Delete.

Formatting aggregate data values

You can format aggregate data values displayed as subtotals, or grand totals as follows:

- Specify the alignment properties.
- Specify a font type, color, or style.
- Specify an available or custom data format to the numeric data values.

To modify alignment options for the aggregate value, right-click the data cell, and choose Alignment➛Align Left, Align Right, or Align Center from the context menu.

To modify font properties for subtotals or grand total values, select the value, and choose Format➛Font from the context menu. In Font, use the drop-down lists to specify the font type, color, and style.

To apply a number format to the subtotals or grand total values, select the value, and choose Format➛Format Data from the context menu. In Number Column Format, select a format from the list of available formats.

Figure 6-19 displays an example of the context menu that appears when you right-click an aggregate data cell.

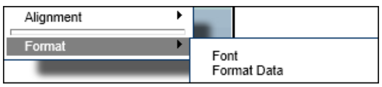


Figure 6-19 Choosing formatting options from the context menu

Filtering aggregate data

In a summary table, the process to define a filter condition is similar to defining a filter condition for a detail table. As in a detail table containing aggregate data values, in a summary table, you can specify whether to display aggregate data, such as subtotals and grand totals, for all the rows retrieved from the data set, or for only data rows that meet the filter condition. This feature is useful when you create a filter for the purpose of excluding certain values, such as a list of geographies with no active marketing campaigns, or comparing a subset of data with the original.

When you define a filter condition on a measure column, BIRT Studio displays a preview of the data rows that meet the specified condition. To display updated aggregate data in the summary table, in Filter, accept the default selection of Recalculate Totals. Any subtotals and grand totals in the summary table are calculated across the filtered data rows in the report design. To calculate aggregate data across all data rows retrieved from the data set, and not just the subset of filtered data, in Filter, deselect Recalculate Totals.

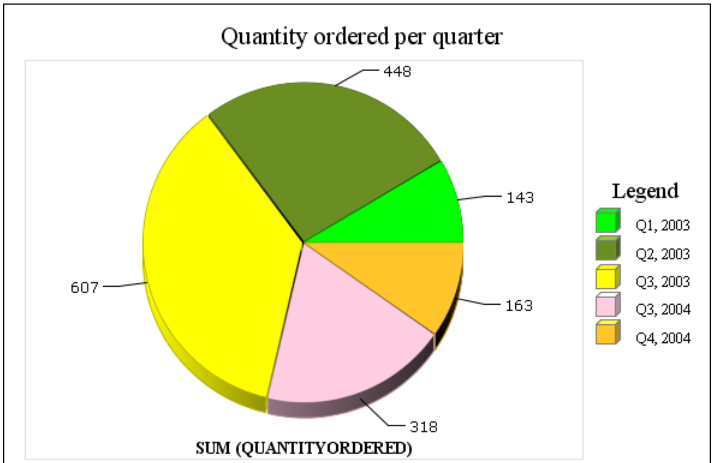
You can specify whether to recalculate totals when creating a simple filter using the Filter dialog box, in Advanced Filter, as well as when filtering the top or bottom values in a summary table.

Inserting a chart in a summary table

You can insert standard BIRT charts at the table level in a summary table. The types of standard BIRT charts you can use in a summary table are the same as those available for use in a detail table.

When you insert a chart in a summary table, BIRT Studio assigns the outermost grouped column as the category or *x*-axis, by default. You can select the value series on the *y*-axis from the available measure columns in the summary table. In addition, you can provide a tooltip, and format the chart by specifying a title, font properties, size, dimension, and legend properties, as you do for standard charts.

The example in Figure 6-20 displays a pie chart showing the total product quantities ordered each quarter. The outermost grouped column is Order Date, which BIRT Studio sets as the category axis. You can choose either of the two measure columns as the *y*-axis, or both, to display multiple value series.



ORDERDATE	COUNTRY	STATUS	SUM (QUANTITYORDERED)	SUM (ORDERTOTAL)
Q4, 2004				
	USA			
		Shipped	163	$14,191.12
	Sub Total (USA)		*163*	*$14,191.12*
Sub Total (Q4, 2004)			*163*	*$14,191.12*
Grand Total			*1679*	*$146,827.12*

Figure 6-20 A summary table displaying a pie chart

Changing table types

In BIRT Studio, you can convert a summary table to a detail table, or a detail table to a summary table. When you change table types, the data in the original report table must be recreated by reinserting the selected data fields.

How to convert a summary table to a detail table

1 In BIRT Studio, choose Turn off Auto Summarize.

A message appears and informs you that if you choose to continue, all data in the current table will be lost, as shown in Figure 6-21. Choose OK.

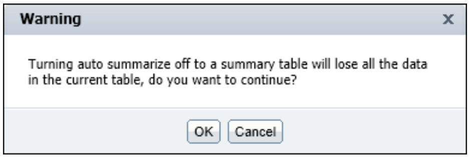


Figure 6-21 Warning dialog

2 Choose Insert to reinsert the selected fields to create a new report design. BIRT Studio displays the chosen data fields in a detail table.

How to convert a detail table to a summary table

1 In BIRT Studio, choose Turn on Auto Summarize.

A message appears and informs you that if you choose to continue, all data in the current table will be lost, as shown in Figure 6-22. Choose OK.

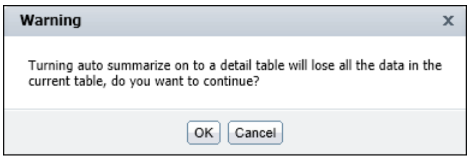


Figure 6-22 Warning dialog

2 Choose Insert to reinsert the selected fields to create a new report design. BIRT Studio displays the chosen data fields in a summary table.

Chapter

7

Presenting data in a cross tab

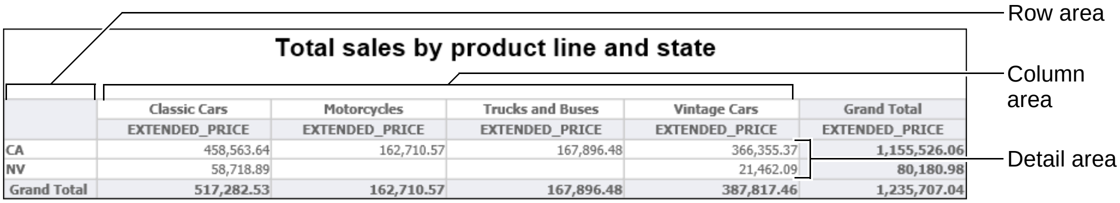
This chapter contains the following topics:

- About cross tabs
- Obtaining data for a cross tab
- Creating a cross tab
- Formatting data in a cross tab
- Analyzing data in a cross tab

About cross tabs

A cross tab displays data in a row-and-column matrix that is similar in appearance to a spreadsheet. The cross tab is ideal for summarizing data in a compact and concise format, and displays summary, or aggregate, values such as sums, counts, or averages. The cross tab groups these values by one set of data listed down the left side of the matrix and another set of data listed across the top of the matrix.

Figure 7-1 shows a cross tab that displays sales totals by state and by product line.



	Classic Cars	Motorcycles	Trucks and Buses	Vintage Cars	Grand Total
	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE
CA	458,563.64	162,710.57	167,896.48	366,355.37	1,155,526.06
NV	58,718.89			21,462.09	80,180.98
Grand Total	517,282.53	162,710.57	167,896.48	387,817.46	1,235,707.04

Figure 7-1 Displaying sales by state and product line in a cross tab

The cross tab uses data from the state, product line, and extended price fields. A cross tab typically uses data from at least three fields as follows:

- One field populates the column headings in the cross tab. There is one column for each unique value in the field. In the example, as shown in Figure 7-1, there are four unique values in the product line field: Classic Cars, Motorcycles, Trucks and Buses, and Vintage Cars. This area is called the column area.
- One field populates the row headings in the cross tab. There is one row for each unique value in the field. In the example, there are two unique values in the state field, CA and NV. This area is called the row area.
- BIRT Studio aggregates the values in one field and displays these values in the cross tab cells. In this example, each cell displays a revenue total by product line and state. BIRT Studio calculates the revenue total using the SUM function on the values in the extended price field. This area is called the detail area.

You can use BIRT Studio to insert a cross tab in a report design, select data for the cross tab, and specify the aggregate data to display. You can also define a bookmark for a cross tab, and then create a hyperlink from another report that links to the bookmarked cross tab element. For more information on creating bookmarks and hyperlinks, see Chapter 9, “Adding interactive viewing features.”

In addition, you can use Actuate BIRT Data Analyzer, which requires a separate license, to edit data in the cross tab, format the data values, and modify cross tab data. You can access Actuate BIRT Data Analyzer from BIRT Studio and Interactive Viewer.

Obtaining data for a cross tab

A cross tab typically derives data from a cube. A cube is a multidimensional data structure that is optimized for analysis and reporting. In BIRT Studio, a cross tab retrieves data from a cube in a data object. A data object typically contains at least one data set and can contain at least one cube, but can contain many data sets and many cubes. Ensure that you select a data object that contains one or more cubes to create the cross tab. When using BIRT Studio as a Java Component, you cannot use the data sets in a data object.

A cube organizes data into dimensions and measures as follows:

- Measures represent values that are counted or aggregated, such as costs or units of products.
- Dimensions are categories, such as products, customers, or sales periods, used to aggregate measures. Dimensions can be hierarchical and contain multiple levels. For example, a region dimension can contain a region-country-state hierarchy. Similarly, a time dimension can contain a year-quarter-month-week hierarchy. Most cubes include a time dimension because, for most reports, showing measures by day, week, month, quarter, or year is essential to analyzing data.

For example, a retail cube can contain data that supports viewing sales volume and cost of goods, which are measures, by store location, time period, and product lines, which are dimensions.

In a cross tab, the row and column areas display the dimensions from a cube. The dimension values form the row and column headings of the cross tab. The detail area contains one or more measures from a cube, displaying the aggregate data.

Creating a cross tab

In BIRT Studio, complete the following tasks to create a cross tab:

- Select data for the cross tab.
- Lay out the cross tab.
- Specify display options for the aggregate values.

The following section describes each of these tasks in detail.

Selecting data for a cross tab

When building a cross tab, you group the aggregate data by at least two dimensions, for example, sales totals by year and product line or sales totals by product line and state. Often, report users need to view aggregate data by more than two dimensions.

Figure 7-2 shows data selected for a cross tab displaying the sales total by state and product line. To create the cross tab, one dimension (State) is inserted in the row area, and one dimension (Productline) is inserted in the column area.

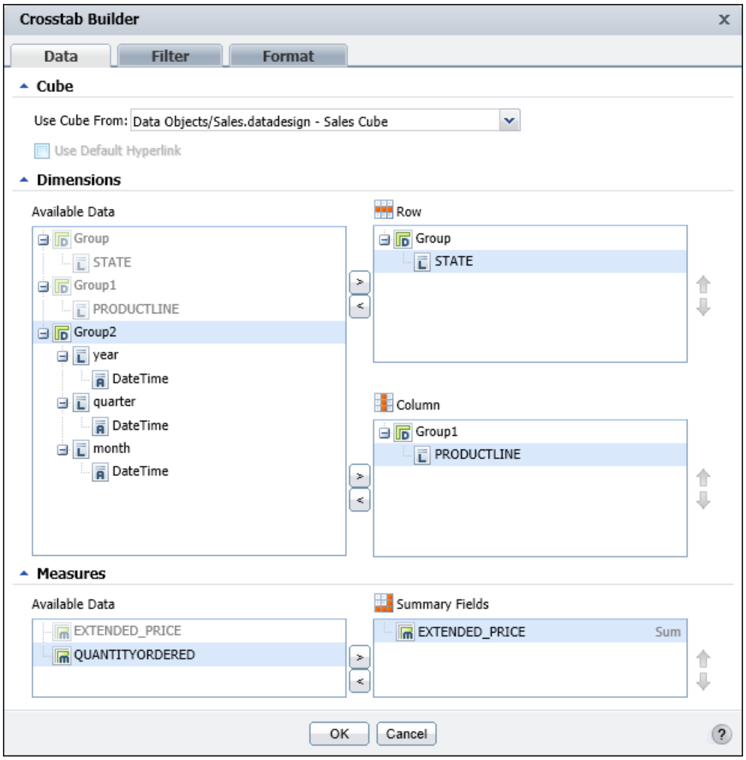


Figure 7-2 Displaying the dimensions and measures for a cross tab

Each additional dimension by which you group data appears as a column or row, and each additional dimension provides a more comprehensive and detailed view of the data. Just as you can define an unlimited number of dimensions for a cube, you can build a cross tab that displays aggregate data by as many dimensions as needed.

Calculating aggregate data by too many dimensions, however, can result in many empty cells, a problem commonly referred to as data sparsity. When designing a cross tab that contains more than two dimensions, make sure that report processing time is not spent calculating zeros. In addition, a cross tab that contains more than two or three dimensions in either the row or column area is difficult to read. Rather than displaying data by too many dimensions in a single cross tab, consider dividing the data into multiple cross tabs, so the data is easier for the user to understand quickly.

How to select data for a cross tab

1 In BIRT Studio, choose New. Report Template appears.

2 In Report Template, complete the following steps:

   1 In Category, select Standard from the drop-down list.

   2 In Available Templates, select Crosstab, as shown in Figure 7-3.

   3 In Themes, select a theme from the drop-down list if your Crosstab template contains associated themes.

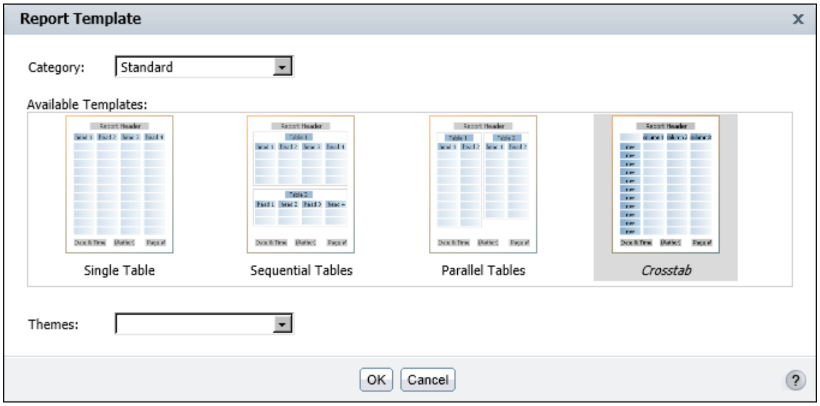


Figure 7-3 Choosing the cross tab template

Choose OK. Crosstab Builder—Data appears.

3 In Cube, in Use Cube From, choose New Data from the drop-down list, as shown in Figure 7-4.

   Select Data appears, displaying the available data objects. If you use customized data objects, ensure that they are in the following folder location of the Actuate BIRT Java Components installation:

```
<context root>\WEB-INF\repository
```

4 In Available Data, select a data object design or data object store file that contains a cube.

   You can select a data object design (.datadesign) file to retrieve data on demand, or a data object store (.data) file to use cached data. Select one or more data objects from the list, and choose the + symbol next to each selection, as shown in Figure 7-5.

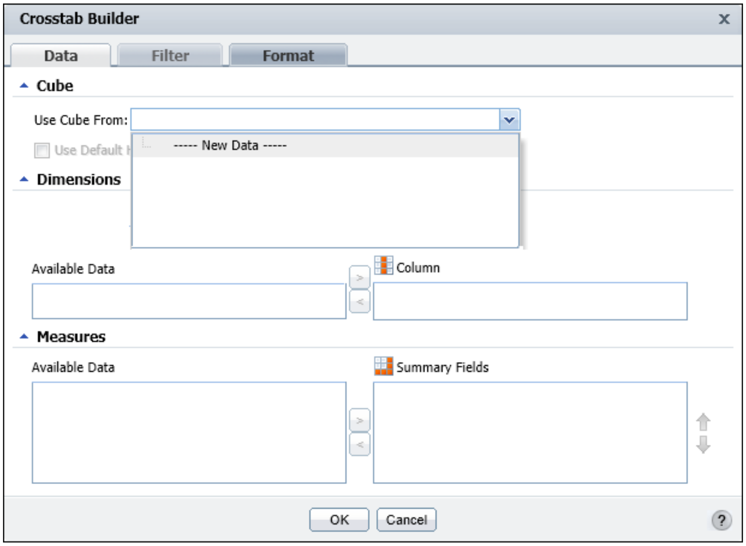


Figure 7-4 Selecting a data object data source

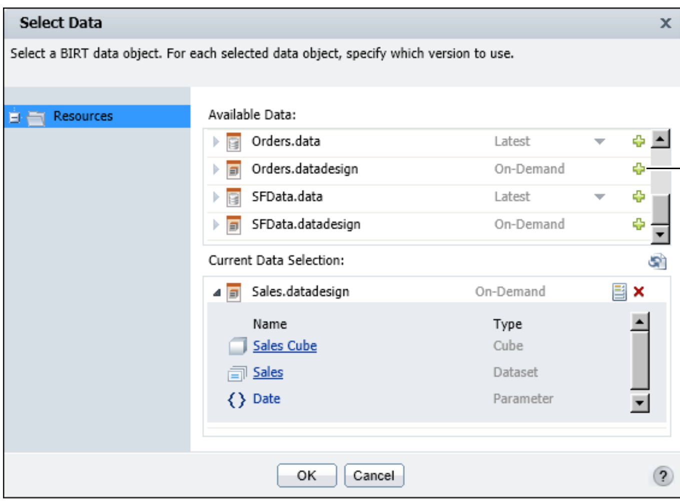


Figure 7-5 Selecting a data object

The data object appears in Current Data Selection, as shown in Figure 7-6. Choose OK. You can modify your selection by choosing Delete, as shown in Figure 7-6, and selecting a new data object in Available Data.

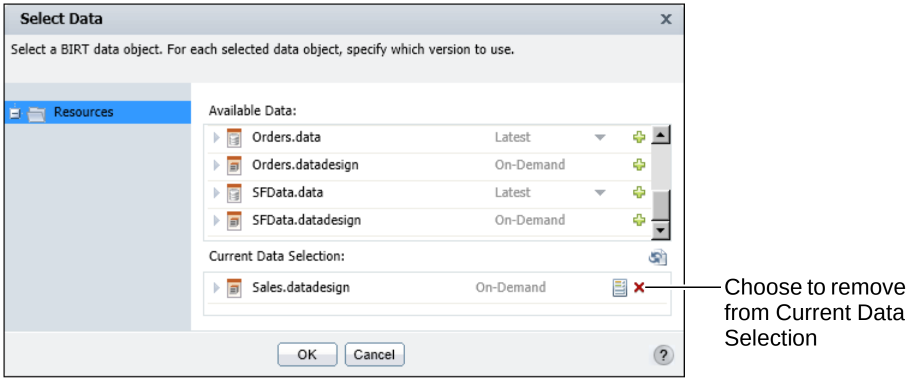


Figure 7-6 Current Data Selection displaying the data object

Crosstab Builder—Data displays the data from the selected data object, organized in dimensions and measures, as shown in Figure 7-7.

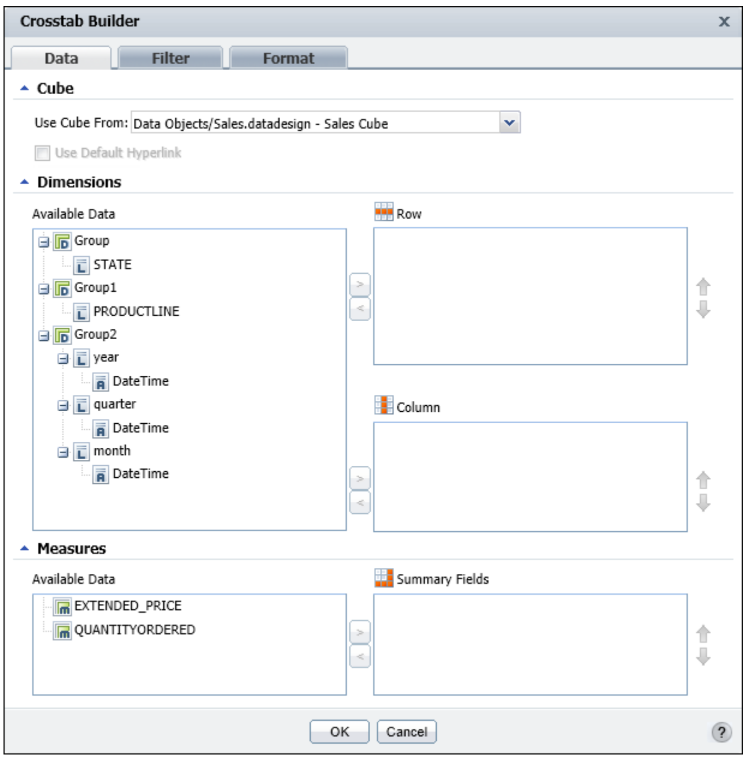


Figure 7-7 Displaying dimensions and measures in the data object

5 To display a subset of the available data in the cross tab, you can specify a filter condition for the data object. To create a filter at the data object level, choose Filter and specify one or more filter conditions, as you would do for a BIRT report. For example, to view data only for Pennsylvania (PA) create a filter on the state dimension, using the Equal To operator, as shown in Figure 7-8.

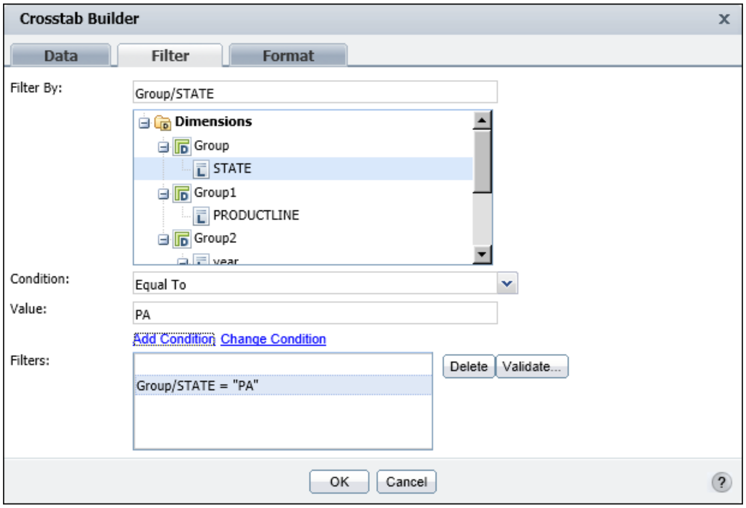


Figure 7-8 Specifying a data object filter

The cross tab displays aggregate extended price data by product line, for only the state of Pennsylvania, as shown in Figure 7-9.

	Classic Cars	Motorcycles	Vintage Cars	Grand Total
	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE
PA	102,856.24	39,025.09	34,925.01	**214,289.43**
Grand Total	**102,856.24**	**39,025.09**	**34,925.01**	**214,289.43**

Figure 7-9 A cross tab displaying filtered data rows

You can also select the Any Of operator in Condition to set a filter condition for more than one value. For example, you can define a filter condition that displays data for Pennsylvania (PA), California (CA), and Massachusetts (MA) in the cross tab.

Laying out data in a cross tab

When you lay out data in a cross tab, provide the following information:

- In Dimensions, specify:
  - The grouped data fields to display as rows
  - The grouped data fields to display as columns
- In Measures, specify the summary fields to display in the cross tab.

How to lay out data in a cross tab

1. In Crosstab Builder—Data, in Dimensions, in Available Data, select a grouped data field, as shown in Figure 7-10. You can select multiple grouped data fields by pressing Ctrl while you select each field.
2. Choose the right arrow next to Row to set the selected field as a row, or choose the right arrow next to Column to set the selected field as a column. The example, as shown in Figure 7-10, specifies State as a row, and Product Line as a column.
3. In Measures, in Available Data, select a data field to aggregate, then choose the right arrow to set the selected field as a summary field. The example in Figure 7-10 sets the Extended Price field as the summary field.

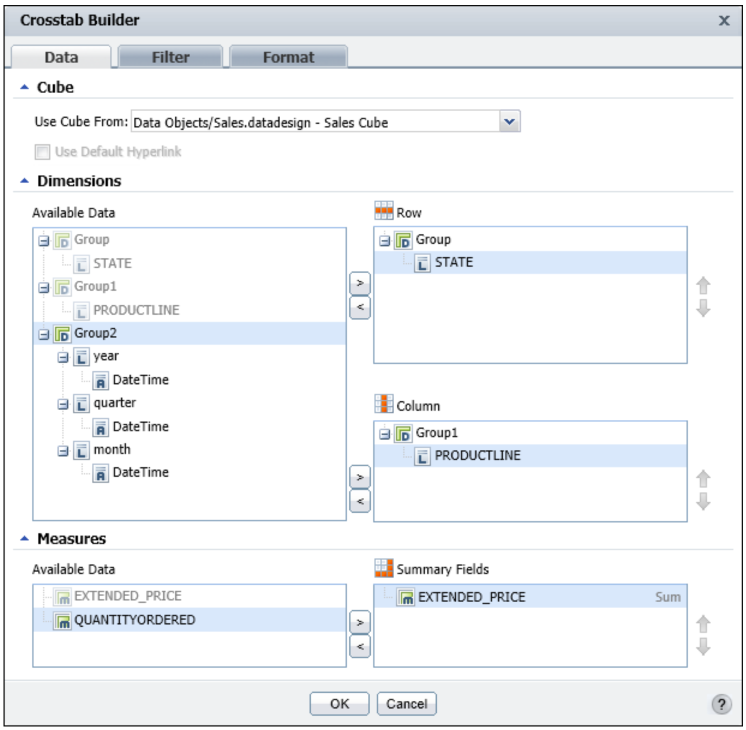


Figure 7-10 Selecting data to display in the cross tab

Choose OK. The selected fields appear in a cross tab in BIRT Studio.

How to provide data from multiple cubes to a cross tab

Choose Manage Data to select a different data object data source. Manage Data appears. Complete the following steps:

In Available Data, choose the + symbol to add a new data object. Repeat this action to make additional cubes available to the cross tab. Choose OK.

The data object you selected appears in Current Data Selection, as shown in Figure 7-11. The selected cubes now appear in Use Cube From on Crosstab Builder. You can now use any of the cubes as a data source for the cross tab.

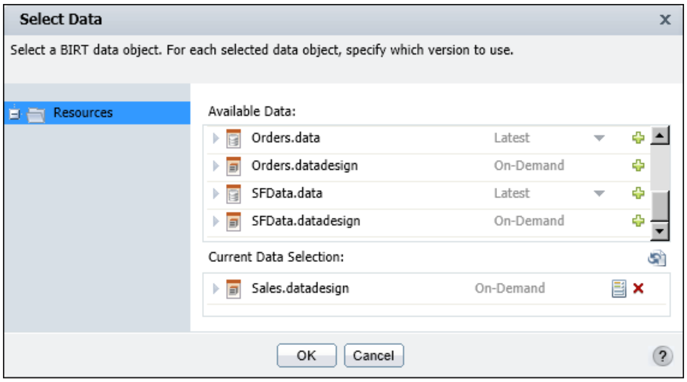


Figure 7-11 Adding a cube to Current Data Selection

Displaying aggregate values

You can display totals for each dimension that you add to a cross tab and for each level within a multilevel dimension. The cross tab in Figure 7-10 displays sales data by product line and state. The rows and columns that display the subtotals and grand totals are highlighted in the cross tab. In the example cross tab you just created, no subtotals appear. Subtotals are displayed when you set a multidimensional data field, as a row or column.

Each number displayed in a cross tab represents an aggregate total. Typically grand totals display the total sales of all products for each state, the total sales of each product, or the total of all sales across products and states. Subtotals display the sales of each product in each state, and so on. BIRT Studio does not create the subtotal and grand total rows and columns by default. You specify the aggregate totals that appear in a cross tab.

How to display aggregate values in a cross tab

1. To specify display properties for the aggregate values in the cross tab, in Crosstab Builder, choose Crosstab Builder—Format, as shown in Figure 7-12.
2. In Grand Totals, do the following:
   - Select Show Grand Totals for Rows to display grand totals for each row.
   - Select Show Grand Totals for Columns to display grand totals for each column.

3 You can display subtotals for multilevel dimensions used in rows or columns. Because the example in Figure 7-10 does not contain multilevel dimensions, the Sub Totals section is not highlighted. In Sub Totals, if highlighted, do the following:

- Select Show Sub Totals for Rows.
- Select Show Sub Totals for Columns.

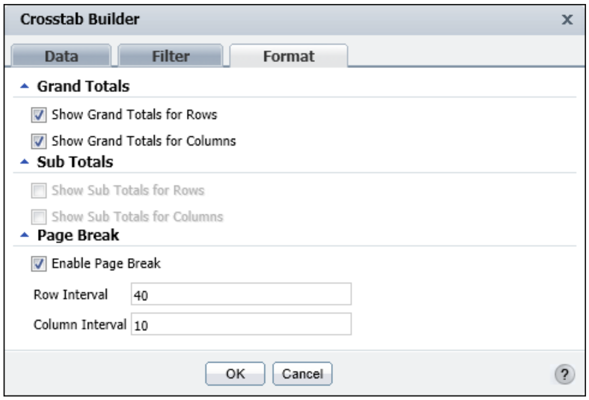
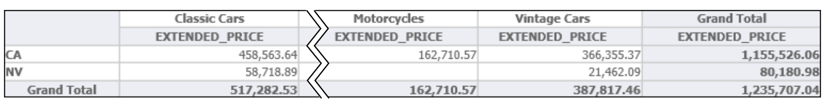


Figure 7-12 Specifying display options for the aggregate values

4 In Page Break, select Enable Page Break to specify pagination properties for the cross tab, do the following:

- In Row Interval, type a value, such as 40, as shown in Figure 7-12.
- In Column Interval, type a value, such as 10, as shown in Figure 7-12.

Choose OK. Sample data appears in the cross tab, displaying grand totals for rows and columns, as shown in Figure 7-13.



	Classic Cars	Motorcycles	Vintage Cars	Grand Total
	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE
CA	458,563.64	162,710.57	366,355.37	1,155,526.06
NV	58,718.89		21,462.09	80,180.98
Grand Total	517,282.53	162,710.57	387,817.46	1,235,707.04

Figure 7-13 Displaying grand totals for rows and columns in a cross tab

Saving and viewing a cross tab

As with other report design files, BIRT Studio displays only a preview of the actual data in the cross tab. You save and view a cross tab in the same way that you save and view a BIRT report design, as described in earlier sections of this document.

How to save and view a cross tab

1 In BIRT Studio, choose Save and view.

2 Save Report Design appears. Specify a file name, and a folder location in which to store the saved file. Choose OK.

The cross tab appears in Actuate BIRT Viewer, as shown in Figure 7-14.

	Classic Cars	Motorcycles	Vintage Cars	Grand Total
	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE
CA	458,563.64	162,710.57	366,355.37	**1,155,526.06**
CT	89,671.28	39,699.67	14,101.07	**159,143.51**
MA	223,366.59	91,024.09	105,384.18	**478,262.84**
NH	69,150.35		39,376.65	**116,449.29**
NJ		35,116.44	9,035.36	**44,151.80**
NV	58,718.89		21,462.09	**80,180.98**
NY	260,619.73	99,514.87	62,342.28	**500,472.88**
PA	102,856.24	39,025.09	34,925.01	**214,289.43**
Grand Total	**1,262,946.72**	**467,090.73**	**652,982.01**	**2,748,476.79**

Figure 7-14 Viewing the cross tab in Actuate BIRT Viewer

Formatting data in a cross tab

As the examples in this document reflect, the data in a cross tab appears as it is stored in the cube. To customize formatting properties for a cross tab, you can use Interactive Viewer. You can use Interactive Viewer to complete the following tasks in a cross tab:

- Display a detail tooltip or a description of the aggregate data values.
- Modify font type, size, color, and style of data.
- Modify alignment properties of text.
- Format data based on its type, for example, format currency as US dollars, as shown in Figure 7-15.

	Classic Cars	Trucks and Buses	Vintage Cars	Grand Total
	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE	EXTENDED_PRICE
CA	$458,563.64	$167,896.48	$366,355.37	**$1,155,526.06**
CT	$89,671.28	$15,671.49	$14,101.07	**$159,143.51**
MA	$223,366.59	$58,487.98	$105,384.18	**$478,262.84**
NH	$69,150.35	$7,922.29	$39,376.65	**$116,449.29**
NJ			$9,035.36	**$44,151.80**
NV	$58,718.89		$21,462.09	**$80,180.98**
NY	$260,619.73	$77,996.00	$62,342.28	**$500,472.88**
PA	$102,856.24	$37,483.09	$34,925.01	**$214,289.43**
Grand Total	**$1,262,946.72**	**$365,457.33**	**$652,982.01**	**2,748,476.79**

Figure 7-15 Cross tab displaying applied currency format

- Launch BIRT Data Analyzer to analyze cross tab data.
- Export data and charts to other file formats.

Analyzing data in a cross tab

From BIRT Studio, you can launch BIRT Data Analyzer if you have purchased a license for it, to manipulate and further analyze cross tab data. Using BIRT Data Analyzer, you can answer questions such as:

- What the sales for a product are over time
- How total sales break down by product
- How many units of a product shipped to all locations in April

You can add, remove, reorganize, and customize the data and its appearance in the cross tab to examine relationships and trends. Using BIRT Data Analyzer, you can perform the following tasks:

- Filter, group, and sort data.
- Format a cross tab.
- Perform calculations.

To launch BIRT Data Analyzer, select any cell in the cross tab, then choose Analyze from the context menu, as shown in Figure 7-16.

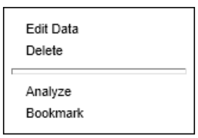


Figure 7-16 Launching BIRT Data Analyzer from BIRT Studio

Chapter

8

Presenting data in a chart

This chapter contains the following topics:

- About charts
- Creating a chart
- Inserting a chart
- Choosing a chart type
- Selecting data for a chart
- Formatting a chart
- Working with Flash charts
- Working with HTML5 charts
- Deleting a chart
- Displaying a chart without the table data

About charts

A chart is a graphical representation of data. Charts are particularly useful for summarizing numeric data and showing the relationship between sets of values, called series. For example, a chart can show sales by region, average temperatures by month, or the price of a stock over three months.

Because a chart presents a picture, it reveals trends that are not as apparent in a table. Figure 8-1 shows an example of a table and a bar chart that display sales data. The chart shows instantly the sales trend for the year.

Month	Sales
January	29000
February	25000
March	42700
April	33250
May	40500
June	45000
July	44000
August	35000
September	42000
October	37000
November	35000
December	22000

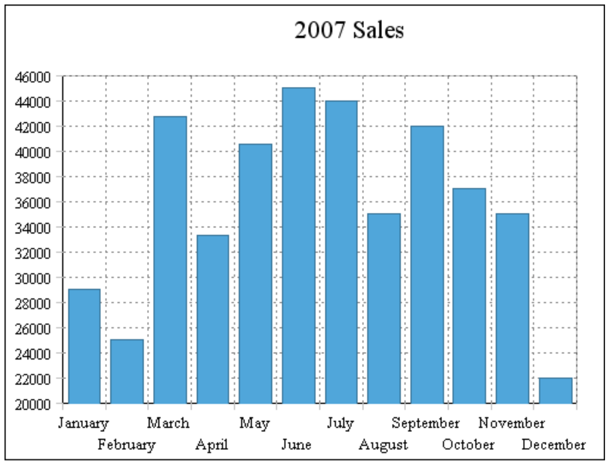


Figure 8-1 Presenting sales data in a table and in a chart

Figure 8-2 shows an example of a pie chart that displays profits by product line. The table in Figure 8-3 provides the data for the pie chart. Typically reports display detail data in a table and summary data in a chart, especially if the detail data spans multiple pages.

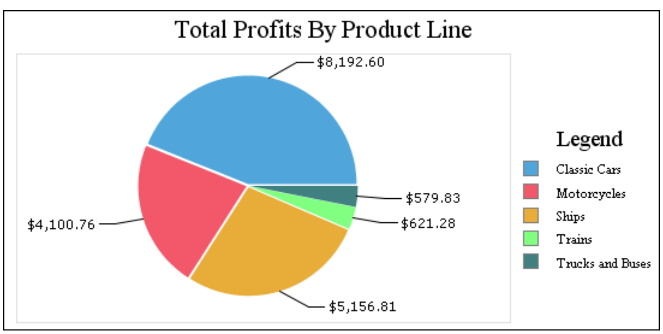


Figure 8-2 Displaying profits by product line in a pie chart

Sales Office	Product Line		Product Name	Total	Profit
San Francisco					
	Classic Cars				
			1970 Chevy Chevelle SS 454	$2,970.40	$656.12
			1970 Plymouth Hemi Cuda	$1,742.88	$976.80
			1970 Triumph Spitfire	$2,929.92	$723.84
			1971 Alpine Renault 1600s	$1,405.35	$363.69
			1976 Ford Gran Torino	$5,864.88	$2,778.30
			1982 Lamborghini Diablo	$687.20	$362.40
			2001 Ferrari Enzo	$5,485.92	$2,331.45
	Sum	*Classic Cars*		*$21,086.55*	*$8,192.60*
	Motorcycles				
			1974 Ducati 350 Mk3 Desmo	$2,755.50	$1,071.60
			1982 Ducati 900 Monster	$1,948.22	$582.32
			1982 Ducati 996 R	$1,274.46	$477.84
			2002 Yamaha YZR M1	$3,472.48	$1,969.00
	Sum	*Motorcycles*		*$9,450.66*	*$4,100.76*
	Ships				
			Pont Yacht	$1,831.44	$532.74
			The Mayflower	$3,314.44	$1,452.54
			The Queen Mary	$3,472.06	$1,434.12
			The Schooner Bluenose	$2,268.94	$874.94
			The Titanic	$1,935.36	$862.47
	Sum	*Ships*		*$12,822.24*	*$5,156.81*
	Trains				
			Collectable Wooden Train	$2,107.60	$621.28
	Sum	*Trains*		*$2,107.60*	*$621.28*
	Trucks and Buses				
			1996 Peterbilt 379 Stake Bed with Outrigger	$1,352.86	$579.83
	Sum	*Trucks and Buses*		*$1,352.86*	*$579.83*
Sum San Francisco				*$46,819.91*	*$18,651.28*
Sum				*$46,819.91*	*$18,651.28*

Figure 8-3 The table that provides the data for the pie chart

About chart technologies

BIRT Studio supports three charting technologies, static BIRT charts, and animated Flash and HTML5 charts. Each technology supports commonly used chart types such as column, bar, pie, line, and area charts. Standard charts are static images whereas Flash and HTML5 charts add motion and provide more visual interest for users. For example, an animated Flash column chart or HTML5 chart can progressively draw its columns from the bottom to the top.

BIRT Studio Release 11 Service Pack 4 introduces a new chart format built on HTML5 technology. HTML5 is an open standard for structuring and presenting content for the World Wide Web, and is increasingly regarded as the alternative to Flash for creating interactive and animated content for traditional and mobile devices.

Software requirements

You can view BIRT charts and HTML5 charts without installing additional software, across operating systems on mobile devices.

Although both Flash charts and HTML5 charts provide animated charts for users to view and interact with, users must first install Flash Player to view Flash charts. Flash Player installs as an ActiveX control or browser plug-in, available at the following location:

```
http://www.adobe.com/products/flashplayer
```

Support for Flash charts on mobile devices is limited to operating systems that support Flash. Additionally, Adobe has discontinued further development of Flash for mobile devices.

Supported output formats

Using Actuate BIRT Viewers, you can export BIRT charts and HTML5 charts to the maximum number of output formats. Supported output formats include HTML, PostScript, PDF, DOC, XLS, PPT, XHTML, and AFP formats. To preserve the animation properties of a Flash chart, export it to HTML or PDF output formats. To view Flash content in a PDF file, download a version of Adobe Reader that supports Flash from the following location:

```
http://www.adobe.com/acrobat
```

When exporting a Flash chart to other formats, such as PostScript, Excel, Word, or XHTML, you can choose to export the chart as a static image. When the chart is viewed in any of these formats, it does not appear animated.

For more information on exporting charts, see *Working with Actuate BIRT Viewers*.

Choosing a chart technology

Table 8-1 provides information that can help you decide which chart format to use in a report.

Table 8-1 Features available in BIRT, Flash, and HTML5 charts

Feature	BIRT	Flash	HTML5
Displays in the Actuate BIRT viewers	✓	✓	✓
Displays in PDF	✓	✓	✓
Displays in other document formats, such as DOC, PPT, XLS, as a static image	✓		✓
Supported on mobile devices	✓	limited	✓
Provides animation		✓	✓

BIRT charts and HTML5 charts support the maximum number of chart types in BIRT Studio. If you are creating a meter chart, or a stock chart, you must choose to create a BIRT chart, because these type of charts are not supported by Flash or HTML5 technologies. To select a chart type that is available in both Flash and

HTML5 technologies, such as doughnut or pie charts, selecting HTML5 is an advantage for users who want to export the chart to other output formats, or view the chart on a mobile device.

In later releases, BIRT Studio will eventually use HTML5 charts to replace BIRT and Flash charts. If you select a chart type that is available in all three technologies, such as column, bar, or line charts, use the HTML5 version of the chart.

Creating a chart

The data for a chart comes from the data in a table. You must create the table before you create a chart. Users typically design reports using this sequence. To display only a chart in your report, hide the table after you create the chart.

When you create a chart, you perform the following tasks:

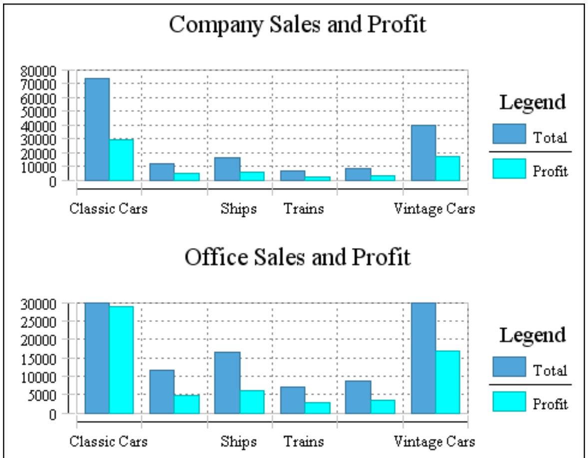
1. Insert a chart.
2. Choose a chart type.
3. Specify the data to present in the chart.
4. Format the chart.

Inserting a chart

The location in which you insert a chart determines what data the chart displays. You can insert a chart in the following locations:

- A table. A chart at the table level summarizes data for the entire table, and the chart appears at the top of every page.
- A section header, if the report organizes data in sections. A chart at the section header level summarizes data for the section, and the chart appears at the top of each section. If a section's data spans multiple pages, the chart appears on every page.

If your report design organizes data in groups instead of sections, you can create only one chart per table. Figure 8-4 shows one page of a 76-page report that displays a chart at the table level and a chart at the level of the sales office section.



Sales Office		San Francisco	
Product Line		Classic Cars	
Order Number	Product Name	Total	Profit
10135	1948 Porsche 356-A Roadster	$3,215.52	$628.32
10182	1949 Jaguar XK 120	$3,140.64	$1,439.64
10135	1956 Porsche 356A Coupe	$6,534.41	$1,914.31
10135	1957 Corvette Convertible	$4,151.52	$1,983.69
10182	1957 Ford Thunderbird	$2,696.76	$1,191.52

Figure 8-4 Placing charts in different locations in a report design

The upper chart, inserted at the table level, displays sales and profit totals for the company. The lower chart, inserted at the sales office section level, displays sales and profit totals for the Tokyo office.

How to insert a chart

To insert a chart in the report, complete the following tasks:

- To insert a chart at the table level, select the table, then choose Chart.
- To insert a chart at the table header level, select a column header in the table header row, then choose Chart. If your report design contains sections, selecting a column header does not enable Chart on the toolbar. You must first delete the existing section or sections, select a column header and insert the chart, then recreate the section or sections.
- To insert a chart at a section level, select the section heading, then choose Chart.

After you insert a chart, BIRT Studio displays the chart editor, as shown in Figure 8-5. You use the chart editor to choose a chart type, select data for the chart, and format the chart.

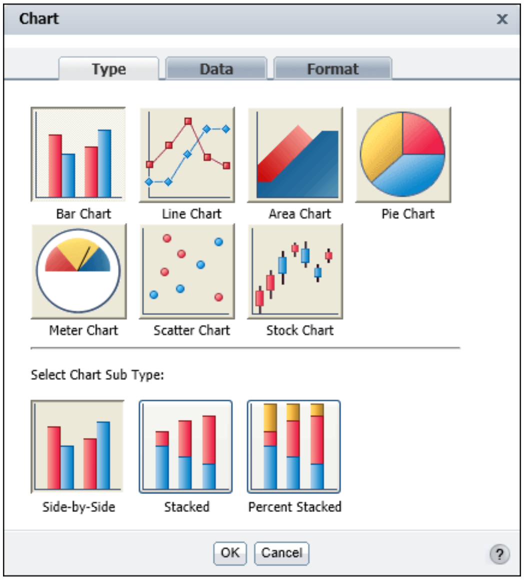


Figure 8-5 The chart editor displaying the Chart—Type page

Choosing a chart type

BIRT Studio provides a wide array of chart types that you can use in a report design. After you insert a chart, the first step is to choose the most suitable chart type for the data. You can use any chart type to display most data, but the data is easier to read when you choose a chart type that has the best visual characteristics for a particular set of data.

For example, to show what percentage each product line contributes to a company's total sales, use a pie chart, which is ideal for showing how parts relate to a whole. To compare the sales of each product line in the current year and the previous year, use a bar chart, which supports side-by-side comparisons. To show how each product line has been selling over the course of five years, a line chart is appropriate for displaying trends in linear data.

Another item to consider when choosing a chart type is the number of data values to display. Some charts, such as scatter charts, reveal trends more clearly when there are more data values. Other charts, such as pie charts, are more effective when there are fewer data values. For example, the pie chart in Figure 8-6 is difficult to read, because it displays too many data values.

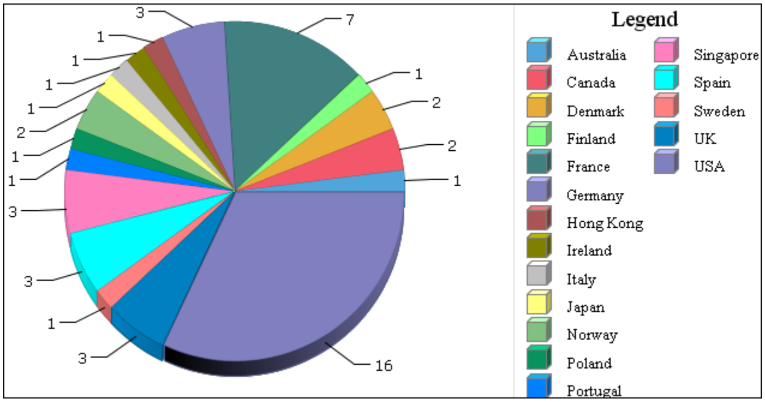


Figure 8-6 A pie chart that displays too many data values

The following sections describe the chart types BIRT Studio supports. Several of the chart types include subtypes.

About area charts

An area chart displays data values as a set of points, connected by a line, with the area below the line filled. You typically use an area chart to present data that occurs over a continuous period of time. There are three types of area charts, described in the following sections.

Stacked area chart

In a stacked area chart, multiple series are stacked vertically, as shown in Figure 8-7. The example shows that the stacked area chart is suitable for the data because the chart displays totals for all series as well as the proportion that each series contributes to the total. The height of the top line shows the total value for each quarter. Each shaded area represents the sales amount for a specific region.

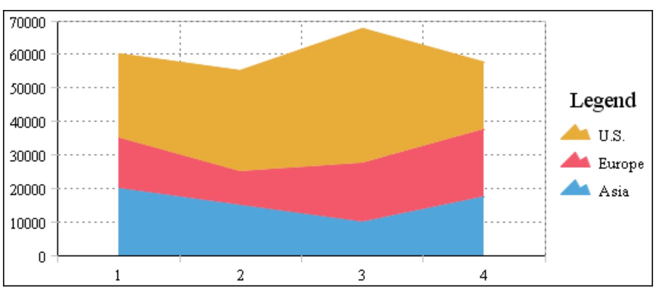


Figure 8-7 A stacked area chart

Overlay area chart

In an overlay area chart, the areas of multiple series overlap, as shown in Figure 8-8. As the example shows, this chart subtype is not suitable for showing multiple series if the data values overlap. In the example, the data for the U.S. obscures the data for Europe and Asia, because the U.S. numbers are the highest for every quarter. Use the overlay area chart if you want to show only one series, for example, only sales for Asia.

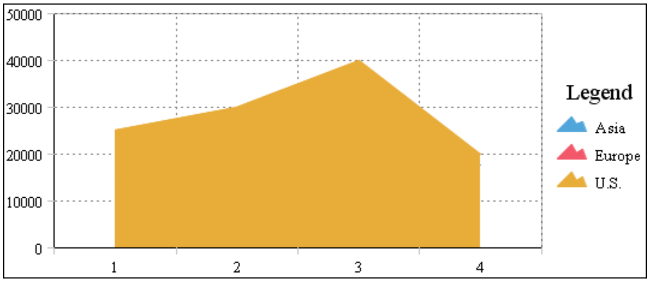


Figure 8-8 An overlay area chart

Percent stacked area chart

In a percent stacked area chart, multiple series are stacked vertically, and the values appear as a percentage of the whole. As Figure 8-9 shows, the sales values are displayed in percentages, instead of the actual numbers, as shown in the previous area charts.

The percent stacked area chart is meaningful only when displaying and comparing multiple series. If displaying only one series, the percent stacked area chart looks like the example in Figure 8-10. The sales percentage of one region, compared to the whole, is 100%.

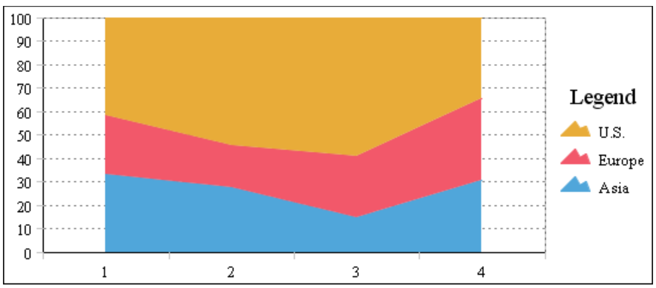


Figure 8-9 A percent stacked area chart displaying three series

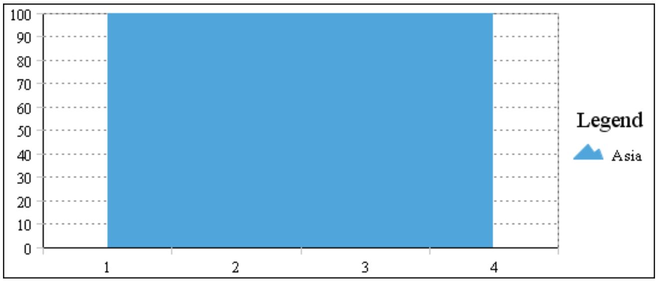


Figure 8-10 A percent stacked area chart displaying one series

About bar charts

A bar chart, by default, displays data values as a set of vertical bars, but you can transpose the axes to display horizontal bars. A bar chart is useful for displaying data side-by-side for easy comparison. There are three subtypes of bar charts. Two of the subtypes, stacked bar chart and percent stacked bar chart, are functionally similar to the stacked area chart and percent stacked area chart subtypes.

Side-by-side bar chart

In a side-by-side bar chart, multiple series appear as side-by-side bars, as shown in Figure 8-11. This bar chart uses the same data as the area charts, as shown in the earlier sections.

Stacked bar chart

In a stacked bar chart, multiple series are stacked vertically, as shown in Figure 8-12. The stacked bar chart shows totals for each category, each quarter in this example, as well as the proportion that each series contributes to the total.

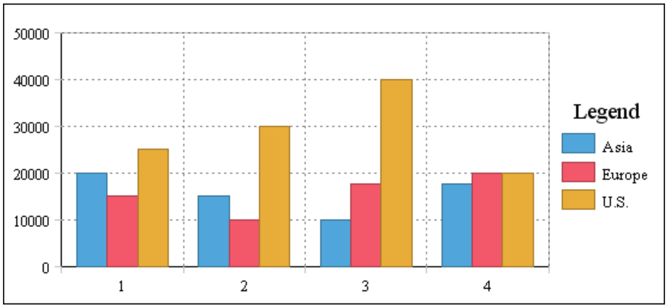


Figure 8-11 A side-by-side bar chart

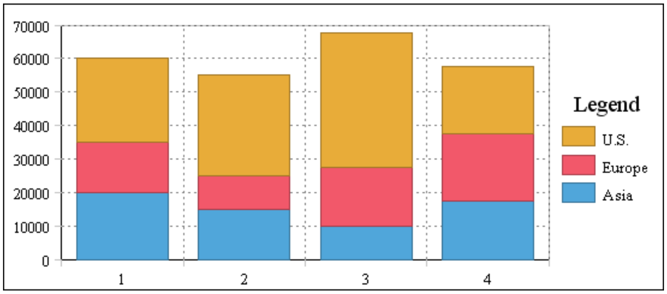


Figure 8-12 A stacked bar chart

Percent stacked bar chart

In a percent stacked bar chart, multiple series are stacked vertically, and the values are shown as a percentage of the whole. As you can see in Figure 8-13, the sales values are shown in percentages instead of the actual numbers, as shown in the previous bar charts.

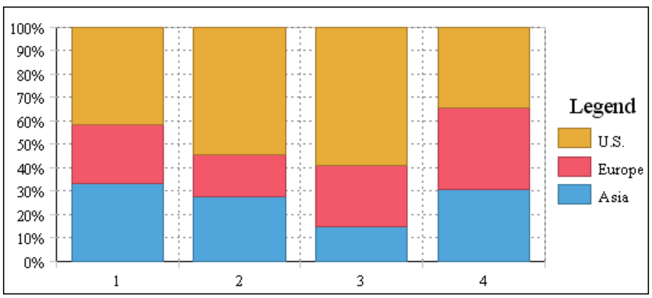


Figure 8-13 A percent stacked bar chart displaying three series

Like the percent stacked area chart, the percent stacked bar chart is meaningful only when displaying and comparing multiple series. Do not use this chart subtype if you are displaying only one series, for example, only sales for Asia. Figure 8-14 shows how a percent stacked bar chart looks when it displays only one region series. All the bars show a value of 100%, because that is the sales percentage of one region when it is compared to itself.

Figure 8-14 A percent stacked bar chart displaying one series

Horizontal bar chart

For all the bar chart subtypes, you can transpose, or flip, the axes to display data as horizontal bars. Figure 8-15 shows an example of a side-by-side bar chart with flipped axes. This feature is useful for showing data that contains many categories. For example, if you want to show sales numbers for 5 regions for 12 months, displaying the data requires 60 bars, which is crowded for a side-by-side bar chart that displays data as vertical bars.

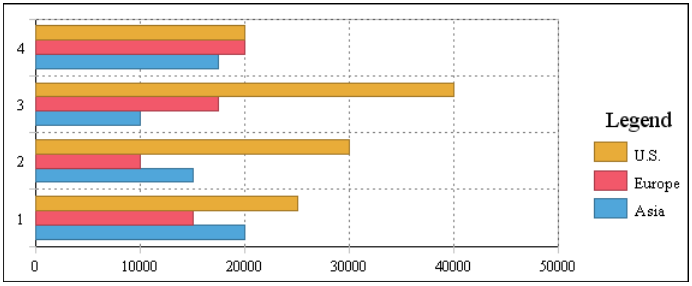


Figure 8-15 A horizontal bar chart

About line charts

A line chart displays data values as a set of points that are connected by a line. You typically use line charts to present large amounts of data that occur over a continuous period of time. A line chart is the most basic type of chart in finance. Figure 8-16 shows an example of a line chart that shows the value of the Euro against the US Dollar over 10 days.

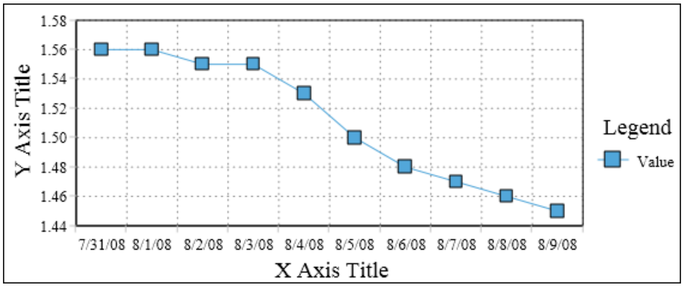


Figure 8-16 A line chart

A line chart is similar to an area chart, except that the line chart does not fill in the area below the line, and it uses a square marker for each data value.

There are three subtypes of line charts, which are functionally similar to the area chart and bar chart subtypes. The line chart subtypes in the following sections use the same data as the area charts and bar charts in earlier sections. You can compare how different chart types present the same set of data.

Overlay line chart

In an overlay line chart, multiple series appear as overlapping lines, as shown in Figure 8-17. A square marker indicates each data value.

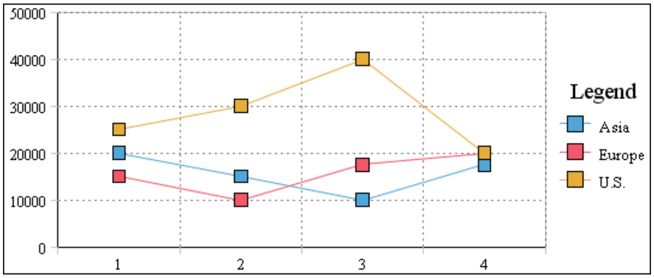


Figure 8-17 An overlay line chart

Stacked line chart

In a stacked line chart, multiple series are stacked vertically, as shown in Figure 8-18. The stacked line chart shows totals for each series, as well as the proportion that each series contributes to the grand total. In the example, the top line shows the total sales amounts for each quarter. The empty area between each line represents the sales amount for a region.

The stacked line chart is not as effective as the stacked area chart for comparing the amount that each region series contributes to the total, because the areas are not filled. In a stacked area chart, the filled-in areas provide a clear visual cue that each part is compared to the whole.

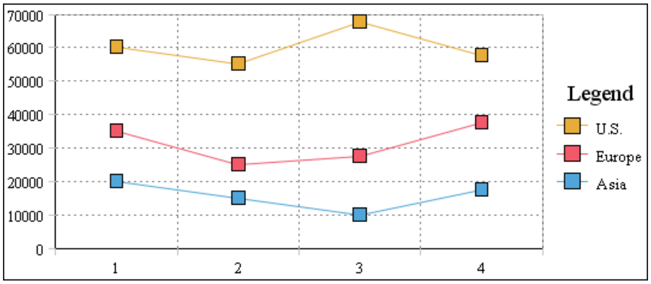


Figure 8-18 A stacked line chart

In addition, as the example shows, a user can easily misinterpret the data in a stacked line chart. There is no obvious indication that the top line shows the total sales amount for each quarter, and the middle line shows the difference in the sales amount between Europe and Asia. A user looking at this chart can mistakenly think that the top line represents the sales data for the U.S., the middle line represents the sales data for Europe, and the bottom line represents the sales data for Asia.

Percent stacked line chart

In a percent stacked line chart, multiple series are stacked vertically and the values are shown as a percentage of the whole. As shown in Figure 8-19, the sales values appear in percentages instead of numbers. Like the percent stacked area chart, the percent stacked line chart makes sense only when displaying and comparing multiple series. Do not use this chart subtype if you are displaying only one series, for example, only sales for Europe.

Like the stacked line chart, the percent stacked line chart is not as effective as its counterpart, the percent stacked area chart, for the reasons cited in the previous section.

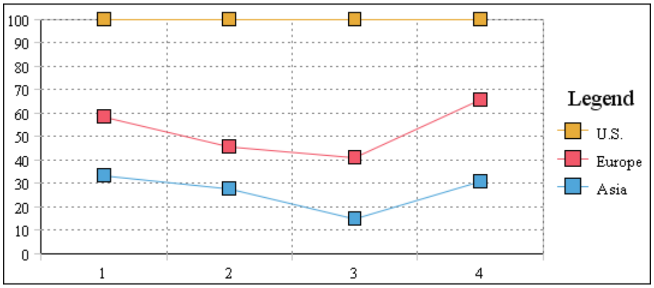


Figure 8-19 A percent stacked line chart

About meter charts

A meter chart displays a value as a needle pointer on a semicircle, called a dial. As Figure 8-20 shows, a meter chart resembles a speedometer, with tick marks and numbers showing a range of values. In this example, the meter chart displays two pointers that represent two values: projected sales total and actual sales total. Use a meter chart to display a small set of values.

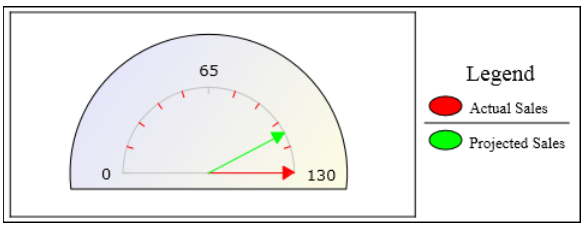


Figure 8-20 A meter chart

Superimposed meter chart

A superimposed meter chart displays multiple values in a single dial, as shown in Figure 8-20. Use the superimposed meter chart type when there are few values to display and when each value is distinct. Duplicate values result in overlapping needles.

Standard meter chart

A standard meter chart displays multiple values in multiple dials, where each dial displays a single value. The meter chart, as shown in Figure 8-21, displays the same data as the chart in Figure 8-20, using two dials instead of one.

The standard meter chart typically is used to create a dashboard effect, which can be visually compelling.

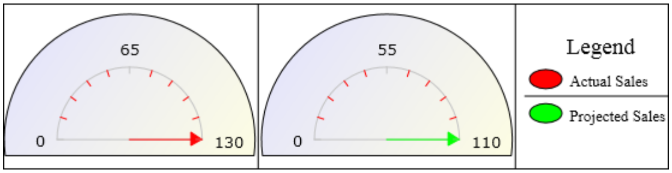


Figure 8-21 A standard meter chart displaying values in two dials

About pie charts

A pie chart is a circular chart that is divided into sectors or slices. Each sector represents a value that is proportional to the sum of the values. Use a pie chart when you want to show the relationship of parts to the whole, for example, the amount each product line contributes to a company's total sales, as shown in Figure 8-22.

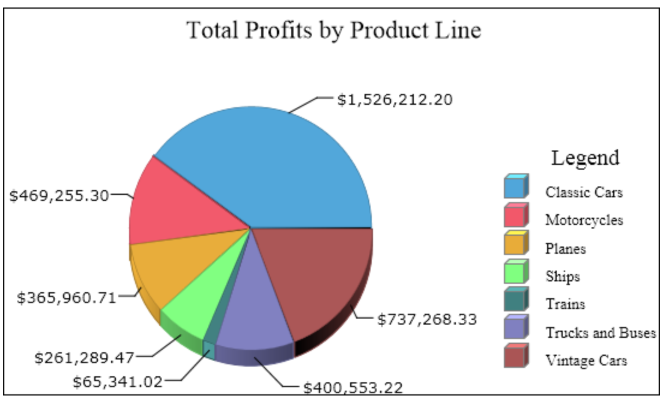


Figure 8-22 A pie chart

About scatter charts

A scatter chart presents data as *x-y* coordinates by combining two sets of numeric values into single data points. A scatter chart typically is used to display scientific and statistical data, because it shows if there is a relationship between two sets of measurements. Use a scatter chart to compare, for example, salaries and years of experience, weight and body fat, rainfall amounts and pollen levels, or test scores and hours of study. The more data values you include in a scatter chart, the clearer the trends the data reveals.

The scatter chart in Figure 8-23 shows the relationship between salary and years of experience. Each pair of values, salary and years of experience, is plotted as a single *x-y* value. The chart reveals a positive, or direct, relationship between salary and years of experience. As the number of years increases, the salary also increases. The chart also displays the salaries earned by men and women. In this example, the chart reveals that men consistently earn more than women.

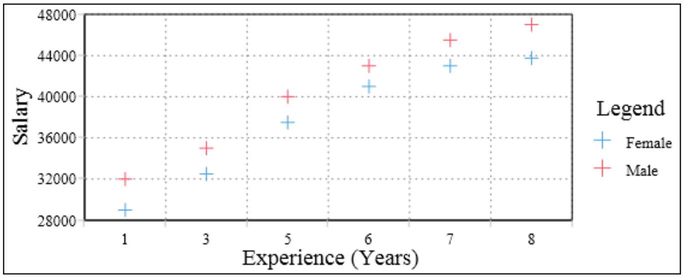


Figure 8-23 A scatter chart

About stock charts

A stock chart displays a stock's open, close, high, and low values for a set of trading dates. A stock chart can show the data for one stock or for multiple stocks. Although a stock chart is typically used to display stock data, you can also use it to chart other values that can be set up in a similar fashion. For example, you can use a stock chart to show four daily temperature values for a set of dates: high, low, sunrise, sunset.

Candlestick stock chart

A candlestick stock chart consists of a series of boxes with lines extending up and down from the ends, as shown in Figure 8-24. The top and bottom points of each line indicate the high and low values, respectively. The top and bottom of each box indicate the open and close values. If the close value is higher than the open value, the box is white. If the open value is higher than the close value, the box is shaded. This style enables you to see immediately whether a stock posted a gain or a loss for a given day. The chart in Figure 8-24 shows that the stock posted a gain every day in the trading week except the fourth day.

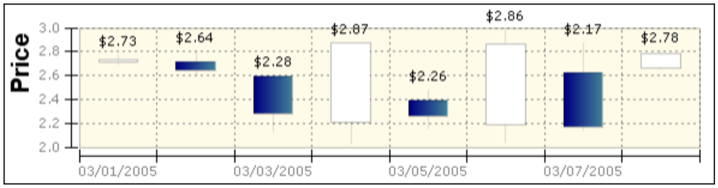


Figure 8-24 A candlestick stock chart

Bar stick stock chart

A bar stick stock chart consists of a series of vertical bars with horizontal tick marks, as shown in Figure 8-25. The top and bottom points of each bar indicate the high and low values, respectively. The horizontal tick marks indicate the open and close values. The tick mark on the left of the bar is the open value. The tick mark on the right of the bar is the close value.

The chart in Figure 8-25 shows the same data as the chart in Figure 8-24. Observe in Figure 8-25 that for every day, except the fourth, the tick mark on the left of the bar is lower than the tick mark on the right. On the fourth day, the tick mark on the left is higher than the tick mark on the right. This difference indicates a stock's gain or loss for a given day. As you can see, the candlestick stock chart shows the gain or loss pattern more clearly than the bar stick stock chart.

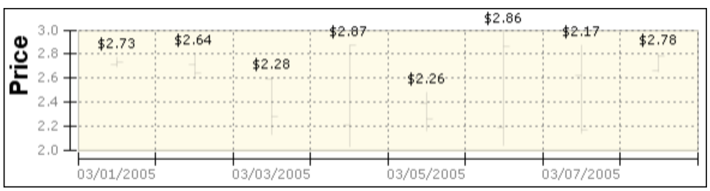


Figure 8-25 A bar stick stock chart

Selecting data for a chart

A chart shows the relationship between sets of values called series. There are two types of series: value and category. A value series contains numeric values, such as prices, sales totals, and salaries. These values determine, for example, the height of a bar in a bar chart or the size of a sector in a pie chart. For charts that use axes to display data, value series values appear on the *y*-axis.

A category series determines how to group the values, for example, by year, by country, or by product line. For charts that use axes, category series values appear on the *x*-axis. For all chart types, you must select one column as the category series and at least one column as the value series. You can select two columns to display two value series. The bar chart in Figure 8-26 displays two value series, total and profit.

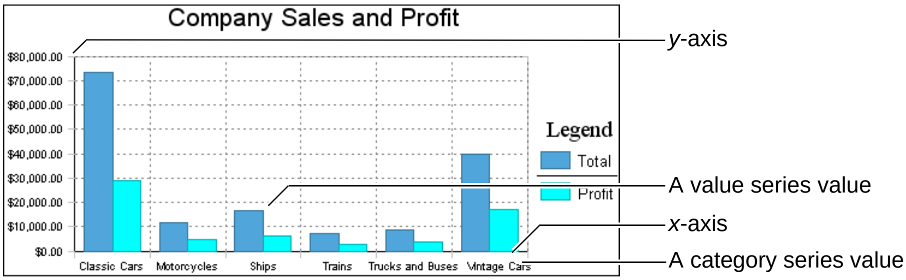


Figure 8-26 Parts of a chart

Then, depending on the table data that a chart uses, you might need to group and aggregate the values, so that the chart can display the data properly. Suppose the table displays the data in Figure 8-27. The table shows sales details by product, product line, and sales office. In Figure 8-27, the table shows only 10 rows of sample data, but the generated report contains 75 pages of data.

Sales Office			San Francisco	
Product Line			Classic Cars	
Product Name		**Total**		**Profit**
1970 Plymouth Hemi Cuda		$1,742.88		$976.80
1976 Ford Gran Torino		$5,864.88		$2,778.30
1982 Lamborghini Diablo		$687.20		$362.40
Sum	*Classic Cars*	*$8,294.96*		*$4,117.50*
Product Line			Motorcycles	
Product Name		**Total**		**Profit**
1957 Vespa GS150		$2,238.30		$755.55
1982 Ducati 900 Monster		$1,948.22		$582.32
Sum	*Motorcycles*	*$4,186.52*		*$1,337.87*
Product Line			Trucks and Buses	
Product Name		**Total**		**Profit**
1940 Ford Pickup Truck		$4,973.50		$2,115.33
1958 Setra Bus		$2,554.44		$918.54
1996 Peterbilt 379 Stake Bed with Outrigger		$1,352.86		$579.83
Sum	*Trucks and Buses*	*$8,880.80*		*$3,613.70*
Product Line			Vintage Cars	
Product Name		**Total**		**Profit**
1939 Cadillac Limousine		$2,163.50		$1,006.50
Sum	*Vintage Cars*	*$2,163.50*		*$1,006.50*
Sum	*San Francisco*	*$23,525.78*		*$10,075.57*
Sum		*$23,525.78*		*$10,075.57*

Figure 8-27 A table organizing detailed sales data in sections

Suppose you create a bar chart based on this table data. You want the chart to show the overall sales total and profit for each product line, just like the chart in Figure 8-26. You select the product line column as the category series and the total and profit columns as the value series, as shown in Figure 8-28.

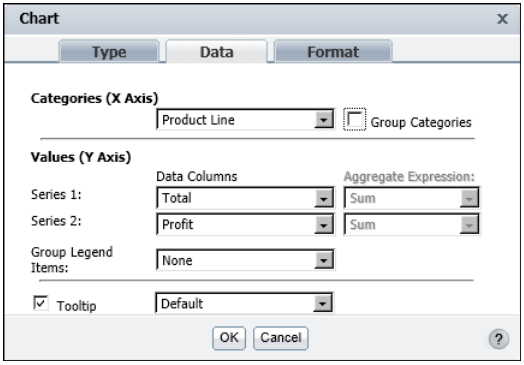


Figure 8-28 Chart—Data showing columns selected for the category and value series

Based on these selections, the bar chart looks like the one in Figure 8-29. Because you did not group or aggregate the data for the chart, the chart plots every total value and every profit value.

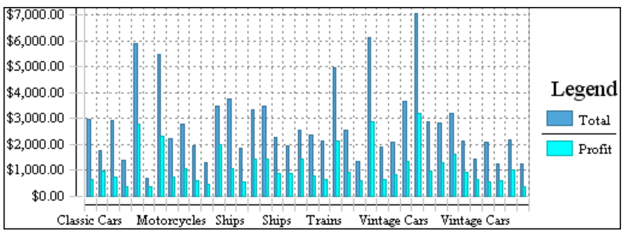


Figure 8-29 A chart that does not group or aggregate data displaying too many data points

To create the chart, as shown in Figure 8-26, which is based on the table data in Figure 8-27, group the chart data by product line and use the Sum function to calculate the grand total of the sales amounts and profits. Figure 8-30 shows the correct way to select and organize data for the chart.

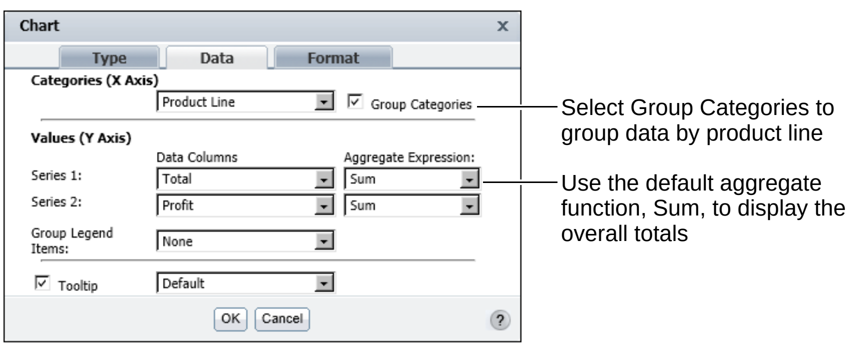


Figure 8-30 Grouping and aggregating data for a chart

The following sections expand on the concepts in this section and provide examples for selecting and organizing data, for different types of table data and for the different types of charts.

Selecting data for an area, bar, or line chart

Area charts, bar charts, and line charts can often be used interchangeably to display the same types of data. This section shows examples of selecting and organizing data for these chart types. Each example shows the following items:

- The area, bar, and line charts
- The table data on which the charts are based
- The selections on Chart—Data

Example 1

The charts in Figure 8-31 show sales numbers by month.

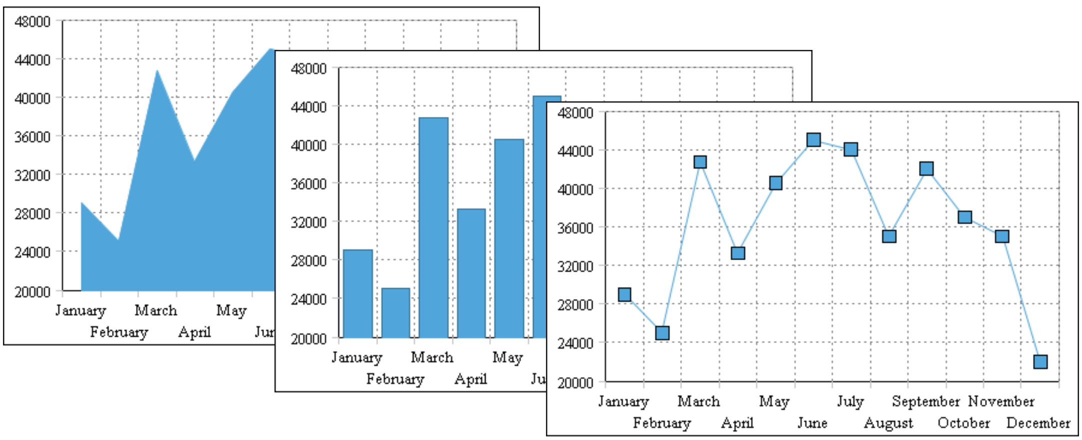


Figure 8-31 An area chart, a bar chart, and a line chart displaying sales numbers by month

Figure 8-32 shows the table data that the charts use. The data in the table is already aggregated. The table shows the total sales per month. Each value is plotted in each of the charts.

Figure 8-33 shows how the data is selected for the chart. You do not need to group or aggregate the data. As Figure 8-33 shows, all you do is select the Month column as the category series, and the Sales column as the value series.

Month	Sales
January	29000
February	25000
March	42700
April	33250
May	40500
June	45000
July	44000
August	35000
September	42000
October	37000
November	35000
December	22000

Figure 8-32 The table data used by the charts in Figure 8-31

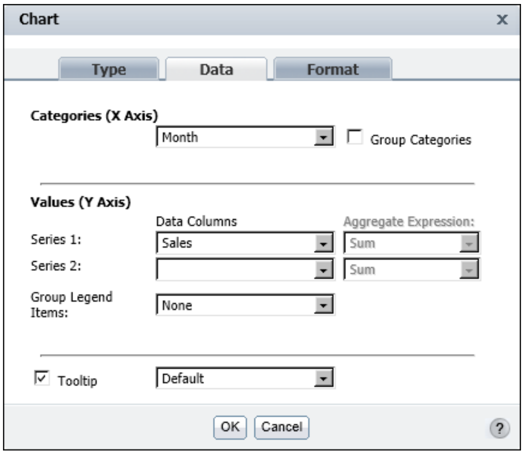


Figure 8-33 The data selected for the charts in Figure 8-31

Example 2

The charts in Figure 8-34 show sales totals by quarter and by region.

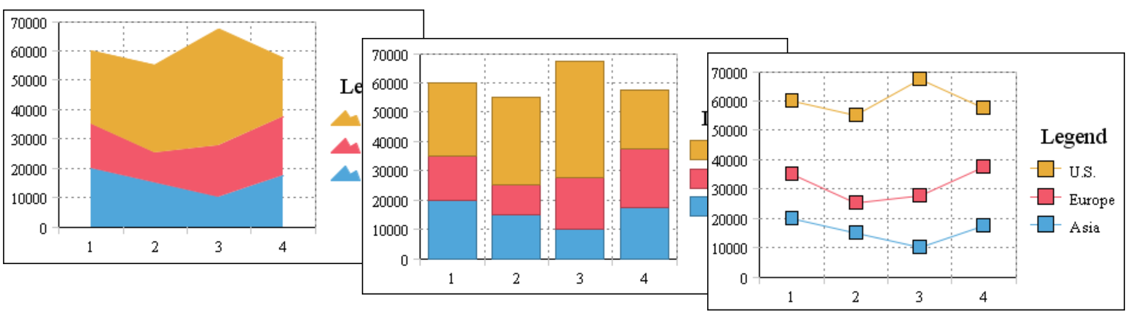


Figure 8-34 An area chart, a bar chart, and a line chart displaying sales by quarter and region

Figure 8-35 shows the table data that the charts use.

Region	Quarter	Sales
U.S.	1	25000
U.S.	2	30000
U.S.	3	40000
U.S.	4	20000
Europe	1	15000
Europe	2	10000
Europe	3	17500
Europe	4	20000
Asia	1	20000
Asia	2	15000
Asia	3	10000
Asia	4	17500

Figure 8-35 The table data used by the charts in Figure 8-34

Figure 8-36 shows how the data is selected for the chart. The Quarter column is the category series, and the Sales column is the value series. In addition, Region is the column selected for the Group Legend Items option. Without this option selected, the chart plots every value in the Quarter and Sales columns, as shown in Figure 8-37.

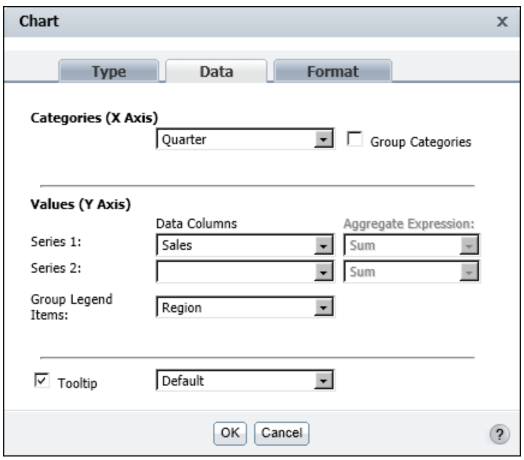


Figure 8-36 The data selected for the charts in Figure 8-34

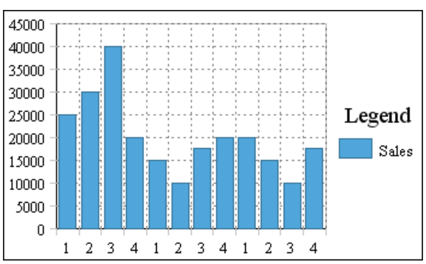


Figure 8-37 The resulting bar chart when data is not grouped by region

Selecting data for a pie chart

Unlike other chart types, a pie chart can display only one value series. For example, a pie chart can show sales totals by product line, but it cannot show sales totals and profit totals by product line. This section shows examples of selecting and organizing data for pie charts. Each example shows the following items:

- The pie chart
- The table data on which the chart is based
- The selections made in Chart—Data

Example 1

The pie chart in Figure 8-38 shows sales by quarter.

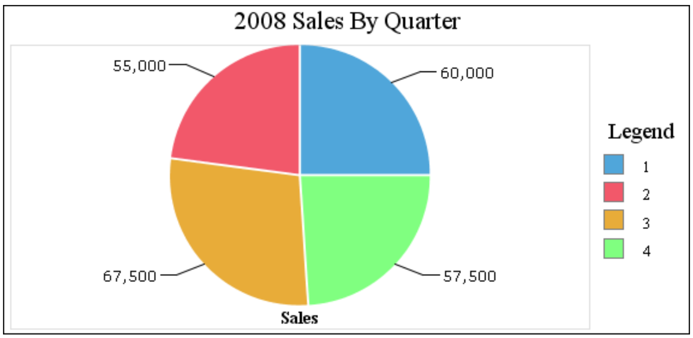


Figure 8-38 A pie chart showing sales by quarter

Figure 8-39 shows the table data that the pie chart uses.

Region	Quarter	Sales
U.S.	1	25000
U.S.	2	30000
U.S.	3	40000
U.S.	4	20000
Europe	1	15000
Europe	2	10000
Europe	3	17500
Europe	4	20000
Asia	1	20000
Asia	2	15000
Asia	3	10000
Asia	4	17500

Figure 8-39 The table data used by the chart in Figure 8-38

Figure 8-40 shows how the data is selected for the chart. The Quarter column is the category series, and the Sales column is the value series. In addition, the data is grouped by quarter, and the Sum function calculates the grand total of the sales, across regions, for each quarter.

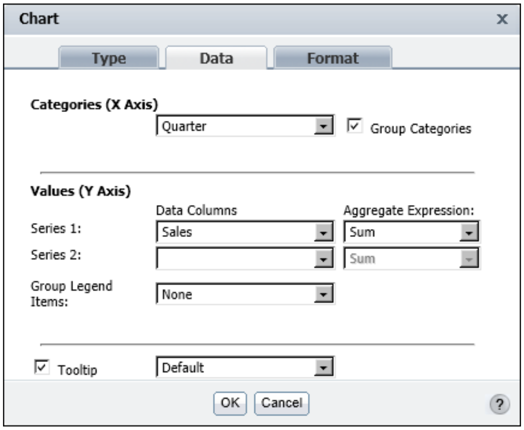


Figure 8-40 The data selected for the chart in Figure 8-38

Example 2

Figure 8-41 shows a series of pie charts. Each pie chart shows the sales by quarter for a particular region.

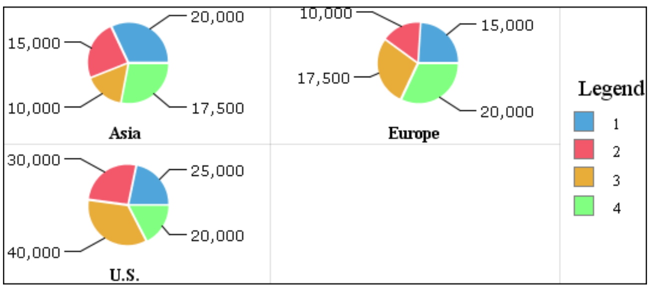


Figure 8-41 A series of pie charts showing sales by quarter and by regions

Figure 8-42 shows the table data that the series of pie charts use. It is the same set of data used in the previous pie chart example.

Region	Quarter	Sales
U.S.	1	25000
U.S.	2	30000
U.S.	3	40000
U.S.	4	20000
Europe	1	15000
Europe	2	10000
Europe	3	17500
Europe	4	20000
Asia	1	20000
Asia	2	15000
Asia	3	10000
Asia	4	17500

Figure 8-42 The table data used by the charts in Figure 8-41

Figure 8-43 shows how the data is selected for the chart. The Quarter column is the category series and the Sales column is the value series. Unlike the previous pie chart example, the category series (quarter) values are not grouped. Instead, the value series data is grouped by the region column.

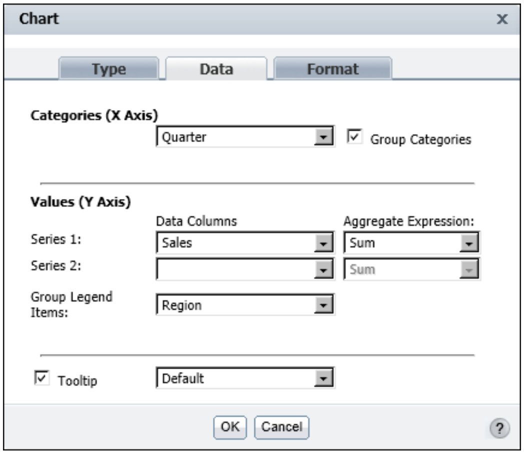


Figure 8-43 The data selected for the charts in Figure 8-41

Example 3

The pie chart in Figure 8-44 shows the number of products in each product line.

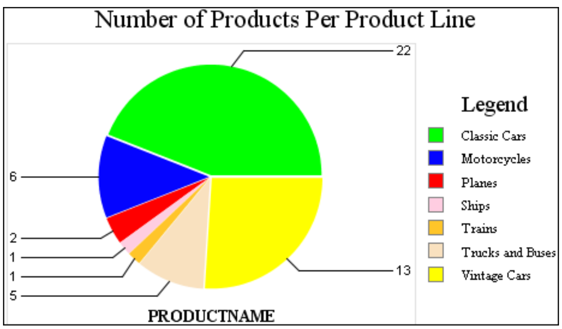


Figure 8-44 A pie chart showing the number of products in each product line

Figure 8-45 shows a sample of the table data that the pie chart uses. The actual data spans two pages.

Figure 8-46 shows how the data is selected for the chart. The PRODUCTLINE column is the category series, and the PRODUCTNAME column is the value series. In addition, the data is grouped by product line, and the Count function counts the number of products in each product line.

Product Line	Product	Stock	Buy Price	Inventory Value
Classic Cars				
	1952 Alpine Renault 1300	7305	$98.58	$720,126.90
	1972 Alfa Romeo GTA	3252	$85.68	$278,631.36
	1962 LanciaA Delta 16V	6791	$103.42	$702,325.22
	1968 Ford Mustang	68	$95.34	$6,483.12
	2001 Ferrari Enzo	3619	$95.59	$345,940.21
	1969 Corvair Monza	6906	$89.14	$615,600.84
	1968 Dodge Charger	9123	$75.16	$685,684.68
	1969 Ford Falcon	1049	$83.05	$87,119.45
	1970 Plymouth Hemi Cuda	5663	$31.92	$180,762.96
	1969 Dodge Charger	7323	$58.73	$430,079.79
	1993 Mazda RX-7	3975	$83.51	$331,952.25
Motorcycles				
	1969 Harley Davidson Ultimate Chopper	7933	$48.81	$387,209.73
	1996 Moto Guzzi 1100i	6625	$68.99	$457,058.75
	2003 Harley-Davidson Eagle Drag Bike	5582	$91.02	$508,073.64
	2002 Suzuki XREO	9997	$66.27	$662,501.19
Trucks and Buses				
	1958 Setra Bus	1579	$77.90	$123,004.10
	1957 Chevy Pickup	6125	$55.70	$341,162.50
	1940 Ford Pickup Truck	2613	$58.33	$152,416.29
Vintage Cars				
	1937 Lincoln Berline	8693	$60.62	$526,969.66
	1936 Mercedes-Benz 500K Special Roadster	8635	$24.26	$209,485.10

Figure 8-45 The table data used by the chart in Figure 8-44

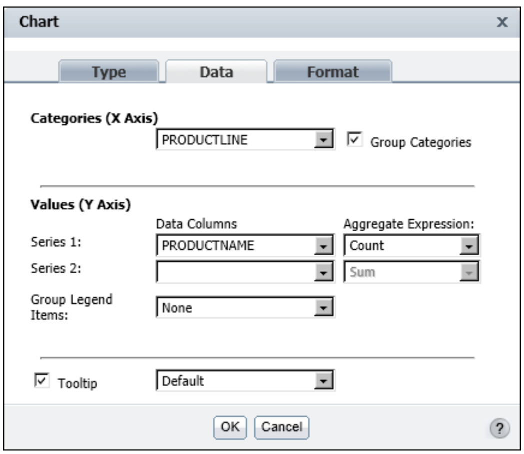


Figure 8-46 The data selected for the chart in Figure 8-44

Selecting data for a meter chart

Unlike most chart types, a meter chart does not display data on a *y* or *x*-axis. Instead, a meter chart displays each data value as a needle on a dial. A meter chart is used to emphasize a small number of values. This section shows examples of selecting and organizing data for meter charts. The example shows the following items:

- The meter chart

- The table data on which the chart is based
- The selections made in Chart—Data

Example

The superimposed meter chart in Figure 8-47 shows two values, actual sales and projected sales.

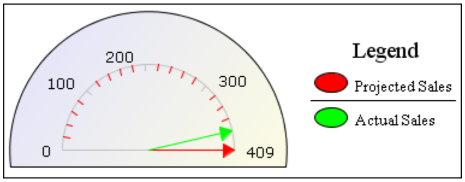


Figure 8-47 A meter chart showing actual sales and projected sales

Figure 8-48 shows the table data that the chart uses. The meter chart displays the aggregate values in the measure columns of the table. Figure 8-49 shows how the data is selected for the chart.

Year	Category	Projected Sales	Actual Sales	
2006				
	Keyboards	70	80	
	Monitors	39	60	
	Printers	110	90	
	Motherboards	56	30	
	CPU	134	120	
		Total 409	*Total*	380

Figure 8-48 The table data used by the chart in Figure 8-47

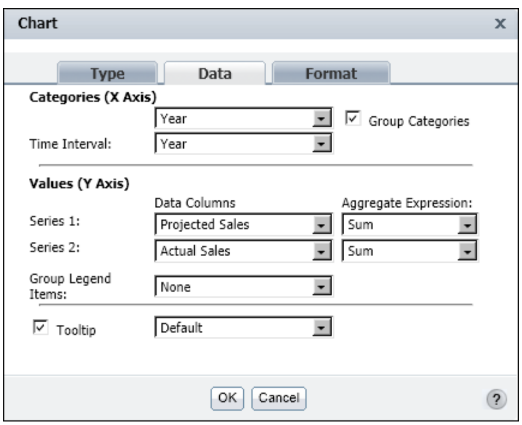


Figure 8-49 The data selected for the chart in Figure 8-47

A category series is typically not essential to a meter chart, but is required in this chart that displays Projected Sales and Actual Sales totals for 2006, so the Year column is selected. The aggregate data in the Projected Sales column and the Actual Sales column are the value series.

Selecting data for a scatter chart

A scatter chart displays data as *x-y* coordinates. It combines each pair of numeric values into single data points. You select columns that contain numeric values for both the category and the value series. This section shows examples of selecting and organizing data for scatter charts. Each example shows the following items:

- The scatter chart
- The table data on which the chart is based
- The selections made in Chart—Data

Example 1

The scatter chart in Figure 8-50 shows the relationship between salaries and years of experience.

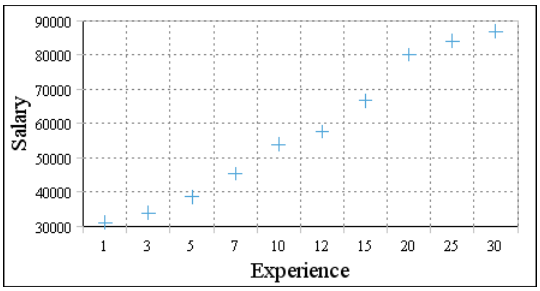


Figure 8-50 A scatter chart showing the relationship between salaries and years of experience

Figure 8-51 shows the table that contains the data that the scatter chart uses. The chart uses data from the Years_Experience and Average Salary columns.

Average Salary	Years_Experience	Average Salary (Female)	Average Salary (Male)
31000	1	30000	32000
33750	3	32500	35000
38500	5	37000	40000
45250	7	43000	47500
53750	10	52500	55000
57500	12	55000	60000
66500	15	63000	70000
80000	20	75000	85000
84000	25	78000	90000
86750	30	80000	93500

Figure 8-51 The table data used by the chart in Figure 8-50

Figure 8-52 shows how the data is selected for the chart.

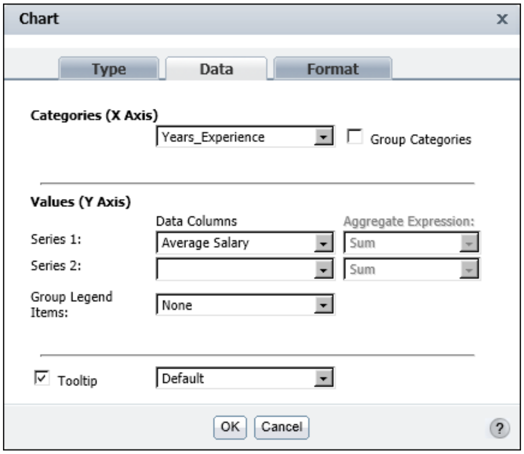


Figure 8-52 The data selected for the chart in Figure 8-50

The Years_Experience column is the category series, and the Average Salary column is the value series. The data is neither grouped nor aggregated, because the scatter chart plots every value in the Years_Experience and Average Salary columns.

Example 2

Like the scatter chart in the previous example, the scatter chart in Figure 8-53 shows the relationship between salaries and years of experience. In this example, however, the chart displays two value series: the average salaries for men and the average salaries for women.

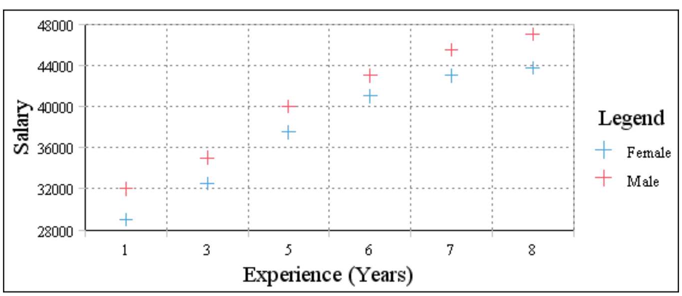


Figure 8-53 A scatter chart showing the relationship between salaries and years of experience, by gender

Figure 8-54 shows the table containing the data that the scatter chart uses. Unlike the table in the previous scatter chart example, this table does not show the average salaries by years of experience or by gender. Instead, this table shows salary and experience data for each employee. To display the average salaries by years of experience and by gender, the data must be grouped and aggregated.

Employee	Gender	Years_Experience	Salary
Sally Firth	Female	1	30000
Maria Jose	Female	1	28000
Robert Miller	Male	1	33000
Richard Wade	Male	1	31000
George Smith	Male	3	35500
Joe Manetta	Male	3	34500
Jose Vargas	Male	3	35000
So-Young Kim	Female	3	33000
Maya Salvatore	Female	3	32000
Kim Smith	Female	5	38000
Tyler Oliver	Male	5	40000
Alex Chang	Male	5	39000
Sandra Au	Female	5	37000
Joe Bennett	Male	5	41000
James Lee	Male	6	43000
Sarah Jones	Female	6	41000
David Taylor	Male	7	45000
Bob Matthews	Male	7	46000
Susan Volt	Female	7	43000
John Smithers	Male	8	47000
Mary Anderson	Female	8	44000
Barbara Billings	Female	8	43500

Figure 8-54 The table data used by the chart in Figure 8-53

Figure 8-55 shows how the data is selected, grouped, and aggregated for the chart.

The Years_Experience column is the category series and the Salary column is the value series. The values are grouped by years of experience and gender. The Average aggregate function is selected, so that the chart calculates and displays the average salary for each group.

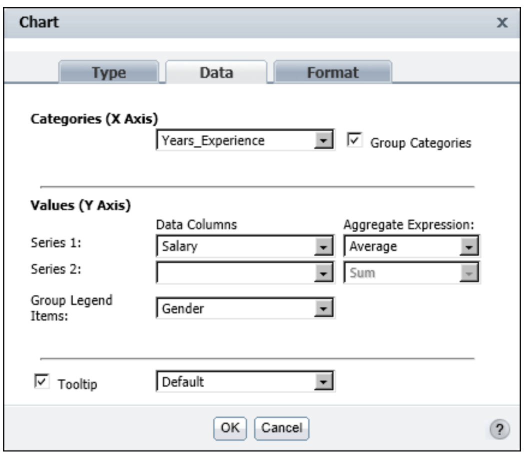


Figure 8-55 The data selected for the chart in Figure 8-53

Selecting data for a stock chart

A stock chart requires the following data:

- Date values as the categories series
- High values, low values, open values, and close values as the value series

Example

The stock chart in Figure 8-56 shows a stock's high, low, open, and close values for five days. Figure 8-57 shows the table data that the stock chart uses. Figure 8-58 shows how the data is selected for the chart. The Date column is the category series. The High, Low, Open, and Close columns are the value series.

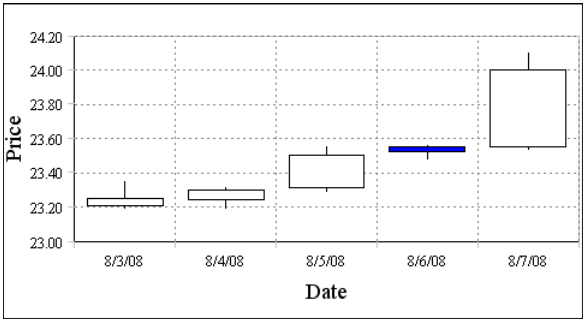


Figure 8-56 A stock chart showing high, low, open, and close values

Company	Date	Open	Close	Low	High
MYCO	8/3/08	23.21	23.25	23.19	23.35
MYCO	8/4/08	23.24	23.3	23.19	23.31
MYCO	8/5/08	23.31	23.5	23.29	23.55
MYCO	8/6/08	23.55	23.52	23.48	23.56
MYCO	8/7/08	23.55	24	23.54	24.1

Figure 8-57 The table data used by the chart in Figure 8-56

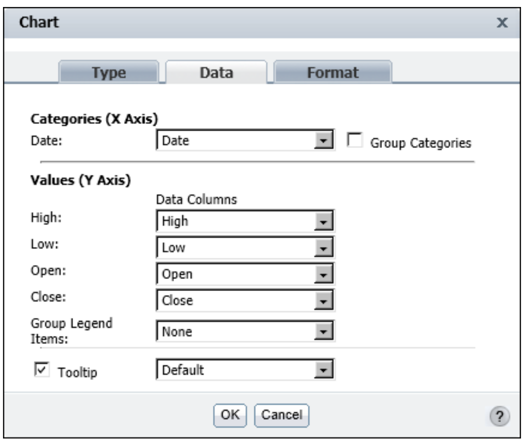
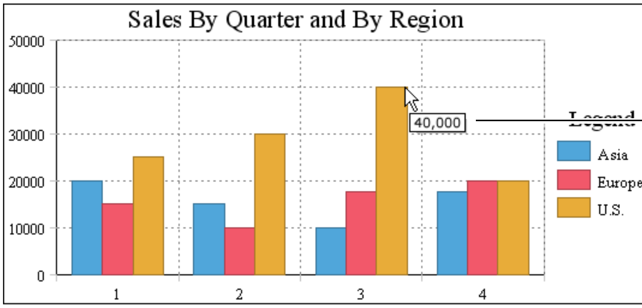


Figure 8-58 The data selected for the chart in Figure 8-56

Displaying additional series information

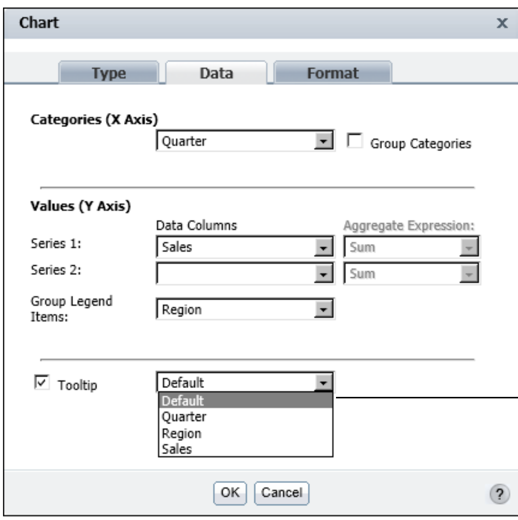
By now, you may have noticed the Tooltip option on the Chart—Data page of the chart editor. The Tooltip option is selected by default, as shown in Figure 8-58. A tooltip displays the value series value when a user places the mouse pointer over a bar in a bar chart, a sector in a pie chart, or a data-point marker in a line chart. Figure 8-59 shows a tooltip displaying 40,000 when the mouse is placed over a bar in a bar chart.



By default, a tooltip displays a value series value

Figure 8-59 A tooltip displaying a value series value

You can disable the tooltip by deselecting the Tooltip option. You also can display a different tooltip value. For the bar chart in Figure 8-59, instead of displaying the sales value, you can display the region or the quarter. To display a different tooltip value, select a different column from the drop-down list, as shown in Figure 8-60.



Select the column whose value to display as a tooltip

Figure 8-60 Selecting a value to display as a tooltip

Formatting a chart

When you create a chart and use the default formatting options, the chart type and the data you select determines the basic look of the chart. The template or the software determines styles, such as fonts and colors. Figure 8-61 shows an example of a bar chart that uses default formats. You can improve the chart by editing the placeholder text for the chart title, *y*-axis title, and *x*-axis title, and by changing the size or position of the *x*-axis labels to display all the city labels.

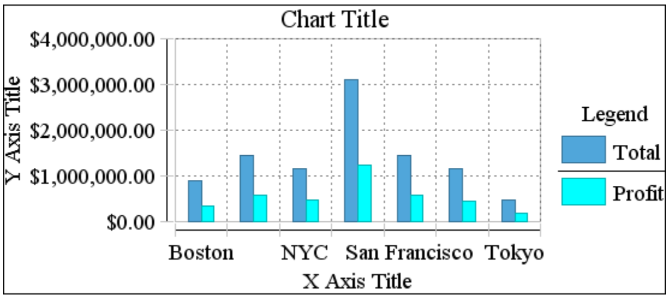


Figure 8-61 A bar chart using default formats

To change the appearance of a chart, change the option settings on Chart—Format. Figure 8-62 shows Chart—Format displaying the default option settings for a bar chart.

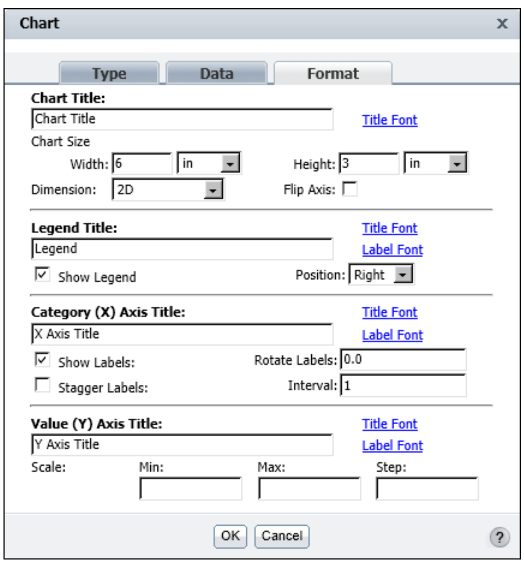


Figure 8-62 Chart—Format displaying the default options for a bar chart

The options that appear differ depending on the chart type. For example, a pie chart does not have axes, so the Category (*X*) Axis Title and Value (*Y)* Axis Title options do not appear on Chart—Format for a pie chart.

Changing the size of a chart

You can change a chart's width and height. If a chart looks too big, you can reduce its width, or height, or both. For example, a superimposed meter chart that displays data on one dial can look oversized. Conversely, increase the size of a chart if items in the chart look too crowded. For example, if a bar chart displays many bars, and many of the *x*-axis labels do not appear, you can start by increasing the width of the chart to see if all the items fit.

Setting a chart's width

To set the width of a chart, on Chart—Format, in Chart Size, type a number in Width. A chart's width is limited by the width of the table if you select Fixed Width for the Layout Preference option in Page Setup. If you specify a chart width that exceeds the width of the table, the chart appears truncated. Figure 8-63 shows a chart where the legend appears truncated. If Layout Preference is set to Auto Expand Width, the table expands to accommodate the chart width you set. In either case, if you increase a chart's width, confirm that the chart still fits the page. Always check the output in the viewer and in PDF format.

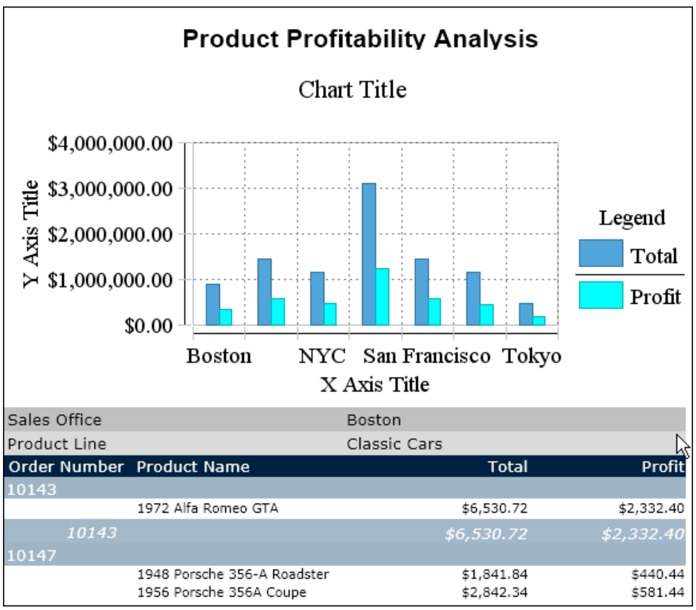


Figure 8-63 A report where the chart width exceeds the table width

Setting a chart's height

To set the height of a chart, in Chart—Format, in Chart Size, type a number in Height. When you increase or decrease a chart's height, you increase or decrease the visual contrast between data values. Compare the charts in Figure 8-64 and Figure 8-65. Both present the same data. By changing the chart height in Figure 8-64, each chart presents a different impression of the data. The taller chart shows more contrast between the data values. If you look at the image alone you think there is a greater disparity between salaries earned by men and women than in the chart in Figure 8-64.

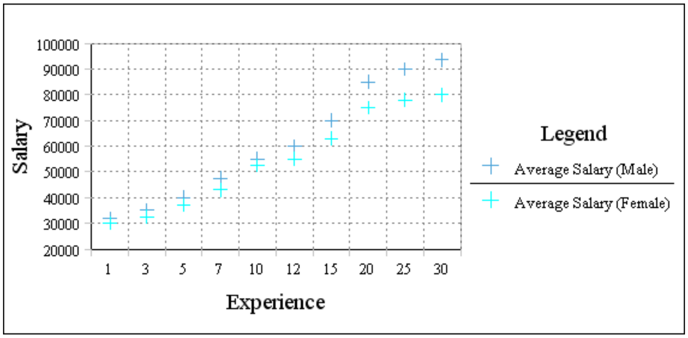


Figure 8-64 A scatter chart using the default chart size

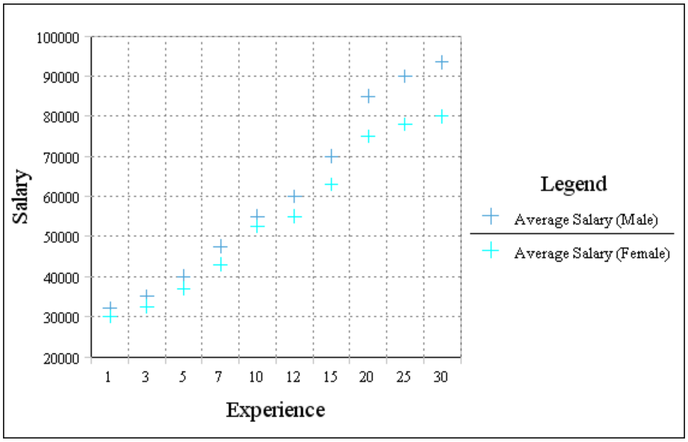


Figure 8-65 A scatter chart with its height increased

Editing and formatting titles and labels

By default, every chart displays Chart Title as its title. If a chart has axes, the chart also displays *X* Axis Title and *Y* Axis Title. To edit these titles, in Chart—Format, type new titles in the text boxes in Chart Title, Category (X) Axis Title, and Value (Y) Axis Title, respectively. If you do not want to display any titles, delete the text in the text boxes. You also can change the font attributes of every title and label in the chart. To format a title, choose Title Font. To format a label, choose Label Font. Figure 8-66 shows the font attributes you can change.

You change the font attributes of a title or labels for aesthetic reasons or for practical reasons. For example, if the *x*-axis does not display all the labels, as shown in Figure 8-63, reducing the font size of the Category (X) Axis labels is one way to solve the problem. Figure 8-67 shows the effect of reducing the size of the labels on the *x*-axis. All the city names appear.

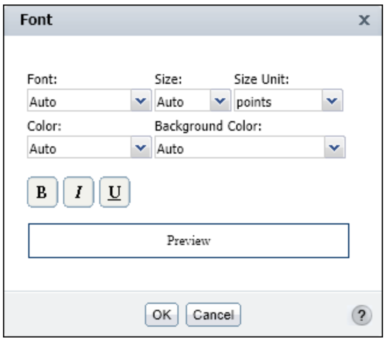


Figure 8-66 Setting font attributes

The chart in Figure 8-67 also displays the *y*-axis and legend labels in a smaller size than the same labels in the chart in Figure 8-63. Typically, when you change the font attributes of labels in one area, you apply the same font attributes to labels in other areas, so that the labels in the chart have a consistent appearance.

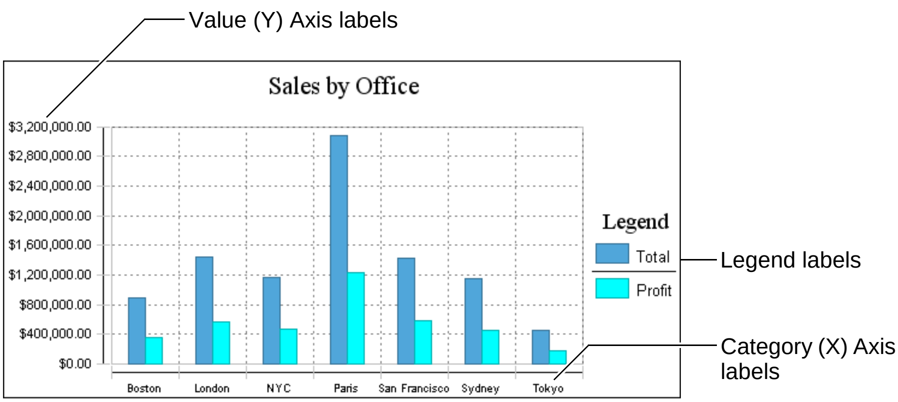


Figure 8-67 Labels in a bar chart set to the same font attributes

Making a chart look three dimensional

A chart that uses the default formats appears as a two-dimensional chart. You can make a chart appear three dimensional by selecting 2D With Depth in Dimension, as shown in Figure 8-68.

Figure 8-69 shows a bar chart that uses the 2D With Depth setting. Not all chart types support the 2D With Depth setting. For example, this setting is not available for meter charts or line charts.

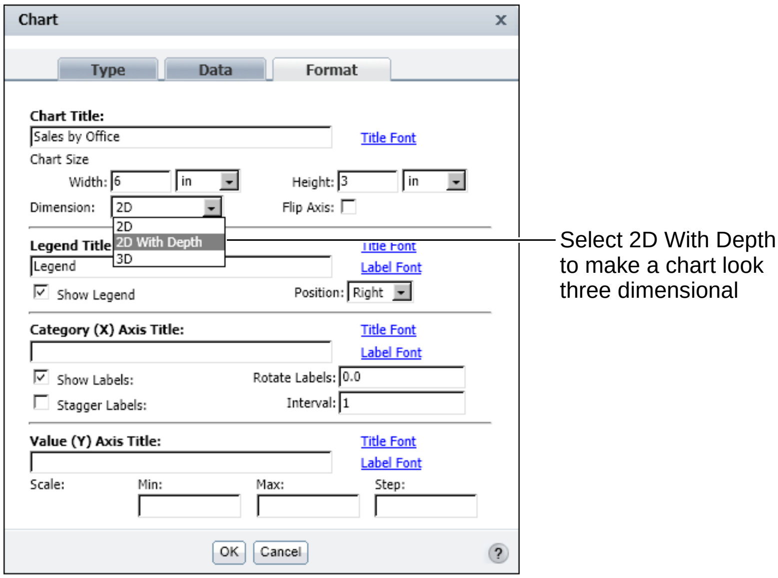


Figure 8-68 Setting the Dimension option

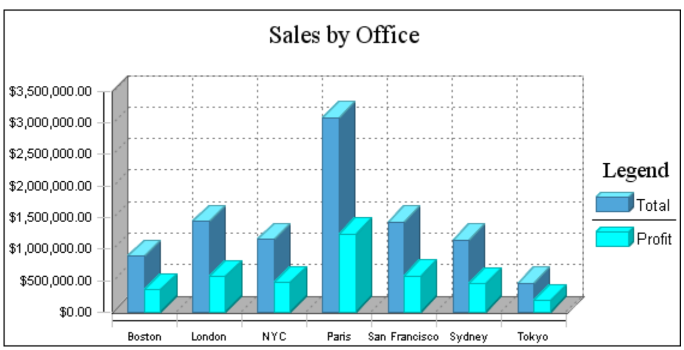


Figure 8-69 A bar chart using the 2D With Depth setting

Changing the position of the legend

By default, a legend appears on the right side of a chart. You can change the position of the legend to appear above, below, or on the left side of the chart. If a chart is too wide, positioning the legend above or below the chart saves space horizontally. To reposition a legend, select Above, Below, Left, Right, or Inside, as shown in Figure 8-70.

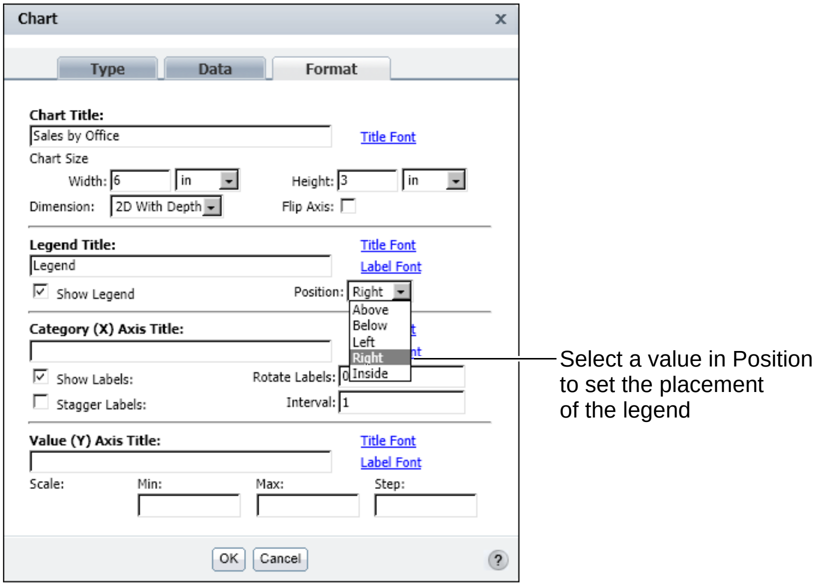


Figure 8-70 Setting the position of a legend

You can also choose to remove a legend by deselecting Show Legend. For example, it is not necessary to display a legend if the chart shows only one value series. The bar chart in Figure 8-71, for example, does not need a legend.

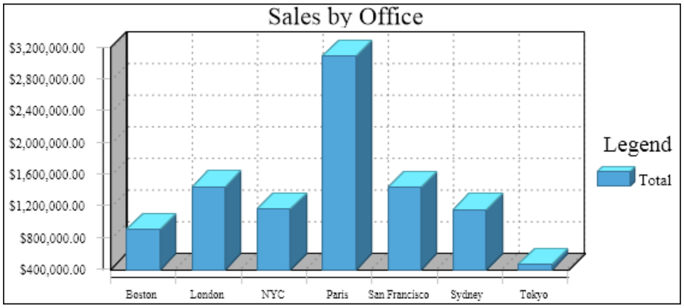


Figure 8-71 A bar chart showing an unnecessary legend

Formatting labels to fit on the *x*-axis

A chart that displays numerous category values sometimes cannot fit all the values on the *x*-axis. As discussed previously, one way to format *x*-axis labels so that they all appear is to reduce the font size. Other techniques include rotating the labels, staggering the labels, and displaying the labels at set intervals. You can also use a combination of these techniques.

Rotating *x*-axis labels

You can rotate labels by a specified number of degrees. On Chart—Format, in Rotate Labels, type the number of degrees to indicate the amount of rotation. To rotate labels in a clockwise direction, use a negative number. To rotate labels in a counter-clockwise direction, use a positive number. Figure 8-72 shows four charts, each with Rotate Labels set to a different value.

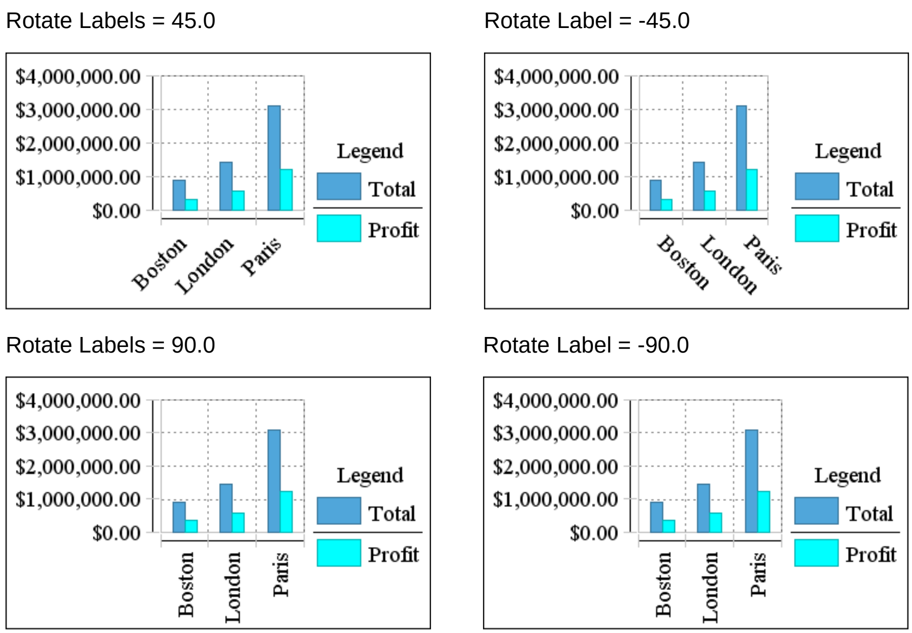


Figure 8-72 Charts using different values for the Rotate Labels option

Staggering *x*-axis labels

On Chart—Format, you can select the Stagger Labels option to place the labels in a zigzag arrangement, as shown in Figure 8-73. The first chart shows the results of staggering, but not rotating, the *x*-axis labels. The second chart shows the results of staggering and rotating the labels 30 degrees.

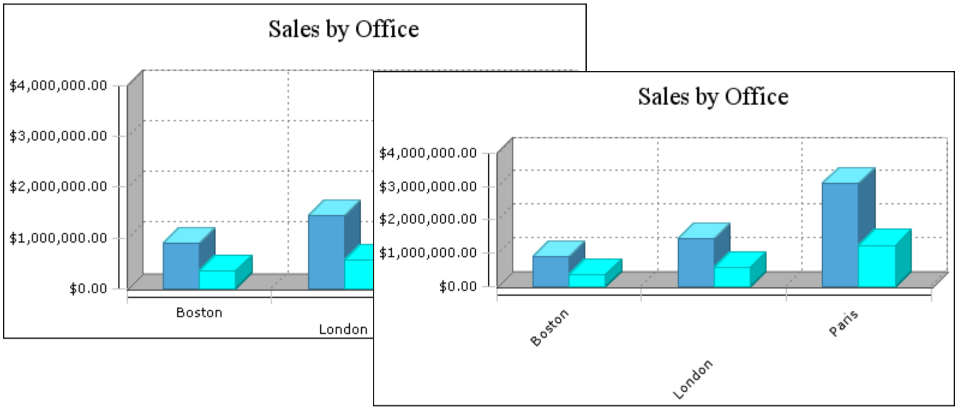


Figure 8-73 Charts using different values for the Stagger Labels and Rotate Labels options

Displaying *x*-axis labels by interval

By default, a chart displays every category value on the *x*-axis, unless the label does not fit in the space allocated to each category value. As described in the previous sections, to make all the labels fit, you can reduce the font size, rotate the labels, or stagger the labels. These techniques, however, can make the *x*-axis look crowded. If it is not essential to display every category value, you can specify that the *x*-axis display alternate values, or every third value, and so on.

On Chart—Format, the Interval option is set to 1 by default, which means every value appears. Set Interval to a different number to specify the interval at which values appear. For example, to display alternate values, set Interval to 2.

Figure 8-74 shows a stock chart that displays open, close, high, and low prices for every trading day in August. Although Interval is set to 1, there is not enough space to display every date as a label on the *x*-axis, so the *x*-axis displays alternate values. Figure 8-75 shows the same stock chart, but this time, Interval is set to 5 to display the first trading date per week.

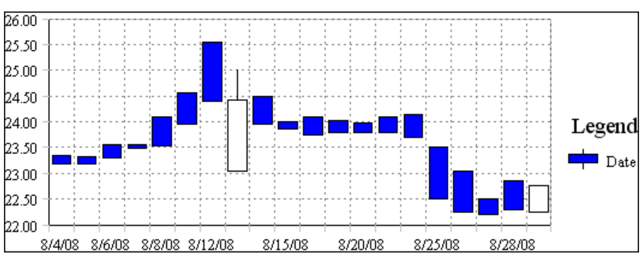


Figure 8-74 A stock chart displaying every trading date value

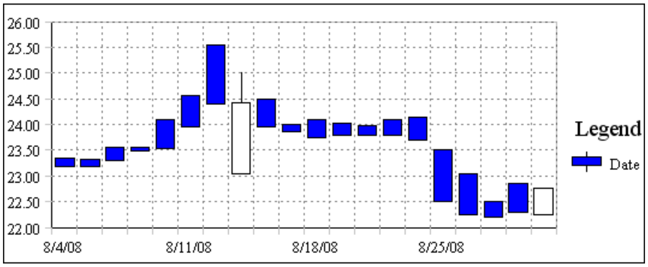


Figure 8-75 A stock chart displaying the first trading date per week

As the chart in Figure 8-75 shows, it makes sense to skip category values only if the values are consecutive numbers or dates, where the user can intuitively fill in the missing values.

Changing the range of values on the *y*-axis

The number and range of values on the *y*-axis depends on several factors, including the smallest value series value, the largest value series value, the height of the chart, and the amount of vertical space the chart items require.

Compare the charts in Figure 8-76. Both charts present the same data, but the *y*-axis values are different. In the chart on the left, the *y*-axis displays six values, ranging from 0 to 3500000. In the chart on the right, the *y*-axis displays five values, ranging from 0 to 4000000. Both charts have the same height value, so in these examples, the different *y*-axis values are a result of the different amounts of space the *x*-axis labels require.

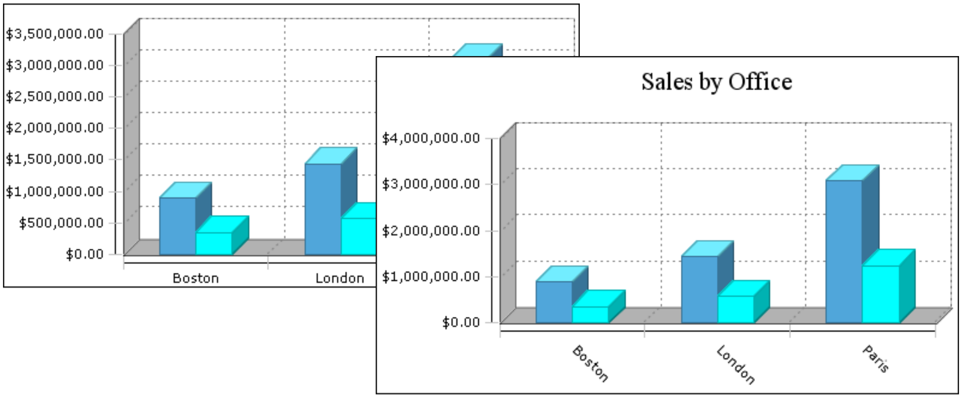


Figure 8-76 Charts displaying different ranges of values on the *y*-axis

You can control the range of values that appear on the *y*-axis, as shown in Figure 8-77.

On Chart—Format, in Scale, you can set the following options:

- Min. Type a number that represents the lowest value to display on the *y*-axis. This number appears at the bottom of the *y*-axis.
- Max. Type a number that represents the highest value to display on the *y*-axis. This number appears at the top of the *y*-axis.
- Step. Type a number that represents the increment between each value.

Figure 8-77 shows the Min, Max, and Step options set to 0, 3500000, and 500000, respectively. These settings change the *y*-axis values of the second chart in Figure 8-76 to the *y*-axis values in Figure 8-78. Notice that the *y*-axis values in Figure 8-78 are closer together. Because the chart height is the same, the chart has to display more values in the same space. When you change the scale of *y*-axis values, you typically adjust the height of the chart.

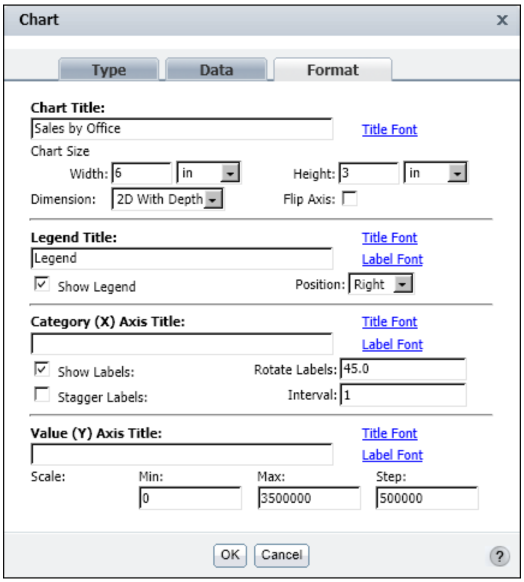


Figure 8-77 Chart—Format showing modified scale values

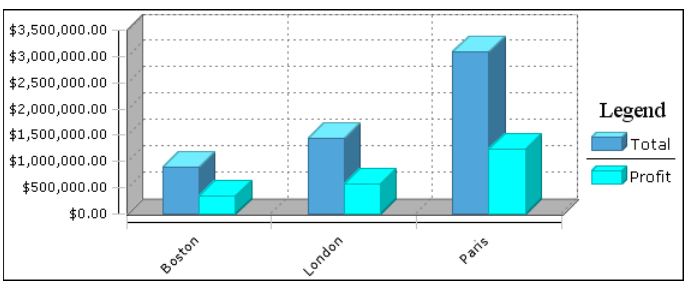


Figure 8-78 The chart using the modified scale values

Working with Flash charts

Flash charts are similar to BIRT charts and HTML5 charts, and are designed to be used in a similar manner. The examples shown in Figure 8-79 and Figure 8-80, display a standard doughnut chart, and a segmented Flash doughnut chart, respectively, viewed in Actuate BIRT Viewer. In the animated Flash doughnut chart, as you select a segment, it slices away from the rest of the chart.

When you format a Flash chart, you can specify animation properties such as rotation and slicing, as you can in an HTML5 chart.

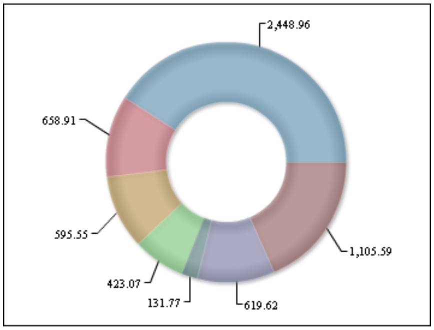


Figure 8-79 Standard doughnut chart

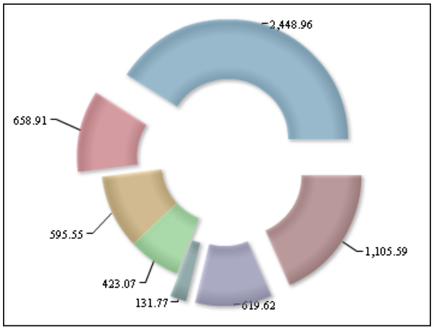


Figure 8-80 Segmented Flash doughnut chart

In BIRT Studio you can insert a Flash chart at the table level, or at the section level, if your report organizes data in sections. You can choose from the following types of Flash charts:

- Column
- Bar
- Line

- Pie
- Doughnut

Creating a Flash chart

In BIRT Studio, you can insert a Flash chart at the table level or section level, just as for regular charts. The procedure to create a Flash chart is similar to the one to create a standard chart. To create a Flash chart, perform the following steps:

1 On BIRT Studio, select the report table, or report section, and choose Chart➛Insert Flash Chart, as shown in Figure 8-81. Flash Chart appears.

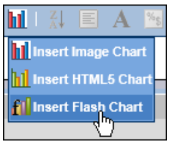


Figure 8-81 Inserting a Flash chart

2 In Flash Chart—Type, select a type of chart. Figure 8-82 shows the types of Flash charts available in BIRT Studio.

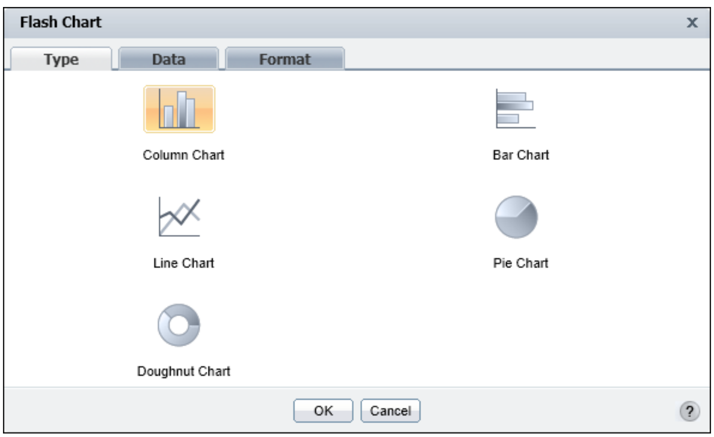


Figure 8-82 Selecting the type of Flash chart

3 On Flash Chart—Data, specify the data to present in the chart, in the same way you did for a standard chart.

4 On Flash Chart—Format, specify the formatting properties for the chart.The available formatting options depend on the chart type, and are similar to the formatting options available for standard charts.

The following sections describe how to select data to display in a Flash chart, and how to format a Flash chart.

Selecting data for a Flash chart

The process to select data for a Flash chart is similar to the process to do so for standard charts. When you select data for a column, line, or bar Flash chart, you specify the value series and the category series, if applicable. You then set the grouping options and select a value to display for the tooltip from the drop-down list.

When you select data for a pie or doughnut Flash chart, you specify the values to display as sectors of the pie or doughnut. You then specify the grouping options and select a value to display for the tooltip from the drop-down list. Figure 8-83 displays Flash Chart—Data.

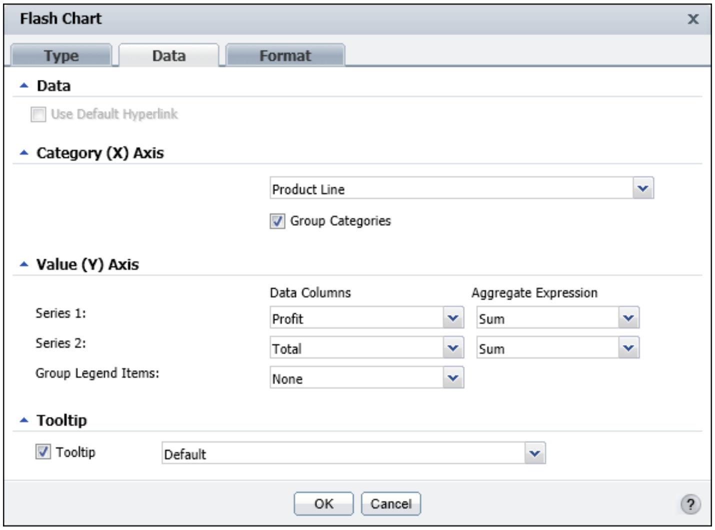


Figure 8-83 Selecting data for a Flash chart

How to select data for a column, line, or bar Flash chart

In Flash Chart—Data, complete the following steps:

1. In Category (X) Axis, select a column from the drop-down list, to set as the *x*-axis.
2. Select Group Categories, to group the values in this column. When selected, only the groups are displayed on the *x*-axis in the chart.
3. In Value (Y) Axis, complete the following steps:
   1. In Series 1, select a column from the first drop-down list. If necessary, specify an aggregate function to aggregate the data in this column.
   2. In Series 2, select a column from the second drop-down list. If necessary, specify an aggregate function to aggregate the data in this column.

3 In Group Legend Items, select a column from the drop-down list. Alternately, accept the default selection of None.

4 In Tooltip, select Tooltip, then select a column from the drop-down list, whose values are displayed as the tooltip. Deselect Tooltip if you do not want to display a tooltip for values in the chart.

   When selected, the value series values are displayed when the user hovers a mouse pointer over a column, bar, or line in the chart.

How to select data for a pie Flash chart

In Flash Chart—Data, complete the following steps:

1 In Slice, in Category, select a column from the drop-down list to display as sectors of the pie or doughnut. Select Group Categories to display groups as sectors.

2 In Value, select a column from the drop-down list. Select an aggregate function to aggregate the values in this column.

3 In Tooltip, select Tooltip, then select a column from the drop-down list, whose values are displayed as the tooltip. Deselect Tooltip if you do not want to display a tooltip for values in the chart.

Formatting a Flash chart

The following section describes the formatting options available for column, line, and bar charts, and pie and doughnut charts.

Formatting a column, line, or bar chart

When you format a column, line, or bar chart, you specify the properties in Flash Chart—Format in a similar way that you do for a standard chart.

In Flash Chart—Format you can specify the title, chart size, dimension, and font properties, and legend properties. In addition, you can format the *x*-axis title, and label properties. You can also format the *y*-axis title, minimum and maximum values, as well as step value, as you would for a regular chart.

Specifying chart properties

You can specify a chart title, width, height, as well as font properties for Flash charts using BIRT Studio. Figure 8-84 shows an example of a line chart that uses the default font properties. When using a line chart, you cannot specify a chart subtype, or dimension, as you can for bar or column charts.

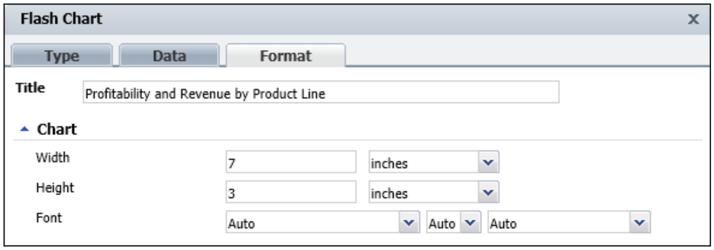


Figure 8-84 Specifying chart properties

Figure 8-85 shows an example of a column chart that uses the default font properties. In addition to specifying the size for bar and column charts, you can select a subtype, dimension, and specify font properties, as you do for standard charts.

Figure 8-85 Specifying chart properties for a bar or column chart

When working with two-dimensional column or bar charts, you can also specify whether to use glass style to display the chart. Select Use Glass Style to display the bars or columns in the chart with smooth, rounded edges. Deselect Use Glass Style to display angular, sharp edges. Figure 8-86 shows two charts that use the same data. The chart on the left does not use glass style, while the chart on the right uses the glass style option.

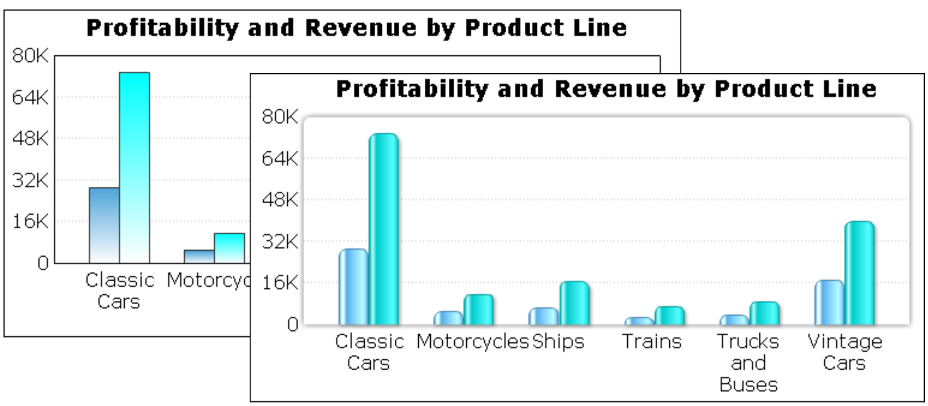


Figure 8-86 Deselecting and selecting glass style in a 2D column chart

Formatting the legend

Using BIRT Studio, you can specify whether to display the legend for a Flash chart. If you choose to display the legend, you can specify a title, as well as specify whether to display the legend to the right of the chart, or below the chart, as shown in Figure 8-87.

Figure 8-87 Formatting the legend

Formatting the category (x) axis

You can specify a title for the category axis and specify whether to display labels for *x*-axis values. If you choose to display labels, you can further specify whether to stagger them at an interval, or display them at an angle, as shown in Figure 8-88.

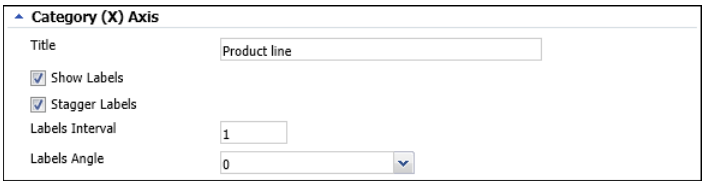


Figure 8-88 Specifying category (x) axis properties

Formatting the value (y) axis

You can specify a title for the value axis, as well as the minimum and maximum value to display. In addition, you can choose Auto Step to enable BIRT Studio to determine the step interval, as shown in Figure 8-89, or use the menu to specify a step interval for the displayed values.

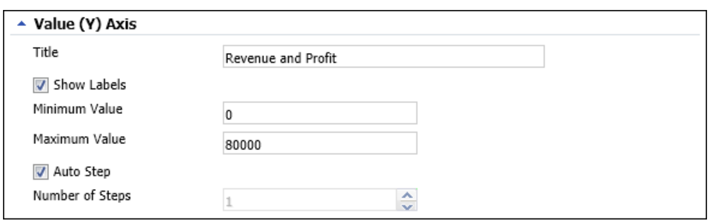


Figure 8-89 Specifying value (y) axis properties

Formatting a pie or doughnut chart

When you format a pie or a doughnut chart, you specify a chart title, size, dimension, angle of rotation, radius of the pie or width of the doughnut ring, font, and legend properties.

When you format a pie chart, in addition to specifying the title, size, dimension, and legend, complete the following steps:

1. Specify the rotation angle in degrees. Figure 8-90 displays an angle of rotation set to 180 degrees.
2. Specify the size of the pie:
   - Select Auto Radius to enable BIRT Studio to set a radius for the pie, as shown in Figure 8-90.
   - Deselect Auto Radius, then specify the size of the pie in pixels.
3. Specify font properties using the drop-down lists to set font type, size, and color.

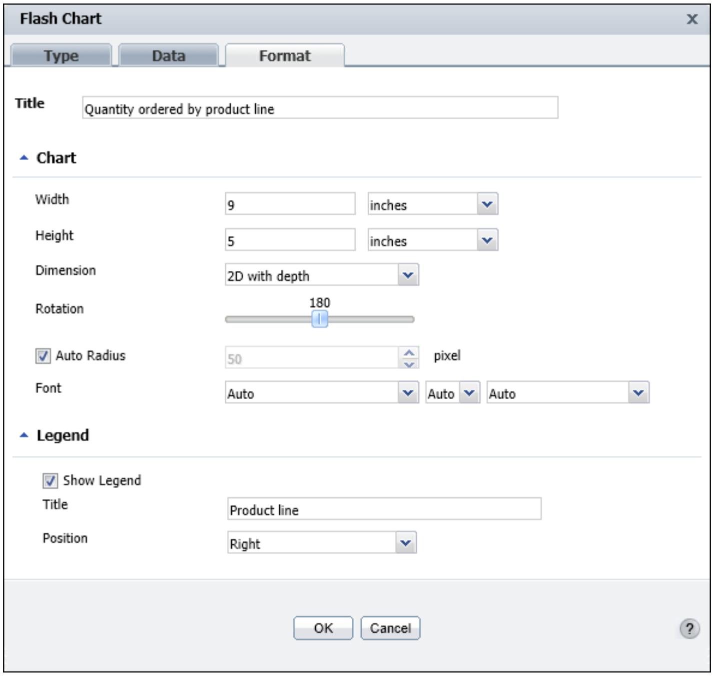


Figure 8-90 Formatting a pie Flash chart

When you format a doughnut Flash chart, you can specify the width of the doughnut ring as follows:

- Accept the default selection of Auto Outer Radius, and Auto Inner Radius to enable BIRT Studio to determine the width of the doughnut, as shown in Figure 8-91.

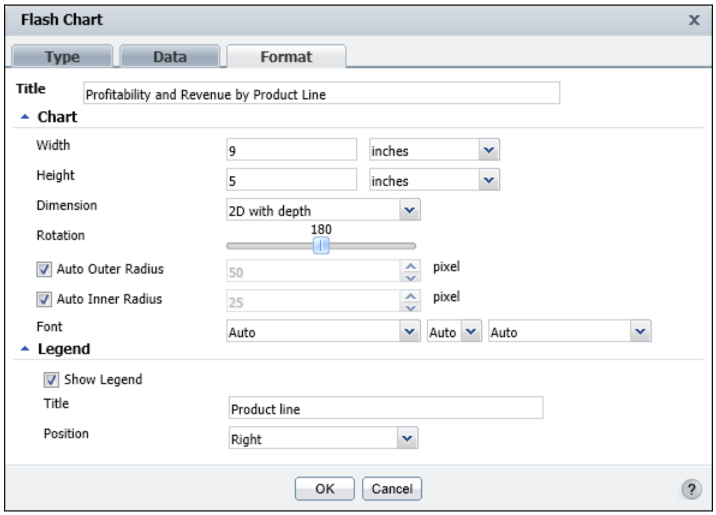


Figure 8-91 Formatting the width of a doughnut Flash chart

- Specify a different value after first deselecting Auto Outer Radius and Auto Inner Radius, by using the menus to set the width of the doughnut.

Working with HTML5 charts

As you create and format HTML5 charts, you will also discover that many of the properties available to BIRT charts are also available to HTML5 charts. In Release 11 Service Pack 4, the chart types supported include column, bar, line, area, pie, doughnut, and scatter charts.

HTML5 charts provide the following benefits:

- Animated charts that display on all computers and mobile devices. Flash charts are not supported on all mobile devices. The standard BIRT charts are designed for static, print-based documents.
- Highly customizable charts whose presentation you can control through the user interface. Flash charts support only a limited number of properties in comparison, while standard BIRT charts do not support animation.

Creating an HTML5 chart

The procedure you use to create an HTML5 chart is similar to the procedure used to create a BIRT chart. To create an HTML5 chart, perform the following tasks:

- In BIRT Studio, select Chart➛Insert HTML5 Chart.
- In the chart builder, choose a chart type.
- Select the data to use in the chart.
- Format the chart.

The example in Figure 8-92 shows an HTML5 chart displaying monthly profit values. BIRT Studio supports the following HTML5 chart types:

- Column
- Bar
- Line
- Pie
- Doughnut
- Scatter
- Area

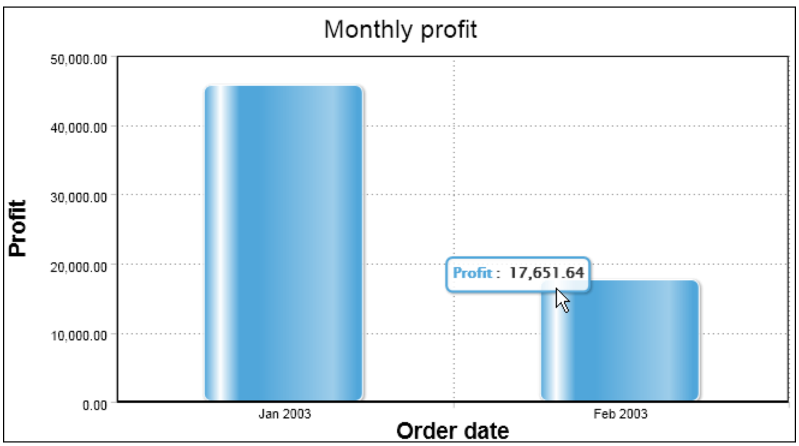


Figure 8-92 HTML5 chart displaying monthly profits for each quarter

Selecting data for an HTML5 chart

You select data for an HTML5 chart in the same way as for a regular chart.

Formatting an HTML5 chart

As in a standard chart, the following formatting options are available for an HTML5 chart:

- Specifying the chart title and size
- Specifying a subtype when applicable
- Displaying the chart using glass style
- Displaying the legend and specifying its position
- Specifying Category (X) axis properties
- Specifying Value (Y) axis properties

In addition to the formatting properties that are available for BIRT charts, HTML5 charts provide the following additional options:

- Specifying style properties for the chart by selecting a theme
- Specifying an option for the user to view and analyze details in category or value axis data

Applying a chart theme

Themes provide a flexible way to define and maintain styles in one place and reuse them for any chart that you create. BIRT Studio provides several predefined themes for HTML5 charts. On the Format Chart page, Theme displays the theme currently being used by the chart, or None, if no theme is in use. To apply a theme or to modify an existing theme, select a theme from the drop-down list.

Using a theme provides a consistent style across different types of charts. For example, a theme that uses corporate styles might define general attributes, such as color schemes for the chart background and plot areas, font styles for chart titles, value labels, or axis labels, border styles, and legend styles. The charts shown in the following examples in Figure 8-93 and Figure 8-94 display two of several themes available in BIRT Studio.

Themes are designed by the report developer, who then makes them available to users. Figure 8-93 shows a chart that uses the Chart Blue theme, while Figure 8-94 shows the same chart using the Chart Gray theme. The chart in the example in Figure 8-92 uses the Chart Grid theme.

Figure 8-93 Chart using the Chart Blue theme

Figure 8-94 Chart using the Chart Gray theme

Examining details for axis data

BIRT Studio enables you to closely analyze sections of a chart using the zoom feature available in HTML5 charts. Select this feature to enable users to zoom into data presented as categories or values.

To use this feature, complete the following steps.

In Allow Zoom into, select:

- Category (X) Axis to view details for data values on the *x*-axis
- Value (Y) Axis to view details for data values on the *y*-axis

In the example shown in Figure 8-95, the Format page displays the option to zoom into a region on the Value (Y) axis.

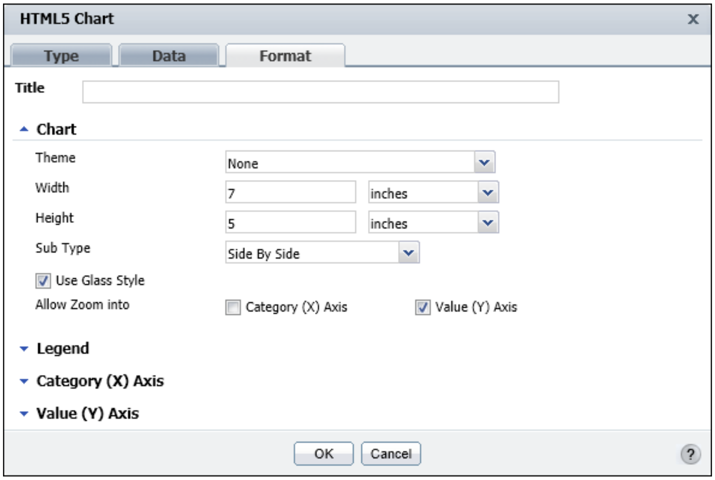


Figure 8-95 Zooming into data values to view details

On the chart, use your mouse pointer to select the region on the Value (Y) axis for which you want to view details. The example in Figure 8-96 shows a selected region on the *y*-axis.

BIRT Studio displays detailed data in the chart, for the selected section. To view the entire chart once again, select Reset Zoom, which appears on the top right corner of the chart.

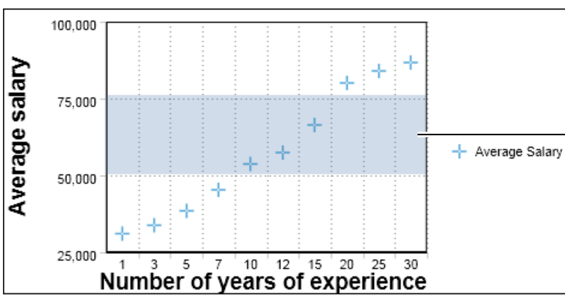


Figure 8-96 Zooming into the Value (Y) axis to view details

Deleting a chart

To delete a BIRT chart, a Flash chart, or an HTML5 chart from a report design, select the chart at the report table or section level, and choose Delete from the context menu. The chart no longer appears in the report design.

Displaying a chart without the table data

A report design that contains a lot of information in a series of tables with charts can often be more effective if it displays only the charts. To display only a chart, right-click anywhere in the table and choose Hide Table. Choose Show All to re-display the table data. You can hide the table for report designs that contain regular charts and Flash charts.

Chapter

9

Adding interactive viewing features

This chapter contains the following topics:

- About interactive viewing options
- Using bookmarks
- Using hyperlinks

About interactive viewing options

When designing a BIRT report, a report developer can provide interactive features in the report to enable end users to navigate report data more easily, and provide quick access to data in lengthy reports. In BIRT Studio, you can use the following interactive viewing features:

- Add bookmarks for users to access specific elements or sections in a report design.
- Add hyperlinks for users to access data within the same report design, other report designs in the Encyclopedia volume, and external web sites.

For example, you can use a bookmark within a report to enable a user to mark the place of a specific report element such as a chart or section within the report design. Bookmarks used with hyperlinks enable a user to navigate information within a report, or in other reports. You create bookmarks and hyperlinks using Actuate BIRT Studio, and Actuate BIRT Designer Professional. These bookmarks and hyperlinks can be used in BIRT Studio, Actuate BIRT Viewers, BIRT Dashboards, and in exported output formats such as PDF or HTML.

When BIRT Studio uses a data object as the data source for a report design, the data object can contain data fields with links to additional information such as subreports, charts, and so on. You can use BIRT Studio to either enable an end user to access these hyperlinks, or disable these hyperlinks.

The following sections describe how you create bookmarks, hyperlinks, and enable or disable default hyperlinks that can be present in BIRT data objects.

About bookmarks

A bookmark defines a position in a report design. You can use BIRT Studio to create a bookmark for a report table, a regular chart or Flash chart in a report table or section, a summary table, and a cross tab.

About hyperlinks

When designing a BIRT report, a report developer can add hyperlinks in the report. Hyperlinks enable users to navigate between information in one part of a report and related information in another part of the same report. Hyperlinks also enable users to view information stored in a different report. For example, a company sales report can contain a hyperlink that points to a more detailed report, breaking down the sales by geography and product line. In BIRT Studio, a report developer can create a hyperlink in a data column, or a BIRT chart.

Using bookmarks

When you define a bookmark, you specify a name that is used by applications such as BIRT Studio, or BIRT Dashboards to identify the bookmarked content. A bookmark defines a position in a report design, and a hyperlink provides the connection that helps the user navigate to the bookmarked element. You use hyperlinks in conjunction with bookmarks, either within the same report, or in external reports stored in the repository.

Creating a bookmark

When defining a bookmark, you specify a name that determines how BIRT Studio generates the bookmark. To create a bookmark for use in all supported BIRT applications, the bookmark name must begin with a letter ([A-Za-z]) and may be followed by any number of letters, digits ([0-9]), hyphens ("-"), underscores ("_"), colons (":"), and periods ("."). Do not enclose the name in single or double quotation marks, or use spaces or the percent symbol ("%") in the name. The letters in the name are case-sensitive. When naming a bookmark, provide a name representative of the report element being bookmarked, such as CrossTab, SummaryTable, Chart, and so on.

How to create a bookmark for a report table

1 Select the target report element for which you want to define a bookmark, such as the report table, summary table or cross tab element, and choose Bookmark from the context menu.

2 Bookmark appears, as shown in Figure 9-1. In Enter Bookmark, type a name for the bookmark.

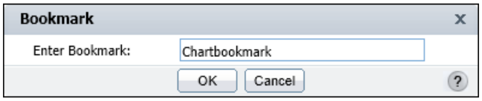


Figure 9-1 Creating a bookmark

Choose OK.

How to create a bookmark for a report element in a section

1 Select the chart or other report element in the section, and choose Bookmark from the context menu.

2 Bookmark appears, as shown in Figure 9-2. Select Generate Dynamic Bookmark Based on Section Value.

Choose OK.

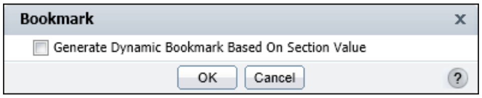


Figure 9-2 Creating a section-level bookmark

To edit a defined bookmark, select the element containing a bookmark, and choose Bookmark from the context menu. Do one of the following:

- For a table-level bookmark, in Bookmark, edit the name of the element in Enter Bookmark.
- For a section-level bookmark, in Bookmark, deselect Generate Dynamic Bookmark Based on Section Value.

To delete a defined bookmark, in Bookmark, delete the name of the element in Enter Bookmark.

The report elements for which you define bookmarks in BIRT Studio can be displayed in BIRT Dashboards using a Reportlet gadget. For more information, see *Building BIRT Dashboards.*

Modifying a bookmark

To edit an existing bookmark, select the element containing the bookmark, and choose Bookmark from the context menu. Then, for a table-level bookmark, in Bookmark, edit the name of the element in Enter Bookmark.

Deleting a bookmark

To delete an existing bookmark at the report table level, in Bookmark, delete the name of the element in Enter Bookmark. If you are working with a bookmark at the report section level, deselect Generate Dynamic Bookmark Based on Section Value.

Using hyperlinks

When reports are too long and complex in terms of the data they display, it becomes difficult for users to easily find the information they need. If these reports are used online, add interactive viewing features to help users easily navigate and explore detailed and lengthy reports. For example, if you have a multipage report design containing a chart in a report table, using a hyperlink from a data column to the chart helps users access the chart easily, while navigating the report, eliminating the need to page back and forth in the report design.

BIRT Studio supports the creation of dynamic data-driven expressions using EasyScript. EasyScript is an expression syntax similar to the syntax used in Excel formulas. Like Excel, EasyScript provides functions for performing calculations on report data. In BIRT Studio, EasyScript is supported in Hyperlink Builder. For example, when a drill-through hyperlink is created to an external report containing static filter parameters, you can create an EasyScript expression to specify the data field for which a user selects parameter values. Similarly, when creating a hyperlink that links to a URL, you can specify an EasyScript expression to URL link for each row in a data field. In Release 11 Service Pack 4, BIRT Studio does not support creating a dynamic data-driven expression for required parameters where no default value is supplied.

You can also create a hyperlink to a bookmark in an external report, as described later in this document. When linking to an external report, you can create a hyperlink to a report design, or report document file, or a file in an output format supported when exporting a report from the Actuate BIRT Viewers. Supported formats include PDF, PostScript, HTML, Microsoft Word, Excel, PowerPoint, and so on. BIRT Studio does not support creating hyperlinks in Flash charts.

Choosing the hyperlink in the source report launches the target cross tab report displaying sales details for all countries, as shown in Figure 9-3. The target report must be stored in the repository.

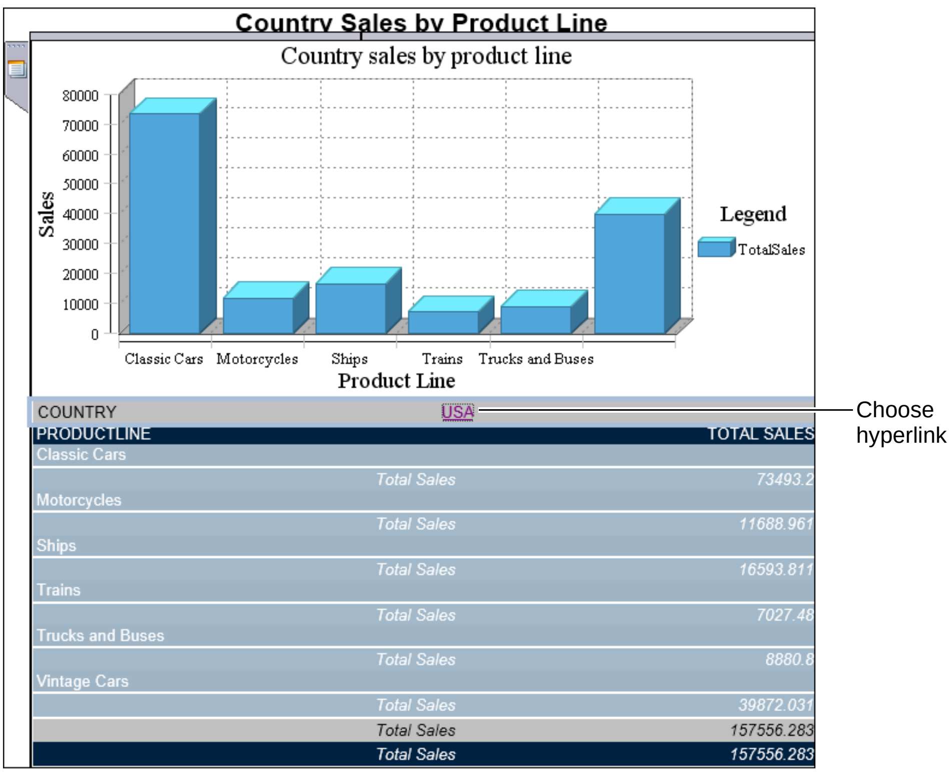
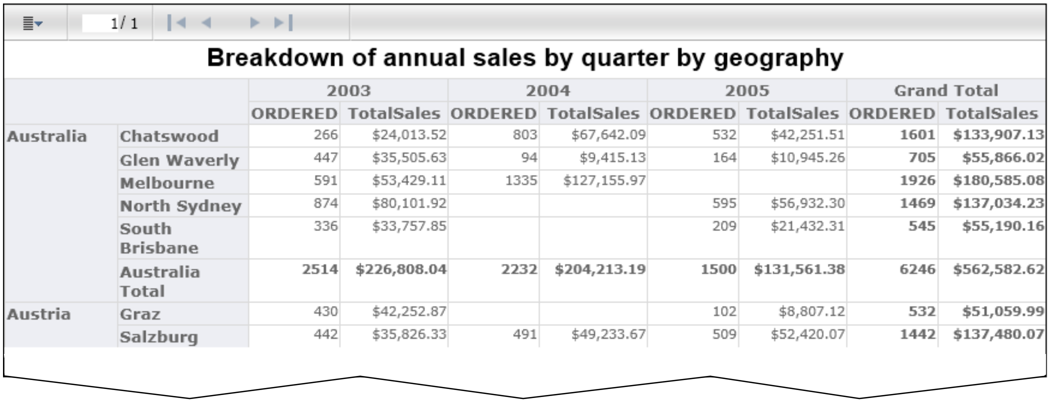


Figure 9-3 Choosing a hyperlink in the source report

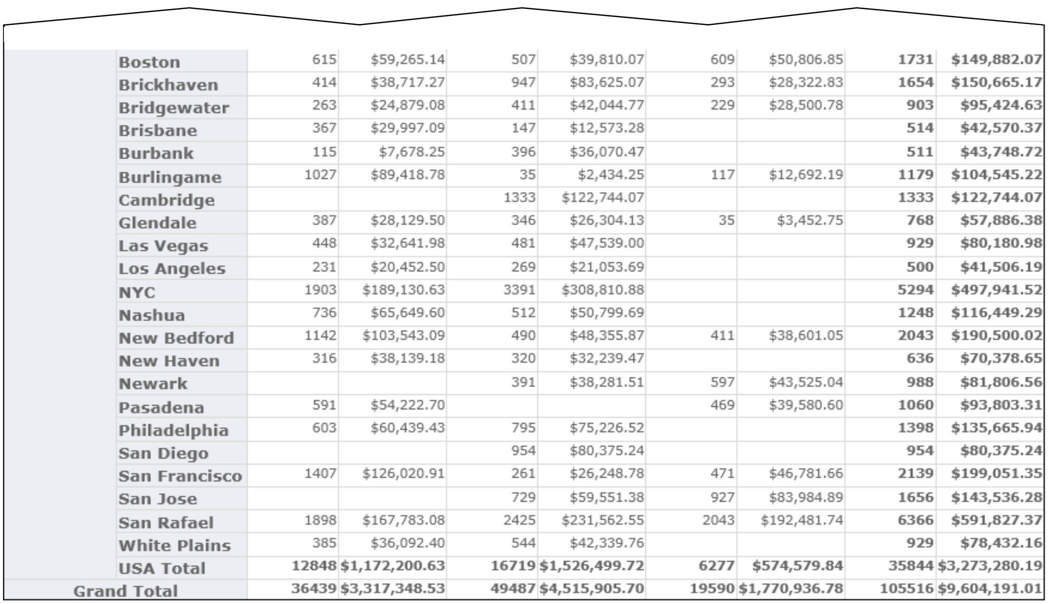
Figure 9-4 and Figure 9-5 display the quarterly sales and order quantity details by city, and country for three years. When a user selects the hyperlink in the source report, Actuate BIRT Viewer displays the target cross tab report.



Breakdown of annual sales by quarter by geography

		2003		2004		2005		Grand Total	
		ORDERED	TotalSales	ORDERED	TotalSales	ORDERED	TotalSales	ORDERED	TotalSales
Australia	Chatswood	266	$24,013.52	803	$67,642.09	532	$42,251.51	1601	$133,907.13
	Glen Waverly	447	$35,505.63	94	$9,415.13	164	$10,945.26	705	$55,866.02
	Melbourne	591	$53,429.11	1335	$127,155.97			1926	$180,585.08
	North Sydney	874	$80,101.92			595	$56,932.30	1469	$137,034.23
	South Brisbane	336	$33,757.85			209	$21,432.31	545	$55,190.16
	Australia Total	2514	$226,808.04	2232	$204,213.19	1500	$131,561.38	6246	$562,582.62
Austria	Graz	430	$42,252.87			102	$8,807.12	532	$51,059.99
	Salzburg	442	$35,826.33	491	$49,233.67	509	$52,420.07	1442	$137,480.07

Figure 9-4 Viewing bookmarked content in the target cross tab report



	Boston	615	$59,265.14	507	$39,810.07	609	$50,806.85	1731	$149,882.07
	Brickhaven	414	$38,717.27	947	$83,625.07	293	$28,322.83	1654	$150,665.17
	Bridgewater	263	$24,879.08	411	$42,044.77	229	$28,500.78	903	$95,424.63
	Brisbane	367	$29,997.09	147	$12,573.28			514	$42,570.37
	Burbank	115	$7,678.25	396	$36,070.47			511	$43,748.72
	Burlingame	1027	$89,418.78	35	$2,434.25	117	$12,692.19	1179	$104,545.22
	Cambridge			1333	$122,744.07			1333	$122,744.07
	Glendale	387	$28,129.50	346	$26,304.13	35	$3,452.75	768	$57,886.38
	Las Vegas	448	$32,641.98	481	$47,539.00			929	$80,180.98
	Los Angeles	231	$20,452.50	269	$21,053.69			500	$41,506.19
	NYC	1903	$189,130.63	3391	$308,810.88			5294	$497,941.52
	Nashua	736	$65,649.60	512	$50,799.69			1248	$116,449.29
	New Bedford	1142	$103,543.09	490	$48,355.87	411	$38,601.05	2043	$190,500.02
	New Haven	316	$38,139.18	320	$32,239.47			636	$70,378.65
	Newark			391	$38,281.51	597	$43,525.04	988	$81,806.56
	Pasadena	591	$54,222.70			469	$39,580.60	1060	$93,803.31
	Philadelphia	603	$60,439.43	795	$75,226.52			1398	$135,665.94
	San Diego			954	$80,375.24			954	$80,375.24
	San Francisco	1407	$126,020.91	261	$26,248.78	471	$46,781.66	2139	$199,051.35
	San Jose			729	$59,551.38	927	$83,984.89	1656	$143,536.28
	San Rafael	1898	$167,783.08	2425	$231,562.55	2043	$192,481.74	6366	$591,827.37
	White Plains	385	$36,092.40	544	$42,339.76			929	$78,432.16
	USA Total	12848	$1,172,200.63	16719	$1,526,499.72	6277	$574,579.84	35844	$3,273,280.19
Grand Total		36439	$3,317,348.53	49487	$4,515,905.70	19590	$1,770,936.78	105516	$9,604,191.01

Figure 9-5 Viewing the target cross tab report

Creating a hyperlink

BIRT Studio provides a convenient Hyperlink Builder wizard to create a hyperlink. You can link two sections or elements in the same report, link a column or element to an external report, or link a column or element to an external web site. The following section explains the procedure to create a hyperlink.

How to create a hyperlink

To create a hyperlink, do the following:

1. Select the report column or element, then choose Hyperlink from the context menu. Hyperlink Builder appears. Use Hyperlink Builder to specify the properties for the hyperlink you want to create.
2. On Hyperlink Builder, first select the type of hyperlink to use, based on your reporting needs.
   - To deactivate any existing hyperlinks in the selected column or element, select No link.
   - To direct a user to information in an external web site, select URL.
   - To enable a user to navigate a lengthy report containing several report elements more easily, select Bookmark in this report.
   - To enable a user to view additional information in an external report, select Drill-Through.
3. Depending on the type of hyperlink you choose, BIRT Studio prompts you to provide additional information such as the bookmark to use, if any, the location of the target external report, the format of the external file, and an optional tooltip to assist the user.

Figure 9-6 shows an example of Hyperlink Builder with options specified to create a bookmark within the same report.

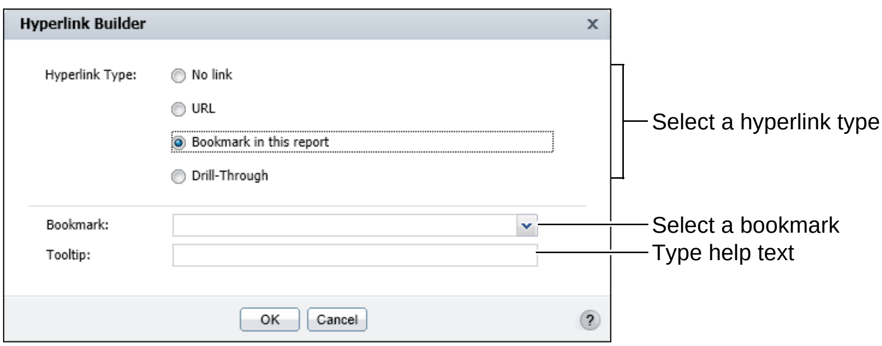


Figure 9-6 Creating a hyperlink

The following section discusses the types of hyperlinks available in BIRT Studio and describes the process to create each type of hyperlink.

Linking two sections in a report

To link one report element to another element in the same report, use a bookmark and a hyperlink. First, define a bookmark for the target report element. Then, define a hyperlink in the source report element. Perform these tasks in the recommended sequence because to successfully create a hyperlink you need the bookmark information.

For information about how to create a bookmark, see "Creating a bookmark," earlier in this chapter. After creating the bookmark, you create a hyperlink that links to the bookmark. Figure 9-7 shows an example that uses a bookmark and hyperlink to link two charts in the report.

Because the report is a multipage report, the section-level pie chart uses a hyperlink for users to easily navigate between the current page, and the bar chart at the beginning of the report. The bar chart contains a bookmark, ReportTableChart, that the hyperlink uses.

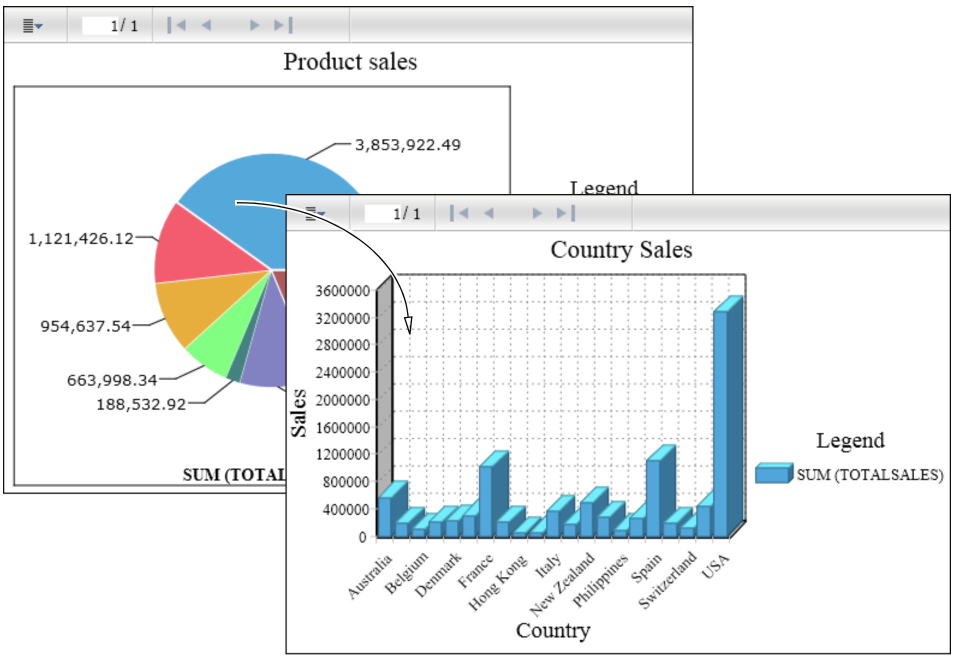


Figure 9-7 Viewing a hyperlink source and target report

How to create a hyperlink that links two sections in a report

1 Select the report column or report element for which you want to create a hyperlink. Then choose Hyperlink from the context menu.

2 Hyperlink Builder appears. On Hyperlink Builder, do the following:

   1 In Hyperlink Type, select Bookmark in this report.

2 In Bookmark, select the bookmark from the list. Figure 9-8 uses the ReportTableChart bookmark.

3 In Tooltip, optionally type the text to display when the user hovers the mouse pointer over the hyperlink. Choose OK.

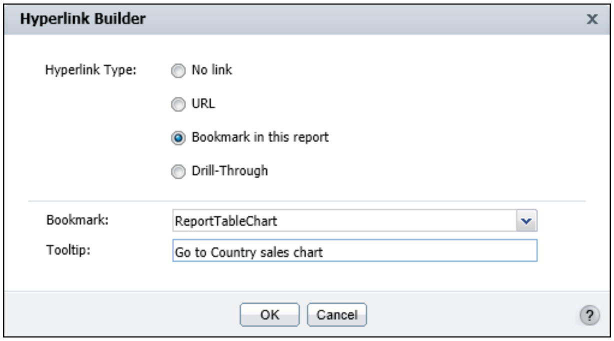


Figure 9-8 Creating a bookmark to link sections in the same report

The report design displays the hyperlink based on the options you specified.

The example in Figure 9-7 displays the result of the hyperlink options shown in Figure 9-8.

Linking to a section in a different report

When you create a link from one report to a different report, or a section in a different report, the procedure you follow is similar to creating a link between sections or elements in the same report. You begin by creating a bookmark in the target report. Then create a hyperlink in the source report. This type of hyperlink is called a drill-through hyperlink. For drill-through hyperlink expressions, the EasyScript builder validates expressions using data bindings that the target report design and the source report design share. The EasyScript builder does not validate expressions containing target report data bindings that the source report design does not share.

For information about how to create a bookmark, see "Creating a bookmark," earlier in this chapter. After creating the bookmark, you create a hyperlink that links to the bookmark. In this example, you create a drill-through hyperlink to a report stored in the repository that displays sales details by state in the USA.

How to create a drill-through hyperlink

1 Select the report column or report element for which you want to create a hyperlink. Then choose Hyperlink from the context menu.

2 Hyperlink Builder appears. On Hyperlink Builder, do the following:

   1 In Hyperlink Type, select Drill-Through.

2 In Report, select the folder icon. Open File appears, as shown in Figure 9-9.

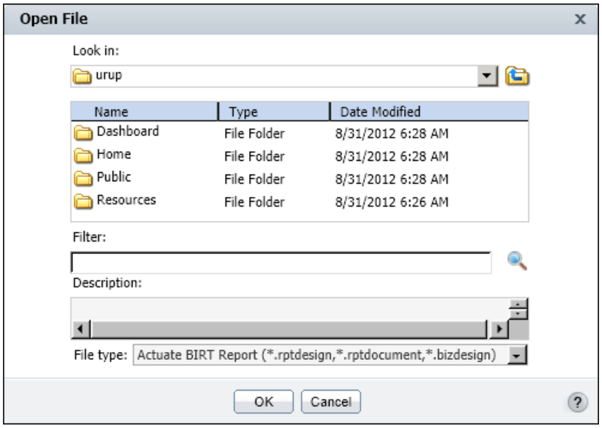


Figure 9-9 Navigating the folder structure to select a target report

3 In Open File, complete the following steps:

  1 Navigate through the file repository to select a folder containing the target report.

  2 If you know the name of the file you want to specify, type the name, or relevant keywords in Filter. Then, choose Find. Open File displays the reports that match the string you provided.

  3 In Description, optionally type a description for the report.

  4 In File type, select Actuate BIRT Report if the target report is a BIRT design (.rptdesign) or BIRT document (.rptdocument) file.

  Choose OK.

4 In Hyperlink Builder, choose More. Hyperlink Builder displays additional options.

5 In Bookmark, select the bookmark in the target report to which to link. If you do not specify a bookmark, the hyperlink points to the beginning of the target report.

6 In Target, do one of the following:

- Select Current browser window or tab.
- Select New browser window or tab.

7 In File Format, if necessary, select a format in which to display the target report.

8 In Tooltip, optionally type the text to display when the user hovers the mouse pointer over the hyperlink, as shown in Figure 9-10.

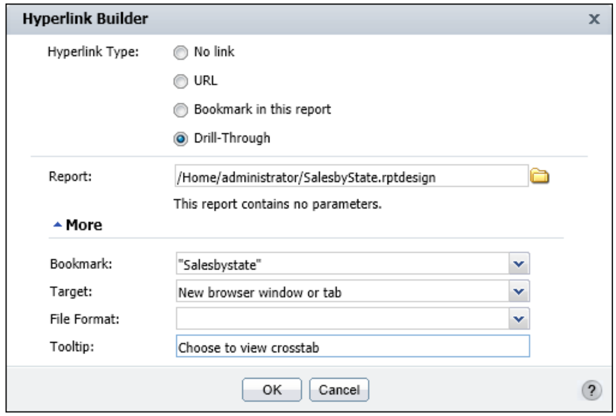


Figure 9-10 Creating a drill-through hyperlink to another report

Choose OK.

The report design displays the hyperlink based on the options you specified.

You can customize the data displayed in the target report and improve performance by using static filter parameters in the target report. For example, instead of generating a target report that shows sales for all states in a country, use parameters to filter report data by state in the target report. The report displays sales details only for the state that the user specifies in the hyperlink builder dialog box, shown in Figure 9-11. Generating a filtered report is typically more efficient than generating a report that displays all the data.

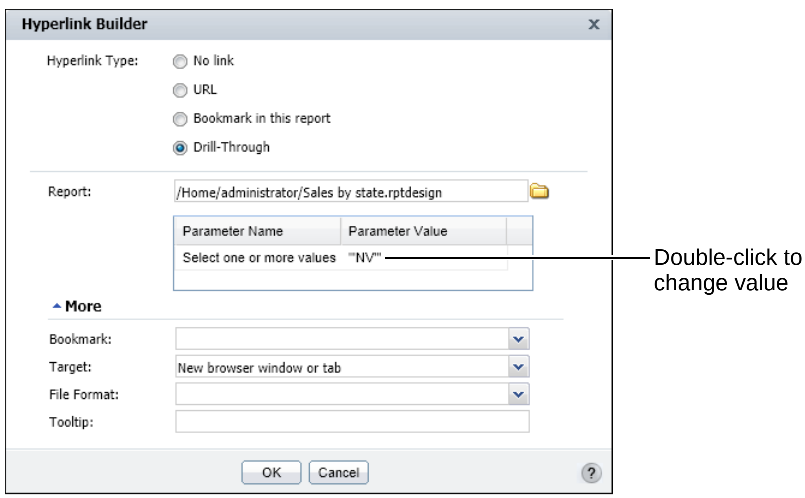


Figure 9-11 Modifying a parameter value

For the value specified in the example shown in Figure 9-11, Actuate BIRT Viewer displays sales details for Nevada, as shown in Figure 9-12.

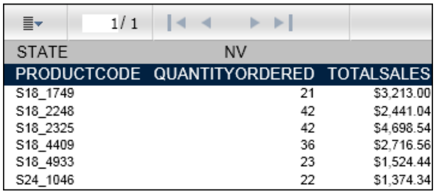


Figure 9-12 Viewing a target report based on a parameter value

To set a parameter value based on a data field in the report, you can use the BIRT Studio EasyScript builder to specify the field to use. Complete the following steps:

1. Double-click the value enclosed in double quotation marks (" "), as shown in Figure 9-11. The EasyScript builder appears.
2. On the EasyScript builder, in Enter Expression, type the left bracket ([), and select a data field from the list of fields that appears. Enclose the value in double quotation marks (" "). The example in Figure 9-13 shows the selection of the data field, State.
3. Select validate to make sure the syntax for parameter value expression is correct.
4. After a message appears confirming that the expression is valid, as shown in Figure 9-13, choose OK.

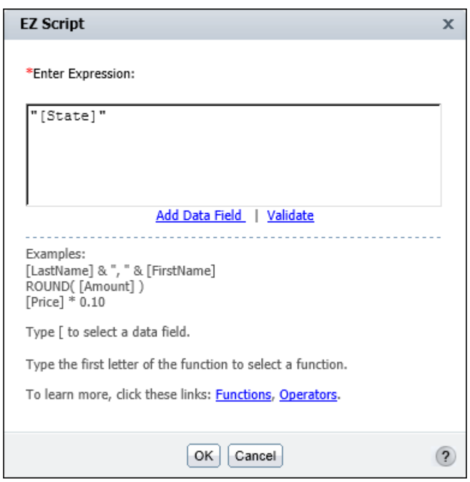


Figure 9-13 Expression syntax confirmation message

5. Hyperlink Builder displays the parameter expression you specified, as shown in Figure 9-14.

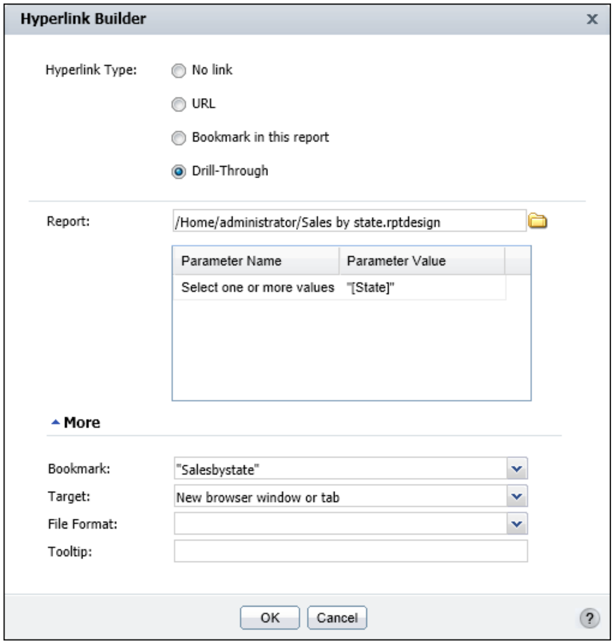


Figure 9-14 Displaying a parameter expression

6 Run the report to make sure that the data displayed is what you expect.

When a user chooses the hyperlink, the target report uses the specified data field to prompt the user to specify a parameter value to retrieve data to display. For example, to view sales details for Massachusetts, select Massachusetts from the list of values in Parameter. In the target report, make sure that you set a default value for the static filter parameter, if the parameter is required. BIRT Studio does not require you to specify a default value for an optional static filter parameter.

Linking to a URL

You can also use hyperlinks to open an external document or web page by providing a Uniform Resource Locator (URL). These links provide users with easy access to additional sources of information creating more interactive reports.

For example, Figure 9-15 shows a report displaying weekly stock performance. To provide a user with additional information about the company, you can create a hyperlink so that choosing the name of the company launches the company web site.

Figure 9-16 shows the options you specify to create a hyperlink that directs a user to an external URL. In the example shown in Figure 9-15, choosing MYCO directs a user to the web site, www.mycompany.com.

Company	Date	Open	Close	Low	High
MYCO	8/3/08	23.21	23.25	23.19	23.35
MYCO	8/4/08	23.24	23.3	23.19	23.31
MYCO	8/5/08	23.31	23.5	23.29	23.55
MYCO	8/6/08	23.55	23.52	23.48	23.56
MYCO	8/7/08	23.55	24	23.54	24.1

Figure 9-15 Displaying a hyperlink to an external URL

How to create a hyperlink to an external URL

1 Select the column for which you want to create a hyperlink, then select Hyperlink from the context menu.

2 Hyperlink Builder appears. On Hyperlink Builder, shown in Figure 9-16, do the following:

   1 In Hyperlink Type, select URL.

   2 In Location, using the EasyScript builder, do the following

      - Type the URL address to link all values of the data field.
      - Construct an expression for each data row to link to a unique URL.

   3 In Target, do one of the following:

      - Select Current browser window or tab.
      - Select New browser window or tab.

   4 In Tooltip, optionally type the text to display when the user hovers the mouse pointer over the hyperlink.

   Choose OK. The report design displays the hyperlink based on the options you specified.

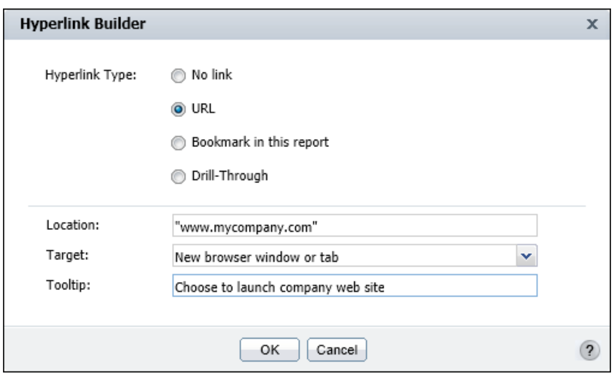


Figure 9-16 Linking to an external URL

Modifying a hyperlink

To modify a hyperlink, select the report element containing the hyperlink, then select Hyperlink from the context menu. On Hyperlink Builder, modify the type of hyperlink and other options as needed. Choose OK.

The report element displays the modified hyperlink.

Deleting a hyperlink

To delete a hyperlink, select the report element containing the hyperlink, then select Hyperlink from the context menu. On Hyperlink Builder, in Hyperlink Type, select No link. Choose OK.

The report element no longer displays the hyperlink.

Chapter

10

Functions and operators

This chapter contains the following topics:

- Functions
- Functions used in computed column expressions
- Functions used in aggregate calculations
- Operators

Functions

This section is a complete reference to all of the functions in BIRT Studio. This reference organizes the functions into two sections based on those used in expressions when creating a computed column and functions that are used to perform aggregate calculations.

Functions used in computed column expressions

The following list of functions appear when you create expressions to compute column data. Each function entry includes a general description of the function, its syntax, the arguments to the function, the result the function returns, and an example that shows typical usage. Use this reference to find information about a function that you want to use when you insert a computed column to display calculated data in a report design.

ABS( )

Returns the absolute value of a number without regard to its sign. For example, 6 is the absolute value of 6 and -6.

Syntax ABS(num)

Argument **num**
The number, or numeric expression that specifies the number, for which you want to find the absolute value.

Returns A number that represents the absolute value of num.

Example The following example returns the absolute value for each number in the TemperatureCelsius data field:

```
ABS([TemperatureCelsius])
```

ADD_DAY( )

Adds a specified number of days to a date value.

Syntax ADD_DAY(date, daysToAdd)

Arguments **date**
The date or date expression that represents the start date.

daysToAdd
The number of days to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date value that results from adding the specified number of days to the start date.

Example The following example adds 15 days to each date value in the InvoiceDate data field:

```
ADD_DAY([InvoiceDate], 15)
```

ADD_HOUR( )

Adds a specified number of hours to a date value.

Syntax ADD_HOUR(date, hoursToAdd)

Arguments **date**
The date or date expression that represents the start date. If a start date does not have a time value, the function assumes the time is midnight, 12:00 AM.

hoursToAdd
The number of hours to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date-and-time value that results from adding the specified number of hours to the start date.

Example The following example adds eight hours to each date value in the ShipDate data field:

```
ADD_HOUR([ShipDate], 8)
```

ADD_MINUTE( )

Adds a specified number of minutes to a date value.

Syntax ADD_MINUTE(date, minutesToAdd)

Arguments **date**
The date or date expression that represents the start date. If a start date does not have a time value, the function assumes the time is midnight, 12:00 AM.

minutesToAdd
The number of minutes to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date-and-time value that results from adding the specified number of minutes to the start date.

Example The following example subtracts 30 minutes from each date in the StartTime data field:

```
ADD_MINUTE([StartTime], -30)
```

ADD_MONTH( )

Adds a specified number of months to a date value.

Syntax ADD_MONTH(date, monthsToAdd)

Arguments **date**
The date or date expression that represents the start date.

monthsToAdd
The number of months to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date value that results from adding the specified number of months to the start date. This function always returns a valid date. If necessary, the day part of the resulting date is adjusted downward to the last day of the resulting month in the resulting year. For example, if you add one month to 1/31/08, ADD_MONTH( ) returns 2/29/08, not 2/31/08 or 2/28/08, because 2008 is a leap year.

Example The following example adds two months to each date value in the InitialRelease data field:

```
ADD_MONTH([InitialRelease], 2)
```

ADD_QUARTER( )

Adds a specified number of quarters to a date value.

Syntax ADD_QUARTER(date, quartersToAdd)

Arguments **date**
The date or date expression that represents the start date.

quartersToAdd
The number of quarters to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date value that results from adding the specified number of quarters to the start date. A quarter is equal to three months. For example, if you add two quarters to 9/22/08, ADD_QUARTER( ) returns 3/22/09.

Example The following example adds two quarters to each date value in the ForecastClosing data field:

```
ADD_QUARTER([ForecastClosing], 2)
```

ADD_SECOND( )

Adds a specified number of seconds to a date value.

Syntax ADD_SECOND(date, secondsToAdd)

Arguments **date**
The date or date expression that represents the start date. If a start date does not have a time value, the function assumes the time is midnight, 12:00 AM.

secondsToAdd
The number of seconds to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date-and-time value that results from adding the specified number of seconds to the start date.

Example The following example adds 30 seconds to each date value in the StartTime data field:

```
ADD_SECOND([StartTime], 30)
```

ADD_WEEK( )

Adds a specified number of weeks to a date value.

Syntax ADD_WEEK(date, weeksToAdd)

Arguments **date**
The date or date expression that represents the start date.

weeksToAdd
The number of weeks to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date value that results from adding the number of weeks to the start date.

Example The following example adds two weeks to each date value in the OrderDate data field:

```
ADD_WEEK([OrderDate], 2)
```

ADD_YEAR( )

Adds a specified number of years to a date value.

Syntax ADD_YEAR(date, yearsToAdd)

Arguments **date**
The date or date expression that represents the start date.

yearsToAdd
The number of years to add to the start date. If you specify a negative number, the result appears to subtract the number from the start date.

Returns The date value that results from adding the number of years to the start date.

Example The following example adds five years to each date value in the HireDate data field:

```
ADD_YEAR([HireDate], 5)
```

BETWEEN( )

Tests if a value is between two specified values.

Syntax BETWEEN(value, upperBound, lowerBound)

Arguments **value**
The value to test. The value can be a string, numeric, or date value.

upperBound
The first value in the range of values to which to compare. String and date values must be enclosed in double quotation marks (" ").

lowerBound
The second value in the range of values to which to compare. String and date values must be enclosed in double quotation marks (" ").

Returns True if value is between upperBound and lowerBound, or equal to upperBound or lowerBound; returns false otherwise.

Examples The following example tests each value in the SalesTotal data field to see if the value is between 10000 and 20000:

```
BETWEEN([SalesTotal], 10000, 20000)
```

The following example tests each value in the CustomerName data field to see if the value is between A and M:

```
BETWEEN([CustomerName], "A", "M")
```

The following example tests each value in the ReceiptDate data field to see if the value is between 10/01/07 and 12/31/07:

```
BETWEEN([ReceiptDate], "10/01/07 12:00 AM", "12/31/07 12:00 AM")
```

The following example uses BETWEEN( ) in conjunction with the IF( ) and ADD_DAY( ) functions to calculate a shipment date. If an orderDate value is in December 2007 (between 12/1/07 and 12/31/07), add five days to the orderDate value. If an orderDate value is in a month other than December, add three days to the orderDate value.

```
IF((BETWEEN([orderDate], "12/1/07 12:00 AM", "12/31/07 12:00
   AM")), (ADD_DAY([orderDate], 5)), (ADD_DAY([orderDate], 3)))
```

CEILING( )

Rounds a number up to the nearest specified multiple.

Syntax CEILING(num, significance)

Arguments

num
The numeric value to round up.

significance
The multiple up to which to round num.

Returns The number that results from the rounding. If num is an exact multiple of significance, no rounding occurs.

Examples CEILING( ) is commonly used to round up prices. For example, to avoid dealing with pennies, if the Price value is 20.52, CEILING( ) returns 20.55. You can round prices in a Price data field up to the nearest nickel with the following expression:

```
CEILING([Price], 0.05)
```

If the Price value is 20.52, CEILING( ) returns 20.60. If the Price value is 20.40, CEILING( ) returns 20.40. No rounding occurs because 20.40 is already a multiple of 0.1. The following example rounds prices up to the nearest dime:

```
CEILING([Price], 0.1)
```

The following example rounds prices up to the nearest dollar. If the Price value is 20.30, CEILING( ) returns 21.0.

```
CEILING([Price], 1)
```

DAY( )

Returns a number from 1 to 31 that represents the day of the month.

Syntax DAY(date)

Argument **date**
The date or date expression from which you want to extract the day.

Returns The number of the day of the month for the specified date value.

Example The following example gets the number of the day for each date value in the ShipDate data field:

```
DAY([ShipDate])
```

DIFF_DAY( )

Calculates the number of days between two date values.

Syntax DIFF_DAY(date1, date2)

Arguments **date1**
The first date or date expression to use in the calculation.

date2
The second date or date expression to use in the calculation.

Returns The number of days between date1 and date2.

Example The following example calculates the number of days between each value in the invoiceDate data field and each value in the paymentDate data field:

```
DIFF_DAY([invoiceDate],[paymentDate])
```

The results show how long it takes to pay invoices.

The following example calculates the number of days from an order date to Christmas:

```
DIFF_DAY([orderDate], "12/25/08")
```

The following example calculates the number of days from the current date to Christmas. TODAY( ) is a function that returns the current date:

```
DIFF_DAY(TODAY(), "12/25/08")
```

DIFF_HOUR( )

Calculates the number of hours between two date values.

Syntax DIFF_HOUR(date1, date2)

Arguments **date1**
The first date or date expression to use in the calculation. If the date does not have a time value, the function assumes the time is midnight, 12:00 AM.

date2
The second date or date expression to use in the calculation. If the date does not have a time value, the function assumes the time is midnight, 12:00 AM.

Returns The number of hours between date1 and date2.

Example The following example calculates the number of hours between each value in the startTime data field and each value in the finishTime data field:

```
DIFF_HOUR([startTime],[finishTime])
```

The following example calculates the number of hours from the current date to Christmas. NOW( ) is a function that returns the current date and time. If you supply a literal date as an argument, you must include the time value, as shown in the following example:

```
DIFF_HOUR(NOW(), "12/25/08 12:00 AM")
```

DIFF_MINUTE( )

Calculates the number of minutes between two date values.

Syntax DIFF_MINUTE(date1, date2)

Arguments **date1**
The first date or date expression to use in the calculation. If the date does not have a time value, the function assumes the time is midnight, 12:00 AM.

date2
The second date or date expression to use in the calculation. If the date does not have a time value, the function assumes the time is midnight, 12:00 AM.

Returns The number of minutes between date1 and date2.

Example The following example calculates the number of minutes between each value in the startTime data field and each value in the finishTime data field:

```
DIFF_MINUTE([startTime],[finishTime])
```

The following example calculates the number of minutes from the current date to Christmas. NOW( ) is a function that returns the current date and time. If you supply a literal date as an argument, you must include the time value, as shown in the following example:

```
DIFF_MINUTE(NOW(), "12/25/08 12:00 AM")
```

DIFF_MONTH( )

Calculates the number of months between two date values.

Syntax DIFF_MONTH(date1,date2)

Arguments **date1**
The first date or date expression to use in the calculation.

date2
The second date or date expression to use in the calculation.

Returns The number of months between date1 and date2. The function calculates the difference by subtracting the month number of date1 from the month number of date2. For example, if date1 is 8/1/08 and date2 is 8/31/08, DIFF_MONTH( ) returns 0. If date1 is 8/25/08 and date2 is 9/5/08, DIFF_MONTH( ) returns 1.

Example The following example calculates the number of months between each value in the askByDate data field and each value in the ShipByDate data field:

```
DIFF_MONTH([askByDate],[shipByDate])
```

The following example calculates the number of months from each value in the hireDate data field to the end of the year:

```
DIFF_MONTH([hireDate], "1/1/09")
```

DIFF_QUARTER( )

Calculates the number of quarters between two date values.

Syntax DIFF_QUARTER(date1, date2)

Arguments **date1**
The first date or date expression to use in the calculation.

date2
The second date or date expression to use in the calculation.

Returns The number of quarters between date1 and date2. DIFF_QUARTER calculates the difference by subtracting the month number of date1 from the month number of date2. A difference of 3 months is equal to 1 quarter. For example, if date1 is 8/1/08 and date2 is 10/31/08, DIFF_QUARTER( ) returns 0. If date1 is 8/25/08 and date2 is 11/5/08, DIFF_QUARTER( ) returns 1.

Example The following example calculates the number of quarters between each value in the PlanClosing data field and each value in the ActualClosing data field:

```
DIFF_QUARTER([PlanClosing],[ActualClosing])
```

The following example calculates the number of quarters from each value in the orderDate data field to the end of the year:

```
DIFF_QUARTER([orderDate], "1/1/09")
```

DIFF_SECOND( )

Calculates the number of seconds between two date values.

Syntax DIFF_SECOND(date1, date2)

Arguments

date1
The first date or date expression to use in the calculation. If the date does not have a time value, the function assumes the time is midnight, 12:00 AM.

date2
The second date or date expression to use in the calculation. If the date does not have a time value, the function assumes the time is midnight, 12:00 AM.

Returns The number of seconds between date1 and date2.

Example The following example calculates the number of seconds between each value in the startTime data field and each value in the finishTime data field:

```
DIFF_SECOND([startTime],[finishTime])
```

The following example calculates the number of seconds from the current date to Christmas. NOW( ) is a function that returns the current date and time. If you supply a literal date as an argument, you must include the time value, as shown in the following example:

```
DIFF_SECOND(NOW(), "12/24/08 12:00 AM")
```

DIFF_WEEK( )

Calculates the number of weeks between two date values.

Syntax DIFF_WEEK(date1, date2)

Arguments

date1
The first date or date expression to use in the calculation.

date2
The second date or date expression to use in the calculation.

Returns The number of weeks between date1 and date2. The function calculates the difference by subtracting the week number of date1 from the week number of date2. For example, if date1 is 1/3/08 (week 1 of the year), and date2 is 1/7/08 (week 2 of the year), DIFF_WEEK( ) returns 1.

Example The following example calculates the number of weeks between each value in the askByDate data field and each value in the shipByDate data field:

```
DIFF_WEEK([askByDate],[shipByDate])
```

The following example calculates the number of weeks from each value in the orderDate data field to the end of the year:

```
DIFF_WEEK([orderDate], "1/1/09")
```

DIFF_YEAR( )

Calculates the number of years between two date values.

Syntax DIFF_YEAR(date1, date2)

Arguments

date1
The first date or date expression to use in the calculation.

date2
The second date or date expression to use in the calculation.

Returns The number of years between date1 and date2. The function calculates the difference by subtracting the year number of date1 from the year number of date2. For example, if date1 is 1/1/08 and date2 is 12/31/08, DIFF_YEAR( ) returns 0. If date1 is 11/25/08 and date2 is 1/5/09, DIFF_YEAR( ) returns 1.

Example The following example calculates the number of years between each value in the HireDate data field and each value in the TerminationDate data field:

```
DIFF_YEAR([HireDate],[TerminationDate])
```

The following example calculates the number of years from each value in the HireDate data field to the current date. TODAY( ) is a function that returns the current date.

```
DIFF_YEAR([HireDate], TODAY())
```

FIND( )

Finds the location of a substring in a string.

Syntax FIND(strToFind, str)

FIND(strToFind, str, startPosition)

Arguments

strToFind
The substring to search for. The search is case-sensitive.

str
The string in which to search.

startPosition
The position in str where the search starts.

Returns The numerical position of the substring in the string. The first character of a string starts at 1. If the substring is not found, FIND( ) returns 0.

Example The following example searches for the substring, Ford, in each ProductName value:

```
FIND("Ford", [ProductName])
```

If the product name is 1969 Ford Falcon, FIND( ) returns 6.

The following example searches for the first hyphen (-) in each product code:

```
FIND("-", [ProductCode])
```

If the product code is ModelA-1234-567, FIND( ) returns 7.

The following example uses FIND( ) in conjunction with the LEFT( ) function to display the characters that precede the hyphen in a product code. The LEFT( ) function extracts a substring of a specified length, starting from the first character. In this example, the length of the substring to display is equal to the numerical position of the hyphen character.

```
LEFT([ProductCode], FIND("-", [ProductCode]))
```

If the product code is ModelA-1234, LEFT( ) returns the following string:

```
ModelA
```

IF( )

Returns one value if a specified condition evaluates to TRUE, or another value if the condition evaluates to FALSE.

Syntax IF(condition, doIfTrue, doIfFalse)

Arguments **condition**
The condition to test.

doIfTrue
The value to return if condition evaluates to TRUE.

doIfFalse
The value to return if condition evaluates to FALSE.

Returns Returns the doIfTrue value if condition is TRUE or the doIfFalse value if condition is FALSE.

Example The following example calculates and displays different discount amounts based on the value in the Total data field. If the Total value is greater than 5000, the discount is 15%. Otherwise, the discount is 10%.

```
IF([Total]>5000, [Total]*15%, [Total]*10%)
```

The following example uses IF( ) in conjunction with the BETWEEN( ) and ADD_DAY( ) functions to calculate a shipment date. If an orderDate value is in December 2007 (between 12/1/07 and 12/31/07), add 5 days to the orderDate value. If an orderDate value is in a month other than December, add 3 days to the orderDate value.

```
IF((BETWEEN([orderDate], "12/1/07 12:00 AM", "12/31/07 12:00
   AM")), (ADD_DAY([orderDate], 5)), (ADD_DAY([orderDate], 3)))
```

The following example checks each value in the Office data field. If the value is Boston, San Francisco, or NYC, the computed column displays U.S. If the value is something other than Boston, San Francisco, or NYC, the computed column displays Europe and Asia Pacific.

```
IF([Office]="Boston" OR [Office]="San Francisco" OR
   [Office]="NYC", "U.S.", "Europe and Asia Pacific")
```

IN( )

Tests if a value is equal to a value in a list.

Syntax IN(value, check1,..., checkN)

Arguments

value
The value to test. The value can be a string, numeric, or date value.

check1, ..., checkN
The value or values to which to compare.

Returns True if value is equal to one of the check values; returns false otherwise.

Example The following example tests if New Haven, Baltimore, or Cooperstown are values in the city data field. If any one of the cities is in the data field, IN( ) returns true.

```
IN([city], "New Haven", "Baltimore", "Cooperstown")
```

The following example tests if 9/15/08 or 9/30/08 are values in the payDate data field. If you supply a literal date as an argument, you must include the time value, as shown in the following example:

```
IN([payDate], "9/15/08 12:00 AM", "9/30/08 12:00 AM")
```

The following example uses IN( ) in conjunction with the IF( ) function to test if Ships or Trains are values in the ProductLine data field. If Ships or Trains is a value in the field, the computed column displays Discontinued Item; otherwise, the product line value is displayed as it appears in the field.

```
IF(IN([ProductLine], "Ships", "Trains"),"Discontinued Item",
   [ProductLine])
```

ISNULL( )

Tests if a value in a specified data field is a null value. A null value means that no value exists.

Syntax ISNULL(value)

Argument **value**
The data field in which to check for null values.

Returns True if a value in the specified data field is a null value; returns false otherwise.

Example The following example uses ISNULL( ) in conjunction with the IF( ) function to test for null values in the BirthDate data field. If there is a null value, the computed column displays No date specified; otherwise the BirthDate value is displayed.

```
IF(ISNULL([BirthDate]), "No date specified", [BirthDate])
```

LEFT( )

Extracts a substring from a string, starting from the leftmost, or first, character.

Syntax LEFT(str)

LEFT(str, n)

Arguments **str**
The string from which to extract a substring.

n
The number of characters to extract, starting from the first character.

Returns A substring of a specific length:

- If you omit n, the number of characters to extract, the function returns the first character only.
- If n is zero, the function returns an empty string.
- If n is greater than the length of the string, the function returns the entire string.

Example The following example displays the first letter of each name in the CustomerName data field:

```
LEFT([CustomerName])
```

The following example uses the LEFT( ) and FIND( ) functions to display the characters that precede the hyphen in a product code:

```
LEFT([ProductCode], FIND("-", [ProductCode]))
```

If the product code is ModelA-1234, LEFT( ) returns the following string:

```
ModelA
```

LEN( )

Counts the number of characters in a string.

Syntax LEN(str)

Argument **str**
The string expression to evaluate.

Returns The number of characters in the specified string.

Example The following example returns the length of each value in the ProductCode data field:

```
LEN([ProductCode])
```

The following example uses LEN( ) in conjunction with the RIGHT( ) and FIND( ) functions to display the characters that appear after the hyphen in a product code. RIGHT( ) extracts a substring of a specified length, starting from the last character. In this example, the length of the entire string returned by LEN( ) minus the length up to the hyphen is the number of characters to display.

```
RIGHT( [PRODUCTNAME], LEN([PRODUCTNAME] ) - (FIND("-"
   ,[PRODUCTNAME] )))
```

If the product code is ModelA-Ford, RIGHT( ) returns the following string:

```
A-Ford
```

LIKE( )

Tests if a string matches a pattern.

Syntax LIKE(str, pattern)

str
The string to evaluate.

pattern
The string pattern to match. You must enclose the pattern in double quotation marks (" "). The match is case-sensitive. You can use the following special characters in a pattern:

- A percent character (%) matches zero or more characters. For example, %ace% matches any string value that contains the substring ace, such as Facebook, and MySpace. It does not match Ace Corporation because this string contains a capital A, and not the lowercase a.

- An underscore character (_) matches exactly one character. For example, t_n matches tan, ten, tin, and ton. It does not match teen or tn.

To match a literal percent (%), underscore (_), precede those characters with two backslash (\\) characters. For example, to see if a string contains M_10, specify the following pattern:

```
"%M\\_10%"
```

Returns True if the string matches the pattern; returns false otherwise.

Example The following example returns true for values in the customerName field that start with D:

```
LIKE([customerName], "D%")
```

The following example returns true for productCode values that contain the substring Ford:

```
LIKE([productCode], "%Ford%")
```

The following example uses two LIKE( ) expressions to look for the substrings "Ford" or "Chevy" in each ProductName value. If a product name contains either substring, the computed column displays U.S. Model; otherwise, it displays Imported Model.

```
IF(((LIKE([ProductName], "%Ford%") = TRUE) OR (LIKE([ProductName],
   "%Chevy%") = TRUE)), "U.S. model", "Imported Model")
```

LOWER( )

Converts all letters in a string to lowercase.

Syntax LOWER(str)

Argument **str**
The string to convert to lowercase.

Returns The specified string in all lowercase letters.

Example The following example displays all the string values in the productLine data field in lowercase:

```
LOWER([productLine])
```

MATCH( )

Returns a Boolean indicating whether a pattern exists within a string.

Syntax MATCH(source, pattern)

Arguments **source**
The string to evaluate.

pattern
The string pattern to match. The pattern uses ECMAScript (JavaScript) syntax, as defined in Section 15.10 of Standard ECMA-262.

Returns True if the pattern matches, false otherwise.

Example The following example uses ECMAScript syntax in the pattern to detect any set of characters followed by the letter C, and returns true:

```
MATCH("ABC",".*C")
```

The following example checks whether the string starts with the letter X, followed by any single character, and ending with C. It returns false.

```
MATCH("ABC","X.C")
```

MOD( )

Returns the modulo value for a number and a divisor.

Syntax MOD(number, divisor)

Arguments **number**
The number from which to derive the mod value.

divisor
The divisor for the mod function.

Returns Returns the remainder value of number divided by divisor.

Example The following example computes the remainder of PriceEstimate data field divided by 12, returning an integer. For example, if the PriceEstimate value is 27365, MOD( ) returns 5.

```
MOD([PriceEstimate], 12)
```

MONTH( )

Returns the month for a specified date value.

Syntax MONTH(date)

MONTH(date, option)

Arguments **date**
The date or date expression whose month to get.

option
A number that represents the month format to return. Use one of the following values:

- 1 to get the month as a number from 1 to 12.
- 2 to get the full month name, for example, January. The result is locale-specific.
- 3 to get the abbreviated month name, for example, Jan. The result is locale-specific.

If you omit option, MONTH( ) returns the month as a number.

Returns The month for a specified date value.

Example The following example returns the month, 1–12, for each value in the ShipDate data field:

```
MONTH([ShipDate])
```

The following example returns the full month name for each ShipDate value:

```
MONTH([ShipDate], 2)
```

NOT( )

Negates a Boolean expression.

Syntax NOT(expression)

Argument **expression**
The Boolean value or expression to negate.

Returns True if the expression evaluates to FALSE, and false if the expression evaluates to TRUE.

Example The following example uses NOT( ) in conjunction with the IF( ) and LIKE( ) functions. It tests if the value in the State data field is not CA. If the value is not CA, it returns the value of the Markup data field multiplied by 10%. If the value is CA it returns the value of the Markup data field multiplied by 15%.

```
IF(NOT(LIKE([State], "CA")),[Markup]*10%,[Markup]*15%)
```

The previous IF( ) statement is semantically equivalent to the following statement:

```
IF(LIKE([State], "CA"),[Markup]*15%,[Markup]*10%)
```

NOTNULL( )

Tests if a value in a specified data field is a non-null value.

Syntax NOTNULL(value)

Argument **value**
The data field in which to check for non-null values.

Returns True if a value in the specified data field is not a null value; returns false otherwise.

Example The following example uses NOTNULL( ) in conjunction with the IF( ) function to test for non-null values in the BirthDate data field. If there is a non-null value, the BirthDate value is displayed; otherwise the string "No date specified" is displayed.

```
IF(NOTNULL([BirthDate]), [BirthDate], "No date specified")
```

NOW( )

Returns the current date and time.

Syntax NOW( )

Returns The current date and time. For example:

```
Sep 23, 2008 11:56 AM
```

Example The following example uses the DIFF_MINUTE( ) and NOW( ) functions to calculate the number of minutes from the current date and time to Christmas:

```
DIFF_MINUTE(NOW(), "12/25/08 12:00 AM")
```

QUARTER( )

Returns the quarter number for a specified date value.

Syntax QUARTER(date)

Arguments **date**
The date or date expression whose quarter number to get.

Returns A number from 1 to 4 that represents the quarter for a specified date value. Quarter 1 starts in January.

Example The following example displays the quarter number for each value in the CloseDate data field:

```
QUARTER([CloseDate])
```

The following example displays a string—Q1, Q2, Q3, or Q4—for each value in the CloseDate data field:

```
"Q" & QUARTER([CloseDate])
```

RIGHT( )

Extracts a substring from a string, starting from the rightmost, or last, character.

Syntax RIGHT(str)

RIGHT(str, n)

Arguments **str**
The string from which to extract a substring.

n
The number of characters to extract, starting from the last character.

Returns A substring of a specific length.

- If you omit n, the number of characters to extract, the function returns the last character only.
- If n is zero, the function returns an empty string.
- If n is greater than the length of the string, the function returns the entire string.

Example The following example displays the last four characters of each value in the ProductCode data field:

```
RIGHT([ProductCode], 4)
```

The following example uses RIGHT( ) in conjunction with the LEN( ) and FIND( ) functions to display the characters that appear after the hyphen in a product code. This example assumes that the number of characters after the hyphen varies. Therefore, the length of the entire string (returned by LEN( )) minus the length up to the hyphen (returned by FIND( )) is the number of characters to display.

```
RIGHT([ProductCode], (LEN([ProductCode]) - FIND("-" ,
   [ProductCode])))
```

If the product code is ModelA-Ford, RIGHT( ) returns Ford. If the product code is ModelB-Toyota, RIGHT( ) returns Toyota.

ROUND( )

Rounds a number to a specified number of digits.

Syntax ROUND(num)

ROUND(num, dec)

Arguments **num**
The number to round.

dec
The number of digits up to which to round num. If you omit dec, ROUND( ) assumes 0.

Returns A number rounded to a specified number of digits.

Example The following example rounds the numbers in the PriceEstimate data field to return an integer. For example, if the PriceEstimate value is 1545.50, ROUND( ) returns 1546. If the PriceEstimate value is 1545.25, ROUND( ) returns 1545.

```
ROUND([PriceEstimate])
```

The following example rounds the numbers in the PriceEstimate data field to one decimal place. For example, if the PriceEstimate value is 1545.56, ROUND( ) returns 1545.6. If the PriceEstimate value is 1545.23, ROUND( ) returns 1545.2.

```
ROUND([PriceEstimate], 1)
```

The following example rounds the numbers in the PriceEstimate data field to one digit to the left of the decimal point. For example, if the PriceEstimate value is 1545.56, ROUND( ) returns 1550. If the PriceEstimate value is 1338.50, ROUND( ) returns 1340.

```
ROUND([PriceEstimate], -1)
```

ROUNDDOWN( )

Rounds a number down to a specified number of digits.

Syntax ROUNDDOWN(num)

ROUNDDOWN(num, dec)

Arguments **num**
The number to round down.

dec
The number of digits up to which to round num down. If you omit dec, ROUND( ) assumes 0.

Returns A number rounded down to a specified number of digits.

Example The following example rounds down the numbers in the PriceEstimate data field to return an integer. For example, if the PriceEstimate value is 1545.25, ROUNDDOWN( ) returns 1545. If the PriceEstimate value is 1545.90, ROUNDDOWN( ) returns 1545.

```
ROUNDDOWN([PriceEstimate])
```

The following example rounds down the numbers in the PriceEstimate data field to one decimal place. For example, if the PriceEstimate value is 1545.56, ROUNDDOWN( ) returns 1545.5. If the PriceEstimate value is 1545.23, ROUNDDOWN( ) returns 1545.2.

```
ROUNDDOWN([PriceEstimate], 1)
```

The following example rounds the numbers in the PriceEstimate data field down to one digit to the left of the decimal point. For example, if the PriceEstimate value is 1545.56, ROUNDDOWN( ) returns 1540. If the PriceEstimate value is 1338.50, ROUNDDOWN( ) returns 1330.

```
ROUNDDOWN([PriceEstimate], -1)
```

ROUNDUP( )

Rounds a number up to a specified number of digits.

Syntax ROUNDUP(num)

ROUNDUP(num, dec)

Arguments **num**
The number to round up.

dec
The number of digits up to which to round num up. If you omit dec, ROUND( ) assumes 0.

Returns A number rounded up to a specified number of digits.

Example The following example rounds up the numbers in the PriceEstimate data field to return an integer. For example, if the PriceEstimate value is 1545.25, ROUNDUP( ) returns 1546. If the PriceEstimate value is 1545.90, ROUNDUP( ) returns 1546.

```
ROUNDUP([PriceEstimate])
```

The following example rounds up the numbers in the PriceEstimate data field to one decimal place. For example, if the PriceEstimate value is 1545.56, ROUNDUP( ) returns 1545.6. If the PriceEstimate value is 1545.23, ROUNDUP( ) returns 1545.3.

```
ROUNDUP([PriceEstimate], 1)
```

The following example rounds up the numbers in the PriceEstimate data field to one digit to the left of the decimal point. For example, if the PriceEstimate value is 1545.56, ROUNDUP( ) returns 1550. If the PriceEstimate value is 1338.50, ROUNDUP( ) returns 1340.

```
ROUNDUP([PriceEstimate], -1)
```

SEARCH( )

Finds the location of a substring in a string. The substring can contain wildcard characters.

Syntax SEARCH(pattern, str)

SEARCH(pattern, str, startPosition)

Arguments **pattern**
The string pattern to search for. You must enclose the pattern in double quotation marks (" "). You can use the following special characters in a pattern:

- An asterisk (*) matches zero or more characters, including spaces. For example, t*n matches tn, tin, and teen.
- A question mark (?) matches exactly one character. For example, t?n matches tan, ten, tin, and ton. It does not match teen or tn.

str
The string in which to search.

startPosition
The position in str where the search starts.

Returns The numerical position of the string pattern in the string. The first character of a string starts at 1. If the substring is not found, SEARCH( ) returns 0.

Example The following example searches for the string pattern, S*A, in each product code. If the product name is KBS5412A, SEARCH( ) returns 3.

```
SEARCH("S*A", [ProductCode])
```

The following example uses SEARCH( ) in conjunction with the LEFT( ) function to display the characters that precede the first space character in a product name. The LEFT( ) function extracts a substring of a specified length, starting from the first character. In this example, the length of the substring to display is equal to the numerical position of the space character.

```
LEFT([ProductName], SEARCH(" ", [ProductName]))
```

If the product name is 1969 Ford Falcon, the expression returns 1969.

SQRT( )

Calculates the square root of a number.

Syntax SQRT(num)

Argument **num**
The number, or numeric expression that specifies the number, for which you want to find the square root. The number must be a positive number.

Returns A number that is the square root of num.

Example The following example calculates the square root of each value in the LotSize data field:

```
SQRT([LotSize])
```

The following example uses SQRT( ) to calculate the actual distance travelled uphill, given the base distance and elevation values. This example applies the Pythagorean theorem, which states that $a^2 + b^2 = c^2$. Using this theorem, the actual distance travelled is c, which means we want to calculate:

$$c = \sqrt{a^2 + b^2}$$

which translates to the following expression:

```
SQRT((([Distance] * [Distance]) + ([Elevation] * [Elevation])))
```

TODAY( )

Returns the current date that includes a time value of midnight, 12:00 AM.

Syntax TODAY( )

Returns The current date in the following format:

```
Sep 25, 2008 12:00 AM
```

Example The following example calculates the number of days from the current date to Christmas:

```
DIFF_DAY(TODAY(), "12/25/08")
```

The following example calculates the number of years from each value in the HireDate data field to the current date:

```
DIFF_YEAR([HireDate], TODAY())
```

TRIM( )

Removes the leading and trailing blanks from a specified string. TRIM( ) does not remove blank characters between words.

Syntax TRIM(str)

Argument **str**
The string from which to remove leading and trailing blank characters.

Returns A string with all leading and trailing blank characters removed.

Example The following example uses TRIM( ) to remove all leading and trailing blank characters from values in the FirstName and LastName data fields. The expression uses the & operator to concatenate each trimmed FirstName value with a space, then with each trimmed LastName value.

```
TRIM([FirstName]) & " " & TRIM([LastName])
```

TRIMLEFT( )

Removes the leading blanks from a specified string.

Syntax TRIMLEFT(str)

Arguments **str**
The string from which to remove the leading blank characters.

Returns A string with all leading blank characters removed.

Example The following example concatenates a literal string with each value in the customerName data field. TRIMLEFT( ) removes all blank characters preceding the customerName value so that there are no extra blank characters between the literal string and the customerName value.

```
"Customer name: " & TRIMLEFT([customerName])
```

TRIMRIGHT( )

Removes the trailing blanks from a specified string.

Syntax TRIMRIGHT(str)

Argument **str**
The string from which to remove the trailing blank characters.

Returns A string with all trailing blank characters removed.

Example The following example concatenates each value in the Comment data field with a semicolon, then with a value in the Action data field. TRIMRIGHT( ) removes all blank characters after the Comment value so that there are no extra blank characters between the Comment string and the semicolon.

```
TRIMRIGHT([Comment]) & "; " & [Action]
```

UPPER( )

Converts all letters in a string to uppercase.

Syntax UPPER(str)

Argument **str**
The string to convert to uppercase.

Returns The specified string in all uppercase letters.

Example The following example displays all the string values in the customerName data field in all uppercase:

```
UPPER([customerName])
```

WEEK( )

Returns a number from 1 to 52 that represents the week of the year.

Syntax WEEK(date)

Argument **date**
The date or date expression whose week of the year to get.

Returns A number that represents the week of the year for the specified date value.

Example The following example gets the week number of the year for each date value in the ShipDate data field:

```
WEEK([ShipDate])
```

WEEKDAY( )

Returns the day of the week for a specified date value.

Syntax WEEKDAY(date, option)

Arguments **date**
The date or date expression from which you want to get the day of the week.

option
A number that represents the weekday format to return. Use one of the following values:

- 1 to get the day as a number from 1 (Sunday) to 7 (Saturday).
- 2 to get the day as a number from 1 (Monday) to 7 (Sunday).
- 3 to get the day as a number from 0 (Monday) to 6 (Sunday).
- 4 to get the full weekday name, for example, Wednesday. The result is locale-specific.
- 5 to get the abbreviated weekday name, for example Wed. The result is locale-specific.

If you omit option, WEEKDAY( ) assumes option 1.

Returns The day of the week for a specified date value.

Example The following example gets the full weekday name for each date value in the DateSold data field:

```
WEEKDAY([DateSold], 4)
```

YEAR( )

Returns the four-digit year value for a specified date value.

Syntax YEAR(date)

date
The date or date expression from which you want to extract the year part.

Returns The number that represents the four-digit year for the specified date value.

Example The following example gets the four-digit year for each date value in the ShipDate data field, and adds 15 to the four-digit year. For example, if the ShipDate value is Sep 16, 2008, YEAR( ) returns 2023.

```
(YEAR([ShipDate]) + 15)
```

Functions used in aggregate calculations

Table 10-1 describes the range of functions that perform aggregate calculations. In BIRT Studio, you can perform aggregate calculations across the data rows in a group, section, or across an entire report table.

Table 10-1 Aggregate functions

Aggregate function	Description
AVERAGE	Returns the average, or mean for a set of data rows. For example, if a report column contains values 5, 2, 7, and 10, AVERAGE returns 6.
COUNT	Counts the number of data rows. If a column contains values 5, 2, 7, and 10, COUNT returns 4.
COUNTDISTINCT	Counts the number of unique values in a set of data rows. If a report column contains values 5, 2, 5, 7, and 10, COUNTDISTINCT returns 4.
FIRST	Returns the first value in set of data rows. If a report column contains data rows 5, 2, 7, and 10, FIRST returns 5.
LAST	Returns the last value in a set of data rows. If a report column contains data row values 2, 5, 7, and 10, LAST returns 10.

Table 10-1 Aggregate functions

Aggregate function	Description
MAX	Returns the largest value in a set of data rows. If a report column contains data row values 5, 2, 7, and 10, MAX returns 10. For string values, MAX returns the last alphabetical value. For date values, MAX returns the latest date.
MEDIAN	Returns the median, or middle value in a set of data rows. If a report column contains values, 5, 2, 7, and 10, MEDIAN returns 6.
MIN	Returns the smallest value in a set of data rows. If a report column contains data row values 5, 2, 7, and 10, MIN returns 2. For string data, MIN returns the first alphabetical value. For date values, MIN returns the earliest date.
MODE	Returns the mode, or the value that occurs most frequently in a set of data rows. If a report column contains values, 5, 2, 5, 7, and 10, MODE returns 5.
QUARTILE	Returns the quartile value in a set of data rows, given a specified quart (0-4). A quartile can be defined as any three values that divide a set of values into four equal parts, such that each part represents 1/4 of the set of values. MIN, MEDIAN, and MAX return the same value as QUARTILE when quart is equal to 0, 2, and 4 respectively. If a set of data rows contains 50, 75, 80, 90, and 95, and you specify a quart of 2, QUARTILE returns 80.
STDDEV	Returns the standard deviation of a set of data rows. Standard deviation is a statistic that shows how widely values disperse from the mean value. If a set of data rows contains 50, 75, 80, 90 and 95, STDDEV returns 17.536.
SUM	Adds all the values in a set of data rows. If a report column contains 50, 75, 80, 90, and 95, SUM returns 390.
VARIANCE	Returns the variance of a set of data rows. Variance is a statistical measure expressing large the size of the differences between the values. The variance increases as the differences between the numbers increase. If a set of data rows contains 50, 75, 80, 90, and 95, VARIANCE returns 307.5. If a set of data rows contains 5, 2, 5, 7, and 10, VARIANCE returns 8.7.
WEIGHTEDAVE	Returns the weighted average value in a set of data rows, given weights specified in another set of values. In a weighted average, each number is assigned a weight or degree of importance. These weights determine the relative importance of each number on the average. Grades are often computed using a weighted average. For example, for a set of scores 50, 75, 80, 90, and 95, with respective weights, 10, 25, 15, 30, and 20, WEIGHTEDAVE returns 81.75.

Operators

This section is a complete reference to all of the operators that you can use when you create expressions. This reference organizes the operators into the following categories:

- Operators in computed column expressions
- Operators in conditional formatting and filter condition expressions

Operators in computed column expressions

Table 10-2 lists the operators you can use when you write expressions for a computed column.

Table 10-2 Operators in computed column expressions

Operator	Use to	Example
+	Add two or more numeric values.	[OrderAmount] + [SalesTax]
-	Subtract one numeric value from another.	[OrderAmount] - [Discount]
*	Multiply numeric values.	[Price] * [Quantity]
/	Divide numeric values.	[Profit]/12
^	Raise a numeric value to a power.	[Length]^2
%	Specify a percent.	[Price] * 80%
=	Test if two values are equal.	IF([ProductName] = "1919 Ford Falcon", "Discontinued Item", [ProductName])
>	Test if one value is greater than another value.	IF([Total] > 5000, [Total]*15% , [Total]*10%)
<	Test if one value is less than another value.	IF([SalePrice] < [MSRP], "Below MSRP", "Above MSRP")
>=	Test if one value is greater than or equal to another value.	IF([Total] >= 5000, [Total]*15% , [Total]*10%)
<=	Test if one value is less than or equal to another value.	IF([SalePrice] <= [MSRP], "Below or equal to MSRP", "Above MSRP")
<>	Test if two values are not equal.	IF([Country] <> "USA", "Imported product", "Domestic product")
AND	Test if two or more conditions are true.	IF(([Gender] = "Male" AND [Salary] >= 150000 AND [Age] < 50), "Match found", "No match")

Table 10-2 Operators in computed column expressions

Operator	Use to	Example
OR	Test if any one of multiple conditions is true.	IF(([City] = "Boston") OR ([City] = "San Francisco"), "U.S.", "Europe and Asia")
&	Concatenate string values.	[FirstName] & " " & [LastName]

Operators in conditional formatting and filter condition expressions

Table 10-3 lists the operators you can use when you create expressions for conditional formatting and filter conditions.

Table 10-3 Operators in conditional formatting and filter condition expressions

Operator	Use to	Example
Between	Test if a column value is between two specified values.	Profit Between 1000 and 2000
Bottom N	Test if a column value is among the lowest *n* values.	SalesAmount Bottom N 10
Bottom Percent	Test if a column value is in the bottom *n* percent of all values.	SalesAmount Bottom Percent 5
Equal to	Test if a column value is equal to a specified value.	Country Equal to France
Greater Than	Test if a column value is greater than a specified value.	Total Greater Than 5000
Greater Than or Equal to	Test if a column value is greater than or equal to a specified value.	Total Greater Than or Equal to 5000
In	Test if a column value is in the list of specified values. Usage similar to the Any Of operator.	Country In USA, Canada, Mexico
Is Blank	Test if a column value is blank (" "). This operator applies only to string values.	E-mail Is Blank
Is False	Test if a column value is false.	LoanApproved Is False
Is Not Blank	Test if a column value is not blank. This operator applies only to string values.	Email Is Not Blank

(continues)

Table 10-3 Operators in conditional formatting and filter condition expressions (continued)

Operator	Use to	Example
Is Not Null	Test if a column value is not a null value. A null value means that no value is supplied.	CreditLimit Is Not Null
Is Null	Tests if a column value is a null value.	CreditLimit Is Null
Is True	Test if a column value is true.	LoanApproved Is True
Less Than	Test if a column value is less than a specified value.	Total Less Than 5000
Less Than or Equal to	Test if a column value is less than or equal to a specified value.	Total Less Than or Equal to 5000
Like	Test if a column value matches a string pattern.	ProductName Like %Ford%
Match	Test if a column value matches a string pattern.	ProductCode Match S20
Not Between	Test if a column value is not between two specified values.	Profit Not Between 1000 and 2000
Not Equal to	Test if a column value is not equal to a specified value.	Country Not Equal to France
Not In	Test if a column value is not in the specified list of values	Country Not In USA, Canada, Mexico
Not Like	Test if a column value does not match a string pattern.	ProductName Not Like %Ford%
Not Match	Test if a column value does not match a string pattern.	Product Code Not Match S10
Top N	Test if a column value is among the top *n* values.	SalesAmount Top N 10
Top Percent	Test if a column value is in the top *n* percent of all values.	SalesAmount Top Percent 5

Part Two

Customizing and integrating BIRT Studio

Chapter

11

Introduction to BIRT Studio customization

This chapter contains the following topics:

- About BIRT Studio and the Actuate Java Components
- Required Actuate software
- Understanding the context root

About BIRT Studio and the Actuate Java Components

A user accesses BIRT Studio from any Actuate Java Component, a web-based application that provides reporting services. Actuate Java Components support accessing, running, and viewing reports. BIRT Studio is a licensed option of Actuate Java Components. Figure 11-1 shows the relationship between BIRT Studio and Actuate Java Components.

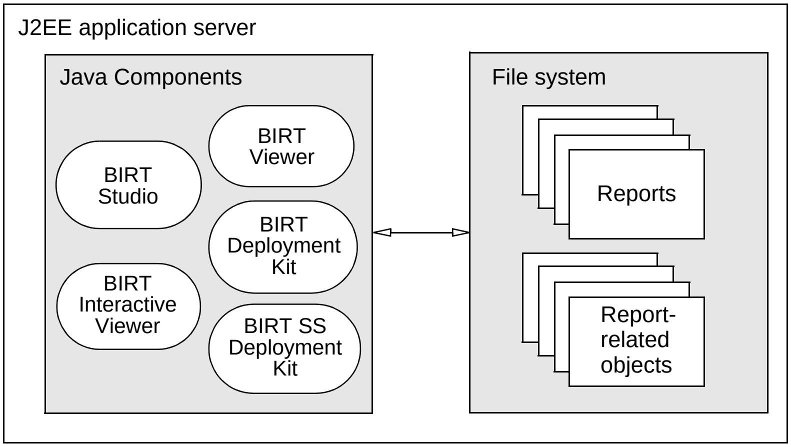


Figure 11-1 Actuate Java Components architecture

An Actuate Java Component is a web application that supports accessing and working with report information using a web browser. You deploy Actuate Java Components on an application server. Actuate Java Components access documents in a file system repository.

When accessed, BIRT Studio appears in your web browser, as shown in Figure 11-2.

Required Actuate software

To work with BIRT Studio, you must have the following software:

- An Actuate Java Components installation with BIRT Studio license option
- An application server or JSP or servlet engine such as Actuate embedded servlet engine or IBM WebSphere where Actuate Java Components are deployed
- Actuate BIRT Designer Professional to create report templates

You can find information about how to install and configure Actuate Java Components in *Installing an Actuate Java Component.*

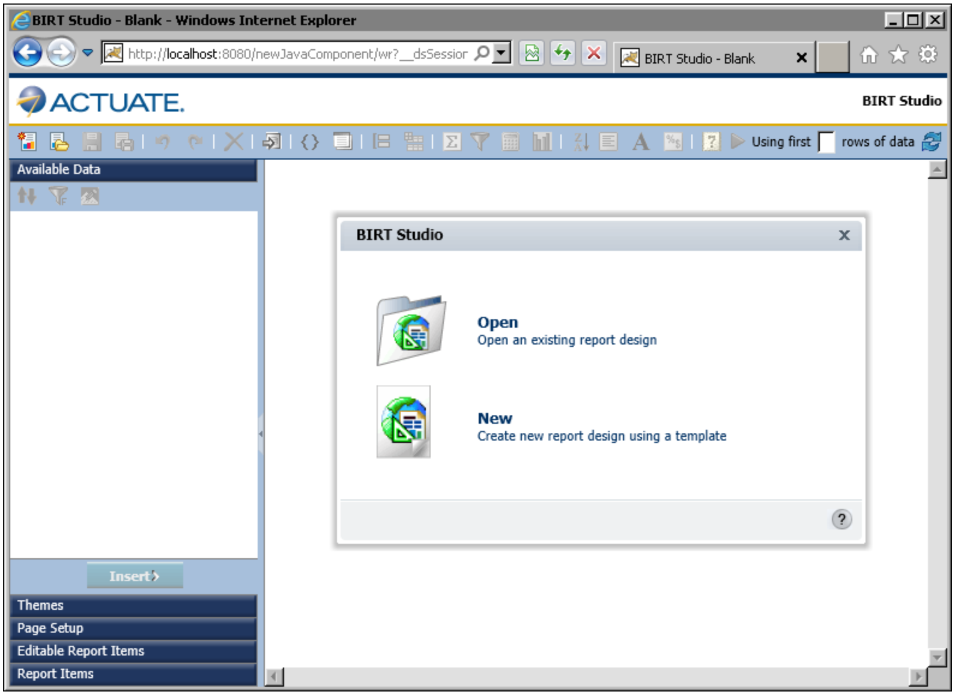


Figure 11-2 BIRT Studio

Understanding the context root

Actuate Java Components are packaged and distributed as a web archive file. When deployed to an application server container, the context root is name of the web archive (.war) file without the .war file extension. For example, if your web archive (.war) file were named DeploymentKit.war, the URL to access BIRT Studio is:

```
http://<web server>:<port>/DeploymentKit/wr
```

The context root for Actuate Java Components is the root directory of the web archive (.war) file when it is extracted.

For more information about Actuate Java Components customization refer to *Actuate BIRT Java Components Developer Guide*.

Chapter

12

Creating and publishing report templates

This chapter contains the following topics:

- About report templates
- Design considerations
- Creating a report template
- Providing data with a report template
- Creating themes for a report template
- Publishing a template
- Setting the default template category
- Publishing resources associated with a template

About report templates

A report template defines a basic report structure on which new reports are based. The BIRT Studio user always creates a new report by selecting a template first.

BIRT Studio includes several predefined templates. Often, however, users request custom templates that better suit the data they want to present. In addition, organizations typically require reports with a particular look and feel. You create templates using the Report Design perspective in Actuate BIRT Designer Professional, an Eclipse-based application for creating reports.

Design considerations

A template typically contains visual elements, such as tables, charts, and labels. It can, however, also contain defined data sets, which specify the data to display in a report. A template can even be a complete report that presents professionally formatted data.

Before you begin creating templates, gather the following basic information:

- What data will the BIRT Studio user use in a report?
- How does the user want to present the data?
- What does the user need as a starting point for a report?

The rest of this section provides tips for creating effective templates.

Separating or combining visual and data elements

When designing a template, one of the key decisions you make is whether to include both visual and data elements in the template or keep them separated. Good design typically dictates that templates contain only visual elements, and that information objects contain the data. You might decide, however, to create templates that include data, because some users, particularly inexperienced users, prefer to view a report without having to do anything other than select the template.

Templates that contain only visual elements are more versatile. A single template can be used for different reports that present different data. The user can mix and match information objects with templates. From a development and administrative perspective, separating presentation from data can be efficient and optimal, because template design and data retrieval can be accomplished by developers with design expertise and data retrieval expertise, respectively. This strategy, however, requires coordination between the template designer and the data architect to ensure that the templates and data are suitable for use with each other.

Designing themes

BIRT Studio provides several options for formatting the contents of a report. The standard toolbar provides the user with formatting options to modify the font, color, style, text alignment, and other properties of individual report items. The report design area on the left side of BIRT Studio displays themes, if any are provided, that the user can select to apply a set of styles to the entire report.

A theme is a collection of styles. The concept and functionality of styles are similar to styles in Microsoft Word and cascading style sheets (CSS). Designers create a theme to apply a consistent style, or look and feel, to a report. A theme, for example, can specify the colors used in a report, the font used for different text, label, or data items, the spacing or alignment of items, border styles, and so on.

BIRT Studio provides three themes with the default templates, as shown in Figure 12-1.

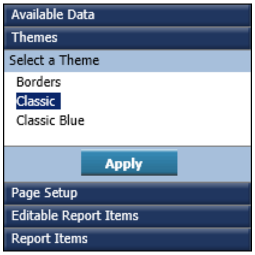


Figure 12-1 Themes provided with the default templates

When you create a template, consider creating different themes, so the user can choose from multiple styles. While the creation of a theme is optional, it is standard practice among designers, similar to the use of cascading style sheets with web pages. Themes are stored in a BIRT library file, separate from the template file.

Defining all the styles in a theme within a library, rather than applying formats to individual report items in the template, makes it easier to maintain and update the appearance of a template. When a user requests new or modified styles to use with a particular template, all you need to do is update the theme in the library, then publish the latest version of the library. You do not need to modify the template file.

Improving usability

A template should be intuitive to use. The user should be able to quickly determine how to use the elements in a template, and be able to freely edit most elements. The following are some guidelines for improving the usability of a template:

- Set the general properties of the template file:
  - The display name of the template
  - A brief description of the template
  - The image to use as the thumbnail image of the template

  These properties are not required. However, if you do not specify a display name, the name of the template file is used. If you do not supply a description, the word “null” appears, and if you do not specify an image, the user sees a grey box with the x icon. Figure 12-2 shows what the user sees on the Report Template page when a template’s properties are set and when the properties are not set.

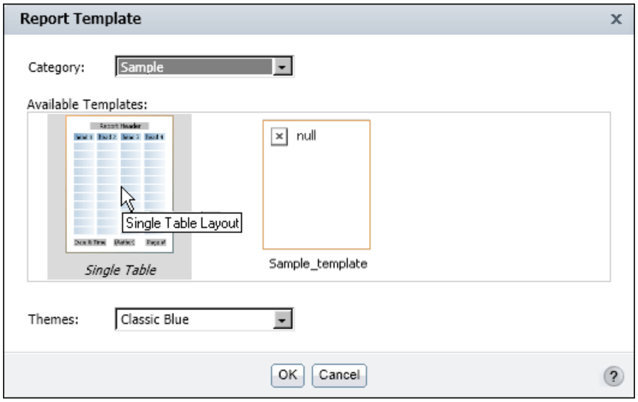


Figure 12-2 Displaying two templates, one with properties set, one without properties set

- Decide which report elements in the template are editable. Examples of editable elements include label elements for displaying report titles, section titles, or column headings, and empty tables into which users insert data. Examples of non-editable elements include company logos and standard copyright statements.
- Provide meaningful names for each report element, so the BIRT Studio user can easily identify the type and purpose of the element. If you do not specify a name, BIRT Studio displays the name of the element type, such as Text or Label. If your template contains three labels, and you do not specify a name for any of the labels, BIRT Studio displays Label three times under Report Items. Report Items lists all the elements in the template, so users can choose whether to display the elements in the report. Figure 12-3 shows one of the default templates with several elements listed under Report Items. The elements have descriptive names.

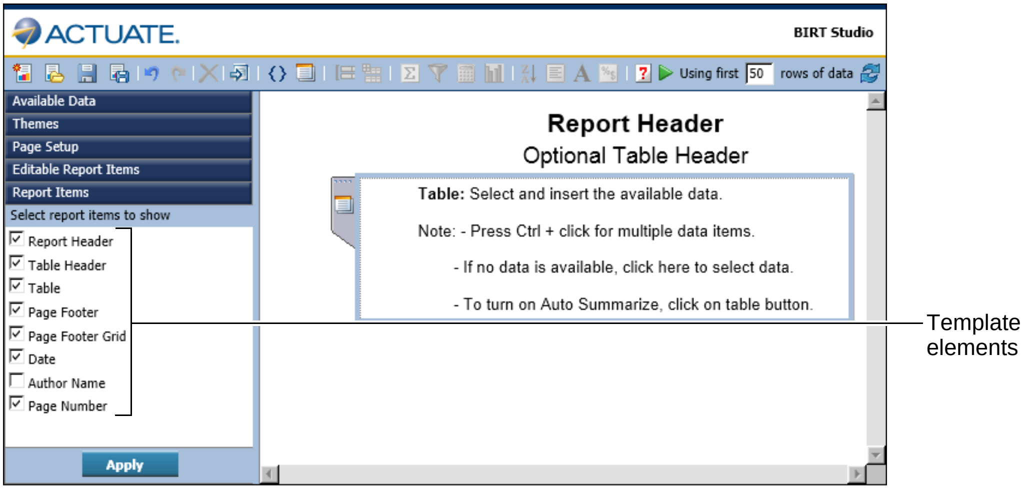


Figure 12-3 Report Items listing all the template elements

- Provide instructions for using each editable element. For example, a table can display a message, such as "To insert data in this report, drag the columns you want from Available Data and drop them in this table." Figure 12-3 shows a table with detailed instructions.

Creating a report template

You use Actuate BIRT Designer Professional to create report templates. You design a template in the same way that you design a BIRT report. In fact, you can create a report design then save it as a template. The file-name extension for a template file is .rpttemplate. If you are creating templates for users worldwide, you can localize the text in the templates the same way you localize text in a BIRT report. The templates that ship with BIRT Studio are localized, and the text in each template appears in the language specified by the locale the user chooses when logging in.

This section describes the key steps for creating a template but does not provide information about the report elements you can use in a template. For information about designing BIRT reports and templates, see the book that accompanies open source BIRT Report Designer, *BIRT: A Field Guide.*

How to create a report template

1 In the Report Design perspective, create a new template using the following procedure:

  1 Choose File➛New➛Other.

2. On New, expand Business Intelligence and Reporting Tools, then select Template. Choose Next.
3. In New Template—Template, select the folder in which to create the template file, specify a file name, then choose Next.
4. In New Template—Set Template Property:
    1. In Display Name, specify a display name for the template. This name identifies the template when the template is displayed on the Report Template page in BIRT Studio.
    2. In Description, provide a description of the template. This description appears as a tooltip when the user hovers the mouse pointer over the template in BIRT Studio.
    3. In Template Image, browse to the thumbnail image of the template. This step assumes that you have already created the image you want to use as the thumbnail image and placed it in Shared Resources. You must use a path that is relative to the <context root>\resources folder. The following example is a typical example of a relative path:

        ```
        \expense_rpt_template.jpg
        ```

        The path example means that BIRT Studio expects the image file to be in the following location in the file system:

        ```
        <context root>\resources\expense_rpt_template.jpg
        ```

    4. Choose Finish. A blank report page appears in the layout editor.

2 Drag elements from the palette, and drop them in the layout editor.

3 For elements that you want BIRT Studio users to edit for their report designs, identify those elements as template report items. You can only edit labels and tables in BIRT Studio:

1. Right-click the element, then choose Create Template Report Item.
2. Specify a descriptive name for the element, so the BIRT Studio user can easily identify the purpose of the element.
3. Provide instructions for using the element. Figure 12-4 shows an example name and instructions for using the element.

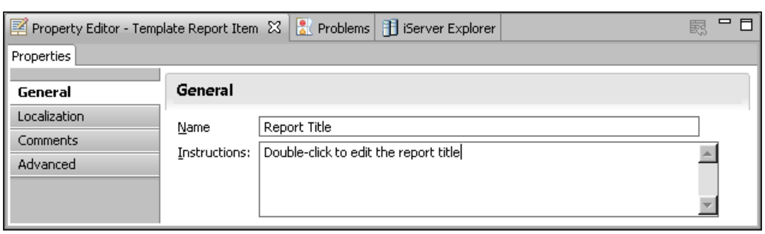


Figure 12-4 Specifying name and instructions for an editable element

4 For elements that you do not want the BIRT Studio user to edit, you should also specify a descriptive name, so that the BIRT Studio user can easily identify the purpose of the element and decide whether to include the element in the report. In Property Editor, type a name for the Name property. Figure 12-5 shows setting a label's name as Copyright.

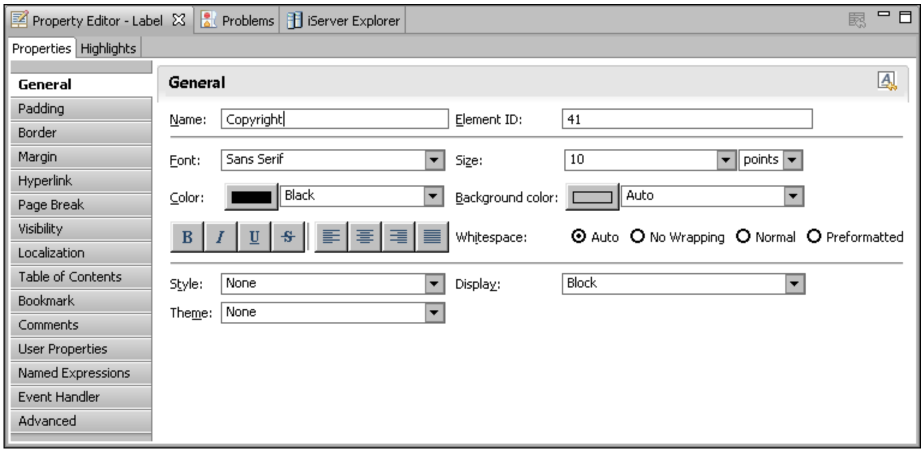


Figure 12-5 Specifying a name for a label element that users cannot edit

You can use the BIRT Thumbnail feature to manage the thumbnail image. You can change the thumbnail image property from the General properties in Property Editor, as shown in Figure 12-6.

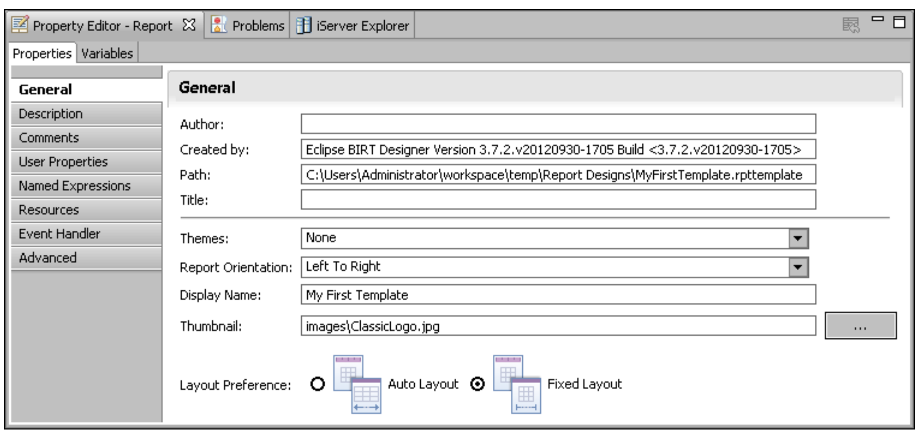


Figure 12-6 General properties

Selecting the thumbnail button gives you three options for selecting the thumbnail image, as shown on Figure 12-7:

- The Generate from the report option auto-generates the image using the template design view and embeds the image in the template itself. It is a convenient feature for template developers. In this case you do not deploy the thumbnail image, you deploy only the template.

- The Browse from file system option allows you to select the image from the file system. This option uses an absolute path to the image file and you have to ensure this path is accessible when you deploy the template to your Java Components installation.
- The Import from shared resources option allows you to select an image file from the BIRT designer resource folder and embed the selected image in the template design. If you use this option you do not need to publish the image file.

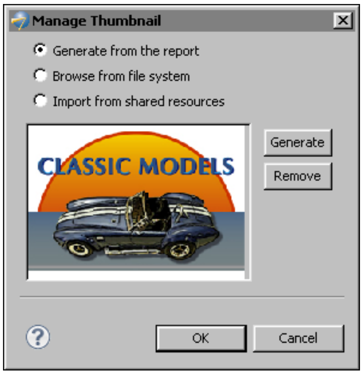


Figure 12-7 Manage Thumbnail

Providing data with a report template

If you are creating a template that includes data, create a data source and data set that specify the data that the BIRT Studio user can display in the report. You can define multiple data sources and multiple data sets in a template. When the user selects a template with multiple data sets, BIRT Studio prompts the user to select one of the data sets to use for the report. For information about defining data sources and data sets, see *BIRT: A Field Guide.*

Using a CSV file as a data source

A BIRT Studio report design can use a comma-separated values (CSV) file as a data source if the CSV file is a predefined data set in a report template. To use the file as a data source, you must copy the CSV file to the appropriate directory. To determine which directory to use, download the .rptdesign file to a local directory as an XML file. In the XML code, locate the <data-sources> element, as shown in the following example:

```
<data-sources>
<oda-data-source
extensionID="org.eclipse.datatools.connectivity.oda.flatfile"
name="Data Source" id="266">
<text-property name="displayName">Data Source
</text-property>
<property name="HOME">C:\</property>
<property name="CHARSET">UTF-8</property>
<property name="INCLTYPELINE">YES</property>
</oda-data-source>
</data-sources>
```

The HOME property shows the directory in which to place the CSV file.

Excluding a data set

You can exclude a data set in a template from the Select Data dialog in BIRT Studio. For example, you want to display stock quote data from a web service in the report, but you do not want the user to manipulate the data. To exclude a data set from the Select Data dialog, set the data set's UsageInBRS property to excluded in the template's XML representation. For example, the following code excludes the Orders data set:

```
<oda-data-set extensionID="org.eclipse.birt.report.data.oda.jdbc
   .JdbcSelectDataSet" name="Orders" id="8">
   <list-property name="userProperties">
      <structure>
         <property name="name">UsageInBRS</property>
         <property name="type">string</property>
         <property name="isVisible">true</property>
      </structure>
   </list-property>
   <property name="UsageInBRS">excluded</property>
```

Creating themes for a report template

As described earlier in this chapter, you create themes to provide the user with different sets of styles to apply to a report. You use Actuate BIRT Designer Professional to create the themes in a library. After you create the library, you associate the library with the template.

A library can contain any number of themes, and a theme can contain any number of styles. Actuate BIRT Designer Professional provides support for two types of styles. You are able to do the following:

- Create a custom style, and apply it to a report element. For example, you can create a style named Report Title, then apply the style to a label that displays the report title.
- Apply style properties to predefined style names, or selectors. These predefined style names correspond to the different types of report elements. For example, you can apply style properties to a predefined style named table-header, and all table headers in the report are formatted accordingly. This technique is similar to defining styles in CSS where you associate styles with HTML elements, such as <H1> or <P>.

How to create a theme

1. Create a library:
   1. Choose File➛New➛Other.
   2. On New, expand Business Intelligence and Reporting Tools, then select Library. Choose Next.
   3. In New Library, specify the folder in which to create the library, specify a file name, then choose Finish. If a message box appears, choose OK.
2. Choose Outline view. Outline view, as shown in Figure 12-8, displays the types of report elements you can add to a library.

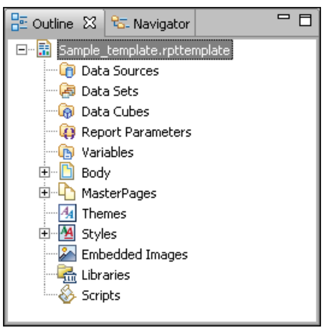


Figure 12-8 Outline view

3. In Outline view, expand Themes. A default theme, named defaultTheme, has already been created.
4. Right-click defaultTheme, and choose Rename to change the name of the theme.
5. Right-click the theme, and choose New Style to create a style for the theme.
6. On New Style, select one of the following options:
   - To apply style properties to a specific type of report element, select Predefined Style, and select a style from the drop-down list.

- To create a user-named style, select Custom Style, and specify a unique descriptive name, such as Report Title or Copyright.

7. Set the desired properties for the style by selecting a property category on the left and specifying property values.
8. When you finish setting style properties, choose OK to save the style.
9. Repeat steps 5 to 8 to create additional styles for the theme.
10. To create a new theme, right-click Themes, and choose New Theme.

How to associate a library with a template

1. If the BIRT resource folder is not the current project folder, place the library in the BIRT resource folder, to make it available to the template. To specify a folder as the resource folder:
    1. Choose Window➛Preferences.
    2. In Preferences, expand Report Design, then choose Resource, as shown in Figure 12-9.

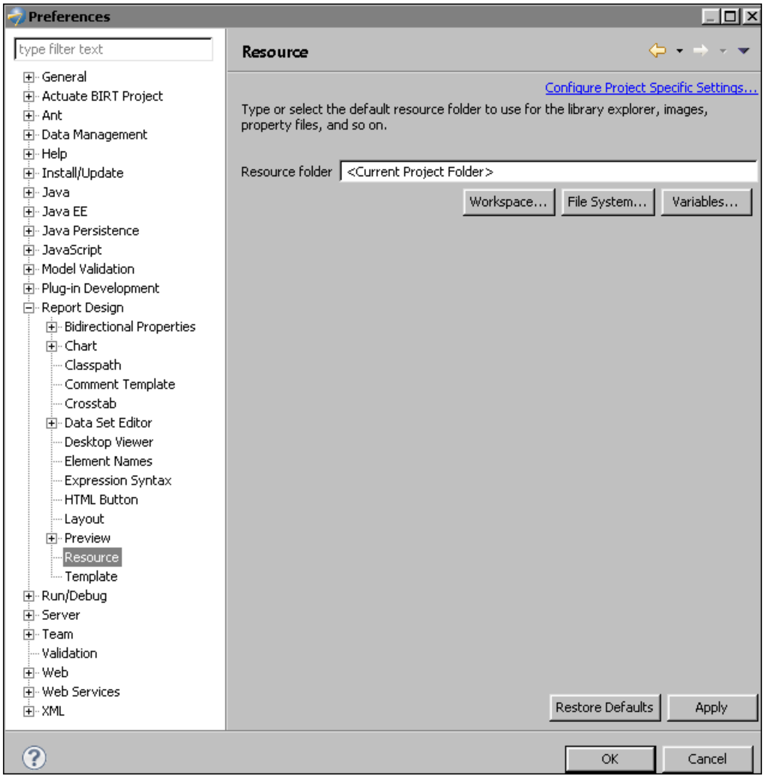


Figure 12-9 Specifying the location of the resource folder in Preferences

3 Choose File System to select a folder to use as the resource folder.

4 On Directory Selection, navigate to a folder on your computer or on the network, or choose Make New Folder to create a new folder.

5 Choose OK to confirm your folder selection. Preferences displays the path to the resource folder.

6 Choose OK to save the resource folder location information, and close the Preferences window.

7 In the layout editor or Navigator, select the library, then choose File➛Place Library in Resource Folder. Publish Library displays the library name and the location of the resource folder.

8 Choose Finish to confirm placing a copy of the library in the resource folder.

2 Open the template file, and choose Outline view.

3 In Outline view, right-click Libraries, and choose Use Library, as shown in Figure 12-10.

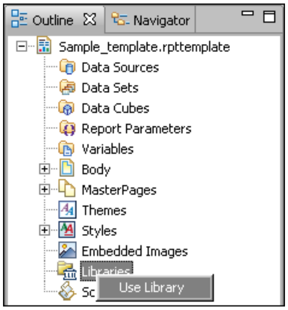


Figure 12-10 Choosing Use Library in the template file's outline view

4 In Use Library, expand Shared Resources to display the libraries in the BIRT resource folder. Figure 12-11 shows an example of Use Library.

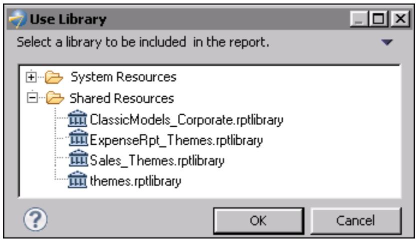


Figure 12-11 Displaying libraries in the resource folder

5 Select the library that contains the themes you want to use with the template, then choose OK.

Publishing a template

Templates must be published in specific locations. Otherwise, BIRT Studio cannot display the templates. BIRT Studio displays templates by categories. The default user interface has only two categories, named Standard and Sample. When you publish a template, you can create a new category or select an existing category in which to display your template.

If you create and publish a wide variety of templates, you can organize the templates into different categories. You can, for example, organize templates by report types (budget reports, expense reports, stock purchase plan reports) or by departments in your organization (Human Resources, Sales, Customer Support).

Figure 12-12 shows an example of BIRT Studio customized to display three template categories by report types.

The Standard category appears at the top of the list because it is the default category supplied with BIRT Studio. All other categories that you create are listed in alphabetical order. You can designate a different template category as the default category. This task is described later in this section.

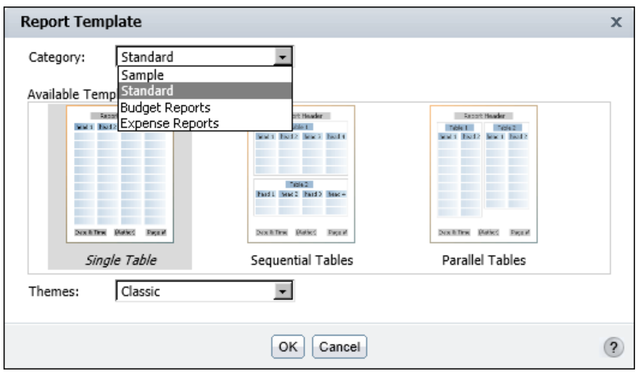


Figure 12-12 Report Template displays three template categories

Template files must be copied to the appropriate folders under the BizRDRptTemplates folder. You must have direct access to the machine on which Actuate BIRT Java Components run, because you need to manually place the template files in the following location on the file system:

```
<context root>\resources\BizRDRptTemplates
```

How to publish a report template

1 Create the proper folder structure on the machine where Actuate BIRT Java Components runs. If you want to organize the templates in categories, create one folder under the BizRDRptTemplates folder for each category. For example, to display templates in categories named Sales and Marketing, you create the following folders:

```
<context root>\resources\BizRDRptTemplates\Sales
<context root>\resources\BizRDRptTemplates\Marketing
```

2 Place the template files into their category folders.

3 Restart your Java Component application every time you create a new category for templates.

Setting the default template category

By default, the Report Template dialog displays the templates in the predefined category, Standard, as shown earlier in Figure 12-12. To view and select a template in another category, the user has to select a different category from the Category drop-down list. Typically, you want to make your custom templates more visible. You can configure BIRT Studio so that the Report Template dialog displays a different category of templates by default.

How to set the default template category

1 Open web.xml for editing. This file is in the following location:

```
<context root>\WEB-INF
```

2 Change the value of the DEFAULT_REPORT_TEMPLATE_CATEGORY _NAME parameter from Standard to the name of the category whose templates you want the Report Template dialog to display by default. The following example shows the Sales category set as the default template category:

```
<param-name>DEFAULT_REPORT_TEMPLATE_CATEGORY_NAME
   </param-name>
<param-value>Sales</param-value>
```

3 Restart your Java Component application for the change to take effect.

Publishing resources associated with a template

Typically, each template uses the following external resources that you must also publish to specific locations:

- An image file that provides the user with a thumbnail image of the template
- A library file that contains the themes the user can select to apply to the report

If a template contains localized text and you have created resource files that translate text into different languages, you must also publish these resource (.properties) files.

All resources used by a template must be copied to the Actuate Java Components resource folder in the following location:

```
<context root>\resources
```

You do not need to restart Actuate Java Components for the resource files to take effect. All you need to do is to log out of Actuate BIRT Java Components, then log in again.

Chapter

13

Providing data

This chapter contains the following topics:

- Types of data sources
- Creating a custom data source

Types of data sources

To create a new report, the first step the BIRT Studio user performs is selecting a template on which to base the report. The second step is selecting the data to display. You provide BIRT Studio users access to data in either of the following ways:

- Create and publish report templates that include defined data sets. Use this method if you want to provide templates that combine both presentation and data elements.
- Create a custom data source and integrate it with BIRT Studio. Use this method if the data is stored in a proprietary system or if you want to design a custom user interface for selecting data.

This chapter describes the second method. The first method is described in Chapter 12, "Creating and publishing report templates."

Creating a custom data source

You can provide users access to data by creating a custom user interface to a data source and integrating it with BIRT Studio. BIRT Studio supports the usage of BIRT and custom data sources. There are three supported scenarios you can implement:

- You use a BIRT JDBC data source, and the out-of-box functionality to configure the user interface for selecting the data.
- You use a BIRT driver to connect to your data source. You create your own user interface for selecting the data. This scenario is appropriate in all cases when your data source is not a BIRT JDBC data source, but it is still supported by BIRT, such as XML, Web Services, etc.
- You use your own custom open data access (ODA) driver to connect to your data source and you build your own user interface for selecting the data.

Actuate provides two examples that reference the supported scenarios:

- Sample ODA data source. The example demonstrates how to configure a user interface for selecting data when using the BIRT JDBC driver.
- Extended sample ODA data source. The example creates a custom user interface for selecting data and uses a BIRT driver to connect to the data source.

Sample ODA data source

Creating a custom user interface to connect to a BIRT JDBC data source does not require additional coding. You must configure the data source and data set and describe the data set columns in the erni_config.xml file, as shown in Listing 13-1.

Listing 13-1 Configuring the sample data source in erni_config.xml

```
<odaconfig>
   <name>OdaSample</name>
   <displayName>Sample ODA data source</displayName>
   <description>A sample ODA data source</description>
   <datasourceExtensionId>org.eclipse.birt.report.data.oda.jdbc
   </datasourceExtensionId>
   <datasourceDisplayName>CLASSICMODELS</datasourceDisplayName>
      <datasetExtensionId>org.eclipse.birt.report.data.oda.jdbc
      .JdbcSelectDataSet</datasetExtensionId>
   <datasetDisplayName>CLASSICMODELS.PAYMENTS</datasetDisplayName>
   <odaDriverClass>org.eclipse.birt.report.data.oda.sampledb
   .Driver</odaDriverClass>
   <odaURL>jdbc:classicmodels:sampledb</odaURL>
   <odaUser>ClassicModels</odaUser>
   <!--<odaPassword></odaPassword>-->
   <!-- Data Type can be DECIMAL(3)/INTEGER(4)/FLOAT(6)
   /DATE-TIME(93)/DATE(91)/TIME(92)/BOOLEAN(16)/STRING(12)-->
   <odaColumns>
      <odaColumn>
         <name>CUSTOMERNUMBER</name>
         <dataType>INTEGER</dataType>
         </odaColumn>
      <odaColumn>
         <name>PAYMENTDATE</name>
         <dataType>DATE</dataType>
      </odaColumn>
      <odaColumn>
         <name>CHECKNUMBER</name>
         <dataType>STRING</dataType>
      </odaColumn>
      <odaColumn>
         <name>AMOUNT</name>
         <dataType>FLOAT</dataType>
      </odaColumn>
   </odaColumns>
   <dataObject>CLASSICMODELS.PAYMENTS</dataObject>
```

```
<!--<queryText></queryText>-->
   <enabledInWorkgroupMode>true</enabledInWorkgroupMode>
   <enabledInEnterpriseMode>false</enabledInEnterpriseMode>
   <entryPoint></entryPoint>
</odaconfig>
```

The sample ODA data source with BIRT Studio is enabled by default. The setting of the data source's <enabledInWorkgroupMode> attribute in erni_config.xml is set to true. The erni_config.xml file is stored in the following location:

```
<context root>\WEB-INF
```

The context root for Actuate Java Components is the root directory of the web archive (.war) file when it is extracted.

If you create a new data source you must always enable the sample data source, and then restart the application server, and open BIRT Studio.

You test this custom data source by starting BIRT Studio. When prompted, select one of the standard report templates. Based on the XML definition in erni_config.xml, the product displays a sample ODA editor with a table, containing the configured database columns, as shown in Figure 13-1. The check boxes appearing next to the column names allow the user to select the data for the report. The editor gives the user the ability to create a parameter for filtering the data in the last column.

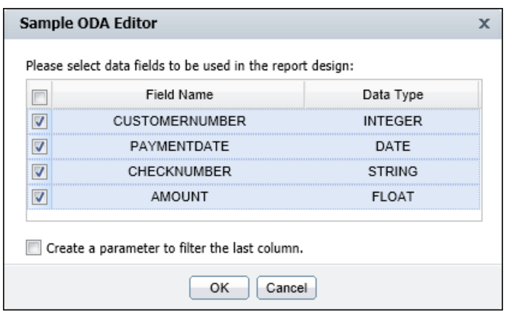


Figure 13-1 Selecting data in the sample ODA editor

Configuring a sample ODA data source

Table 13-1 lists the configuration attributes and their descriptions.

Table 13-1 Attributes of a sample ODA data source

Attribute	Description
<name>	The unique name of the data source. This name follows certain naming conventions. There can be no spaces, for example.

Table 13-1 Attributes of a sample ODA data source

Attribute	Description
<displayName>	The data source name that appears in the Data Source dialog box, as shown in Figure 13-2.
<description>	The data source description that appears in the user interface.
<datasetExtensionId>	Identifies the ID of the data set extension of the data driver.
<datasourceExtensionId>	Identifies the ID of the data source extension of the data driver.
<odaURL>	The database URL to use to connect to the data source.
<odaUser>	The user name, used to connect to the database.
<odaPassword>	The database password.
<odaColumn><name>	The name of the database column to be included in the user interface.
<odaColumn><dataType>	The type of the database column. Can be one of the following: `DECIMAL(3), INTEGER(4), FLOAT(6),DATE-TIME(93),DATE(91), TIME(92), BOOLEAN(16), STRING(12)`
<dataObject>	The name of the database table if single table is used.
<queryText>	The text of the query if the columns are from different tables.
<enabledInWorkgroupMode>	Indicates whether this data source is available to users.
<enabledInEnterpriseMode>	Not used.
<entryPoint>	A URL or servlet that points to the first web page of your custom query builder.

Configuring the ODA data source requires knowledge about your JDBC data source and data set. If you do not know this information, you can obtain the database connection properties from your database administrator. It is a good practice to create first a sample report in BIRT Designer Professional and validate the connection and the database query. You use the XML source of the created report to identify the correct values for the sample ODA configuration. Look for the attribute values in the <data-sources> and <data-sets> tags at the beginning of the report XML. Listing 13-2 shows portions of the report XML that contain the configuration attributes and their values.

Listing 13-2 Report XML source

```
<data-sources>
   <oda-data-source
   extensionID="org.eclipse.birt.report.data.oda.jdbc"
   name="Data Source" id="7">
   …
   <property
   name="odaDriverClass">org.eclipse.birt.report.data.oda.
   sampledb.Driver</property>
   <property name="odaURL">jdbc:classicmodels:sampledb</property>
   <property name="odaUser">ClassicModels</property>
   </oda-data-source>
</data-sources>
<data-sets>
         <oda-data-set
   extensionID="org.eclipse.birt.report.data.oda.jdbc.
   JdbcSelectDataSet" name="Data Set" id="8">
   …
   <xml-property name="queryText"><![CDATA[select *
   from CLASSICMODELS.PAYMENTS]]></xml-property>
```

Extended sample ODA example

To implement an extended custom data source, you perform the following programming and configuration tasks:

- Develop an ODA data driver to connect to and retrieve data from a data source. When you use BIRT drivers you skip this step.
- Develop a query builder that prompts the user to select data, processes the user's selection, and creates the query to get the data.
- Configure the custom data source for use with BIRT Studio.

An example of an extended ODA data source installs with BIRT Studio. The data source accesses data from an integrated sample database named ClassicModels. To test this custom data source with BIRT Studio, you must first enable the data source, named OdaSampleExt, by setting the data source's <enabledInWorkgroupMode> attribute in erni_config.xml to true. The erni_config.xml file is stored in the following location:

```
<context root>\WEB-INF
```

Listing 13-3 shows in bold the <enabledInWorkgroupMode> attribute whose value you change from false to true. Look for the ODA data source named OdaSampleExt.

Listing 13-3 Configuring the extended sample data source in erni_config.xml

```
<odaconfig>
   <name>OdaSampleExt</name>
   <displayName>Extended sample ODA data source</displayName>
   <description>An extended sample ODA data source</description>
   <datasourceExtensionId>org.eclipse.birt.report.data.oda.jdbc</
   datasourceExtensionId>

   <datasetExtensionId>org.eclipse.birt.report.data.oda.jdbc.JdbcS
   electDataSet</datasetExtensionId>
   <enabledInWorkgroupMode>true</enabledInWorkgroupMode>
   <enabledInEnterpriseMode>true</enabledInEnterpriseMode>
   <entryPoint>/OdaSample</entryPoint>
</odaconfig>
```

After you enable the extended sample data source, restart the application server, and open BIRT Studio. When prompted, select one of the standard report templates. The Data Source dialog box, as shown in Figure 13-2, prompts you to select a data source. Select Extended sample ODA data source.

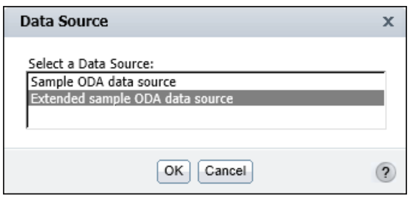


Figure 13-2 Selecting a data source

A query builder page, as shown in Figure 13-3, prompts you to select the data to use in the report.

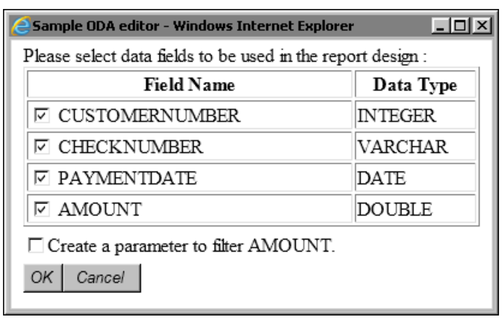


Figure 13-3 Building a query

After you select the data and choose OK, the query builder closes, and the data you selected appears under Available Data, as shown in Figure 13-4.

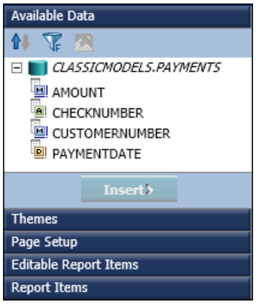


Figure 13-4 Selected data appears under Available Data

Developing an ODA data driver

You develop an ODA data driver by implementing run-time interfaces that the ODA framework defines. The ODA framework is an Eclipse Data Tools Platform (DTP) project component that provides a way for an application to access data from standard and proprietary data sources. For information about ODA and developing data drivers, go to the following site:

```
http://www.eclipse.org/datatools/project_connectivity
   /connectivity_doc/OdaOverview.htm
```

Another resource is Addison-Wesley's *Integrating and Extending BIRT* book.

The extended data source example uses a predefined ODA JDBC driver to connect to the ClassicModels sample database. The source code for this driver (org.eclipse.birt.report.data.oda.jdbc) is available in the Eclipse DTP CVS repository.

After you develop a custom ODA driver, you must copy the driver to the following folder:

```
<context root>\WEB-INF\platform\dropins
```

Developing a custom query builder

You develop a query builder using standard web application components, JavaServer Pages (JSPs) and servlets. After creating the necessary components, you perform a series of tasks to integrate the components with Actuate Java Components. This section describes the sample query builder, on which you can base your own query builder. This section also describes the procedures for compiling and deploying your own query builder components.

The sample query builder consists of the following components:

- ClassicModelQueryBuilder.jsp. This JSP displays the page in Figure 13-3. It also contains JavaScript functions to build the string that contains information about the data that the user selected and to send the request to a servlet. To view the code, see ClassicModelQueryBuilder.jsp in the following location:

```
<context root>\bizRD\oda\sample
```

- SampleServlet.java. This servlet communicates with ClassicModelQueryBuilder.jsp, creates a design session, and creates the query using information from the JSP. To view the code, see SampleServlet.java in the following location:

```
<context root>\iportal\examples\oda\classes\com
   \actuate\erni\oda\ClassicModels
```

Creating the servlet

A servlet performs the main tasks for getting and managing data for BIRT Studio users, and it is the key piece of any custom query builder. The example servlet, SampleServlet.java, extends the HTTPServlet class, and performs the following tasks:

- Manages design sessions for concurrent BIRT Studio users
- Uses ODA API to define the basic methods for constructing a query, result set, and column objects
- Processes the information sent by ClassicModelQueryBuilder.jsp, and creates a query
- Returns dynamic content that appears in Available Data, as shown in Figure 13-4

The servlet contains declarations of the ODA data driver's data source and data set extensions.

```
private final static String Datasource_Extension_Id =
   "org.eclipse.birt.report.data.oda.jdbc";

private final static String Dataset_Extension_Id =
   "org.eclipse.birt.report.data.oda.jdbc.JdbcSelectDataSet";
```

These extension IDs are used to construct the definitions of the data source connection and data set query in the report design.

The servlet supports the GET method only. Its doGet method retrieves request parameters, creates a unique ODA design session, and stores a session ID in the ODA session, so that other requests in the same session can access the same ODA session object. In any given session, the servlet can receive multiple requests with different parameter values.

Table 13-2 describes the supported parameters.

Table 13-2 Session request parameters

Parameter	Description
inedit	Indicates whether the user has started editing the values on the query builder page. Values are null or true.
state	Shows the editing status. Values are null, ok, or cancel.
selection	Contains information about the selected data fields.
sessionId	Contains the session ID.

The servlet executes a different action, depending on the parameter values. Table 13-3 describes the actions taken with the different inedit and state values.

Table 13-3 Actions corresponding to inedit and state values

inedit value	state value	Action
null	any	Generates a new session ID. Stores the ID and the callback URL in the session map. Changes the inedit parameter to true and sends a response to the calling page.
not null	null	Redirects the response to ClassicModelQueryBuilder.jsp.
not null	ok	The user has finished selecting the data fields. The servlet deletes the session ID, and parses the value in the selection parameter to build the query. The response is redirected to a BIRT Studio page.
not null	cancel	The user cancelled out of the query builder page. The servlet deletes the session ID.

Compiling the servlet

After you develop your servlet, you must compile the class. You can use a javac compiler from the command prompt or any Java IDE, such as Eclipse. To compile a servlet class, the following JAR files must be in your Java classpath:

- com.actuate.iportal.jar
- org.eclipse.emf.common.jar
- org.eclipse.emf.ecore.jar
- org.eclipse.datatools.connectivity.oda.design.jar

  These files are in the following location:

  ```
  <context root>\WEB-INF\lib
  ```

- servlet.jar

  You can find this file in different places, depending on the application server you use. For example, if you use Apache Tomcat 6.0 as an application server, you can use servlet-api.jar from the following location:

  ```
  Tomcat 6.0\lib
  ```

Deploying the servlet

After you compile the servlet class, deploy the servlet to your application. You can deploy your servlet as a class file, or packaged as a JAR file. The SampleServlet.class servlet is deployed to your application packaged in iportal.jar.

If you deploy the servlet using a JAR file, copy the JAR file to the following location:

```
<context root>\WEB-INF\lib
```

If you deploy the servlet as a class file, copy the servlet class to the following location:

```
<context root>\WEB-INF\classes
```

Registering the servlet

After you compile your servlet, you also need to register the servlet with the web application. To register the servlet, you add two entries to web.xml, which is stored in the following location:

```
<context root>\WEB-INF
```

The first entry, under the <servlet> element, defines a name for the servlet and specifies the compiled class that executes the servlet. The following example shows the <servlet> entry for the sample servlet:

```
<servlet>

  <servlet-name>OdaSampleServlet</servlet-name>

  <servlet-class>
   com.actuate.erni.oda.ClassicModels.SampleServlet
   </servlet-class>

</servlet>
```

The second entry, under the <servlet-mapping> element, defines the URL pattern that calls this servlet. The following example shows the <servlet-mapping> entry for the sample servlet:

```
<servlet-mapping>
  <servlet-name>OdaSampleServlet</servlet-name>
  <url-pattern>/OdaSample</url-pattern>
</servlet-mapping>
```

Configuring the extended custom data source

After you finish developing all the components of a custom data source, you configure the data source for use with BIRT Studio. Data sources available to BIRT Studio are defined in the following configuration file:

```
<context root>\WEB-INF\erni_config.xml
```

Listing 13-3 shows the definition of the extended data source. To add your custom data source, create a new <odaconfig> element in erni_config.xml. Table 13-4 describes each attribute, within the <odaconfig> element, that you configure.

Table 13-4 Attributes of an extended custom data source

Attribute	Description
<name>	The unique name of the data source. This name follows specific naming conventions. There can be no spaces, for example.
<displayName>	The data source name that appears in the Data Source dialog box, as shown in Figure 13-2.
<description>	The data source description that appears in the user interface.
<datasetExtensionId>	Identifies the ID of the data set extension that was created in the ODA data driver. The value should match the data set extension ID that is specified in the servlet.
<datasourceExtensionId>	Identifies the ID of the data source extension that was created in the ODA data driver. The value should match the data source extension ID that is specified in the servlet.
<enabledInWorkgroupMode>	Indicates whether this data source is available to users.
<enabledInEnterpriseMode>	Not used.
<entryPoint>	A URL or servlet that points to the first web page of your custom query builder.

Chapter

14

Configuring BIRT Studio

This chapter contains the following topics:

- Enabling or disabling functionality
- Configuring the application environment

Enabling or disabling functionality

BIRT Studio provides a full range of tools to support the report design process. For example, if the BIRT Studio users you support have little experience designing reports, you can simplify the design process by disabling more advanced functionality, such as creating calculated columns, and aggregating data. Alternatively, if you want users to format report content only by selecting a corporate-designed theme, you can disable the formatting functionality.

Configuring toolbar and context menu items

You configure the toolbar and context menu functionality that is available to users by editing attributes in the BIRT Studio configuration file, erni_config.xml. This file is located in the following location:

```
<context root>\WEB-INF
```

Customizations you make to erni_config.xml apply at the application level. If you want different sets of functionality available to different groups of users, you need to create multiple instances of the web application, then customize the functionality of each BIRT Studio instance.

In erni_config.xml, the <actionSets> element defines all the user actions that can be enabled or disabled. The actions are organized by category, for example, file operations, calculations, and formatting. The <actionSet> element defines the category, and the <action> element defines a specific action. Listing 14-1 shows the hierarchy of elements.

Listing 14-1 An example of an <actionSet> element in erni_config.xml

```
<actionSets>
   <actionSet>
      <name>FileOperations</name>
      <visible>true</visible>
      <action>
         <name>New</name>
         <enabled>true</enabled>
      </action>
      <action>
         <name>Open</name>
         <enabled>true</enabled>
      </action>
      <action>
         <name>Save</name>
         <enabled>true</enabled>
      </action>
      <action>
```

```
      <name>SaveAs</name>
      <enabled>true</enabled>
    </action>
  </actionSet>
```

By default, all actions are enabled. You can disable actions in the following ways:

- To disable a particular action, change the action's <enabled> attribute from true to false.
- To disable all actions within a category in one step, change the action set's <visible> attribute from true to false.

For the changes to take effect, restart the appropriate web application. When you relaunch BIRT Studio, the toolbar displays different buttons, and the context menus display different items, depending on which action or actions you disabled.

Configurable actions

Table 14-1 lists the toolbar and context menu actions that you can enable or disable. Some of the actions appear in both the toolbar and context menus, and some appear in context menus only. While you can disable any of the actions defined in erni_config.xml, it does not make sense to disable all the actions. For example, disabling both the New and Open actions under file operations prevents a user from creating or opening reports.

Table 14-1 User actions you can enable or disable through erni_config.xml

Action set	Action	Description
Calculations	Aggregation	Performs a calculation over a specified set of data rows.
	Calculation	Creates a calculated column, based on a specified expression.
	ChangeSubtotal	Changes the subtotal function, applied to a column in a summary table.
	Chart	Inserts a chart.
	DataFields	Shows the data fields in the report, and supports adding or deleting fields in the report.
	EditCalculation	Changes a calculated column.
	Filter	Filters table rows, based on a specified condition.

(continues)

Table 14-1 User actions you can enable or disable through erni_config.xml (continued)

Action set	Action	Description
ColumnHeader Operations	DeleteRow	Deletes the row of the selected column header.
	InsertRow	Inserts a row above or below the selected column header.
	Merge	Merges the selected column header with the header on the right, left, or above.
	Split	Splits the selected merged columns.
ColumnOperations	ColumnWidth	Changes the width of the selected column.
	HideColumn	Hides the selected column.
	MergeColumns	Merges the selected columns.
	MoveToDetail	Moves the selected item in the group header row to the table's detail row.
	MoveToGroup	Moves the selected item in the table's detail row to the group header row.
	NoRepeat	If duplicate data values appear in the selected column, displays only the first instance.
	ReorderColumns	Changes the order of the columns in the table.
	RepeatValues	Displays duplicate data values in the selected column.
	ShowColumns	Shows the selected columns.
CrosstabOperations	Analyze	Opens the Data Analyzer.
	Delete	Deletes a cross tab.
	Edit	Opens the cross tab builder dialog.
	SwitchView	Switches the cross tab view.
DeleteColumn	DeleteColumn	Deletes a column from a table in the report.
EditText	EditText	Enables editing of the selected static text.
FileOperations	New	Creates a new report design file.

Table 14-1 User actions you can enable or disable through erni_config.xml (continued)

Action set	Action	Description
	Open	Opens an existing report design.
	Save	Saves the current report design.
	SaveAs	Saves the current report design file under a different name or in a new location.
Formatting	AlignCenter	Centers the text in the selected column.
	AlignLeft	Aligns the left sides of text in the selected column.
	AlignRight	Aligns the right sides of text in the selected column.
	Border	Draws a border around the selected column.
	ConditionalFormat	Formats data in the selected column, based on a specified condition.
	Data	Formats the display of data in the selected column.
	Font	Formats the font of data in the selected column.
GeneralOperations	Parameter	Displays the parameters, if any, for the current report.
	SwitchSummary Mode	Switches between summary table mode and detail table mode for the selected table.
	TableBuilder	Enables the Table Builder wizard.
Grouping	CreateSection	Adds a report section, which provides an additional level of data grouping.
	DeleteSection	Removes the selected report section.
	GroupBy	Groups table rows by values in the selected column.
	HideDetail	Hides the detail rows in a report section.

(continues)

Table 14-1 User actions you can enable or disable through erni_config.xml (continued)

Action set	Action	Description
Grouping *(continued)*	PageBreak	Adds page breaks before or after a report section.
	ShowDetail	Shows the detail rows in a report section.
	UngroupBy	Removes groups in the selected column.
Help	Help	Shows help information.
ManageData	ManageData	Opens the manage data dialog.
PageLayout	PageLayoutIn Toolbar	Displays page layout toggle under toolbar. Disabled by default.
Preview	PreviewHTML	Shows a preview of the report in HTML format.
ReportItem Operations	Bookmark	Assigns a bookmark to a report item.
	Hyperlink	Assigns a hyperlink to a report item.
SectionOperations	SectionHeading	Shows the data fields in the report, and supports adding fields to the selected section heading.
Sorting	AdvancedSort	Sorts the table rows by the values of multiple columns.
	SortAscending	Sorts, in ascending order, the table rows by the values of the selected column.
	SortDescending	Sorts, in descending order, the table rows by the values of the selected column.
TemplateTable Operations	AutoSummarize On	If set to true, creates a summary table by default. If set to false, creates a detail table by default.
UndoRedo	Redo	Redo the last action.
	Undo	Undo the last action.

Configuration examples

This section provides examples of editing attributes in erni_config.xml, and the resulting changes to the BIRT Studio page. Figure 14-1 shows the default BIRT Studio page with all actions enabled. The formatting actions on the toolbar and context menu are called out, so that you can see the difference in the toolbar and context menu when these actions are disabled.

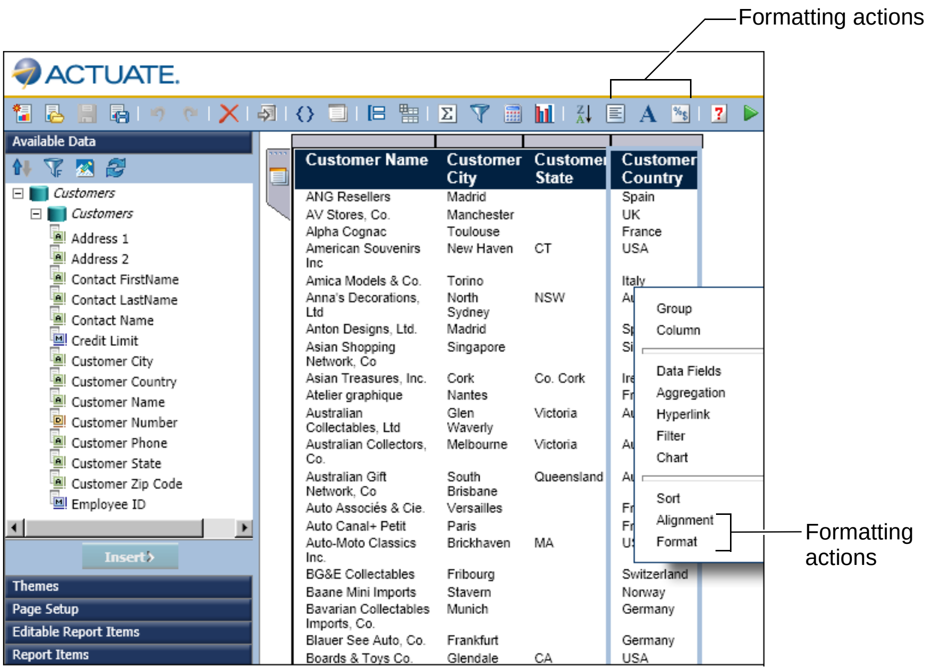


Figure 14-1 Default BIRT Studio page

Listing 14-2 shows a change to the Formatting action set. Its <visible> attribute, shown in bold, is set to false. Note, however, that all the actions under the Formatting action set are still enabled.

Listing 14-2 Visibility of the Formatting action set, changed to false

```
<actionSet>
   <name>Formatting</name>
   <visible>false</visible>
   <action>
      <name>AlignLeft</name>
      <enabled>true</enabled>
   </action>
   <action>
      <name>AlignCentert</name>
```

```
        <enabled>true</enabled>
    </action>
    <action>
        <name>AlignRight</name>
        <enabled>true</enabled>
    </action>
    <action>
        <name>Font</name>
        <enabled>true</enabled>
    </action>
    <action>
        <name>Border</name>
        <enabled>true</enabled>
    </action>
    <action>
        <name>ConditionalFormat</name>
        <enabled>true</enabled>
    </action>
    <action>
        <name>Data</name>
        <enabled>true</enabled>
    </action>
</actionSet>
```

Figure 14-2 shows the updated BIRT Studio page.

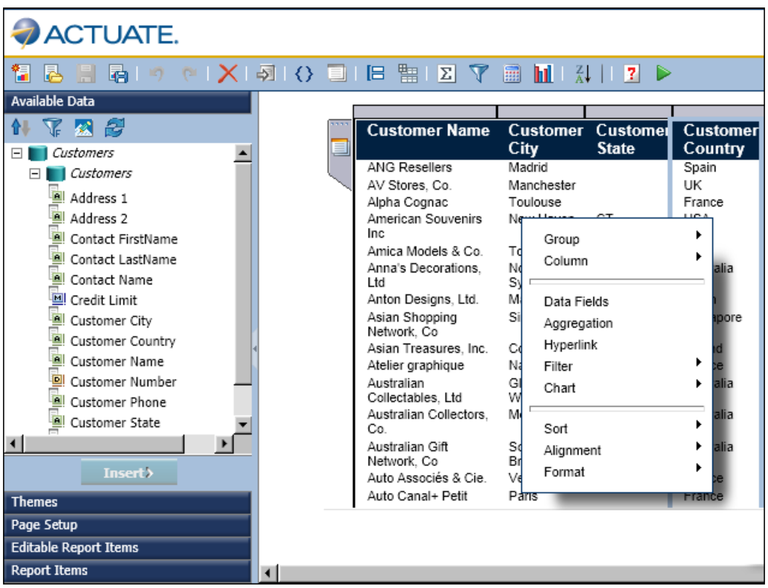


Figure 14-2 Updated toolbar and context menu, without any formatting functions

None of the formatting actions appear in the toolbar or the context menu. Setting the <visible> attribute of an action set to false disables all actions within the action set.

Listing 14-3 shows changes to the Font and Data actions within the Formatting action set. The Formatting action set's <visible> attribute is set to true. The Font and Data actions are disabled. The other actions in the action set are still enabled.

Listing 14-3 Font and Data (formatting) actions disabled

```
<actionSet>
   <name>Formatting</name>
   <visible>true</visible>
   <action>
      <name>AlignLeft</name>
      <enabled>true</enabled>
   </action>
   <action>
      <name>AlignCentert</name>
      <enabled>true</enabled>
   </action>
   <action>
      <name>AlignRight</name>
      <enabled>true</enabled>
   </action>
   <action>
      <name>Font</name>
      <enabled>false</enabled>
   </action>
   <action>
      <name>Border</name>
      <enabled>true</enabled>
   </action>
   <action>
      <name>ConditionalFormat</name>
      <enabled>true</enabled>
   </action>
   <action>
      <name>Data</name>
      <enabled>false</enabled>
   </action>
</actionSet>
```

Figure 14-3 shows the updated BIRT Studio page. The alignment actions are available on the toolbar and on the context menu, but not the Font and Data formatting actions.

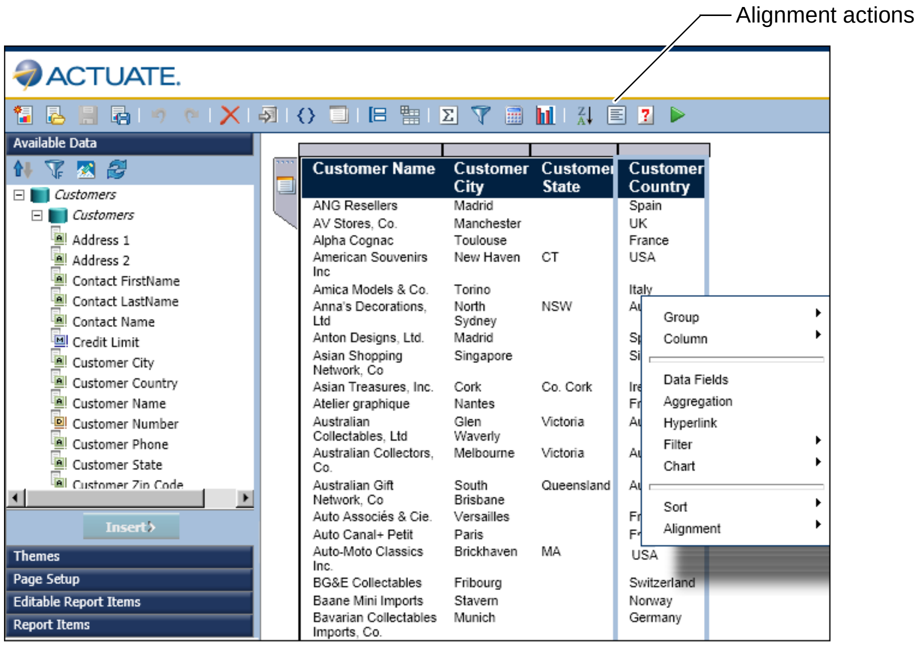


Figure 14-3 Updated toolbar and context menu, without the Font and Data formatting actions

Specifying the default position of aggregate values

In Release 10 and earlier, aggregate values were placed in a group's footer by default. In Release 11 and later, aggregate values are placed in a group's header by default. To revert to the behavior prior to Release 11, set EnableNewAggregationStyle to false in erni_config.xml:

```
<featureConfigs>
   <featureConfig>
      <name>EnableNewAggregationStyle</name>
      <value>false</value>
   </featureConfig>
</featureConfigs>
```

Configuring the application environment

You can change the values of the configuration parameters in the following file:

```
<context root>\WEB-INF\web.xml
```

BIRT Studio parameters control how BIRT Studio operates a web application and how it interoperates with any Java Components packaged with it in a WAR file. Table 14-2 describes the configuration parameters for BIRT Studio.

Table 14-2 BIRT Studio web.xml parameters

Parameter name	Description
BIRT_RESOURCE_PATH	Path to Actuate BIRT shared resources, including libraries and templates for Actuate BIRT report designs and BIRT Studio. The default value is <context root>\resources.
BIRT_SCRIPT_LIB_PATH	Path to script libraries. The default value is <context root>\scriptlib.
DEFAULT_DATA_CACHE_ROW_COUNT	The number of data rows to display in BIRT Studio when designing a report. The default value is 100.
DEFAULT_LOCALE	The default locale. The default locale is en_US. Users can select a locale when they log in.
DEFAULT_PAGE_BREAK_INTERVAL	The number of rows to display on one page when viewing a report. A value of 0 indicates no page breaks.
DEFAULT_REPORT_TEMPLATE_CATEGORY_NAME	The default BIRT Report template category to load when a user opens BIRT Studio. The default value is Standard.
DEFAULT_TIMEZONE	The default time zone. The default time zone is Pacific Standard Time (PST).
DEFAULT_WORKGROUP_FUNCTIONALITY_ROLE	Specifies the functionality role for all users of BIRT Studio, and the BIRT Viewer. The default value is Active Portal Intermediate.
ENABLE_CASE_SENSITIVE_STRING_FILTER	Specifies whether the filter dialog considers case. True specifies case sensitivity, for example, CA matches CA only, not ca or Ca. The default value is false.
MAX_BRSTUDIO_DESIGN_SESSION	The maximum number of designs a user can edit concurrently in BIRT Studio. The default is 10.
MAX_BRSTUDIO_USER_SESSION	The maximum number of concurrent BIRT Studio sessions on the server. The default is 256.
MAX_DATA_CACHE_ROW_COUNT	Limits the number of data rows that a user can choose to display in Actuate BusinessReport Studio when designing a report. The default value is 200.

(continues)

Table 14-2 BIRT Studio web.xml parameters (continued)

Parameter name	Description
MAX_NUMBER_OF_ VALUES_FOR_ DYNAMIC_PARAMETER	The number of values shown in the parameter dialog box for a dynamic value parameter in BIRT Studio: ■ A positive number value N means only the first N values appear the parameter dialog box. ■ A value of 0 means all values from the data source appear in the parameter dialog box. The default value is 0. ■ A value of -1 means only the first N values appear where N is the current data cache row count setting for the current design session. MAX_NUMBER_OF_VALUES_FOR_DYNAMIC_ PARAMETER only applies to a dynamic value parameter. All the values appear for a static value parameter no matter how many values it has. For a static value parameter, the full list appears in the parameter dialog box when the user chooses Save and View.
MEMORY_DATA_ CACHE_ROW_COUNT	Specifies the number of data rows to cache in memory. The default value is 50.
MORE_VALUE_ROW_ COUNT	Specifies the number of rows to fetch when a user chooses to filter a report on a column in BIRT Studio. The default value is 200.
SEARCH_ENABLE_ COLUMN_HEADERS	Indicates whether to include column headings in report search results when the output format is CSV or TSV. Set this parameter to true, the default value, to include column headings.
SEARCH_USE_ QUOTE_DELIMITER	Indicates whether to enclose search results in quotation marks when the output format is CSV or TSV. The default value is true, which encloses the results in quotation marks.

Chapter

15

Actuate BIRT Studio URIs

This chapter contains the following topics:

- Accessing BIRT Studio using a URI
- Using the BIRT Studio servlet
- Using the BIRT Studio URLs
- BIRT Studio file manager URIs overview
- BIRT Studio file manager URIs quick reference
- Common URI parameters
- BIRT Studio file manager URIs reference
- Actuate BIRT Viewer URIs reference

Accessing BIRT Studio using a URI

BIRT Studio is a web application that is initiated by a Java servlet. The BIRT studio servlet manages binary content and performs tasks such as uploading and downloading binary files.

You invoke the BIRT Studio servlet using the following syntax:

```
http://<application server>:<port>/<context root>/wr
```

where

- application server is the name of the machine hosting the application server.
- port is the port on which the application server listens for requests.
- context root is the BIRT Studio context root.
- wr is the name to which the servlet is mapped in the web application's web.xml file. A typical location for web.xml is <context root>\WEB-INF.

Servlet names are case-sensitive. Do not modify the servlets, their names, or their mapping in web.xml.

Using the BIRT Studio servlet

The BIRT Studio servlet loads the BIRT Studio user interface and establishes a connection to a report repository. A report repository is required in order to use the servlet.

Name com.actuate.erni.servlet.ERNIViewerServlet

Invoke the BIRT Studio servlet as:

```
http://<web server>:<port>/<context root>/wr?<parameters>
```

URI parameters The BIRT Studio servlet requires a repository parameter in order to operate. Table 15-1 lists and describes the URI parameter for the BIRT Studio servlet.

Table 15-1 BIRT Studio URI parameter

URI parameter	Description
repositoryType	The repository type. Use workgroup for the local file system.

Using the BIRT Studio URLs

You can log in to BIRT Studio by typing a URL in a web browser's address field. In addition to the initial BIRT Studio page, you can open BIRT Studio with a specific report design.

In the example URLs in the following topics, special characters are represented by codes, as shown in Table 15-2.

Table 15-2 Codes for special characters in URLs

Character	Code
Colon (:)	%3a
Slash (/)	%2f
Period (.)	%2e
Space ( )	%20

How to open BIRT Studio and load an existing report design

To open an existing report design in BIRT Studio, use a URL like the one shown in the following example:

```
http://localhost:8080/DeploymentKit/wr?
__report=%2fHome%2fPublic%2fCustomers%2erptdesign
&pCountry=USA
```

where

- __report=%2fHome%2fPublic%2fCustomers%2erptdesign is the path to the report design to use.
- pCountry=USA is a parameter-value pair for the report design.

BIRT Studio file manager URIs overview

The following sections describe the Java Component BIRT Studio JSPs that manage files and folders. For a repository managed with iServer, Information Console provides this functionality. For information about Information Console URIs, see *Information Console Developer Guide.*

The following sections provide quick reference tables and detailed reference information about Actuate BIRT Studio URIs. An Actuate BIRT Studio URI is a directive to Actuate BIRT Studio to perform an action, such as showing a list of files, rather than change the appearance of the application.

File manager pages use the .do extension for the Struts action mapping to a page. The complete page name appears as part of the reference material. BIRT Studio page and folder names are case-sensitive.

BIRT Studio file manager URIs quick reference

Table 15-3 lists the BIRT Studio file manager URIs.

Table 15-3 Actuate BIRT Studio URI pages

BIRT Studio page	Description
banner page	Displays a banner at the top of each BIRT Studio file manager page.
browse file page	Provides file and folder browsing functionality for the submit request pages.
browse page	See browse file page.
delete file status page	Displays whether a file was successfully deleted.
detail page	Supports error handling and presenting object details.
drop page	Supports deleting files or cancelling running jobs.
error page	Retrieves an error message from the exception or the request and displays it.
execute report page	Submits a run report job request to the server.
executereport page	See execute report page.
getfiledetails page	See file or folder detail page.
getfolderitems page	See file and folder index page.
list page	Supports listing files and folders.
page not found page	Displays an error message when a JSP is unavailable in BIRT Studio file manager.
parameters page	Presents a list of the request parameters.

Common URI parameters

All Actuate BIRT Studio file manager URIs have the parameters shown in Table 15-4. String values that are too long are truncated for all parameters. The web browser that you use determines the length of parameters.

Table 15-4 Common Actuate BIRT Studio URI parameters

URI parameter	Description
serverURL	Contains the URI that accesses the Actuate web application, such as http://Services:8000>
timezone	The current user's time zone
volume	The volume to which the user is connected

BIRT Studio file manager URIs reference

This section provides the detailed reference for BIRT Studio file manager URIs. In the definitions, <context root> represents the name of your Actuate BIRT Studio context root.

Table 15-5 lists the topics this chapter covers and the file names discussed in each topic. All pages are under the BIRT Studio context root.

Table 15-5 Actuate BIRT Studio pages

Topic	BIRT Studio file
banner page	iportal\activePortal\private\common\banner.jsp
browse file page	browsefile.do
	iportal\activePortal\private\query\browse.jsp
delete file status page	iportal\activePortal\private\filesfolders \deletefilestatus.jsp
detail page	
■ error detail page	iportal\activePortal\errors\detail.jsp
	getfiledetails.do
■ file or folder detail page	iportal\activePortal\private\filesfolders\filedetail.jsp
error page	errors\error.jsp
	iportal\activePortal\private\common\errors \error.jsp
execute report page	executereport.do
file or folder drop page	deletefile.do

(continues)

Table 15-5 Actuate BIRT Studio pages (continued)

Topic	BIRT Studio file
file and folder index page	getfolderitems.do iportal\activePortal\private\filesfolders \filefolderlist.jsp
file and folder list page	getfolderitems.do iportal\activePortal\private\filesfolders \filefolderlist.jsp
page not found page	iportal\activePortal\errors\pagenotfound.jsp
parameters page	iportal\activePortal\private\newrequest \parameters.jsp
viewer page for Actuate BIRT reports and business reports	IVServlet

banner page

Provides the banner that appears across the top of all Actuate BIRT Studio web pages. The default banner displays the Actuate logo, user name, cluster name, and volume name, and provides links for Logout, Options, and Help. The banner page obtains the user name, cluster name, and volume name from variables maintained by the authenticate page.

Name <context root>\iportal\activePortal\private\common\banner.jsp

Used by
iportal\activePortal\private\login.jsp
iportal\activePortal\private\channels\channelnoticelist.jsp
iportal\activePortal\private\channels\channeloperationstatus.jsp
iportal\activePortal\private\filesfolders\deletefilestatus.jsp
iportal\activePortal\private\filesfolders\filedetail.jsp
iportal\activePortal\private\filesfolders\filefolderlist.jsp
iportal\activePortal\private\jobs\getjobdetails.jsp
iportal\activePortal\private\jobs\joboperationstatus.jsp
iportal\activePortal\private\jobs\selectjobs.jsp
iportal\activePortal\private\newrequest\newrequest.jsp
iportal\activePortal\private\newrequest\newrequest2.jsp
iportal\activePortal\private\newrequest\submitjobstatus.jsp
iportal\activePortal\private\options\options.jsp
iportal\activePortal\private\query\create.jsp
iportal\activePortal\private\query\execute.jsp

browse file page

Contains file and folder browsing functionality used by other submit request pages.

Name <context root>\browsefile.do

<context root>\iportal\activePortal\private\query\browse.jsp

Parameters workingFolder is the name of the folder for which to display contents in the folder browser window. The browse file page also uses the common URI parameters.

Used by iportal\activePortal\private\newrequest\browse.jsp
iportal\activePortal\private\query\browse.jsp

delete file status page

Summarizes the result of a deletion performed by the drop page and indicates whether a file was successfully deleted. The delete file status page includes authenticate to obtain user session data. BIRT Studio performs the deletion as part of an action and then forwards to the delete file status page.

Name <context root>\iportal\activePortal\private\filesfolders\deletefilestatus.jsp

Used by Not applicable.

detail page

Displays detailed information about repository objects. There are two detail pages:

<context root>\iportal\activePortal\errors

<context root>\iportal\activePortal\filesfolders

error detail page

Provides a template error page that can be embedded in another page.

Name <context root>\iportal\activePortal\errors\detail.jsp

Used by iportal\activePortal\private\common\errors\error.jsp
iportal\activePortal\viewer\print.jsp
iportal\activePortal\viewer\saveas.jsp
iportal\activePortal\viewer\searchframe.jsp

iportal\activePortal\viewer\viewdefault.jsp
iportal\activePortal\viewer\viewtoc.jsp

file or folder detail page

Displays detailed information about the selected viewable folder or file. Users request file details by choosing the magnifying glass icon to the right of files listed on the folder page, or folder details by choosing the magnifying glass icon to the right of the folder name in the breadcrumb. Users can request another viewable document or delete the current file or folder from the file or folder detail page. filedetail.jsp uses the HTML code in <context root>\iportal\activePortal\private\filesfolders\filedetailcontent.jsp to display the information.

Name <context root>\getfiledetails.do

<context root>\iportal\activePortal\private\filesfolders\filedetail.jsp

Parameters Table 15-6 describes the parameters for the file or folder detail page. The file or folder detail page also uses the common URI parameters.

Table 15-6 file or folder detail URI parameters

URI parameter	Description
name	The full path name of the repository object for which to show details. This parameter is ignored if objectID is also specified.
objectID	The repository object's unique identifier.
version	The repository object's version number. The default is the latest version.

Used by Not applicable.

drop page

Deletes one or more files or folders.

file or folder drop page

Deletes the specified file or folder. The file or folder drop page includes the banner page to obtain user session data.

Name <context root>\deletefile.do

Parameters Table 15-7 describes the parameters for the file or folder drop page. The file or folder drop page also uses the common URI parameters.

Table 15-7 file or folder drop URI parameters

URI parameter	Description
ID	The unique identifier of the repository object to delete.
name	The full path name of the repository object to delete. Multiple name parameters, to delete more than one file or folder at a time, are allowed. This parameter is ignored if ID is also specified.
redirect	URI to which to redirect the job deletion page. The default redirect page is processedaction_status.

Used by Not applicable.

error page

Displays the specified error message. BIRT Studio uses two pages. All BIRT Studio code uses <context root>\iportal\activePortal\private \common\errors\error.jsp.

Name <context root>\iportal\activePortal\errors\error.jsp

<context root>\iportal\activePortal\private\common\errors\error.jsp

Used by iportal\activePortal\private\login.jsp
iportal\activePortal\private\common\closewindow.jsp
iportal\activePortal\private\common\sidebar.jsp
iportal\activePortal\private\common\errors\errorpage.jsp
iportal\activePortal\private\options\options.jsp
iportal\activePortal\private\query\create.jsp
iportal\activePortal\private\query\execute.jsp
iportal\activePortal\private\templates\template.jsp
iportal\activePortal\viewer\closewindow.jsp
iportal\activePortal\viewer\print.jsp
iportal\activePortal\viewer\saveas.jsp
iportal\activePortal\viewer\searchframe.jsp
iportal\activePortal\viewer\searchreport.jsp
iportal\activePortal\viewer\viewframeset.jsp

execute report page

Submits a run report job request.

When executing a report job or query, a Cancel button appears after a specified wait time passes. You can change the time by setting the EXECUTE_REPORT_WAIT_TIME configuration parameter in the appropriate BIRT Studio configuration file.

For reports that accept run-time parameters, you can set the parameter in the URL by adding an ampersand (&), the parameter name, and an equal (=) sign, followed by the parameter value in quotes.

Name <context root>\executereport.do

Parameters Table 15-8 describes the parameters for the execute report page. The execute report page also uses the common URI parameters.

Table 15-8 execute report URI parameters

URI parameter	Description
__ageDays	Use with __ageHours to determine how long output objects exist before they are automatically deleted. Use only if __archivePolicy is set to Age. __ageDays can be any positive number.
__ageHours	Use with __ageDays to determine how long output objects exist before they are automatically deleted. Use only if __archivePolicy is set to Age. __ageHours can be any positive number.
__executableName	The name of the executable file for this request.
invokeSubmit	Controls whether the browser is redirected to the parameter screen or whether the report job is run immediately. If true, the report job is executed without displaying the parameters. If false, the parameters are displayed. False is the default.
__outputDocName	The name and path of the resulting BIRT document. This parameter is only usable for BIRT reports when the BIRT_SAVE_REPORT_DOCUMENT_ENABLED parameter is set to true in web.xml. If the given path is absolute, then executereport saves the report to that path. If the given path is relative, then executereport saves the report to the path set in the BIRT_SAVE_REPORT_DOCUMENT_PATH web.xml parameter.

Table 15-8 execute report URI parameters

URI parameter	Description
__priority	Specifies the job submission priority. Values are High, Medium, and Low.
__priorityValue	Specifies a number ranging from 1 to 1000 and corresponding to the job submission priority. You can only specify values allowed by your functionality level.
__progressive	Indicates whether to display the report document after it generates. If false, the report document displays after it generates. If true, the report document displays progressively, as it generates.
__serverURL	Contains the URI that accesses the JSP engine, such as http://Services:8900.
__wait	If "wait", BIRT Studio waits for the report generation to be completed before displaying it. If "nowait", BIRT Studio displays the first page right away even if the report job is not completed.

For example, the following URL executes the Sales By Territory.rptdesign report immediately with the Territory run-time parameter set to EMEA:

```
http://localhost:8080/ActuateJavaComponent/executereport.do?
   __requesttype=immediate&__executableName=/Public
   /BIRT%20and%20BIRT%20Report%20Studio%20Examples
   /Sales%20by%20Territory.rptdesign&userid=anonymous&
   __saveOutput=false&Territory=%22EMEA%22&invokeSubmit=True
```

The following parameter names are reserved for internal use only by the execute report page:

- doframe
- inputfile
- jobType
- name
- selectTab

Used by Not applicable.

index page

Provides the entry point and structure for the parts of BIRT Studio generated from multiple files.

file and folder index page

The default entry point to the BIRT Studio web application. The file and folder index page provides the entry point and structure to support the Files and Folders functionality. The structure is a table that BIRT Studio uses to format and present files and folders data. Page content varies depending on the BIRT Studio directive.

The file and folder index page uses the banner page to provide the reporting web page banner. filefolderlist.jsp uses the HTML code in <context root>\iportal\activePortal\private\filesfolders\filefolderlistcontent.jsp to display files and folders data.

Name <context root>\getfolderitems.do

<context root>\iportal\activePortal\private\filesfolders\filefolderlist.jsp

Parameters Table 15-9 describes the parameters for file and folder index page. The file and folder index page also uses the common URI parameters.

Table 15-9 file and folder index URI parameters

URI parameter	Description
startUpMessage	Specifies a message to appear when BIRT Studio calls this page.
subpage	Specifies the content of the page. Possible values are: ■ _list: include list ■ _detail: include detail Specifying any other value for subpage invokes the page not found page.

list page

Lists files in a container, such as a folder.

file and folder list page

Presents a list of objects that reside in the current working repository folder. Users request folder listings by choosing links on the reporting web page. The file and folder list page includes a filter section where users specify criteria for viewing report documents.

When users access a repository for the first time, BIRT Studio displays their home folder, if they have one, or the top folder in the repository. All files and folders in that folder that they have permission to view appear in the BIRT Studio listing page. Users can specify a filter to choose the types of files to view.

Name <context root>\getfolderitems.do

<context root>\iportal\activePortal\private\filesfolders\filefolderlist.jsp

Parameters Table 15-10 describes the parameters for the file and folder list page. The file and folder list page also uses the common URI parameters.

Table 15-10 file and folder list URI parameters

URI parameter	Description
applyFilter	If true, apply filter. If false, filter not applied.
filter	The filter specifying the file and folder names to list. Filter is a string. The default is "".
folder	The folder for which to list the contents. Folder name is a string. If no folder is specified, List uses the last working folder known for the session if cookies are enabled. If cookies are not enabled, List uses the user's home folder as specified in the user settings.
onlyLatest	If true, show only the latest version of a file if multiple versions exist. If false, show all versions of a file if multiple versions exist. The default is false.
resetFilter	Any non-null value for resetFilter causes the filter to return to its original state. Users can reset the filter by choosing the Default button on the listing page.
showDocument	If true, show all viewable documents. If false, do not show viewable documents. The default is true.
showExecutables	If true, show all report executables. If false, do not show report executables. The default is true.
showFolders	If true, show all folders. If false, do not show folders. The default is true.

Used by Not applicable.

page not found page

Displays an error message when BIRT Studio cannot find the page that a user specifies. This page is a BIRT Studio page only.

Name <context root>\iportal\activePortal\errors\pagenotfound.jsp

Used by Not applicable.

parameters page

Displays job parameters. Users access the parameters list by choosing Save and View for a .rptdesign that contains parameters.

Parameters looks like Figure 15-1.

Name <context root>\iportal\activePortal\private\newrequest\parameters.jsp

Used by iportal\activePortal\private\newrequest\newrequestpage.jsp

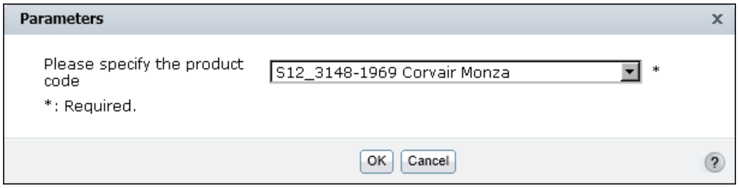


Figure 15-1 Parameters page

Actuate BIRT Viewer URIs reference

To view and interact with Actuate BIRT reports, you use the Actuate BIRT servlet. All BIRT Viewer options and varieties use the same URL. For detailed information about the BIRT servlet URL, see *Working with Actuate BIRT Viewers*.

Index

Symbols

Numerics

A

B

C

D

G

H

I

J

L

M

Q

R

S

T

U

V

W

X

Y